

# Wavelet-based Edge Multiscale Parareal Algorithm for subdiffusion equations with heterogeneous coefficients in a large time domain

Guanglian Li\*

July 14, 2023

## Abstract

We present the Wavelet-based Edge Multiscale Parareal (WEMP) Algorithm, recently proposed in [Li and Hu, *J. Comput. Phys.*, 2021], for efficiently solving subdiffusion equations with heterogeneous coefficients in long time. This algorithm combines the benefits of multiscale methods, which can handle heterogeneity in the spatial domain, and the strength of parareal algorithms for speeding up time evolution problems when sufficient processors are available. Our algorithm overcomes the challenge posed by the nonlocality of the fractional derivative in previous parabolic problem work by constructing an auxiliary problem on each coarse temporal subdomain to completely uncouple the temporal variable. We prove the approximation properties of the correction operator and derive a new summation of exponential to generate a single-step time stepping scheme, with the number of terms of  $\mathcal{O}(|\log \tau_f|^2)$  independent of the final time, where  $\tau_f$  is the fine-scale time step size. We establish the convergence rate of our algorithm in terms of the mesh size in the spatial domain, the level parameter used in the multiscale method, the coarse-scale time step size, and the fine-scale time step size. Finally, we present several numerical tests that demonstrate the effectiveness of our algorithm and validate our theoretical results.

**Keywords:** multiscale, heterogeneous, long time, edge, wavelets, parareal, time-fractional, diffusion

## 1 Introduction

We consider in this paper a new efficient multiscale parareal algorithm for time-fractional diffusion problems with heterogeneous coefficients, which reads to find  $u \in V := H_0^1(D)$ , satisfying

$$\begin{cases} \partial_t^\alpha u = \nabla \cdot (\kappa \nabla u) + f & \text{in } D, \quad t \in (0, T] \\ u = 0 & \text{on } \partial D, \quad t \in (0, T] \\ u(\cdot, 0) = u_0 & \text{in } D. \end{cases} \quad (1.1)$$

Here,  $\partial_t^\alpha u$  ( $\alpha \in (0, 1)$ ) refers to the left-sided Caputo fractional derivative of order  $\alpha$  of the function  $u(\cdot, t)$ , defined by (see, e.g. [21, p. 91, (2.4.1)] or [32, p. 78]),

$$\partial_t^\alpha u(\cdot, t) = \frac{1}{\Gamma(1-\alpha)} \int_0^t \frac{1}{(t-s)^\alpha} \frac{\partial u}{\partial s}(\cdot, s) ds \quad (1.2)$$

---

\*Department of Mathematics, The University of Hong Kong, Pokfulam Road, Hong Kong. (lotusli@maths.hku.hk, lotusli@hku.hk). G.L. acknowledges support from Newton International Fellowships Alumni following-on funding awarded by The Royal Society, Young Scientists fund by NSFC, China, and Early Career Scheme (Project number: 27301921), RGC, Hong Kong.

with  $\Gamma(\cdot)$  being Euler's Gamma function:  $\Gamma(x) = \int_0^\infty s^{x-1}e^{-s}ds$  for  $x > 0$ .

Furthermore,  $D \subset \mathbb{R}^d$  for  $d = 2, 3$  is bounded with certain smoothness and  $\kappa$  is piecewise smooth as described in Assumption 3.1. The force term  $f \in W^{2,\infty}([0, T]; L^2(D))$ , the initial data  $u_0 \in \dot{H}^{2\sigma}(D) \cap H_0^1(D)$  with  $\sigma \in (0, 1]$ . The subspace  $\dot{H}^{2\sigma}(D) \subset L^2(D)$  is a Hilbert space to be defined in Section 2. To simplify the notation, let  $I := [0, T]$ . The main challenges for solving Problem (1.1) numerically lie in: 1) the existence of multiple scales in the coefficient  $\kappa$ , since resolving the problem to the finest scale as mandatory for classical numerical methods would incur huge number of degrees of freedom in the spatial domain. 2) the nonlocality of the fractional derivative  $\partial_t^\alpha$  (1.2) makes the numerical computation and storage extremely costly especially for the case of long time simulation with a large final time  $T$ .

On the one hand, the multiscale nature of the solution in the spatial domain as a consequence of the heterogeneous coefficient  $\kappa$  makes the computation challenging and deteriorates the storage cost. Nevertheless, the accurate description of many important applications, e.g., composite materials, porous media, ground water modeling and reservoir simulation, inevitably involves many heterogeneous parameters with multiple scales. They can have both multiple inseparable scales and high-contrast in order to reflect the nature of the practical applications. Because of this, the classical numerical treatment becomes prohibitively expensive especially for the unsteady problem (1.1). Consequently, various multiscale model reduction techniques have been proposed in the literature over the past few decades. Among them are multiscale finite element methods (MsFEMs), heterogeneous multiscale methods (HMMs), variational multiscale methods, flux norm approach, generalized multiscale finite element methods (GMsFEMs) and localized orthogonal decomposition (LOD) [1, 5, 9, 10, 17, 18, 25, 31]. Wavelet-based Edge Multiscale Finite Element Methods (WEMsFEMs) was proposed recently by the author, c.f. Algorithm 2, which facilitates deriving a rigorous convergence rate with merely mild assumptions [15, 16, 23]. This method utilizes wavelets as the basis functions over the coarse edges to define local multiscale basis functions, which facilitates transforming the approximate rate over the edges to the convergence rate in each local region. Then the Partition of Unity Method (PUM) [35] is applied to derive the global convergence rate. The motivation for using wavelets as the ansatz space is that the existence of heterogeneity causes the solution has a low regularity, and wavelets are known to be efficient in approximating functions with such unfavorable property.

On the other hand, the nonlocality of the fractional derivative makes long-time simulation prohibitive, which are essential in many applications such as the rare event sampling and molecular dynamics simulation. The L1 scheme is one of the well known time stepping schemes that uses piecewise linear approximation on each time step. High order schemes have been designed and analyzed in the literature to obtain a higher order convergence rate when the solution has sufficient temporal regularity. Furthermore, these time stepping schemes can be combined with approximation by exponential sums or the proper orthogonal decomposition method in the history part to reduce the storage costs [20, 38, 39]. The parareal algorithm is one of the most popular and success algorithms for long time simulation, and it has been applied to many problems including the financial mathematics and quantum chemistry [4, 6, 12, 13, 14, 28, 29, 30]. The parareal algorithm facilitates speeding up the numerical simulation of time evolution problems when sufficient processors are available [3] by using an iterative solver based on a cheap inaccurate sequential coarse-scale time stepping scheme and expensive accurate fine-scale time stepping scheme that can be performed in parallel. This algorithm was introduced by Lions, Mayday and Turinici [27]. Coupling of parareal algorithm and some other techniques has been developed in many literature, see [2, 8, 11, 22]. Among them is the coupling of parareal algorithm and model reduction techniques [22], wherein a micro-macro parareal algorithm for the time-parallel integration of multiscale-in-time systems is introduced to solve singularly perturbed ordinary differential equations. A new coupling strategy for the parareal algorithm with multiscale integrators is introduced in [2].

