Numer. Math. Theor. Meth. Appl. Vol. **19**, No. 1, pp. 129-154
doi: 10.4208/nmtma.OA-2025-0030 February 2026

A Fast Fully Discrete Mixed Finite Element Scheme for Fractional Viscoelastic Models of Wave Propagation

Hao Yuan and Xiaoping Xie[*]

School of Mathematics, Sichuan University, Chengdu 610064, P.R. China

Received 14 March 2025; Accepted (in revised version) 19 November 2025

Abstract. Due to the nonlocal feature of fractional differential operators, the numerical solution to fractional partial differential equations usually requires expensive memory and computation costs. This paper develops a fast scheme for fractional viscoelastic models of wave propagation. We first apply the Laplace transform to convert the time-fractional constitutive equation into an integro-differential form that involves the Mittag-Leffler function as a convolution kernel. Then we construct an efficient sum-of-exponentials (SOE) approximation for the Mittag-Leffler function. We use mixed finite elements for the spatial discretization and the Newmark scheme for the temporal discretization of the second time-derivative of the displacement variable in the kinematical equation and finally obtain the fast algorithm. Compared with the traditional L1 scheme for time fractional derivative, our fast scheme reduces the memory complexity from $\mathcal{O}(N_sN)$ to $\mathcal{O}(N_sN_{\rm exp})$ and the computation complexity from $\mathcal{O}(N_sN^2)$ to $\mathcal{O}(N_sN_{\rm exp}N)$, where N denotes the total number of temporal grid points, $N_{\rm exp}$ the number of exponentials in SOE, and N_s the complexity of memory and computation related to the spatial discretization. Numerical experiments confirm the theoretical results.

AMS subject classifications: 65N30, 65M60, 65M12

Key words: Fractional viscoelastic model, wave propagation, Mittag-Leffler function, sum-of-exponentials approximation, Caputo derivative, fast scheme.

1. Introduction

Assume that $\Omega \subset \mathbb{R}^d$ $(d = 2,3)$ is a bounded open domain with boundary $\partial\Omega$, $T > 0$ is the time length, and $\alpha \in (0,1)$ is a constant. Consider the following fractional viscoelastic model of wave propagation which was introduced by Caputo and

[*] Corresponding author. *Email addresses:* `xpxie@scu.edu.cn` (X.-P. Xie), `drifteryuan@qq.com` (H. Yuan)

Mainardi [12]:

$$\begin{cases} \rho \mathbf{u}_{tt} - \mathbf{div}\, \sigma = \mathbf{f}, & (x,t) \in \Omega \times [0,T], \qquad (1.1a) \\ \sigma + \tau_\sigma^\alpha \dfrac{\partial^\alpha \sigma}{\partial t^\alpha} = \mathbb{D}\left(\varepsilon(\mathbf{u}) + \tau_\varepsilon^\alpha \dfrac{\partial^\alpha \varepsilon(\mathbf{u})}{\partial t^\alpha} \right), & (x,t) \in \Omega \times [0,T], \qquad (1.1b) \\ \mathbf{u} = 0, & (x,t) \in \partial\Omega \times [0,T], \qquad (1.1c) \\ \mathbf{u}(x,0) = \mathbf{u}_0, \quad \mathbf{u}_t(x,0) = \mathbf{v}_0, \sigma(x,0) = \sigma_0, & x \in \Omega. \qquad (1.1d) \end{cases}$$

Here $\mathbf{u} = (u_1, \ldots, u_d)^T$ is the displacement field, $\sigma = (\sigma_{ij})_{d\times d}$ the symmetric stress tensor, $\mathbf{div}\, \sigma = (\sum_{i=1}^d \partial_i \sigma_{i1}, \cdots, \sum_{i=1}^d \partial_i \sigma_{id})^T$, $\varepsilon(\mathbf{u}) = (\bigtriangledown \mathbf{u} + (\bigtriangledown \mathbf{u})^{\mathrm{T}})/2$ the strain tensor, τ_σ the relaxation time, τ_ε the retardation time, $\rho(x)$ the mass density, and $\mathbb{D}$ the fourth order symmetric tensor. $\mathbf{f} = (f_1, \ldots, f_d)$ is the body force and $\mathbf{u}_0(x)$, $\mathbf{v}_0(x)$, $\sigma_0(x)$ are initial data. For any function $\mathbf{v}(x,t)$, denote $\mathbf{v}_t := \partial \mathbf{v}/\partial t$ and $\mathbf{v}_{tt} := \partial^2 \mathbf{v}/\partial t^2$, and for $0 < \alpha < 1$, let $\partial^\alpha \mathbf{v}/\partial t^\alpha$ be the α-order Caputo fractional derivative of $\mathbf{v}(x,t)$ defined by

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

We note that the following three classical viscoelastic models correspond to different choices of the relaxation/retardation time in the constitutive equation (1.1b) with $\alpha = 1$: the Kelvin-Voigt model ($\tau_\sigma = 0$, $\tau_\varepsilon > 0$); the Maxwell model ($\tau_\sigma > 0$, $\tau_\varepsilon = 0$) and the Zener model ($\tau_\sigma > 0$, $\tau_\varepsilon > 0$).

Many materials display elastic and viscous kinematic behaviours simultaneously. Such a feature, called viscoelasticity, is commonly characterized by using springs, which obey the Hooke's law, and viscous dashpots, which obey the Newton's law. Different combinations of the springs and dashpots lead to various viscoelastic models, e.g. the Zener model, the Kelvin-Voigt model and the Maxwell model. We refer the reader to [10,15,16,19,24,26,41,42] for several monographs on the development and application of the viscoelasticity theory.

In recent decades, fractional order differential operators, as extension of integer order ones, have been widely used in many scientific and engineering fields such as physics, chemistry, materials science, biology, finance and other sciences, due to their ability to accurately describe states or development processes with memory and hereditary characteristics. As far as the viscoelastic materials with complex rheological properties are concerned, more and more studies indicate that, comparing with the integer order models, time fractional viscoelastic models can more precisely characterize the creep and relaxation dynamic behaviours and capture the effects of "fading" memory [5,6,9,11,22,23,36,40].

There are some works in the literature on the numerical analysis of time fractional viscoelastic models. Enelund and Josefson [17] rewrote the constitutive equation of fractional Zener model (Riemann Liouville type) as an integro-differential equation with a weakly singular convolution kernel by Laplace transform and carried out finite element simulation. Based on the integro-differential form of constitutive equation from [17], Adolfsson *et al.* [1] proposed a piecewise constant discontinuous Galerkin method for a fractional order (Riemann Liouville type) viscoelastic differential equation. Subsequently, they applied a discontinuous Galerkin method in time and a continuous Galerkin finite element method in space to discretize the quasi-static fractional viscoelastic model [2]. Yu *et al.* [46] adopted finite element simulation for a fractional Zener model (Riemann Liouville type) with integro-differential form of constitutive equation in 3D cerebral arteries and aneurysms. Lam *et al.* [32] presented a finite element scheme for 1D fractional Zener model (Caputo type) with integro-differential form of constitutive equation. Liu and Xie [34] proposed a semi-discrete hybrid stress finite element method for a time fractional viscoelastic model, where the corresponding integro-differential equation is of a Mittag-Leffler type convolution kernel, and derived error estimate for the semi-discrete scheme.

The nonlocal feature of fractional differential operators usually means expensive computational cost and memory cost in the numerical simulation of fractional models. To tackle such difficulties, Lubich and Schädle [35] proposed a new algorithm for the evaluation of convolution integral when solving wave propagation problems. The algorithm is based on local SOE approximation for the inverse Laplace transform of kernel function by applying trapezoidal rule to the contour integral. Li [33] presented a locally SOE approximation for the integral representation of the kernel function by using an efficient Q-point Gauss–Legendre quadrature. Yu *et al.* [46] considered an SOE approximation of Mittag-Leffler function by applying trapezoidal rule to the contour integral and applied it to the fractional Zener model. Jiang *et al.* [30] and Yan *et al.* [44] split the convolution integral in the Caputo fractional derivative into a local part and a history part, and presented fast algorithms for time fractional diffusion equations by adopting the SOE approximation (using Gauss-Jacobi quadrature and Gauss-Legendre quadrature) for the history part and L1 (L2-1_σ) formula for the local part. Baffet [3] divided the fractional integral of a function f into a history term (convolution of the history of f and a regular kernel) and a local term, and gave a method for fractional differential equations by using SOE approximation (by Gauss-Jacobi quadrature) for the history part and an implicit scheme for the local part. Zeng *et al.* [47] developed a unified fast time-stepping method for both fractional integral and derivative operators by using truncated Laguerre-Gauss quadrature for the kernel function in history part and a direct convolution method for local part. Lam *et al.* [32] gave an SOE approximation (by Gauss-Legendre quadrature) for the integral representation of Mittag-Leffler function and applied it to a 1D fractional Zener model. We refer to [4,7,13,18,28,29,43,45,48] for some other fast algorithms for time fractional order PDEs.

In this paper, we present an efficient numerical scheme for solving the fractional viscoelastic model (1.1). Our contribution lies in the following aspects:

- The constitutive equation (1.1b) is converted to an integro-differential form with Mittag-Leffler function as the convolution kernel.

- An efficient SOE approximation (different from that of [32]) is proposed for the Mittag-Leffler function and applied to accelerate the evaluation of the convolution. For a given tolerance error ϵ of the proposed SOE approximation, its computation complexity is $N_{\exp} = \mathcal{O}(|\log \epsilon|^2)$.

- An estimate of the truncation error of the SOE approximation is derived. We note that there is no truncation error estimation in [32].
- The proposed SOE approximation is applied to the fractional viscoelastic model to get a fast numerical scheme.
- The resulting fast scheme requires $\mathcal{O}(N_s N_{\exp})$ memory complexity and $\mathcal{O}(N_s N_{\exp} N)$ computation complexity, in contrast to $\mathcal{O}(N_s N)$ and $\mathcal{O}(N_s N^2)$ for the traditional L1 scheme. Here N denotes the total number of temporal grid points and N_s represents the complexity of memory and computation related to the spatial discretization. In particular, if the tolerance error of the SOE approximation is taken as $\epsilon = \triangle t = T/N$, we will have $N_{\exp} = \mathcal{O}(\log^2 N)$ (cf. Remark 2.5).

The rest of this paper is arranged as follows. Section 2 introduces some preliminaries on the SOE approximation of Mittag-Leffler function. Section 3 gives two numerical schemes: the L1-Newmark scheme and the fast scheme with the SOE approximation. Finally, numerical examples are provided in Section 4 to verify the performance of the SOE approximation and the fast scheme.

2. Preliminary results

2.1. Alternative form of the constitutive law and weak formulations

We note that the constitutive equation (1.1b) is of the following differential form:

$$\sigma + \tau_\sigma^\alpha \frac{\partial^\alpha \sigma}{\partial t^\alpha} = \mathbb{D}\left(\varepsilon(\mathbf{u}) + \tau_\varepsilon^\alpha \frac{\partial^\alpha \varepsilon(\mathbf{u})}{\partial t^\alpha}\right). \tag{2.1}$$

In this subsection we shall convert it to an explicit expression of σ when $\tau_\sigma \neq 0$. To this end, we first introduce two basic tools: the Laplace transform and the Mittag-Leffler function.