As a continuation of [24], we incorporate the parareal algorithm into WEMsFEM to numerically calculate the time evolution problems efficiently. The main contribution of this paper is three-fold. Firstly, we derive a new model reduction in the temporal domain by means of summation of exponentials to handle long time situation with the number of terms independent of the final time. Secondly, the stability of the resulting time stepping scheme is rigorously analyzed, with a very mild requirement on the accuracy of the approximation by summation of exponentials. Thirdly, together with the multiscale model reduction in the spatial domain, we obtain a stable iterative scheme with a very fast convergence rate. Furthermore, the convergence analysis is rigorously justified.

Our main algorithm is as follows. We first construct a multiscale space  $V_{\text{ms},\ell}$  based on WEMsFEM with  $\ell$  as the wavelets level parameter. Then we design a new parareal algorithm that corrects the current solution only and treats the history information as a parameter. This parameter is updated once by a formula after obtaining a corrected current solution. In the meanwhile, we will use  $V_{\text{ms},\ell}$  as the ansatz space to save computation and storage costs. Note that two parareal algorithms were proposed in [38], both of which involved correction operator for the history information. On the one hand, our numerical experiments demonstrate much faster convergence for our new parareal algorithm. On the other hand, our new algorithm saves more computational costs, and we can derive an explicit convergence rate. The convergence rate of this algorithm is presented in Theorem 6.2. We proved for any  $\sigma \in (0, 1)$ ,

$$\begin{aligned} \|u(\cdot, T^n) - U_k^n\|_D &\lesssim (\tau_f^{\sigma\alpha} + \eta(H, \ell) |\log \eta(H, \ell)|) \|f\|_{W^{2,\infty}(0,T;L^2(D))} \\ &\quad + \left( \tau_f^{\sigma\alpha} + \eta(H, \ell) |\log \eta(H, \ell)| (T^n)^{-\alpha(1-\sigma)} + \prod_{j=0}^k \left( \frac{\tau_c}{T^{n-j}} \right)^{\alpha\sigma} \right) |u_0|_{2\sigma}. \end{aligned}$$

where  $u(\cdot, T^n)$  and  $U_k^n$  are the exact solution at  $T^n = n \times \tau_c$  for  $n = 2, \dots$  and our proposed WEMP algorithm at iteration  $k$ . The notations  $\tau_c$  and  $\tau_f$  represent for the coarse time step size and fine time step size, respectively.  $H, \ell$  and  $k$  are the space domain mesh size, the level parameter and iteration number. The parameter  $\eta(H, \ell)$  is defined in (3.8) that reflects the approximation property of the multiscale ansatz space  $V_{\text{ms},\ell}$ . This implies that taking  $\ell = \mathcal{O}(|\log H|)$  and  $k = \mathcal{O}(|\log \tau_f / \log \tau_c|)$ , we obtain  $\mathcal{O}(H^2 + \tau_f^{\sigma\alpha})$  error.

To demonstrate the performance of our proposed algorithm we present several numerical tests to first investigate the influence of approximation by exponential sums on the accuracy of our method. Then a series of simulations are made to show the convergence of our method. Finally, we present several numerical results with large final time  $T$ .

Our paper is structured as follows. In Section 2, we introduce the Galerkin-L1-scheme to Problem (1.1), and provide an overview of multiscale model reduction in the spatial domain and the parareal algorithm. In Section 3, we discuss the construction of the multiscale ansatz space  $V_{\text{ms},\ell}$  by WEMsFEMs based upon [15, 16, 23] and recap its approximation properties. Section 4 is devoted to deriving an efficient numerical scheme for the history information in the fractional derivative based on the Sinc approximation, which facilitates designing a new multiscale-SOE-scheme to solve Problem (1.1). The main ingredient of our proposed algorithm is presented in Section 5 and its convergence rate is derived in Section 6. Extensive numerical tests are presented in Section 7. Finally, we complete our paper with concluding remarks in Section 8.

## 2 Problem setting

We first present in Section 2.1 the full discretization of problem (1.1) by the standard conforming Galerkin piecewise linear element in the spatial domain under a tiny fitted mesh  $\mathcal{T}_h$  of mesh size  $h \ll 1$ , and L1 scheme in the temporal domain using a uniform discretization of mesh size  $\tau_f \ll 1$ . Then we propose, on

the one hand, to replace this expensive fitted mesh  $\mathcal{T}_h$  with a much cheaper unfitted mesh  $\mathcal{T}_H$  with mesh size  $H \gg h$  by construction of multiscale ansatz space in Section 2.2, and on the other hand, to further reduce the computational complexity by building iterative time stepping scheme over the coarse subintervals based on the parareal algorithm in Section 2.3.

To start with, we define the Hilbert space  $\dot{H}^s(D)$ , which is analogous to [37, Chapter 3]. Let  $\{(\lambda_m, \phi_m)\}_{m=1}^\infty$  be the eigenpairs of the following eigenvalue problems with the eigenvalues arranged in a nondecreasing order,

$$\begin{aligned} \mathcal{L}\phi_m &:= -\nabla \cdot (\kappa \nabla \phi_m) = \lambda_m \phi_m && \text{in } D \\ \phi_m &= 0 && \text{on } \partial D. \end{aligned}$$

Note that the eigenfunctions  $\{\phi_m\}_{m=1}^\infty$  form an orthonormal basis in  $L^2(D)$ , and consequently, each  $v \in L^2(D)$  admits the representation  $v = \sum_{m=1}^\infty (v, \phi_m)_D \phi_m$  with  $(\cdot, \cdot)_D$  being the inner product in  $L^2(D)$ . The associated norm in  $L^2(D)$  is denoted as  $\|\cdot\|_D$ .

The Hilbert space  $\dot{H}^s(D) \subset L^2(D)$  is defined by

$$\dot{H}^s(D) = \{v \in L^2(D) : \sum_{m=1}^\infty \lambda_m^s |(v, \phi_m)_D|^2 < \infty\} \quad (2.1)$$

with its associated norm being  $|v|_s := \left( \sum_{m=1}^\infty \lambda_m^s |(v, \phi_m)_D|^2 \right)^{1/2}$  for all  $v \in \dot{H}^s(D)$ .

## 2.1 Galerkin-L1-scheme

To discretize problem (1.1), we first introduce fine and coarse discretization in the spatial domain  $D$ . Let  $\mathcal{T}_H$  be a regular partition of the domain  $D$  into finite elements (triangles, quadrilaterals, tetrahedral, etc.) with a mesh size  $H$ , which is determined by the desired accuracy of our numerical method. We refer to this partition and its elements as coarse mesh and the coarse elements. Then each coarse element is further partitioned into a union of connected fitted fine elements. This fine-scale partition is denoted by  $\mathcal{T}_h$  with  $h$  being its mesh size. Let  $\mathcal{F}_h$  (or  $\mathcal{F}_H$ ) be the collection of all edges in  $\mathcal{T}_h$  (or  $\mathcal{T}_H$ ). Over the fine mesh  $\mathcal{T}_h$ , let  $V_h$  be the conforming piecewise linear finite element space:

$$V_h := \{v \in V : v|_E \in \mathcal{P}_1(E) \text{ for all } E \in \mathcal{T}_h\},$$

where  $\mathcal{P}_1(E)$  denotes the space of linear polynomials on the fine element  $E \in \mathcal{T}_h$ .

The time interval  $I := [0, T]$  is decomposed into a sequence of uniform coarse subintervals  $[T^n, T^{n+1}]$  for  $n = 0, 1, \dots, M_c$  of size  $\tau_c$  with  $\tau_c := T/M_c$  for some  $M_c \in \mathbb{N}_+$  and  $T^0 := 0$ . The total number of coarse steps  $M_c$  is governed by the available number of processors and the aimed speedup. Each coarse time interval  $[T^n, T^{n+1}]$  is further uniformly discretized with a fine time step  $\tau_f$  controlled by the expected accuracy in the temporal discretization. Let  $t_n = n \times \tau_f$  for  $n = 0, 1, \dots, M_f$  with  $M_f := T \times \tau_f^{-1}$ . Note that we assume  $\tau_c \gg \tau_f$  throughout of this paper.

The  $L1$  scheme [26, 36] is one of the most popular and simplest discretization schemes for (1.2) the temporal domain  $I$ , which is based on the idea of using piecewise constant approximation to approximate the first derivative. The scheme reads

$$\tilde{\partial}_{\tau_f}^\alpha v(x, t_{n+1}) = \sum_{j=0}^n b_j \frac{v(x, t_{n+1-j}) - v(x, t_{n-j})}{\tau_f^\alpha c_\alpha} \quad (2.2)$$

with  $c_\alpha = \Gamma(2 - \alpha)$  and  $b_j = (j + 1)^{1-\alpha} - j^{1-\alpha}$  for  $j = 0, 1, \dots, n$ .

The next lemma provides a lower bound of the coefficient for the L1 scheme:

**Lemma 2.1.** *The coefficient from (2.2) has the following lower bound,*

$$b_{j-1} - b_j \geq \alpha(1 - \alpha)(n + 1)^{-\alpha-1} \text{ for all } j = 1, \dots, n.$$

*Proof.* Let  $r(x) := (x + 1)^{1-\alpha} - x^{1-\alpha}$  for  $x > 0$ . Then  $r(x)$  is decreasing and convex. Consequently,

$$b_{j-1} - b_j \geq b_{n-1} - b_n \geq -r'(n) = (1 - \alpha)(n^{-\alpha} - (n + 1)^{-\alpha}).$$

Note that  $r_1(x) = x^{-\alpha}$  for  $x > 0$  is decreasing and convex, this yields,

$$b_{j-1} - b_j \geq (1 - \alpha)(n^{-\alpha} - (n + 1)^{-\alpha}) \geq \alpha(1 - \alpha)(n + 1)^{-\alpha-1}.$$

We have proved the desired assertion. □

We use conforming Galerkin method for the discretization in the spatial domain  $D$  throughout this paper. Then the Galerkin-L1-scheme reads to find  $u_h^n \in V_h$  for  $n = 1, 2, \dots, M_f$ , satisfying

$$\begin{cases} (\tilde{\partial}_{\tau_f}^\alpha u_h^n, v_h)_D + a(u_h^n, v_h) = (f(t_n), v_h)_D & \text{for all } v_h \in V_h \\ u_h^0 = I_h u_0. \end{cases} \quad (2.3)$$

Here, the bilinear form  $a(\cdot, \cdot)$  on  $V \times V$  is defined by

$$a(v_1, v_2) := \int_D \kappa \nabla v_1 \cdot \nabla v_2 \, dx \text{ for all } v_1, v_2 \in V$$

with  $I_h$  being the  $L^2(D)$ -projection from  $V$  to  $V_h$ . Furthermore, we define the energy norm  $\|v\|_{H_\kappa^1(D)} := \sqrt{a(v, v)}$  for all  $v \in V$ .

The fine-scale solution  $u_h^n$  will serve as a reference solution in Section 7. Note that due to the presence of multiple scales in the coefficient  $\kappa$ , the fine-scale mesh size  $h$  should be commensurate with the smallest scale and thus it can be very small in order to obtain an accurate solution for each time step. In addition, the well-known limited spatial smoothness property of Problem (1.1) makes high-order methods in the spatial domain infeasible [33]. These result in huge computational complexity, and consequently, more efficient methods are in great demand. Moreover, the storage cost of order  $\mathcal{O}(M_f h^{-d})$  associated with the L1 scheme (2.2) becomes tremendous under such scenario. These make a direct solver for  $u_h^n$  prohibitive.

## 2.2 Multiscale-L1-scheme

Multiscale model reduction methods aim at replacing the fine-scale mesh  $\mathcal{T}_h$  with a coarse-scale mesh  $\mathcal{T}_H$  when solving Problem (2.3) numerically, which, meanwhile, maintains a certain accuracy. To describe it, we need a few notations. The vertices of  $\mathcal{T}_H$  are denoted by  $\{O_i\}_{i=1}^N$ , with  $N$  being the total number of coarse nodes. The coarse neighborhood associated with the node  $O_i$  is denoted by

$$\omega_i := \bigcup \{K \in \mathcal{T}_H : O_i \in \bar{K}\}. \quad (2.4)$$

We refer to Figure 1 for an illustration of neighborhoods and elements subordinated to the coarse discretization  $\mathcal{T}_H$ . Throughout, we use  $\omega_i$  to denote a coarse neighborhood. Furthermore, Let  $\mathcal{F}_h$  (or  $\mathcal{F}_H$ ) be the

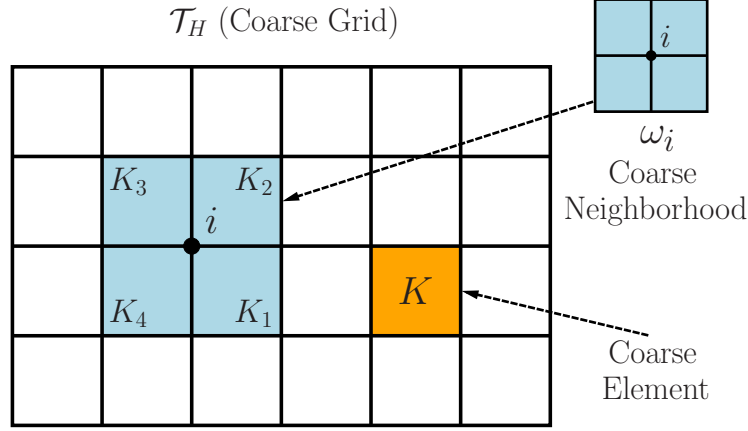


Figure 1: Illustration of a coarse neighborhood and coarse element.

collection of all edges in  $\mathcal{T}_h$  (or  $\mathcal{T}_H$ ) and let  $\mathcal{F}_h(\partial\omega_i)$  (or  $\mathcal{F}_H(\partial\omega_i)$ ) be the restriction of  $\mathcal{F}_h$  on  $\partial\omega_i$  (or  $\mathcal{F}_H$  on  $\partial\omega_i$ ).