Let $f(t)$ be a function defined in $\mathbb{R}^+$. The Laplace transform of $f(t)$ is defined by

$$\hat{f}(s) := \mathcal{L}\big(f(t); s\big) = \int_0^\infty f(t) e^{-st} dt,$$

where $s \in \mathbb{C}$ and Re $s \geq 0$. There holds the following property of the Laplace transform for the Caputo fractional derivative [14,39]:

$$\mathcal{L}\left(\frac{\partial^\alpha f}{\partial t^\alpha}(t); s\right) = s^\alpha \hat{f}(s) - s^{\alpha-1} f(0), \quad \alpha \in (0,1). \tag{2.2}$$

For $\alpha > 0$, and $\beta \in \mathbb{R}$, the two-parameter Mittag-Leffler function is defined by

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

In particular, the one-parameter Mittag-Leffler function is given by

$$E_{\alpha}(z) := E_{\alpha,1}(z) = \sum_{j=0}^{\infty} \frac{z^j}{\Gamma(j\alpha+1)}.$$

There hold the following properties (cf. [14, 31]).

Lemma 2.1. *(1) For $\alpha, \beta > 0$ and $z \in \mathbb{C}$, there holds*

$$E_{\alpha,\beta}(z) = zE_{\alpha,\alpha+\beta}(z) + \frac{1}{\Gamma(\beta)}. \tag{2.3}$$

(2) For $\alpha, \beta > 0, \lambda \geq 0$ and $t > 0$, there holds

$$\mathcal{L}\big(t^{\beta-1}E_{\alpha,\beta}(-\lambda t^{\alpha}); s\big) = \frac{s^{\alpha-\beta}}{s^{\alpha}+\lambda}, \quad \mathrm{Re}\, s \geq 0. \tag{2.4}$$

It has been shown in [25, 37] that the following integral identity for the Mittag-Leffler function of $-t^{\alpha}$ holds for $t > 0$ and $0 < \alpha < 1$:

$$E_{\alpha}(-t^{\alpha}) = \frac{\sin(\alpha\pi)}{\pi} \int_0^{\infty} \frac{s^{\alpha-1}}{s^{2\alpha} + 2s^{\alpha}\cos(\alpha\pi) + 1} e^{-st} ds. \tag{2.5}$$

We are now at a position to derive the explicit expression of σ from the constitutive equation (2.1). To begin with, we Laplace-transform (2.1) and apply (2.2) to obtain

$$\mathbb{D}^{-1}\left(\hat{\sigma} + \tau_{\sigma}^{\alpha}(s^{\alpha}\hat{\sigma} - s^{\alpha-1}\sigma_0)\right) = \varepsilon(\hat{\mathbf{u}}) + \tau_{\varepsilon}^{\alpha}\big(s^{\alpha}\varepsilon(\hat{\mathbf{u}}) - s^{\alpha-1}\varepsilon(\mathbf{u}_0)\big),$$

which yields

$$\begin{aligned}
\mathbb{D}^{-1}\hat{\sigma} &= \frac{1+(\tau_{\varepsilon}s)^{\alpha}}{1+(\tau_{\sigma}s)^{\alpha}}\varepsilon(\hat{\mathbf{u}}) + \frac{s^{\alpha-1}}{1+(\tau_{\sigma}s)^{\alpha}}\left(\tau_{\sigma}^{\alpha}\mathbb{D}^{-1}\sigma_0 + \tau_{\varepsilon}^{\alpha}\varepsilon(\mathbf{u}_0)\right) \\
&= \left(\frac{\tau_{\varepsilon}}{\tau_{\sigma}}\right)^{\alpha} \frac{s^{\alpha-1}}{(\tau_{\sigma})^{-\alpha}+s^{\alpha}}\big(s\varepsilon(\hat{\mathbf{u}}) - \varepsilon(\mathbf{u}_0)\big) + \frac{1}{\tau_{\sigma}^{\alpha}} \frac{s^{-1}}{(\tau_{\sigma})^{-\alpha}+s^{\alpha}}\big(s\varepsilon(\hat{\mathbf{u}}) - \varepsilon(\mathbf{u}_0)\big) \\
&\quad + \frac{s^{\alpha-1}}{(\tau_{\sigma})^{-\alpha}+s^{\alpha}}\mathbb{D}^{-1}\sigma_0 + \frac{1}{\tau_{\sigma}^{\alpha}} \frac{s^{-1}}{(\tau_{\sigma})^{-\alpha}+s^{\alpha}}\varepsilon(\mathbf{u}_0).
\end{aligned}$$

Applying classical convolution theorem in Laplace transform [14, 39], Eqs. (2.3), (2.4) and the inverse-Laplace-transform, we finally get the explicit expression

$$\begin{aligned}
\mathbb{D}^{-1}\sigma = &\left(\left(\frac{\tau_{\varepsilon}}{\tau_{\sigma}}\right)^{\alpha} - 1\right) \int_0^t E_{\alpha}\left(-\left(\frac{t-\tau}{\tau_{\sigma}}\right)^{\alpha}\right) \varepsilon\big(\mathbf{u}_t(\tau)\big) d\tau + \varepsilon(\mathbf{u}_t) \\
&+ E_{\alpha}\left(-\left(\frac{t}{\tau_{\sigma}}\right)^{\alpha}\right) \left(\mathbb{D}^{-1}\sigma_0 - \varepsilon(\mathbf{u}_0)\right).
\end{aligned} \tag{2.6}$$

In what follows we shall give a weak problem of (1.1) based on the alternative constitutive relation (2.6). Let $L^2(\Omega)$ be the space of square integrable functions defined on Ω, and let $\underline{L}^2(\Omega)$ and $\underline{\underline{L}}^2(\Omega)$ be its vector and tensor analogues. We use $\langle\cdot,\cdot\rangle$ to denote the inner product on these three spaces. Define

$$\underline{\mathrm{H}}(\mathbf{div},\Omega,S) := \left\{\chi = (\chi_{ij})_{d\times d} \in \underline{\underline{L}}^2(\Omega) \mid \chi_{ij} = \chi_{ji},\ \mathbf{div}\,\chi \in \underline{L}^2(\Omega)\right\}.$$

In light of Eq. (2.6), we have the following weak formulation for (1.1): Find $\sigma \in \underline{\mathrm{H}}(\mathbf{div},\Omega,S)$ and $\mathbf{u} \in \underline{L}^2(\Omega)$ such that

$$\begin{cases} \langle \rho \mathbf{u}_{tt}, \mathbf{v}\rangle - \langle \mathbf{div}\,\sigma, \mathbf{v}\rangle = \langle \mathbf{f}, \mathbf{v}\rangle, & \forall\, \mathbf{v} \in \underline{L}^2(\Omega), \\ \langle \mathbb{D}^{-1}\sigma, \chi\rangle + \left(\left(\dfrac{\tau_\varepsilon}{\tau_\sigma}\right)^\alpha - 1\right) & \\ \quad \times \displaystyle\int_0^t E_\alpha\left(-\left(\frac{t-\tau}{\tau_\sigma}\right)^\alpha\right)\langle \mathbf{div}\,\chi, \mathbf{u}_t(\tau)\rangle d\tau + \langle \mathbf{div}\,\chi, \mathbf{u}_t\rangle & \\ = E_\alpha\left(-\left(\dfrac{t}{\tau_\sigma}\right)^\alpha\right)\left(\langle \mathbb{D}^{-1}\sigma_0, \chi\rangle + \langle \mathbf{div}\,\chi, \mathbf{u}_0\rangle\right), & \forall\, \chi \in \underline{\mathrm{H}}(\mathbf{div},\Omega,S), \\ \mathbf{u}(x,0) = \mathbf{u}_0, \quad \mathbf{u}_t(x,0) = \mathbf{v}_0. & \end{cases} \tag{2.7}$$

Remark 2.1. From the original model (1.1), we easily have the following weak formulation: Find $\sigma \in \underline{\mathrm{H}}(\mathbf{div},\Omega,S)$ and $\mathbf{u} \in \underline{L}^2(\Omega)$ such that

$$\begin{cases} \langle \rho \mathbf{u}_{tt}, \mathbf{v}\rangle - \langle \mathbf{div}\,\sigma, \mathbf{v}\rangle = \langle \mathbf{f}, \mathbf{v}\rangle, & \forall\, \mathbf{v} \in \underline{L}^2(\Omega), \\ \langle \mathbb{D}^{-1}\sigma, \chi\rangle + \tau_\sigma^\alpha \left\langle \dfrac{\partial^\alpha \mathbb{D}^{-1}\sigma}{\partial t^\alpha}, \chi \right\rangle & \\ \quad + \langle \mathbf{div}\chi, \mathbf{u}\rangle + \tau_\varepsilon^\alpha \left\langle \mathbf{div}\,\chi, \dfrac{\partial^\alpha \mathbf{u}}{\partial t^\alpha}\right\rangle = 0, & \forall\, \chi \in \underline{\mathrm{H}}(\mathbf{div},\Omega,S), \\ \mathbf{u}(x,0) = \mathbf{u}_0, \quad \mathbf{u}_t(x,0) = \mathbf{v}_0, \quad \sigma(x,0) = \sigma_0. & \end{cases} \tag{2.8}$$

2.2. Efficient SOE approximation of Mittag-Leffler function

Notice that there is a term $E_\alpha(-((t-\tau)/\tau_\sigma)^\alpha)$ involved in the weak formulation (2.7). As the Mittag-Leffler function is an infinite series, how to compute such a term efficiently is crucial to the design of fast algorithm for the fractional viscoelastic model (1.1). In this section we aim to construct an efficient sum-of-exponentials approximation of the Mittag-Leffler function $E_\alpha(-t^\alpha)$ based on the Gaussian quadrature rule.

2.2.1. Gaussian quadrature approximation

For a constant $l > 1$, let $g(z)$ be a function of one complex variable which is meromorphic in an open set containing the closure $\overline{B(l)}$ of the disc $B(l) = \{z \in \mathbb{C} : |z| < l\}$ and has only a finite number of simple poles p_m in $B(l)$. Consider the following Gaussian quadrature of $g(x)$ on interval $[-1,1] \subset (-l,l)$:

$$\int_{-1}^{1} g(x)dx = \sum_{j=1}^{J} \omega_j g(\xi_j) - \sum_{m} Y_J(p_m)\mathrm{Res}(g)_{p_m} + R_J(g). \tag{2.9}$$

Here ω_j and ξ_j denote respectively the Gaussian quadrature weights and nodes for $j = 1, 2, \ldots, J$, $\mathrm{Res}(g)_{p_m}$ is the residue of g at the pole p_m. It follows from [21] that

$$R_J(g) = \frac{1}{2\pi i} \int_{|z|=l} Y_J(z)g(z)dz,$$

where

$$Y_J(z) := \frac{1}{P_J} \int_{-1}^{1} \frac{P_J(x)}{z-x}dx, \quad z \in \mathbb{C}\backslash[-1,1]$$

and P_J is the Legendre orthogonal polynomial of degree J.

Remark 2.2. If $g(z)$ is analytic in $\overline{B(l)}$, the Gaussian quadrature (2.9) can be written as follows:

$$\int_{-1}^{1} g(x)dx = \sum_{j=1}^{J} \omega_j g(\xi_j) + R_J(g). \tag{2.10}$$

The following estimate of $R_J(g)$ is from [3].

Lemma 2.2. *There exists a positive integer J_* and a positive constant C, independent of l such that*

$$|R_J(g)| \leq C(l + \sqrt{l^2-1})^{-2J} \max_{|z|=l} |g(z)|, \quad \forall J > J_*. \tag{2.11}$$

2.2.2. SOE approximation of $E_\alpha(-t^\alpha)$

Applying the variable substitution $x = s^{-\alpha}$ in the integral (2.5), we get

$$E_\alpha(-t^\alpha) = \int_0^\infty f(x,t,\alpha)dx \tag{2.12}$$

with

$$f(x,t,\alpha) := \frac{\sin(\alpha\pi)}{\alpha\pi} \frac{e^{-tx^{-1/\alpha}}}{x^2 + 2x\cos(\alpha\pi) + 1}.$$

In order to derive an efficient SOE approximation for $E_\alpha(-t^\alpha)$, we truncate the integral (2.12) to a finite interval, then subdivide the finite interval into a set of non-overlapping intervals

$$(c_k - r_k, c_k + r_k), \quad k = 0, 1, \ldots, K. \tag{2.13}$$

Specifically, we decompose (2.12) into a summation form

$$E_\alpha(-t^\alpha) = \sum_{k=0}^{K} \int_{c_k-r_k}^{c_k+r_k} f(x,t,\alpha)dx + \int_{c_k+r_k}^{\infty} f(x,t,\alpha)dx \tag{2.14}$$

and apply the Gauss quadrature discussed in Section 2.2.1 for the integral over each subinterval $(c_k - r_k, c_k + r_k)$. Following a similar strategy as outlined in [3, 44], we determine the centers $c_0, c_1, \cdots, c_K$ and radii $r_0, r_1, \cdots, r_K$ for the intervals defined in Eq. (2.13) as follows: Let $q > 1$ be a constant,

$$c_0 = r_0 = \frac{1}{2}, \quad c_k = \frac{(q+1)q^{k-1}}{2}, \quad r_k = \frac{(q-1)q^{k-1}}{2}, \quad k = 1, 2, \ldots, K. \tag{2.15}$$

Remark 2.3. Note that the choice of an exponential function q^k in Eq. (2.15) leads to an exponential increase in subinterval length. This then allows, as shown in Lemma 2.3, as few subintervals $(c_k - r_k, c_k + r_k)$ in Eq. (2.14) as possible while keeping the required approximation accuracy of the finite sum. In addition, when applying the Gauss quadrature formula to each term $\int_{c_k-r_k}^{c_k+r_k} f(x,t,\alpha)dx$, as shown in Lemma 2.4, the remaining term can be controlled uniformly.

We now at a position to compute the integral term

$$\int_{c_k-r_k}^{c_k+r_k} f(x,t,\alpha)dx$$

for each k so as to get the desired SOE approximation. To this end, we apply the integration variable substitution $x = r_k y + c_k$ to obtain

$$\int_{c_k-r_k}^{c_k+r_k} f(x,t,\alpha)dx = \int_{-1}^{1} g_k(y,t)dy, \tag{2.16}$$

where

$$g_k(y,t) := \frac{\sin(\alpha\pi)}{\alpha\pi} \frac{r_k e^{-t(r_k y + c_k)^{-1/\alpha}}}{(r_k y + c_k)^2 + 2(r_k y + c_k)\cos(\alpha\pi) + 1}, \quad k = 0, 1, \ldots, K. \tag{2.17}$$

Notice that $0 < \alpha < 1$ and

$$z^2 + 2z\cos(\alpha\pi) + 1 = \big(z + \cos(\alpha\pi) + i\sin(\alpha\pi)\big)\big(z + \cos(\alpha\pi) - i\sin(\alpha\pi)\big),$$

then we easily know that for any k, $g_k(y,t)$ has two simple poles

$$\begin{aligned} \zeta_{k,1} &= -\frac{c_k}{r_k} + \frac{1}{r_k}\big(\cos((1-\alpha)\pi) + i\sin((1-\alpha)\pi)\big), \\ \zeta_{k,2} = \bar{\zeta}_{k,1} &= -\frac{c_k}{r_k} + \frac{1}{r_k}\big(\cos((1-\alpha)\pi) - i\sin((1-\alpha)\pi)\big). \end{aligned} \tag{2.18}$$

Denote

$$\begin{aligned} q_1 &:= \sqrt{5-4\cos((1-\alpha)\pi)}, \\ q_2 &:= \frac{1}{q-1}\sqrt{(q+1)^2-4(q+1)\cos((1-\alpha)\pi)+4}. \end{aligned}$$

For $0<\alpha<1$ we easily have

$$\begin{aligned} q_1 &> \sqrt{5-4}=1, \\ q_2 &> \frac{1}{q-1}\sqrt{(q+1)^2-4(q+1)+4}=1. \end{aligned}$$

Thus, it is reasonable to make the following assumption on l and q:

$$1<l<\min\left\{1+\frac{2}{q}, q_1, q_2\right\}. \tag{2.19}$$

Remark 2.4. From (2.15) we easily know that, for $k=0,1,\ldots,K$,

$$\begin{aligned} |\zeta_{k,1}| = |\zeta_{k,2}| &= \sqrt{\left(-\frac{q+1}{q-1}+\frac{1}{r_k}\cos((1-\alpha)\pi)\right)^2+\frac{1}{r_k^2}\sin^2(1-\alpha)\pi} \\ &= \sqrt{\frac{(q+1)^2}{(q-1)^2}-\frac{4(q+1)}{q^{k-1}(q-1)^2}\cos((1-\alpha)\pi)+\frac{4}{q^{2(k-1)}(q-1)^2}}. \end{aligned}$$

This relation, together with $0<\alpha<1$, $q>1$ and the assumption (2.19), further implies that

$$\begin{aligned} &|\zeta_{k,1}| = |\zeta_{k,2}| = \begin{cases} \sqrt{5-4\cos((1-\alpha)\pi)} = q_1 > l, & \text{if} \quad k=0, \\ \dfrac{1}{q-1}\sqrt{(q+1)^2-4(q+1)\cos((1-\alpha)\pi)+4} & \\ \qquad\qquad\qquad = q_2 > l, & \text{if} \quad k=1, \end{cases} \\ &|\zeta_{k,1}| = |\zeta_{k,2}| \geq \left|\frac{q+1}{q-1}-\frac{2}{q^{k-1}(q-1)}\right| \geq \frac{q+1}{q-1}-\frac{2}{q(q-1)} \\ &\qquad\qquad\qquad\qquad\qquad\qquad = 1+\frac{2}{q} > l, \quad \text{if} \quad k\geq 2. \end{aligned} \tag{2.20}$$

By (2.20) it is easy to see that $g_k(\cdot,t)$ has no poles in the disk $B(l)$ for any k. Thus, from the Gaussian quadrature formula (2.10) we have

$$\int_{-1}^{1} g_k(x,t)dx = \sum_{j=1}^{J} b_{kj}e^{-ta_{kj}} + R_J(g_k), \quad k=0,1,\ldots,K, \tag{2.21}$$

where, for $k=0,1,\ldots,K$ and $j=1,\ldots,J$,

$$\begin{aligned} a_{kj} &= (r_k\xi_j+c_k)^{-1/\alpha}, \\ b_{kj} &= \frac{\sin(\alpha\pi)}{\alpha\pi}\frac{\omega_j r_k}{(r_k\xi_j+c_k)^2+2(r_k\xi_j+c_k)\cos(\alpha\pi)+1}. \end{aligned} \tag{2.22}$$

We recall that ω_j and ξ_j are the Gaussian quadrature weights and nodes, respectively.

Substituting (2.21) and (2.16) into (2.14), we finally get the sum-of-exponentials approximation of the Mittag-Leffler function

$$\begin{aligned} E_\alpha(-t^\alpha) &= \sum_{k=0}^{K}\left(\sum_{j=1}^{J} b_{kj}e^{-ta_{kj}} + R_J(g_k)\right) + \int_{q^K}^{\infty} f(x,t,\alpha)dx \\ &= \sum_{k=0}^{K}\sum_{j=1}^{J} b_{kj}e^{-ta_{kj}} + R_{soe}(t), \end{aligned} \tag{2.23}$$

where

$$R_{soe}(t) := \sum_{k=0}^{K} R_J(g_k) + \int_{q^K}^{\infty} f(x,t,\alpha)dx. \tag{2.24}$$

In what follows we shall estimate the remaining term $R_{soe}(t)$. For the truncation integral term of (2.24), we easily obtain the following conclusion.

Lemma 2.3. *For* $0 < \alpha < 1, q > 1$ *and* $t > 0$*, there holds*

$$\left|\int_{q^K}^{\infty} f(x,t,\alpha)dx\right| \le \frac{1}{q^K - 1}.$$

Proof. Notice that

$$\int_{q^K}^{\infty} f(x,t,\alpha)dx = \frac{\sin(\alpha\pi)}{\alpha\pi}\int_{q^K}^{\infty} \frac{e^{-tx^{-1/\alpha}}}{x^2 + 2x\cos(\alpha\pi) + 1}dx.$$

For $0 < \alpha < 1$ and $t, x > 0$, we have

$$0 < e^{-tx^{-1/\alpha}} \le 1, \quad 0 < \frac{\sin(\alpha\pi)}{\alpha\pi} < 1,$$

and then

$$\begin{aligned} \left|\int_{q^K}^{\infty} f(x,t,\alpha)dx\right| &\le \left|\int_{q^K}^{\infty} \frac{1}{x^2 + 2x\cos(\alpha\pi) + 1}dx\right| \\ &\le \int_{q^K}^{\infty} \frac{1}{x^2 - 2x + 1}dx = \frac{1}{q^K - 1}. \end{aligned}$$

This finishes the proof. □

For the term $R_J(g_k)$ in (2.24), we have the following result.