Next, the multiscale ansatz space  $V_{\text{ms}} \subset V_h$  is constructed that are locally supported on certain coarse neighborhood  $\omega_i$ . Then the multiscale-L1-scheme aims at seeking  $u_{\text{ms}}^n \in V_{\text{ms}}$  for  $n = 1, \dots, M_f$ , s.t.,

$$\begin{cases} (\tilde{\partial}_{\tau_f}^\alpha u_{\text{ms}}^n, v_{\text{ms}})_D + a(u_{\text{ms}}^n, v_{\text{ms}}) = (f(\cdot, t_n), v_{\text{ms}})_D & \text{for all } v_{\text{ms}} \in V_{\text{ms}} \\ u_{\text{ms}}^0 = I_{\text{ms}}u_0. \end{cases} \quad (2.5)$$

Here,  $V_{\text{ms}}$  denotes the multiscale space spanned by these multiscale basis functions and  $I_{\text{ms}}$  is a projection operator from  $V$  to  $V_{\text{ms}}$ .

The construction of  $V_{\text{ms}}$  will be presented in Section 3. Note that we need a very tiny fine-scale time step size  $\tau_f$  to guarantee a reasonable approximation property of  $u_{\text{ms}}^n$  to  $u(\cdot, t_n)$  for  $n = 1, \dots, M_f$  due to, e.g., the singularity of the solution  $u(\cdot, t)$  at  $t = 0$  when the source term  $f$  fails to belong to certain smooth functional space. Consequently, the computational complexity and the storage requirement of the multiscale-L1-scheme (2.5) can be still extremely expensive. For this reason, we introduce in Section 2.3 the parareal algorithm to significantly improve its efficiency.

### 2.3 Parareal algorithm

The main goal of the parareal algorithm is to obtain a high-order approximation of the solution at the coarse time step  $u(\cdot, T^n)$  for  $n = 1, \dots, M_c$  iteratively under the coarse time stepping scheme, which is corrected by parallel local fine time stepping scheme over each coarse subinterval. To demonstrate this algorithm, we need to introduce the one-step coarse propagator and fine propagator

$$\mathcal{G}(T^n, U^n) \text{ and } \mathcal{F}(T^n, U^n)$$

for given initial data  $U^n$  over the coarse interval  $[T^n, T^{n+1}]$ . The outcome of these propagators is an approximation to  $u(\cdot, T^{n+1})$  for  $n = 0, 1, \dots, M_c - 1$ .

---

**Algorithm 1:** Parareal algorithm

---

**Data:** Initial data  $u_0$ ; tolerance  $\delta$  and initial solution  $U_0^n$  for  $n = 0, \dots, M_c$ .

**Result:**  $U_k^n$  for  $n = 0, \dots, M_c$ .

1: Set  $k = 1$ .

2: **do**

3: Set the initial data  $U_k^0 = I_{\text{ms}}u_0$ .

4: Parallel compute  $U_k^n$  on each local time subinterval  $[T^{n-1}, T^n]$  for  $n = 1, \dots, M_c$ ,

$$U_k^n = \mathcal{G}(T^{n-1}, U_{k-1}^{n-1}).$$

5: Parallel compute  $u_k^n$  on each local time subinterval  $[T^{n-1}, T^n]$  for  $n = 1, \dots, M_c$ ,

$$u_k^n = \mathcal{F}(T^{n-1}, U_{k-1}^{n-1}).$$

6: Parallel compute the solution jumps for  $n = 1, \dots, M_c$ ,

$$\mathcal{S}(T^{n-1}, U_{k-1}^{n-1}) = u_k^n - U_k^n.$$

7: Sequentially compute the corrected coarse solutions by propagation of the jumps

$$U_k^n = \mathcal{S}(T^{n-1}, U_{k-1}^{n-1}) + \mathcal{G}(T^{n-1}, U_k^{n-1}).$$

8: Estimate the error

$$\text{err} := 1/M_c \sum_{n=1}^{M_c} \|U_k^n - U_{k-1}^n\|_{\ell_2}.$$

9:  $k \leftarrow k + 1$ .

10: **while**  $\text{err} > \delta$

---

We present in Algorithm 1 the main ingredient of this method, which includes a parallel computation of the local jumps in Step 6 and the sequential propagation of the local jumps in Step 7. One can observe from [33, Theorem 2.1] that the solution operator to problem (1.1) is not a semigroup. Consequently, the fine propagator  $\tilde{\partial}_{\tau_f}^\alpha$  defined by the L1 scheme (2.2) requires all the information from the previous time steps. This results in a huge computational and storage cost. To reduce this cost, we will on the one hand, decrease the number of terms in the history part by means of approximation by exponential sums. On the other hand, the length of each term will be reduced by multiscale approximation.

The goal of this paper is to design an efficient numerical algorithm to approximate  $u(\cdot, t)$  such that it can be solved on the coarse spatial mesh  $\mathcal{T}_H$ , without resolving the microscale information in the coefficient  $\kappa$  using a rather coarse time step  $\tau_c \gg \tau_f$  that can be computationally inexpensive. In the meanwhile, the storage cost can be reduced hugely. This is achieved by a combination of the multiscale method and parareal algorithm as proposed in [24] for parabolic problems with heterogeneous coefficients. Compared with parabolic problems, the main difficulty encountered here is that the solution operator of the time fractional diffusion problem is not a semigroup. This will be overcome in Section 4 by a new approximation by exponential sums.

### 3 Multiscale ansatz space

This section is concerned with the construction of the multiscale space by means of the Wavelet-based Edge Multiscale Finite Element Methods (WEMsFEMs) [15, 16, 23]. The main idea is to use wavelets as the boundary data to construct local multiscale basis functions on each coarse neighborhood  $\omega_i$ . Then the multiscale ansatz space  $V_{\text{ms},\ell}$  is constructed by multiplying these local multiscale basis functions with the partition of unity functions defined over the coarse patches  $\{\omega_i\}_{i=1}^N$ .

Let  $\{\chi_i\}_{i=1}^N$  be the partition of unity functions, which are the standard multiscale basis functions on each coarse element  $K \in \mathcal{T}_H$  defined by

$$\begin{aligned} -\nabla \cdot (\kappa \nabla \chi_i) &= 0 & \text{in } K \\ \chi_i &= g_i & \text{on } \partial K. \end{aligned} \quad (3.1)$$

Here,  $g_i$  is affine over  $\partial K$  with  $g_i(O_j) = \delta_{ij}$  for all  $i, j = 1, \dots, N$ . Recall that  $\{O_j\}_{j=1}^N$  are the set of coarse nodes on  $\mathcal{T}_H$ .