Lemma 2.4. *For* $0 < t \le T$*,* $0 < \alpha < 1$*,* $q > 1$ *and* l *satisfying* (2.19)*, there holds*

$$|R_J(g_k)| \le C_{\alpha,T,q}(l + \sqrt{l^2 - 1})^{-2J}, \quad k = 0, 1, \ldots, K, \tag{2.25}$$

where

$$C_{\alpha,T,q} = \begin{cases} \dfrac{2qe^T}{(q-1)(q_1-l)^2}, & \textit{if} \quad k=0, \\ \dfrac{2qe^T}{(q-1)(q_2-l)^2}, & \textit{if} \quad k=1, \\ \dfrac{2qe^T}{(q-1)(1+q/2-l)^2}, & \textit{if} \quad k\geq 2. \end{cases}$$

Proof. In light of Eq. (2.11), for each $R_J(g_k)$ we only need to estimate the term $\max_{|z|=l}|g_k(z,t)|$. By (2.17) and (2.18) we have

$$\begin{aligned} \max_{|z|=l}|g_k(z,t)| &< \frac{\max_{|z|=l}|e^{-t(r_kz+c_k)^{-1/\alpha}}|}{r_k\min_{|z|=l}|z-\zeta_{k,1}|\,|z-\zeta_{k,2}|} \\ &< \frac{\max_{|z|=l}|e^{-tr_k^{-1/\alpha}(z+(q+1)/(q-1))^{-1/\alpha}}|}{r_k\min_{|z|=l}|z-\zeta_{k,1}|\,|z-\zeta_{k,2}|}. \end{aligned} \tag{2.26}$$

From (2.15) and (2.20) we easily know that

$$\frac{1}{r_k} \leq \max\left\{2, \frac{2}{q-1}\right\} < 2+\frac{2}{q-1} = \frac{2q}{q-1}, \quad k=0,1,2,\ldots$$

and

$$\min_{|z|=l}|z-\zeta_{k,1}|\,|z-\zeta_{k,2}| \geq \begin{cases} (q_1-l)^2, & \text{if} \quad k=0, \\ (q_2-l)^2, & \text{if} \quad k=1, \\ \left(1+\dfrac{q}{2}-l\right)^2, & \text{if} \quad k\geq 2. \end{cases}$$

To estimate the term $\max_{|z|=l}|e^{-tr_k^{-1/\alpha}(z+(q+1)/(q-1))^{-1/\alpha}}|$, we assume that

$$z = l(\cos\theta + i\sin\theta), \quad -\pi < \theta \leq \pi$$

and obtain

$$\begin{aligned} z+\frac{q+1}{q-1} &= l\cos\theta + \frac{q+1}{q-1} + il\sin\theta \\ &= \left[\left(l\cos\theta+\frac{q+1}{q-1}\right)^2 + l^2\sin^2\theta\right]^{1/2} (\cos\tilde{\theta}+i\sin\tilde{\theta}), \end{aligned}$$

where

$$\tilde{\theta} = \arctan\frac{l\sin\theta}{l\cos\theta+(q+1)/(q-1)}.$$

This means

$$\left(z+\frac{q+1}{q-1}\right)^{-1/\alpha} = \left[\left(l\cos\theta+\frac{q+1}{q-1}\right)^2 + l^2\sin^2\theta\right]^{-1/(2\alpha)}$$

$$\times\left(\cos\left(-\frac{\tilde{\theta}}{\alpha}\right)+i\sin\left(-\frac{\tilde{\theta}}{\alpha}\right)\right)$$

and

$$\begin{aligned}-\mathrm{Re}\left(z+\frac{q+1}{q-1}\right)^{-1/\alpha}&=-\left[\left(l\cos\theta+\frac{q+1}{q-1}\right)^2+l^2\sin^2\theta\right]^{-1/(2\alpha)}\cos\left(-\frac{\tilde{\theta}}{\alpha}\right)\\&\le\left[\left(l\cos\theta+\frac{q+1}{q-1}\right)^2+l^2\sin^2\theta\right]^{-1/(2\alpha)}\le\left(l+\frac{q+1}{q-1}\right)^{-1/\alpha}.\end{aligned}$$

Thus, we have

$$\begin{aligned}&\max_{|z|=l}\left|e^{-tr_k^{-1/\alpha}(z+(q+1)/(q-1))^{-1/\alpha}}\right|\\=&\max_{|z|=l}\left|e^{-tr_k^{-1/\alpha}\mathrm{Re}(z+c_k/r_k)^{-1/\alpha}}\right|\\\le\ &e^{T(2q/(q-1))^{1/\alpha}((q+1)/(q-1)+l)^{-1/\alpha}}\\\le\ &e^{T(2q/(q+1+l(q-1)))^{1/\alpha}}\le e^T.\end{aligned}\tag{2.27}$$

Finally, combining Eq. (2.11) and the inequalities (2.26)-(2.27) gives the desired estimate (2.25). □

Recalling (2.22), we give a compact form of the SOE approximation (2.23) as follows:

$$E_\alpha(-t^\alpha)=\sum_{j=1}^{(K+1)J}b_je^{-a_jt}+R_{soe}(t),\tag{2.28}$$

where a_j and b_j are the j-th elements of

$$\begin{aligned}&[a_{01},a_{02},\ldots,a_{0J},a_{11},a_{12},\ldots,a_{1J},\ldots,a_{(K+1)1},\ldots,a_{(K+1)(J-1)},a_{(K+1)J}],\\&[b_{01},b_{02},\ldots,b_{0J},b_{11},b_{12},\ldots,b_{1J},\ldots,b_{(K+1)1},\ldots,b_{(K+1)(J-1)},b_{(K+1)J}],\end{aligned}$$

respectively.

Denote $N_{\exp}=(K+1)J$, we are now at a position to estimate the SOE approximation error

$$R_{soe}(t)=E_\alpha(-t^\alpha)-\sum_{j=1}^{N_{\exp}}b_je^{-a_jt},\quad 0<t\le T.\tag{2.29}$$

In light of Lemmas 2.3 and 2.4 and the relation (2.24), we immediately get the following main conclusion.

Theorem 2.1. *For $0<\alpha<1$, $q>1$ and $1<l<\min\{1+2/q,q_1,q_2\}$, there holds*

$$|R_{soe}(t)|\le C_{\alpha,T,q}(K+1)(l+\sqrt{l^2-1})^{-2J}+\frac{1}{q^K-1},\quad 0<t\le T.\tag{2.30}$$

Moreover, for any $0 < \epsilon < 1$ *there holds*

$$|R_{soe}(t)| = \mathcal{O}(\epsilon), \quad 0 < t \le T, \tag{2.31}$$

provided that

$$K = \mathcal{O}\big(|\log\epsilon|\big), \quad J = \mathcal{O}\big(|\log(\epsilon^{-1}|\log\epsilon|)|\big). \tag{2.32}$$

Remark 2.5. Theorem 2.1 means that for a given tolerance error ϵ, the computation complexity of the SOE approximation (2.28) is

$$N_{\text{exp}} = (K+1)J = \mathcal{O}\big(|\log\epsilon|^2\big).$$

Furthermore, denote $N := T/\triangle t$ with $\triangle t < 1$ being the temporal step size, then for $\epsilon = \triangle t$ we have

$$N_{\text{exp}} = \mathcal{O}\big(\log^2 N\big).$$

Remark 2.6. In view of (2.30) and (2.32), we shall select

$$K = \left\lceil \frac{|\log\varepsilon|}{\log q} \right\rceil, \quad J = \left\lceil \frac{\log\left(\varepsilon^{-1}|\log\varepsilon|\right)}{2\log q \log l} \right\rceil \tag{2.33}$$

in the numerical implementation (cf. Section 4), where $\lceil\cdot\rceil$ denotes the ceiling function, which rounds up to the nearest integer.

3. Numerical schemes for the fractional viscoelastic model

In this section, we present two fully discrete mixed finite element schemes for the fractional viscoelastic model (1.1). One is based on the weak form (2.8) and applies the traditional L1 scheme and the Newmark scheme to discretize the time-fractional derivative and the second time derivative, respectively. The other one is based on the weak form (2.7) and adopts the SOE approximation for the Mittag-Leffler function.

Let $\underline{\underline{\mathrm{H}}}_h \subset \underline{\underline{\mathrm{H}}}(\mathbf{div}, \Omega, S)$ and $\underline{\mathrm{V}}_h \subset \underline{L}^2(\Omega)$ be two finite-dimensional spaces for stress and displacement approximations, respectively. For any positive integer N, let

$$\big\{t_n: \ t_n = n\triangle t, \ 0 \le n \le N\big\}$$

be a uniform partition of the time interval $(0, T]$ with the time step size $\triangle t = T/N$.

3.1. L1-Newmark mixed finite element scheme

In view of the weak form (2.8), the generic semi-discrete mixed conforming finite element scheme for the fractional viscoelastic model (1.1) reads: Find $\sigma_h(t) \in \underline{\underline{\mathrm{H}}}_h$ and

$\mathbf{u}_h(t) \in \underline{\mathrm{V}_h}$ such that

$$\begin{cases} \langle \rho \mathbf{u}_{h,tt}, \mathbf{v}_h \rangle = \langle \mathbf{div}\, \sigma_h, \mathbf{v}_h \rangle + \langle \mathbf{f}, \mathbf{v}_h \rangle, & \forall\, \mathbf{v}_h \in \underline{\mathrm{V}_h}, \\ \langle \mathbb{D}^{-1}\sigma_h, \chi_h \rangle + \tau_\sigma^\alpha \left\langle \dfrac{\partial^\alpha \mathbb{D}^{-1}\sigma_h}{\partial t^\alpha}, \chi_h \right\rangle & \\ \quad + \langle \mathbf{div}\, \chi_h, \mathbf{u}_h \rangle + \tau_\varepsilon^\alpha \left\langle \mathbf{div}\, \chi_h, \dfrac{\partial^\alpha \mathbf{u}_h}{\partial t^\alpha} \right\rangle = 0, & \forall\, \chi_h \in \underline{\underline{\mathrm{H}_h}}, \\ \mathbf{u}_h(0) = I_{\mathbb{V}_h}\mathbf{u}_0, \quad \mathbf{u}_{h,t}(0) = I_{\mathbb{V}_h}\mathbf{v}_0, \quad \sigma_h(0) = I_{\mathbb{H}_h}\sigma_0, & \end{cases} \tag{3.1}$$

where $I_{\mathbb{V}_h}$ and $I_{\mathbb{H}_h}$ denote the projection operators onto $\underline{\mathrm{V}_h}$ and $\underline{\underline{\mathrm{H}_h}}$, respectively.

Let $\{\varphi_i\}_{i=1}^r$ and $\{\kappa_i\}_{i=1}^s$ be bases of $\underline{\underline{\mathrm{H}_h}}$ and $\underline{\mathrm{V}_h}$, respectively, and introduce matrices $\mathbf{A} = (\mathbf{A}_{ij})_{r\times r}$, $\mathbf{B} = (\mathbf{B}_{ij})_{r\times s}$, $\mathbf{C} = (\mathbf{C}_{ij})_{s\times s}$ with

$$\mathbf{A}_{ij} = \langle \mathbb{D}^{-1}\varphi_i, \varphi_j \rangle, \quad \mathbf{B}_{ij} = \langle \mathbf{div}\, \varphi_i, \kappa_j \rangle, \quad \mathbf{C}_{ij} = \langle \rho\kappa_i, \kappa_j \rangle.$$

We write $\sigma_h = \sum_{i=1}^r \beta_i(t)\varphi_i$, $\mathbf{u}_h = \sum_{j=1}^s U_j(t)\kappa_j$, $\eta_j = \langle \mathbf{f}(t), \kappa_j \rangle$, and denote

$$\beta(t) := (\beta_1, \beta_2, \cdots, \beta_r)^{\mathrm{T}}, \quad U(t) := (U_1, U_2, \cdots, U_s)^{\mathrm{T}}, \quad \eta(t) := (\eta_1, \eta_2, \cdots, \eta_s)^{\mathrm{T}}.$$

Then we can rewrite (3.1) in the matrix form

$$\begin{cases} \mathbf{C}U_{tt} - \mathbf{B}^{\mathrm{T}}\beta = \eta, \\ \mathbf{A}\beta + \tau_\sigma^\alpha \mathbf{A}\dfrac{\partial^\alpha \beta}{\partial t^\alpha} + \mathbf{B}U + \tau_\varepsilon^\alpha \mathbf{B}\dfrac{\partial^\alpha U}{\partial t^\alpha} = 0 \end{cases} \tag{3.2}$$

with the initial data $U(0) = I_{\mathbb{V}_h}\mathbf{u}_0$, $U_t(0) = I_{\mathbb{V}_h}\mathbf{v}_0$, $\beta(0) = I_{\mathbb{H}_h}\sigma_0$.

To discretize the term U_{tt} in (3.2), we choose the Newmark scheme [38] as follows:

$$\begin{cases} U_{tt}(t_n) = \dfrac{1}{\triangle t^2\theta_2}\Big(U(t_n) - U(t_{n-1}) - \triangle t U_t(t_{n-1}) - \dfrac{\triangle t^2}{2}(1-2\theta_2)U_{tt}(t_{n-1})\Big), \\ U_t(t_n) = U_t(t_{n-1}) + \triangle t\left[(1-\theta_1)U_{tt}(t_{n-1}) + \theta_1 U_{tt}(t_n)\right], \end{cases} \tag{3.3}$$

where the choice of parameters (θ_1, θ_2) depends on the requirement of accuracy and stability for the scheme (cf. Remark 3.1). In our numerical experiments in next section we choose $\theta_1 = 1/2$ and $\theta_2 = 1/4$.

Remark 3.1. We list four well-known members of the Newmark method [8, 38] in Table 1.

Table 1: Members of the Newmark method.

Method	θ_1	θ_2	Accuracy
Newmark explicit method	1/2	0	second order
Fox-Goodwin method	1/2	1/12	third order
Linear average acceleration method	1/2	1/6	second order
Constant average acceleration method	1/2	1/4	second order

We note that the constant average acceleration Newmark method ($\theta_1 = 1/2$, $\theta_2 = 1/4$) is second-order accurate and unconditionally stable.

For the discretization of Caputo fractional derivative $\partial^\alpha U/\partial t^\alpha$, the following L1 scheme is commonly used:

$$\frac{\partial^\alpha U}{\partial t^\alpha}(t_n) = \frac{\triangle t^{-\alpha}}{\Gamma(2-\alpha)}\left[a_0^\alpha U(t_n) - \sum_{k=1}^{n-1}\left(a_{n-k-1}^\alpha - a_{n-k}^\alpha\right)U(t_k) - a_{n-1}^\alpha U(0)\right], \tag{3.4}$$

where $a_k^\alpha = (k+1)^{1-\alpha} - k^{1-\alpha}$ for $k = 0, 1, \ldots, n-1$.

Substituting the L1 scheme (3.4) and the Newmark scheme (3.3) into (3.2) leads to the following fully discrete linear system:

$$\begin{aligned}&\left(\frac{\mathbf{C}}{\triangle t^2\theta_2} + \frac{1+L_\varepsilon}{1+L_\sigma}\mathbf{B}^{\mathrm{T}}\mathbf{A}^{-1}\mathbf{B}\right)U(t_n)\\ &= \eta(t_n) + \frac{L_\sigma}{1+L_\sigma}\mathbf{B}^{\mathrm{T}}K_{\sigma,n-1} + \frac{L_\varepsilon}{1+L_\sigma}\mathbf{B}^{\mathrm{T}}\mathbf{A}^{-1}\mathbf{B}K_{u,n-1}\\ &\quad + \frac{\mathbf{C}}{\triangle t^2\theta_2}\left(U(t_{n-1}) + \triangle t U_t(t_{n-1}) + \frac{\triangle t^2}{2}(1-2\theta_2)U_{tt}(t_{n-1})\right), \quad n = 1, 2, \ldots, N,\end{aligned}$$

where

$$\begin{aligned}L_\sigma &:= \frac{\tau_\sigma^\alpha}{\triangle t^\alpha\Gamma(2-\alpha)}, \quad K_{\sigma,n-1} := \sum_{k=1}^{n-1}\left(a_{n-k-1}^\alpha - a_{n-k}^\alpha\right)\beta(t_k) + a_{n-1}^\alpha\beta(0),\\ L_\varepsilon &:= \frac{\tau_\varepsilon^\alpha}{\triangle t^\alpha\Gamma(2-\alpha)}, \quad K_{u,n-1} := \sum_{k=1}^{n-1}\left(a_{n-k-1}^\alpha - a_{n-k}^\alpha\right)U(t_k) + a_{n-1}^\alpha U(0).\end{aligned}$$

Define

$$H_{n-1} := \frac{L_\varepsilon}{1+L_\sigma}\mathbf{B}^{\mathrm{T}}\mathbf{A}^{-1}\mathbf{B}K_{u,n-1} + \frac{L_\sigma}{1+L_\sigma}\mathbf{B}^{\mathrm{T}}K_{\sigma,n-1},$$

we easily have the following recurrence relation:

$$\begin{aligned}H_{n-1} &= \frac{L_\varepsilon}{1+L_\sigma}\mathbf{B}^{\mathrm{T}}\mathbf{A}^{-1}\mathbf{B}K_{u,n-1} + \frac{L_\sigma}{1+L_\sigma}\mathbf{B}^{\mathrm{T}}\left[\sum_{k=1}^{n-1}\left(a_{n-k-1}^\alpha - a_{n-k}^\alpha\right)\beta(t_k) + a_{n-1}^\alpha\beta(0)\right]\\ &= \frac{L_\varepsilon}{1+L_\sigma}\mathbf{B}^{\mathrm{T}}\mathbf{A}^{-1}\mathbf{B}K_{u,n-1} + \frac{L_\sigma}{1+L_\sigma}\sum_{k=1}^{n-1}\left(a_{n-k-1}^\alpha - a_{n-k}^\alpha\right)H_{k-1}\\ &\quad - \frac{L_\sigma(1+L_\varepsilon)}{(1+L_\sigma)^2}\mathbf{B}^{\mathrm{T}}\mathbf{A}^{-1}\mathbf{B}\left[\sum_{k=1}^{n-1}\left(a_{n-k-1}^\alpha - a_{n-k}^\alpha\right)U(t_k) + a_{n-1}^\alpha U(0)\right]\\ &= \frac{L_\varepsilon - L_\sigma}{(1+L_\sigma)^2}\mathbf{B}^{\mathrm{T}}\mathbf{A}^{-1}\mathbf{B}K_{u,n-1} + \frac{L_\sigma}{1+L_\sigma}\sum_{k=1}^{n-1}\left(a_{n-k-1}^\alpha - a_{n-k}^\alpha\right)H_{k-1}.\end{aligned} \tag{3.5}$$

In conclusion, we have the following L1-Newmark mixed finite element algorithm.

Algorithm 3.1 L1-Newmark MFE Scheme

```
Require: $U(0), U_t(0), \eta(0), H_0, U_{tt}(0) = \mathbf{C}^{-1}(\eta(0) - (1+L_\varepsilon)/(1+L_\sigma)\mathbf{B}^{\mathrm{T}}\mathbf{A}^{-1}\mathbf{B}U(0))$.
Ensure: $U(t_N)$
for $n \leftarrow 1, N$ do
    Solve $U(t_n)$ with the scheme
        $\left(\frac{\mathbf{C}}{\triangle t^2\theta_2} + \frac{1+L_\varepsilon}{1+L_\sigma}\mathbf{B}^{\mathrm{T}}\mathbf{A}^{-1}\mathbf{B}\right)U(t_n) = \eta(t_n) + H_{n-1} + \frac{\mathbf{C}}{\triangle t^2\theta_2}\left(U(t_{n-1}) + \triangle t U_t(t_{n-1}) + \frac{\triangle t^2}{2}(1-2\theta_2)U_{tt}(t_{n-1})\right).$
    Calculate and store history variable $H_n$.
    Compute $U_t(t_n)$ and $U_{tt}(t_n)$ through Eq. (3.3).
end for
Return $U(t_N)$.
```

Note that at each time step we need to calculate and store the history variable H_n. This means that Algorithm 3.1 requires $\mathcal{O}(N_sN)$ memory complexity and $\mathcal{O}(N_sN^2)$ computation complexity. Here we simply denote by $\mathcal{O}(N_s)$ the complexities of memory and computation related to the spatial discretization. As N is large, the complexities of memory and computation of Algorithm 3.1 may create obstacles for a long time simulation. Therefore, in the following subsection we shall provide a fast numerical scheme based on the weak form (2.7).

3.2. Fast numerical scheme with SOE approximation

In view of the weak form (2.7), we have the following semi-discrete mixed conforming finite element scheme for the fractional viscoelastic model (1.1): Find $\sigma_h(t) \in \underline{\underline{\mathrm{H}_h}}$ and $\mathbf{u}_h(t) \in \underline{\mathrm{V}_h}$ such that

$$\begin{cases} \langle \rho\mathbf{u}_{h,tt}, \mathbf{v}_h\rangle = \langle \mathbf{div}\,\sigma_h, \mathbf{v}_h\rangle + \langle \mathbf{f}, \mathbf{v}_h\rangle, & \forall\, \mathbf{v}_h \in \underline{\mathrm{V}_h}, \\ \left\langle \mathbb{D}^{-1}\sigma_h, \chi_h\right\rangle + \left(\left(\dfrac{\tau_\varepsilon}{\tau_\sigma}\right)^\alpha - 1\right) & \\ \qquad \times \displaystyle\int_0^t E_\alpha\left(-\left(\frac{t-\tau}{\tau_\sigma}\right)^\alpha\right)\langle \mathbf{div}\,\chi_h, \mathbf{u}_{h,t}(\tau)\rangle d\tau + \langle \mathbf{div}\,\chi_h, \mathbf{u}_{h,t}\rangle & \\ = E_\alpha\left(-\left(\dfrac{t}{\tau_\sigma}\right)^\alpha\right)\left\langle \mathbb{D}^{-1}\sigma_0, \chi_h\right\rangle + E_\alpha\left(-\left(\dfrac{t}{\tau_\sigma}\right)^\alpha\right)\langle \mathbf{div}\,\chi_h, \mathbf{u}_0\rangle, & \forall\, \chi_h \in \underline{\underline{\mathrm{H}_h}}, \\ \mathbf{u}_h(0) = I_{\mathbb{V}_h}\mathbf{u}_0, \quad \mathbf{u}_{h,t}(0) = I_{\mathbb{V}_h}\mathbf{v}_0. & \end{cases}$$

Using the same notations as in Section 3.1, we rewrite this system as the following matrix form:

$$\begin{cases} \mathbf{C}U_{tt} - \mathbf{B}^{\mathrm{T}}\beta = \eta, \\ \mathbf{A}\beta + \left(\left(\dfrac{\tau_\varepsilon}{\tau_\sigma}\right)^{\alpha} - 1\right)\mathbf{B}\displaystyle\int_0^t E_\alpha\left(-\left(\frac{t-\tau}{\tau_\sigma}\right)^{\alpha}\right)U_t(\tau)d\tau + \mathbf{B}U_t = \iota, \end{cases} \tag{3.6}$$

where

$$\iota = \big(\mathbf{A}\beta(0) + \mathbf{B}U(0)\big)E_\alpha\left(-\left(\frac{t}{\tau_\sigma}\right)^{\alpha}\right)$$

with the initial data $U(0) = I_{\mathbb{V}_h}\mathbf{u}_0$, $U_t(0) = I_{\mathbb{V}_h}\mathbf{v}_0$, $\beta(0) = I_{\mathbb{H}_h}\sigma_0$.