The construction of the multiscale test space  $V_{\text{ms},\ell}$  is presented in Algorithm 2. Given the level parameter  $\ell \in \mathbb{N}$ , and the type of wavelets on each edge of the coarse neighborhood  $\omega_i$ , one can obtain the local multiscale space  $V_{i,\ell}$  on  $\omega_i$  by solving  $2^{\ell+2}$  local problems in parallel in Step 2. One local multiscale basis function that is vanishing over  $\partial\omega_i$  is calculated in parallel. In Step 4, we can use these local multiscale space to build the global multiscale space  $V_{\text{ms},\ell}$  by multiplying the partition of unity functions  $\chi_i$ .

---

#### Algorithm 2: Wavelet-based Edge Multiscale Finite Element Method (WEMsFEM)

---

**Data:** The level parameter  $\ell \in \mathbb{N}$ ; coarse neighborhood  $\omega_i$  and its four coarse edges  $\Gamma_{i,k}$  with  $k = 1, 2, 3, 4$ , i.e.,  $\cup_{k=1}^4 \Gamma_{i,k} = \partial\omega_i$ ; the subspace  $V_{\ell,k} \subset L^2(\Gamma_{i,k})$  up to level  $\ell$  on each coarse edge  $\Gamma_{i,k}$ .

**Result:** Multiscale space  $V_{\text{ms},\ell}$ .

- 1: Define the local edge basis functions. Let  $V_\ell^i := \oplus_{k=1}^4 V_{\ell,k} := \text{span}\{v_k : k = 1, \dots, 2^{\ell+2}\}$ . Then the number of basis functions in  $V_\ell^i$  is  $4 \times 2^\ell = 2^{\ell+2}$ .
- 2: Calculate local multiscale basis  $\mathcal{L}_i^{-1}(v_k)$  for all  $k = 1, \dots, 2^{\ell+2}$ . Here,  $\mathcal{L}_i^{-1}(v_k) := v$  satisfies

$$\begin{cases} \mathcal{L}_i v := -\nabla \cdot (\kappa \nabla v) = 0 & \text{in } \omega_i \\ v = v_k & \text{on } \partial\omega_i. \end{cases}$$

- 3: Solve one local multiscale basis function,

$$\begin{cases} -\nabla \cdot (\kappa \nabla v^i) = \frac{\tilde{\kappa}}{\int_{\omega_i} \tilde{\kappa} \, dx} & \text{in } \omega_i \\ -\kappa \frac{\partial v^i}{\partial n} = |\partial\omega_i|^{-1} & \text{on } \partial\omega_i. \end{cases}$$

- 4: Build global multiscale space:  $V_{\text{ms},\ell} := \text{span}\{\chi_i \mathcal{L}_i^{-1}(v_k), \chi_i v^i : 1 \leq i \leq N, 1 \leq k \leq 2^{\ell+2}\}$ .
- 

To analyze this algorithm, we define the weighted coefficient:

$$\tilde{\kappa} = H^2 \kappa \sum_{i=1}^N |\nabla \chi_i|^2. \quad (3.2)$$



Furthermore, let  $\tilde{\kappa}^{-1}$  be defined by

$$\tilde{\kappa}^{-1}(x) := \begin{cases} \tilde{\kappa}^{-1}, & \text{when } \tilde{\kappa}(x) \neq 0, \\ 1, & \text{otherwise.} \end{cases} \quad (3.3)$$

Next, we recap a few results on the approximation properties of the multiscale ansatz space  $V_{\text{ms},\ell}$  which play a key role in the later convergence analysis (Section 6.1). The following assumption on the coefficient  $\kappa$  implies that  $\kappa$  is piecewise smooth.

**Assumption 3.1** (Structure of  $D$  and  $\kappa$ ). *Let  $D$  be a domain with a  $C^{1,\rho}$  ( $0 < \rho < 1$ ) boundary  $\partial D$ , and  $\{D_i\}_{i=1}^m \subset D$  be  $m$  pairwise disjoint strictly convex open subsets, each with a  $C^{1,\rho}$  boundary  $\Gamma_i := \partial D_i$ , and denote  $D_0 = D \setminus \bigcup_{i=1}^m \bar{D}_i$ . Let the permeability coefficient  $\kappa$  be piecewise regular function defined by*

$$\kappa = \begin{cases} \eta_i(x) & \text{in } D_i, \\ 1 & \text{in } D_0. \end{cases} \quad (3.4)$$

Here  $\eta_i \in C^\mu(\bar{D}_i)$  with  $\mu \in (0, 1)$  for  $i = 1, \dots, m$ . Denote  $\eta_{\min} := \min_i \{\min_{x \in D_i} \{\eta_i(x)\}\} \geq 1$  and  $\eta_{\max} := \max_i \{\|\eta_i\|_{C^0(D_i)}\}$ . Furthermore, we assume the diameter of  $D$  is 1.

Assumption 3.1, first proposed in [23] then further utilized in [15], is needed to make the constant appearing in the approximation properties independent of the possible contrast in the coefficient  $\kappa$ .

In the following, we define the  $L^2(\partial\omega_i)$ -orthogonal projection onto the local multiscale space up to level  $\ell$  by  $\mathcal{P}_{i,\ell} : L^2(\partial\omega_i) \rightarrow \mathcal{L}_i^{-1}(V_\ell^i)$  satisfying

$$\mathcal{P}_{i,\ell}(v) := \sum_{j=1}^{2^{\ell+2}} (v, \psi_j)_{\partial\omega_i} \mathcal{L}_i^{-1}(\psi_j) \quad \text{for all } v \in L^2(\partial\omega_i). \quad (3.5)$$

Here, we denote  $\psi_j$  for  $j = 1, \dots, 2^{\ell+2}$  as the hierarchical bases defined on the four edges of  $\omega_i$  of level  $\ell$  and the local operator  $\mathcal{L}_i$  is defined as in Algorithm 2.