According to the SOE approximation (2.31) in Theorem 2.1, we have

$$E_\alpha\left(-\left(\frac{t-\tau}{\tau_\sigma}\right)^{\alpha}\right) = \sum_{j=1}^{N_{\rm exp}} b_j e^{-a_j((t-\tau)/\tau_\sigma)} + \mathcal{O}(\epsilon),$$

which, together with integration by parts, gives

$$\begin{aligned} &\int_0^t E_\alpha\left(-\left(\frac{t-\tau}{\tau_\sigma}\right)^{\alpha}\right)U_t(\tau)d\tau \\ &= \sum_{j=1}^{N_{\rm exp}} b_j\left(U(t) - U(0)e^{-a_j t/\tau_\sigma} - \frac{a_j}{\tau_\sigma}\int_0^t e^{-a_j((t-\tau)/\tau_\sigma)}U(\tau)d\tau\right) + \mathcal{O}(\epsilon). \end{aligned} \tag{3.7}$$

Introduce the auxiliary variable

$$G_j(t) := \int_0^t e^{-a_j((t-\tau)/\tau_\sigma)}U(\tau)d\tau,$$

and we have the following simple recurrence relation at $t = t_n$:

$$\begin{aligned} G_j(t_n) &= e^{-a_j/\tau_\sigma\triangle t}G_j(t_{n-1}) + \int_{t_{n-1}}^{t_n} e^{-a_j((t_n-\tau)/\tau_\sigma)}U(\tau)d\tau \\ &\approx e^{-a_j/\tau_\sigma\triangle t}G_j(t_{n-1}) + T_{1,j}U(t_{n-1}) + T_{2,j}U_t(t_{n-1}) + T_{3,j}U_{tt}(t_{n-1}), \end{aligned} \tag{3.8}$$

where

$$\begin{aligned} &T_{1,j} = \frac{\tau_\sigma}{a_j}\left(1 - e^{-a_j/\tau_\sigma\triangle t}\right), \quad T_{2,j} = \frac{\tau_\sigma}{a_j}\triangle t e^{-a_j/\tau_\sigma\triangle t} + \left(\triangle t - \frac{\tau_\sigma}{a_j}\right)T_{1,j}, \\ &T_{3,j} = \left(\frac{\triangle t^2}{2} - \triangle t\frac{\tau_\sigma}{a_j} + \frac{\tau_\sigma^2}{a_j^2}\right)T_{1,j} + \frac{\tau_\sigma}{a_j}\triangle t e^{-a_j/\tau_\sigma\triangle t}\left(\frac{\triangle t}{2} - \frac{\tau_\sigma}{a_j}\right). \end{aligned}$$

Finally, we apply the Newmark scheme (3.3) to the semi-discrete scheme (3.6) and use (3.8) and (3.7) to obtain the linear system

$$\begin{aligned}&\left[\frac{\mathbf{C}+\theta_1\triangle t\mathbf{B}^{\mathrm{T}}\mathbf{A}^{-1}\mathbf{B}}{\triangle t^2\theta_2}+\left(\left(\frac{\tau_\varepsilon}{\tau_\sigma}\right)^\alpha-1\right)\mathbf{B}^{\mathrm{T}}\mathbf{A}^{-1}\mathbf{B}\right]U(t_n)\\&=\eta(t_n)+E(t_n)+Q_1U(t_{n-1})+Q_2U_t(t_{n-1})+Q_3U_{tt}(t_{n-1})\\&\quad+\left(\left(\frac{\tau_\varepsilon}{\tau_\sigma}\right)^\alpha-1\right)\mathbf{B}^{\mathrm{T}}\mathbf{A}^{-1}\mathbf{B}\sum_{j=1}^{N_{\exp}}\frac{a_jb_j}{\tau_\sigma}G_j(t_n),\end{aligned}\tag{3.9}$$

where

$$\begin{aligned}E(t_n)&=\left(\sum_{j=1}^{N_{\exp}}b_je^{-a_jt_n/\tau_\sigma}\right)\left(\mathbf{B}^{\mathrm{T}}\beta(0)+\left(\frac{\tau_\varepsilon}{\tau_\sigma}\right)^\alpha\mathbf{B}^{\mathrm{T}}\mathbf{A}^{-1}\mathbf{B}U(0)\right),\\Q_1&=\frac{\mathbf{C}+\theta_1\triangle t\mathbf{B}^{\mathrm{T}}\mathbf{A}^{-1}\mathbf{B}}{\triangle t^2\theta_2},\quad Q_2=\frac{\mathbf{C}+(\theta_1-\theta_2)\triangle t\mathbf{B}^{\mathrm{T}}\mathbf{A}^{-1}\mathbf{B}}{\triangle t\theta_2},\\Q_3&=\frac{(1-2\theta_2)\mathbf{C}+\triangle t(\theta_1-2\theta_2)\mathbf{B}^{\mathrm{T}}\mathbf{A}^{-1}\mathbf{B}}{2\theta_2},\end{aligned}$$

and the history variable $G_j(t_n)$ is computed by using the approximation formula (3.8), i.e.

$$G_j(t_n)=e^{-a_j/\tau_\sigma\triangle t}G_j(t_{n-1})+T_{1,j}U(t_{n-1})+T_{2,j}U_t(t_{n-1})+T_{3,j}U_{tt}(t_{n-1}).\tag{3.10}$$

In particular, $G_j(0)=0$. The resulting fast algorithm, i.e. Algorithm 3.2, is given as follows:

Algorithm 3.2 Fast Scheme

```
Require: $U(0), U_t(0), G_j(0)=0, U_{tt}(0)=\mathbf{C}^{-1}(\eta(0)+\mathbf{B}^{\mathrm{T}}\beta(0))$.
Ensure: $U(t_N)$.
Calculate $N_{\exp}, a_j, b_j, T_{i,j}, Q_i, j=1,\ldots,N_{\exp},\ i=1,2,3$.
for $n \leftarrow 1, N$ do
    Calculate by (3.10) and store the history variable $G_j(t_n),\ j=1,\ldots,N_{\exp}$.
    Solve $U(t_n)$ with the scheme (3.9).
    Get $U_{tt}(t_n)$ and $U_t(t_n)$ through Eq. (3.3).
end for
Return $U(t_N)$.
```

Comparing with the L1-Newmark Algorithm 3.1, we easily see that, due to $N >> N_{\exp}$ (cf. Remark 2.5), Algorithm 3.2 reduces the costs of memory and computation from $\mathcal{O}(N_sN)$ and $\mathcal{O}(N_sN^2)$ to $\mathcal{O}(N_{\exp}N_s)$ and to $\mathcal{O}(N_sN_{\exp}N)$, respectively.

4. Numerical results

In this section, we provide some numerical results to verify the efficiency of both the SOE approximation (2.28) (or (2.23)) and the fast scheme (3.2). All the algorithms are implemented by using MATLAB 2023a and executed on a PC equipped with a 3.40GHz processor, 32GB of RAM, and running Windows 10.

Example 4.1 (Test of SOE Approximation Accuracy). In this example, we evaluate the SOE approximation (2.28) for the Mittag-Leffler function $E_\alpha(-t^\alpha)$ under two distinct scenarios:

1) Varying the parameters l and q while keeping the fractional order α and the tolerance error ε fixed.

2) Varying the tolerance error ε while keeping α, l, and q fixed.

According to Theorem 2.1, the parameters q and l are required to satisfy

$$q > 1, \quad 1 < l < \min\left\{1 + \frac{2}{q}, q_1, q_2\right\} \tag{4.1}$$

with

$$\begin{aligned} q_1 &= \sqrt{5 - 4\cos((1-\alpha)\pi)}, \\ q_2 &= \frac{1}{q-1}\sqrt{(q+1)^2 - 4(q+1)\cos((1-\alpha)\pi) + 4}. \end{aligned}$$

The values of $q_3 := \min\{1 + 2/q, q_1, q_2\}$ with $\alpha = 0.2, 0.5, 0.7$ and $q = 2, 8, 9, 10, 11$ are listed in Table 2, based on which we compute the following cases: $(q = 2, l = 1.5)$, $(q = 8, l = 1.1)$, $(q = 9, l = 1.1)$, $(q = 10, l = 1.1)$, $(q = 11, l = 1.09)$.

As mentioned in Remark 2.6, for given ε, l and q the number $N_{\text{exp}} = (K+1)J$ of the SOE approximation is determined by (2.33), i.e.

$$K = \left\lceil \frac{|\log\varepsilon|}{\log q} \right\rceil, \quad J = \left\lceil \frac{\log\left(\varepsilon^{-1}|\log\varepsilon|\right)}{2\log q \log l} \right\rceil.$$

Table 3 lists the results of N_{exp} in different cases. It is noteworthy that, under the same level of tolerance error, the case with $(q = 10, \ l = 1.1)$ yields the smallest N_{exp}.

Table 2: The values of q_3 with different q and α: $1 < l < q_3$.

q	2	8	9	10	11
$\alpha = 0.2$	2	1.25	1.2222	1.2	1.1818
$\alpha = 0.5$	2	1.25	1.2222	1.2	1.1818
$\alpha = 0.7$	1.6275	1.1414	1.1214	1.1063	1.0945

Table 3: Values of $N_{\exp}=(K+1)J$ for different levels of tolerance error ε and different choices of q,l.

q,l	$\varepsilon=10^{-2}$	$\varepsilon=10^{-3}$	$\varepsilon=10^{-4}$
$q=2,\quad l=1.5$	$88\ (K=7,\ J=11)$	$176\ (K=16,\ J=16)$	$315\ (K=14,\ J=21)$
$q=8,\quad l=1.1$	$64\ (K=3,\ J=16)$	$115\ (K=4,\ J=23)$	$174\ (K=5,\ J=29)$
$q=9,\quad l=1.1$	$60\ (K=3,\ J=15)$	$110\ (K=4,\ J=22)$	$168\ (K=5,\ J=28)$
* $q=10,\quad l=1.1$	$42\ (K=2,\ J=14)$	$84\ (K=3,\ J=21)$	$135\ (K=4,\ J=27)$
$q=11,\quad l=1.09$	$45\ (K=2,\ J=15)$	$88\ (K=3,\ J=22)$	$140\ (K=4,\ J=28)$

Numerical results of the SOE approximation error $|R_{soe}(t)|$ in different cases are demonstrated in Fig. 1. Note that by (2.29) $R_{soe}(t)$ is of the form

$$R_{soe}(t)=E_\alpha(-t^\alpha)-\sum_{j=1}^{N_{\exp}}b_je^{-a_jt},$$

and in our actual computation the term $E_\alpha(-t^\alpha)$ is quantified by using the optimal parabolic contour algorithm [20].

From Fig. 1 we have the following observations:

- Figs. 1(a), 1(c), 1(e), 1(g), 1(h), 1(i) plotted the results of $R_{soe}(t)$ against t. We can see that for fixed α and tolerance error ε, the obtained SOE approximation with different choices of q and l satisfying (4.1) is of the accuracy $R_{soe}(t)=\mathcal{O}(\varepsilon)$. This is conformable to the theoretical prediction (2.31) in Theorem 2.1.

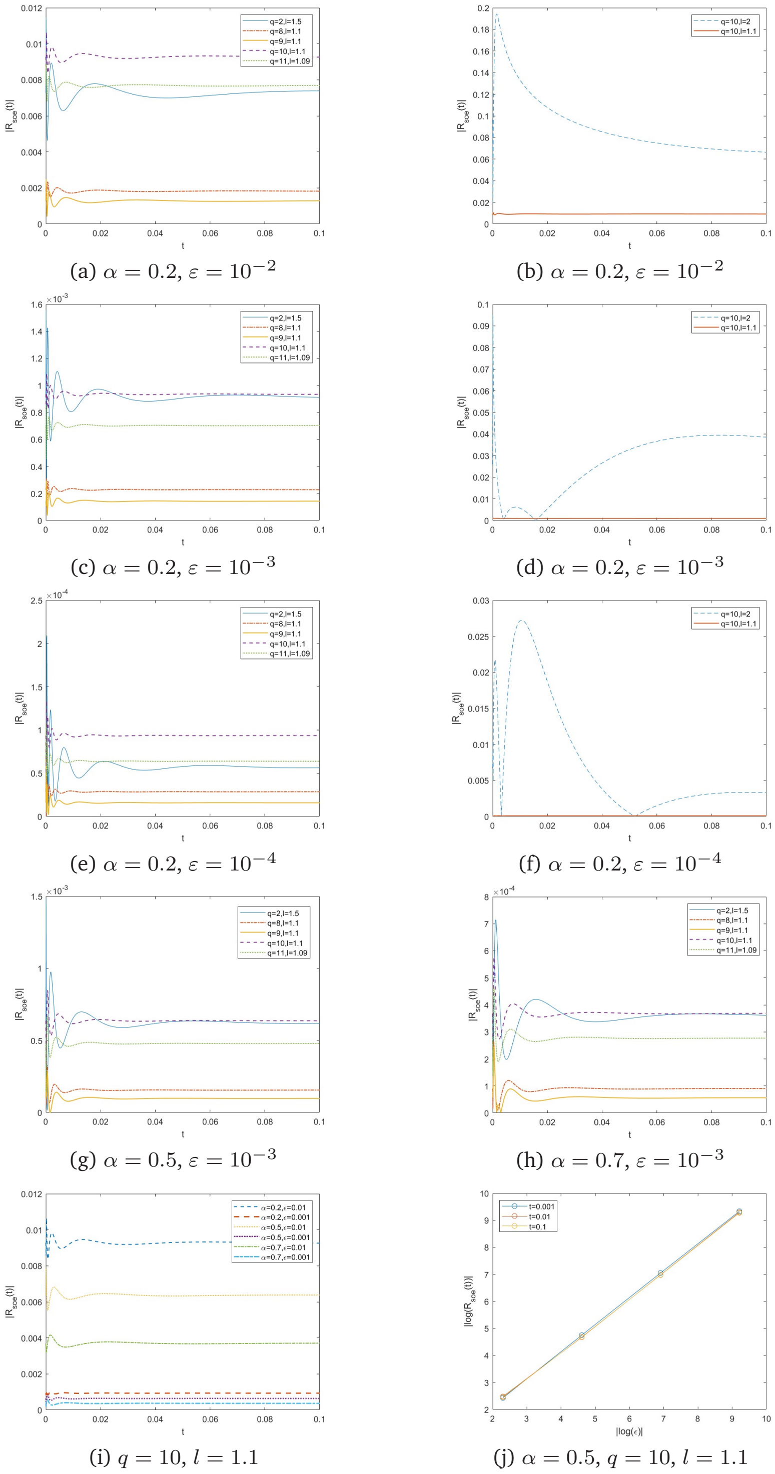
- In particular, the case with $(q=10,\ l=1.1)$ has the smallest $N_{\exp}$ among all the cases (cf. Table 3). As far as the complexity is concerned, this is the best choice of q and l in comparison.

- Figs. 1(b), 1(d), 1(f) also give results of $R_{soe}(t)$ in the case $(q=10,\ l=2)$ not satisfying the condition (4.1). We can see that the approximation accuracy in this case is not as good as that in other cases.

- Fig. 1(i) shows results of $R_{soe}(t)$ at different α and ε. In the relatively best case $(q=10,\ l=1.1)$. We can see that for each α, the smaller the tolerance error ε becomes, the more accurate the SOE approximation will be.

- Fig. 1(j) demonstrates that the logarithmic error of the SOE approximation is proportional to $|\log(\varepsilon)|$ when α is fixed. We can also observe that $R_{soe}(t)$ decreases over t. This indicates that the SOE approximation is particularly suitable for long-time simulations in fractional viscoelastic models.

Example 4.2 (Efficiency Test of Fast Scheme). In the model problem (1.1), we take $\Omega=[0,1]\times[0,1]$, $T=1$, $\alpha=0.5$, $\tau_\sigma=1$, and $\tau_\varepsilon=1$. The elastic medium is assumed



(a) $\alpha = 0.2$, $\varepsilon = 10^{-2}$

(b) $\alpha = 0.2$, $\varepsilon = 10^{-2}$

(c) $\alpha = 0.2$, $\varepsilon = 10^{-3}$

(d) $\alpha = 0.2$, $\varepsilon = 10^{-3}$

(e) $\alpha = 0.2$, $\varepsilon = 10^{-4}$

(f) $\alpha = 0.2$, $\varepsilon = 10^{-4}$

(g) $\alpha = 0.5$, $\varepsilon = 10^{-3}$

(h) $\alpha = 0.7$, $\varepsilon = 10^{-3}$

(i) $q = 10$, $l = 1.1$

(j) $\alpha = 0.5$, $q = 10$, $l = 1.1$

Figure 1: Results of SOE approximation error $R_{soe}(t)$ for $E_\alpha(-t^\alpha)$.

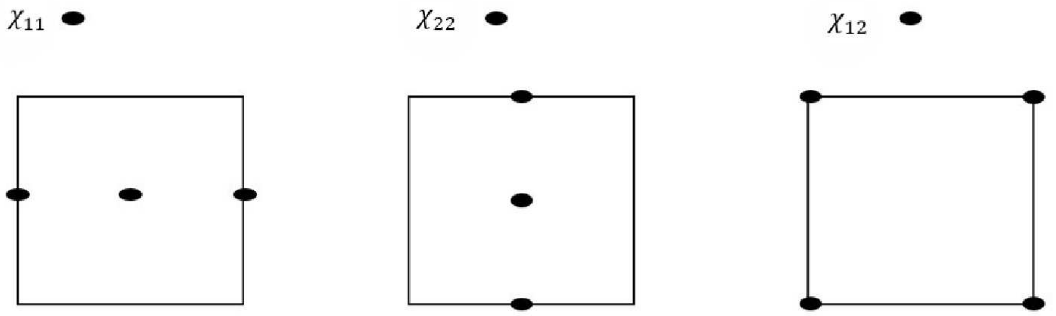


Figure 2: Nodal degrees of freedom for Hu-Man-Zhang's element.

to be isotropic, with material properties $\rho = 1$, $\mu = 1$, and $\lambda = 1$, and the exact displacement field $\mathbf{u}(x,y,t)$ of the model is also assumed to take the form

$$\mathbf{u}(x,y,t) = \begin{pmatrix} e^{-t}(x^2-x)^2(4y^3-6y^2+2y) \\ -e^{-t}(y^2-y)^2(4x^3-6x^2+2x) \end{pmatrix}.$$

In Algorithm 3.2 we use $1/h \times 1/h$ square meshes and N uniform grids for the spatial domain Ω and the time region $[0,T]$. For the spatial discretization, we apply the Hu-Man-Zhang rectangular element [27] spaces, i.e.

$$\underline{\underline{\mathrm{H}_h}} = \Big\{\chi \in \underline{\mathrm{H}}(\mathbf{div},\Omega,S);\ \chi_{11} \in P_{2,0}(T), \chi_{22} \in P_{0,2}(T), \chi_{12} \in Q_1(T),\ \forall T \in \mathcal{T}_h\Big\},$$

$$\underline{\mathrm{V}_h} = \big\{w \in \underline{L}^2(\Omega);\ w_1 \in P_{1,0}(T), w_2 \in P_{0,1}(T),\ \forall T \in \mathcal{T}_h\big\}.$$

The local nodal degrees of freedom for the stress tensor τ are shown in Fig. 2.

In the SOE approximation, we set $q = 10$, $l = 1.1$, and $\varepsilon = 10^{-3}$. Numerical results of the error

$$\|U - \mathbf{u}\|_{l^\infty} := \max_{1 \le n \le N} \|U(t_n) - \mathbf{u}(t_n)\|_{L^2(\Omega)},$$

the memory cost of the history variables H_n in Algorithm 3.1 and G_j in Algorithm 3.2, and the wall time of the total runtime of the algorithms are given in Tables 4-6 (All timings are measured in wall-clock seconds using MATLAB's tic/toc mechanism). Here, the wall time does not include the time required to solve the linear system.

From Tables 4-6 we can see that when the spatial mesh is fixed, the errors of the L1-Newmark MFE scheme and the fast scheme are close. However, the wall time and the memory (Mem) cost of the fast scheme are much less than those of the L1-Newmark MFE scheme. In particular, the memory cost of the fast algorithm remains almost constant as the time step size becomes smaller. This is because the fast algorithm only requires N_{exp} pieces of information from the $(N-1)$-th level (cf. (3.8)) to compute the N-th time level, whereas the L1-Newmark MFE scheme needs all historical information from the previous $N-1$ time levels (cf. (3.5)). Fig. 3 gives results of the wall time and the memory cost of the two algorithms at different time steps when $h = 1/8$, showing that the fast scheme performs excellently.

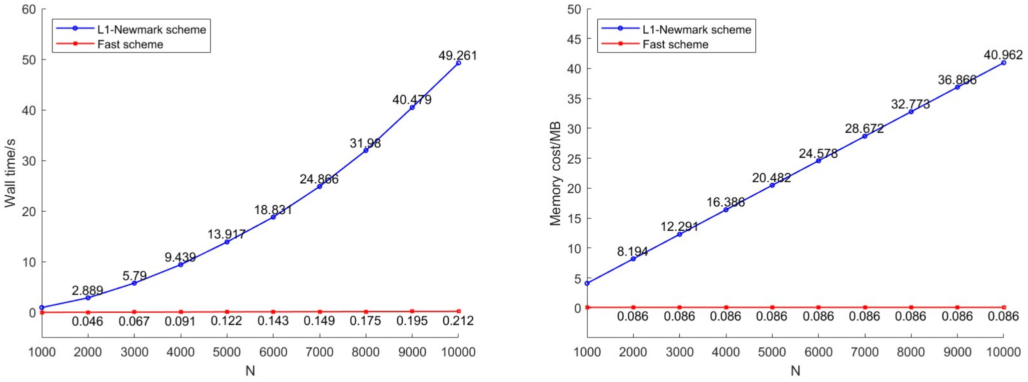


Figure 3: A comparison of wall time and memory cost between the two algorithms for different time steps when $h = 1/8$.

Table 4: Results of wall time, memory cost and error $\|U - \mathbf{u}\|_{l^\infty}$ for $h = 1/8$.