Let  $\mathcal{P}_\ell$  be the projection onto the multiscale space  $V_{\text{ms},\ell}$  given by

$$\mathcal{P}_\ell(v) := \sum_{i=1}^N \chi_i \mathcal{P}_{i,\ell}(v) \quad \text{for all } v \in V. \quad (3.6)$$

Let  $\mathcal{L} := -\nabla \cdot (\kappa \nabla \cdot)$  be the elliptic operator defined on  $V$ , and let its discrete operator  $\mathcal{L}_\ell : V_{\text{ms},\ell} \rightarrow H^{-1}(D)$  be

$$(\mathcal{L}_\ell w_\ell, v_\ell) := (\mathcal{L} w_\ell, v_\ell) = a(w_\ell, v_\ell) \quad \text{for all } v_\ell \text{ and } w_\ell \in V_{\text{ms},\ell}.$$

Then the inverse operator  $\mathcal{L}_\ell^{-1}$  exists, which is self-adjoint, positive semi-definite on  $L^2(D)$ , and positive definite on  $V_{\text{ms},\ell}$ . Further, let  $\mathcal{R}_\ell$  be the Riesz operator associated to  $\mathcal{L}_\ell$  in the multiscale space  $V_{\text{ms},\ell}$ , i.e.,

$$\forall v \in V \text{ and } w_{\text{ms}} \in V_{\text{ms},\ell} : a(v - \mathcal{R}_\ell v, w_{\text{ms}}) = 0,$$

then the following relation holds

$$\mathcal{L}_\ell^{-1} = \mathcal{R}_\ell \mathcal{L}^{-1}.$$

The approximation property of  $\mathcal{L}_\ell^{-1}$  in energy norm is derived in [15, Proposition 5.2]. For any  $v \in L^2(D)$ , it holds

$$\|\mathcal{L}^{-1}v - \mathcal{L}_\ell^{-1}v\|_{H_k^1(D)} \lesssim \eta(H, \ell)\|v\|_{L^2(D)} \quad (3.7)$$

with

$$\eta(H, \ell) := H \|\tilde{\kappa}\|_{L^\infty(D)}^{1/2} + 2^{-\ell/2}\|\kappa\|_{L^\infty(\mathcal{F}_H)}. \quad (3.8)$$

This, together with Assumption 3.1 and orthogonality of the Riesz operator  $\mathcal{R}_\ell$ , leads to

$$\|\mathcal{L}^{-1}v - \mathcal{L}_\ell^{-1}v\|_D \lesssim \eta(H, \ell) \|\mathcal{L}^{-1}v\|_{H_k^1(D)}. \quad (3.9)$$

## 4 Model reduction in the history part

Recall that the time fractional operator  $\partial_t^\alpha$  can be split into a summation of the local component and the history component

$$\begin{aligned} \partial_t^\alpha v(x, t_{n+1}) &= \frac{1}{\Gamma(1-\alpha)} \int_0^{t_{n+1}} \frac{1}{(t_{n+1}-s)^\alpha} v'(x, s) ds \\ &= \frac{1}{\Gamma(1-\alpha)} \int_{t_n}^{t_{n+1}} \frac{1}{(t_{n+1}-s)^\alpha} v'(x, s) ds + \frac{1}{\Gamma(1-\alpha)} \int_0^{t_n} \frac{1}{(t_{n+1}-s)^\alpha} v'(x, s) ds \\ &=: \partial_{\mathcal{N}}^\alpha v(x, t_{n+1}) + \partial_{\mathcal{F}}^\alpha v(x, t_{n+1}). \end{aligned} \quad (4.1)$$

We will approximate the local component  $\partial_{\mathcal{N}}^\alpha v(x, t_{n+1})$  by the L1 scheme and the history component  $\partial_{\mathcal{F}}^\alpha v(x, t_{n+1})$  by a new exponential sums inspired by sinc approximation [7, 34] in Section 4.1. Based upon this, a new local time integrator is presented in Section 4.2 that can handle long time simulation more efficiently.

### 4.1 Approximation by exponential sums

In order to deal with long time situation, we will construct a new approximation by exponential sums inspired by [7], which is essentially based upon Sinc quadrature to analytic, improperly integrable functions in  $\mathbb{R}$ , see [34]. Let  $F(s)$  be an analytic, improperly integrable functions defined on  $\mathbb{R}$  and let  $\tau > 0$  be the step size and  $N \in \mathbb{N}_+$ , then we obtain by integration after Sinc interpolation to  $F(s)$  the following quadrature formula

$$\begin{aligned} \int_{-\infty}^{\infty} F(s) ds &\approx \tau \sum_{j=-\infty}^{\infty} F(j\tau) \\ &\approx \tau \sum_{j=-N}^N F(j\tau). \end{aligned}$$

Here, the second estimate is a direct truncation of the first estimate. The quadrature error can be estimated by the following theorem [34, Theorem 3.2.1].

**Lemma 4.1.** *Let  $F(s)$  be holomorphic in the strip  $\mathcal{D}_\delta := \{z \in \mathbb{C} : \Im z \in (-\delta, \delta)\}$  with  $\delta > 0$  such that*

$$\int_{\partial\mathcal{D}_\delta} |F(s)| |ds| < \infty, \text{ and for some } c, \beta > 0 : |F(s)| < ce^{-\beta|s|} \text{ for all } s \in \mathbb{R}.$$

Then taking  $\tau := \left(\frac{2\pi\delta}{\beta N}\right)^{1/2}$ , we obtain

$$\left| \int_{-\infty}^{\infty} F(s) ds - \tau \sum_{j=-N}^N F(j\tau) \right| \lesssim e^{-\sqrt{2\pi\delta\beta N}}.$$

Here, the hidden constant depends only on  $F(s)$ ,  $\delta$  and  $\beta$ .

We will utilize Lemma 4.1 to derive a new exponential sums approximation to  $\frac{1}{t^{1+\alpha}}$ . To this end, we first rewrite it as an integral over  $\mathbb{R}$  of a analytic, improperly integrable function, which satisfies the conditions listed in Lemma 4.1.

**Theorem 4.1** (Approximation by exponential sums for long time). *For all  $\alpha > 0$  and  $N \in \mathbb{N}_+$ , there holds*

$$\left| \frac{1}{t^{1+\alpha}} - \tau \sum_{\ell=-N}^N F(\ell\tau; t) \right| \leq \left( \frac{(t - \alpha/e)^{-1} + 1}{1 - e^{-\pi\sqrt{\beta N}}} + \frac{2}{\beta} \right) e^{-\pi\sqrt{\beta N}} \text{ for all } t > \alpha. \quad (4.2)$$

Given  $t > 0$ ,  $F(s; t) : \mathbb{R} \rightarrow \mathbb{R}^+$  as a function of  $s \in \mathbb{R}$  is

$$F(s; t) := e^{-t \ln(1+e^s)} (\ln(1+e^s))^\alpha \frac{1}{1+e^{-s}}. \quad (4.3)$$

The step size  $\tau$  and the positive parameter  $\beta$  are

$$\begin{cases} \tau := \frac{\pi}{\sqrt{\beta N}} \\ \beta := \min\{1, t - \alpha\}. \end{cases} \quad (4.4)$$

*Proof.* For any  $t > 0$ , by the definition of Gamma function, we obtain

$$t^{-(1+\alpha)} = \frac{1}{\Gamma(\alpha+1)} \int_0^\infty e^{-tx} x^\alpha dx.$$

Then taking integration by parts and a change of variables yields,

$$\begin{aligned} t^{-(1+\alpha)} &= \frac{1}{\Gamma(\alpha+1)} \int_{-\infty}^{+\infty} e^{-t \ln(1+e^s)} (\ln(1+e^s))^\alpha \frac{1}{1+e^{-s}} ds \\ &= \frac{1}{\Gamma(\alpha+1)} \int_{-\infty}^{+\infty} F(s; t) ds. \end{aligned}$$

Here,  $F(s; t) : \mathbb{R} \rightarrow \mathbb{R}^+$  as a function of  $s$  parameterized by  $t \in (0, \infty)$  is defined in (4.3). In the following, we will verify that  $F(s; t)$  satisfies the conditions listed in Lemma 4.1 when  $t > \alpha$ .

On the one hand, it holds that

$$|F(s; t)| \lesssim e^{-(t-\alpha)|\Re s|}. \quad (4.5)$$

Indeed, Definition (4.3) implies

$$|F(s; t)| = \left| e^{-t \ln(1+e^s)} (\ln(1+e^s))^\alpha \frac{1}{1+e^{-s}} \right| \lesssim e^{-t \Re s} (\Re s)^\alpha \text{ for all } \Re s > 0.$$

This, together with the fact that

$$\ln x \leq \frac{x}{e} \text{ for all } x > 0,$$

leads to

$$|F(s; t)| \leq e^{-(t-\alpha)\Re s} \text{ for all } \Re s > 0.$$

One can observe that

$$|F(s; t)| \lesssim e^{-|\Re s|} \text{ for all } \Re s < 0.$$

Consequently, inequality (4.5) has been proved.

Note that  $F(s; t)$  is holomorphic in the strip  $\mathcal{D} := \{z \in \mathbb{C} : \Im z \in (-\pi/2, \pi/2)\}$ , and one can prove from inequality (4.5),

$$\int_{\partial \mathcal{D}} |F(s; t)| |ds| \lesssim \frac{1}{t - \alpha} + 1.$$

Consequently, all the requirement for Lemma 4.1 is fulfilled and the assertion follows from Lemma 4.1.  $\square$

Note that we attempt to use sum of exponentials to approximate the history component  $\partial_{\mathcal{F}}^{\alpha} v(x, t_{n+1})$ , and thus we demand an estimate (4.2) for  $t > \tau_f$  instead. This, nevertheless, can be derived by a scaling argument.

**Corollary 4.1.** *For any  $\epsilon > 0$  and  $\tau_f \in (0, 1)$ , there exist  $(\omega_j, \lambda_j)_{j=1}^{N_{\text{exp}}}$ , s.t.,*

$$\left| \frac{1}{t^{1+\alpha}} - \sum_{j=1}^{N_{\text{exp}}} \omega_j e^{-\lambda_j t} \right| \lesssim \epsilon \text{ for all } t \in (\tau_f, T), \quad (4.6)$$

with

$$N_{\text{exp}} \approx |\log(\tau_f^{1+\alpha} \epsilon)|^2.$$

*Proof.* Let  $t > \tau_f$ , then the new variable  $s := (\alpha + 1) \frac{t}{\tau_f} > \alpha + 1$ . In view of (4.2), we obtain for  $N \in \mathbb{N}$ ,

$$\left| \frac{1}{s^{1+\alpha}} - \tau \sum_{\ell=-N}^N F(\ell\tau; s) \right| \leq \left( \frac{2}{1 - e^{-\pi\sqrt{N}}} + 2 \right) e^{-\pi\sqrt{N}}$$

with  $\tau = \frac{\pi}{\sqrt{N}}$ .

Next, making a change of variable by  $t = \frac{s\tau_f}{\alpha+1}$  and using (4.3), lead to

$$\left| \frac{1}{t^{1+\alpha}} - \sum_{j=1}^{N_{\text{exp}}} \omega_j e^{-\lambda_j t} \right| \lesssim \tau_f^{-1-\alpha} \left( \frac{2}{1 - e^{-\pi\sqrt{\frac{N_{\text{exp}}-1}{2}}}} + 2 \right) e^{-\pi\sqrt{\frac{N_{\text{exp}}-1}{2}}} \text{ for all } t \in (\tau_f, \infty).$$

Here,

$$N_{\text{exp}} = 2N + 1,$$

$$\omega_j := \tau_f^{-1-\alpha} (\alpha + 1)^{1+\alpha} \frac{\pi}{\sqrt{N}} \left( \ln(1 + e^{(j+N+1)\tau}) \right)^{\alpha} \frac{1}{1 + e^{-(j+N+1)\tau}},$$

$$\lambda_j := \tau_f^{-1} (\alpha + 1) \ln(1 + e^{(j+N+1)\tau}).$$

Consequently, to guarantee

$$\left| \frac{1}{t^{1+\alpha}} - \sum_{j=1}^{N_{\text{exp}}} \omega_j e^{-\lambda_j t} \right| \lesssim \epsilon \text{ for all } t \in (\tau_f, \infty),$$

one only needs to choose  $N_{\text{exp}}$  sufficiently large, s.t.,

$$\tau_f^{-1-\alpha} \left( \frac{2}{1 - e^{-\pi \sqrt{\frac{N_{\text{exp}}-1}{2}}}} + 2 \right) e^{-\pi \sqrt{\frac{N_{\text{exp}}-1}{2}}} \lesssim \epsilon,$$

i.e.,

$$N_{\text{exp}} \approx |\log(\tau_f^{1+\alpha} \epsilon)|^2.$$

This completes the proof.  $\square$

In another word, a number of  $\mathcal{O}(|\log(\tau_f^{1+\alpha} \epsilon)|^2)$  terms is required to have an approximation error below a tolerance  $\epsilon > 0$ . Note that this result is independent of the final time. Further, one can observe that  $\lambda_j = \mathcal{O}(\tau_f^{-1})$  and  $\lambda_j > 0$  for all  $j = 1, \dots, N_{\text{exp}}$ . Denote

$$\gamma := \min_{j=1, \dots, N_{\text{exp}}} \{\lambda_j\}, \quad (4.7)$$

then  $\gamma \gg 1$  and  $\gamma = \mathcal{O}(\tau_f^{-1})$ .