$\triangle t$	L1-Newmark			Fast Scheme		
	time (s)	Mem (MB)	$\|U-\mathbf{u}\|_{l^\infty}$	time (s)	Mem (MB)	$\|U-\mathbf{u}\|_{l^\infty}$
0.01	0.062	0.411	1.816 104 5e−3	0.002	0.086	1.815 976 3e−3
0.005	0.130	0.821	1.815 941 3e−3	0.004	0.086	1.815 867 4e−3
0.001	0.976	4.098	1.815 957 3e−3	0.026	0.086	1.815 934 5e−3
0.0005	2.777	8.194	1.815 952 3e−3	0.043	0.086	1.815 936 5e−3
0.0001	49.261	40.962	1.815 946 8e−3	0.212	0.086	1.815 937 1e−3

Table 5: Results of wall time, memory cost and error $\|U - \mathbf{u}\|_{l^\infty}$ for $h = 1/16$.

$\triangle t$	L1-Newmark			Fast Scheme		
	time (s)	Mem (MB)	$\|U-\mathbf{u}\|_{l^\infty}$	time (s)	Mem (MB)	$\|U-\mathbf{u}\|_{l^\infty}$
0.01	0.351	1.647	1.308 384 1e−3	0.034	0.344	1.308 333 7e−3
0.005	0.742	3.285	1.308 378 6e−3	0.067	0.344	1.308 349 5e−3
0.001	5.264	16.392	1.308 429 2e−3	0.343	0.344	1.308 419 9e−3
0.0005	13.839	32.776	1.308 424 5e−3	0.666	0.344	1.308 418 1e−3
0.0001	216.673	163.848	1.308 421 3e−3	3.276	0.344	1.308 417 2e−3

Table 6: Results of wall time, memory cost and error $\|U - \mathbf{u}\|_{l^\infty}$ for $h = 1/32$.

$\triangle t$	L1-Newmark			Fast Scheme		
	time (s)	Mem (MB)	$\|U-\mathbf{u}\|_{l^\infty}$	time (s)	Mem (MB)	$\|U-\mathbf{u}\|_{l^\infty}$
0.01	2.824	6.586	1.140 568 1e−3	1.153	1.376	1.140 552 9e−3
0.005	5.688	13.141	1.140 580 8e−3	2.152	1.376	1.140 572 3e−3
0.001	33.486	65.569	1.140 729 9e−3	10.728	1.376	1.140 727 2e−3
0.0005	82.683	131.114	1.140 696 8e−3	21.532	1.376	1.140 694 9e−3
0.0001	1003.425	655.387	1.140 686 6e−3	107.819	1.376	1.140 685 8e−3

5. Conclusions

We proposed an efficient SOE approximation for the Mittag-Leffler function, with a rigorous error analysis, and applied it to develop a fast fully discrete mixed finite element scheme for the fractional viscoelastic wave propagation model. Numerical experiments have demonstrated the excellent performance of the developed fast scheme in storage and computing efficiency.

Acknowledgments

This work was supported in part by the National Natural Science Foundation of China (Grant No. 12171340).

References

[1] K. Adolfsson, M. Enelund, and S. Larsson, *Adaptive discretization of fractional order viscoelasticity using sparse time history*, Comput. Methods Appl. Mech. Eng. 193 (2004), 4567–4590.

[2] K. Adolfsson, M. Enelund, and S. Larsson, *Space-time discretization of an integro-differential equation modeling quasi-static fractional-order viscoelasticity*, J. Vib. Control 14 (2008), 1631–1649.

[3] D. Baffet, *A Gauss-Jacobi kernel compression scheme for fractional differential equations*, J. Sci. Comput. 79 (2019), 227–248.

[4] D. Baffet and J. Hesthaven, *A kernel compression scheme for fractional differential equations*, SIAM J. Numer. Anal. 55 (2017), 496–520.

[5] R. Bagley and P. Torvik, *A theoretical basis for the application of fractional calculus to viscoelasticity*, J. Rheol. 27 (1983), 201–210.

[6] R. Bagley and P. Torvik, *Fractional calculus-A different approach to the finite element analysis of viscoelastically damped structures*, AIAA J. 21 (1983), 741–748.

[7] X. Bai, J. Huang, H. Rui, and S. Wang, *Numerical simulation for 2D/3D time fractional Maxwell's system based on a fast second-order FDTD algorithm*, J. Comput. Appl. Math. 416 (2022), Paper No. 114590.

[8] T. Belytschko and T. J. R. Hughes, *A precis of developments in computational methods for transient analysis*, J. Appl. Mech. Trans. ASME 50 (1983), 1033–1041.

[9] G. S. Blair, *Analytical and integrative aspects of the stress-strain-time problem*, J. Sci. Instrum. 21 (2002), 80–84.

[10] D. R. Bland, *The Theory of Linear Viscoelasticity*, International Series of Monographs on Pure and Applied Mathematics, Pergamon Press, 1960.

[11] M. Caputo and F. Mainardi, *A new dissipation model based on memory mechanism*, Pure Appl. Geophys. 91 (1971), 134–147.

[12] M. Caputo and F. Mainardi, *Linear models of dissipation in anelastic solids*, Riv. Nuovo Cimento 1 (1971), 161–198.

[13] L. Chen, J. Zhang, J. Zhao, W. Cao, H. Wang, and J. Zhang, *An accurate and efficient algorithm for the time-fractional molecular beam epitaxy model with slope selection*, Comput. Phys. Commun. 245 (2019), Paper No. 106842.

[14] K. Diethelm, *The Analysis of Fractional Differential Equations*, Springer, 2010.

[15] E. Dill, *Continuum Mechanics: Elasticity, Plasticity, Viscoelasticity*, CRC Press, 2007.

[16] A. D. DROZDOV, *Mechanics of Viscoelastic Solids*, Wiley, 1998.

[17] M. ENELUND AND B. L. JOSEFSON, *Time-domain finite element analysis of viscoelastic structures with fractional derivatives constitutive relations*, AIAA J. 35 (1997), 1630–1637.

[18] Z. FANG, H. SUN, AND H. WANG, *A fast method for variable-order Caputo fractional derivative with applications to time-fractional diffusion equations*, Comput. Math. Appl. 80 (2020), 1443–1458.

[19] Y. C. FUNG, *International Series on Dynamics. (Book Reviews: Foundations of Solid Mechanics)*, Science, 152 (1966).

[20] R. GARRAPPA, *Numerical evaluation of two and three parameter Mittag-Leffler functions*, SIAM J. Numer. Anal. 53 (2015), 1350–1369.

[21] W. GAUTSCHI, *A survey of Gauss-Christoffel quadrature formulae*, in: E. B. Christoffel: The Influence of His Work on Mathematics and the Physical Sciences, P. L. Butzer and F. Fehér (Eds), Birkäuser, (1981), 72–147.

[22] A. GEMANT, *A method of analyzing experimental results obtained from elasto-viscous bodies*, Physics 7 (1936), 311–317.

[23] A. GEMANT, *Frictional phenomena*, J. Appl. Phys. 13 (1942), 290–299.

[24] J. M. GOLDEN AND G. A. C. GRAHAM, *Boundary Value Problems in Linear Viscoelasticity*, Springer, 1988.

[25] R. GORENFLO, A. A. KILBAS, F. MAINARDI, AND S. V. ROGOSIN, *Mittag-Leffler Functions, Related Topics and Applications*, Springer, 2014.

[26] M. E. GURTIN AND E. STERNBERG, *On the linear theory of viscoelasticity*, Arch. Ration. Mech. Anal. 11 (1962), 291–356.

[27] J. HU, H. MAN, AND S. ZHANG, *A simple conforming mixed finite element for linear elasticity on rectangular grids in any space dimension*, J. Sci. Comput. 58 (2014), 367–379.

[28] Y. HUANG, Q. LI, R. LI, F. ZENG, AND L. GUO, *A unified fast memory-saving time-stepping method for fractional operators and its applications*, Numer. Math. Theor. Meth. Appl. 15 (2022), 679–714.

[29] J. JIA, H. WANG, AND X. ZHENG, *A fast algorithm for time-fractional diffusion equation with space-time-dependent variable order,* Numer. Algorithms 94 (2023), 1705–1730.

[30] S. JIANG, J. ZHANG, Q. ZHANG, AND Z. ZHANG, *Fast evaluation of the Caputo fractional derivative and its applications to fractional diffusion equations*, Commun. Comput. Phys. 21 (2017), 650–678.

[31] B. JIN, *Fractional Differential Equations*, Springer, 2021.

[32] P. LAM, H. SO, AND C. CHAN, *Exponential sum approximation for Mittag-Leffler function and its application to fractional zener wave equation*, J. Comput. Phys. 410 (2020), Paper No. 109389.

[33] J. LI, *A fast time stepping method for evaluating fractional integrals*, SIAM J. Sci. Comput. 31 (2010), 4696–4714.

[34] M. LIU AND X. XIE, *A hybrid stress finite element method for integro-differential equations modelling dynamic fractional order viscoelasticity*, Int. J. Numer. Anal. Mod. 21 (2024), 221–243.

[35] C. LUBICH AND A. SCHÄDLE, *Fast convolution for nonreflecting boundary conditions*, SIAM J. Sci. Comput. 24 (2002), 161–182.

[36] F. MAINARDI, *Fractional Calculus and Waves in Linear Viscoelasticity: An Introduction to Mathematical Models*, World Scientific, 2022.

[37] X. MENG AND M. STYNES, *The Green's function and a maximum principle for a Caputo two-point boundary value problem with a convection term*, J. Math. Anal. Appl. 461 (2018), 198–218.

[38] N. M. Newmark, *A method of computation for structural dynamics*, J. Eng. Mech. Divis. 85 (1959), 67–94.

[39] I. Podlubny, *Fractional Differential Equations: An Introduction to Fractiorlal Derivatives, Fractiorlal Differential Eqnations, to Methods of Their Solutiori and Some of Their Applications*, Academic Press, 1998.

[40] Y. Rabotnov, F. Leckie, and W. Prager, *Creep problems in structural members*, J. Appl. Mech. 37 (1970), p. 249.

[41] R. A. Schapery, *Nonlinear viscoelastic solids*, Int. J. Solids Struct. 37 (2000), 359–366.

[42] P. C. M. Severino and J. C. Guillermo, *Computational Viscoelasticity*, Springer, 2012.

[43] K. Wang and J. Huang, *A fast algorithm for the Caputo fractional derivative*, East Asian J. Appl. Math. 8 (2018), 656–677.

[44] Y. Yan, Z. Sun, and J. Zhang, *Fast evaluation of the Caputo fractional derivative and its applications to fractional diffusion equations: A second-order scheme*, Commun. Comput. Phys. 22 (2017), 1028–1048.

[45] B. Yin, Y. Liu, H. Li, and F. Zeng, *A class of efficient time-stepping methods for multi-term time-fractional reaction-diffusion-wave equations*, Appl. Numer. Math. 165 (2021), 56–82.

[46] Y. Yu, P. Perdikaris, and G. Karniadakis, *Fractional modeling of viscoelasticity in 3D cerebral arteries and aneurysms*, J. Comput. Phys. 323 (2016), 219–242.

[47] F. Zeng, I. Turner, and K. Burrage, *A stable fast time-stepping method for fractional integral and derivative operators*, J. Sci. Comput. 77 (2018), 283–307.

[48] J. Zhang, Z. Fang, and H. Sun, *Exponential-sum-approximation technique for variable-order time-fractional diffusion equations*, J. Appl. Math. Comput. 68 (2022), 323–347.