**Remark 4.1** (Alternative approximation by exponential sums). *Another approximation by exponential sums are provided in [19, Theorem 2.5], For any  $\epsilon > 0$ , there exist  $(\omega_{1,j}, \lambda_{1,j})_{j=1}^{N_{1,\text{exp}}}$ , s.t.,*

$$\left| \frac{1}{t^{1+\alpha}} - \sum_{j=1}^{N_{1,\text{exp}}} \omega_{1,j} e^{-\lambda_{1,j} t} \right| \lesssim \epsilon \text{ for all } t \in (\tau_f, T), \quad (4.8)$$

with

$$N_{1,\text{exp}} \approx |\ln \epsilon| \left( \ln |\ln \epsilon| + \ln \frac{T}{\tau_f} \right) + |\ln \tau_f| \left( \ln |\ln \epsilon| + |\ln \tau_f| \right).$$

*Note that this approximation is only suitable for cases with a final time of reasonable size due to the term  $\ln \frac{T}{\tau_f}$  in the total number of terms  $N_{1,\text{exp}}$ .*

## 4.2 Local time integrators

We construct in this section local time integrators based upon the approximation by exponential sums developed in Corollary 4.1. The accuracy requirement of this exponential sums approximation is derived in (6.1), which implies that  $\epsilon = \mathcal{O}(T^{-1-\alpha})$ . The fine scale time step  $\tau_f$  can be very tiny due to the possible singularity of the solution  $u(\cdot, t)$  at the original  $t = 0$  due to, for example, insufficient regularity in the source term  $f$  or the initial data  $u_0$ . For this reason, we assume

$$\tau_f \ll \epsilon \quad (4.9)$$

for given tolerance  $\epsilon > 0$  throughout this paper.

Applying the summation of exponentials (Corollary 4.1) to the history component  $\partial_{\mathcal{F}}^\alpha$  after taking integration by parts, we obtain

$$\begin{aligned} & \bar{\partial}_{\tau_f}^\alpha v(x, t_{n+1}) \\ & \approx \frac{v(x, t_{n+1}) - v(x, t_n)}{\tau_f^\alpha c_\alpha} + \frac{1}{\Gamma(1-\alpha)} \left( \frac{v(x, t_n)}{\tau_f^\alpha} - \frac{v(x, t_0)}{t_{n+1}^\alpha} - \alpha \sum_{j=1}^{N_{\text{exp}}} \omega_j \int_0^{t_n} e^{-\lambda_j(t_{n+1}-s)} v(x, s) ds \right) \\ & = \underbrace{\frac{v(x, t_{n+1}) - v(x, t_n)}{\tau_f^\alpha c_\alpha}}_{=:\bar{\partial}_{\tau_f, \mathcal{N}}^\alpha v(x, t_{n+1})} + \underbrace{\frac{1}{\Gamma(1-\alpha)} \left( \frac{v(x, t_n)}{\tau_f^\alpha} - \frac{v(x, t_0)}{t_{n+1}^\alpha} - \alpha \sum_{j=1}^{N_{\text{exp}}} \omega_j \psi_j[n] \right)}_{=:\bar{\partial}_{\tau_f, \mathcal{F}}^\alpha v(x, t_{n+1})}. \end{aligned} \quad (4.10)$$

Here, the parameters  $\psi_j[n]$  for  $j = 1, \dots, N_{\text{exp}}$  are defined by

$$\psi_j[n] := \int_0^{t_n} e^{-\lambda_j(t_{n+1}-s)} v(x, s) ds.$$

which can be written as

$$\psi_j[n] = \int_0^{t_{n-1}} e^{-\lambda_j(t_{n+1}-s)} v(x, s) ds + \int_{t_{n-1}}^{t_n} e^{-\lambda_j(t_{n+1}-s)} v(x, s) ds.$$

This implies the following recurrence relation

$$\begin{aligned} \psi_j[n] &= e^{-\lambda_j(t_{n+1}-t_n)} \psi_j[n-1] + \int_{t_{n-1}}^{t_n} e^{-\lambda_j(t_{n+1}-s)} v(x, s) ds \text{ for } n \geq 1 \\ \psi_j[0] &= 0. \end{aligned}$$

Note that the integrals in the recurrence relation can be estimated by the L1 scheme, which yields

$$\begin{aligned} \int_{t_{n-1}}^{t_n} e^{-\lambda_j(t_{n+1}-s)} v(x, s) ds &\approx \int_{t_{n-1}}^{t_n} e^{-\lambda_j(t_{n+1}-s)} \left( \frac{t_n - s}{\tau_f} v(x, t_{n-1}) + \frac{s - t_{n-1}}{\tau_f} v(x, t_n) \right) ds \\ &=: c_{1,j}^{\tau_f} v(x, t_{n-1}) + c_{2,j}^{\tau_f} v(x, t_n), \end{aligned} \quad (4.11)$$

with

$$c_{1,j}^{\tau_f} := \frac{e^{-\lambda_j \tau_f}}{\lambda_j^2 \tau_f} \left( 1 - e^{-\lambda_j \tau_f} - \lambda_j \tau_f e^{-\lambda_j \tau_f} \right) \text{ and } c_{2,j}^{\tau_f} := \frac{e^{-\lambda_j \tau_f}}{\lambda_j^2 \tau_f} \left( -1 + e^{-\lambda_j \tau_f} + \lambda_j \tau_f \right). \quad (4.12)$$

Note also that the parameters  $c_{i,j}^{\tau_f}$  for  $i = 1, 2$  and  $j = 1, 2, \dots, N_{\text{exp}}$  rely only on  $\{\lambda_j\}_{j=1}^{N_{\text{exp}}}$  and the parameter  $\tau_f$ . Analogously, we can define  $c_{i,j}^{\tau_c}$  with a coarse step size  $\tau_c$ .

We establish an estimate of those parameters in the next lemma that provides a property of the history information.

**Lemma 4.2.** *Let  $c_{i,j}^\tau$  for  $i = 1, 2$  and  $j = 1, 2, \dots, N_{\text{exp}}$  be defined in (4.12) with  $\tau = \tau_f$  or  $\tau = \tau_c$ . Then it holds*

$$\left| \sum_{j=1}^{N_{\text{exp}}} \omega_j c_{i,j}^\tau \right| \leq e^{-\gamma \tau} \left( \tau^{-\alpha} (1-\alpha)^{-1} + \epsilon \tau / 2 \right).$$

*Proof.* A combination of (4.11) and (4.8) lead to

$$\begin{aligned}
\left| \sum_{j=1}^{N_{\text{exp}}} \omega_j c_{1,j}^\tau \right| &= \left| \sum_{j=1}^{N_{\text{exp}}} \omega_j \int_{t_{n-1}}^{t_n} e^{-\lambda_j(t_{n+1}-s)} \frac{t_n - s}{\tau_f} ds \right| \\
&\leq e^{-\gamma\tau} \left| \sum_{j=1}^{N_{\text{exp}}} \omega_j \int_0^\tau e^{-\lambda_j(\tau-s)} \frac{\tau - s}{\tau} ds \right| \\
&\leq e^{-\gamma\tau} \left( \int_0^\tau (\tau - s)^{-1-\alpha} \frac{\tau - s}{\tau} ds + \epsilon \int_0^\tau \frac{\tau - s}{\tau} ds \right) \\
&= e^{-\gamma\tau} (\tau^{-\alpha} (1 - \alpha)^{-1} + \epsilon\tau/2).
\end{aligned}$$

Recall that  $\gamma$  is defined in (4.7). This has proved the case for  $i = 1$ . The other case can be shown similarly.  $\square$

**Remark 4.2** (Efficiency of the new time stepping scheme (4.10)).  $\bar{\partial}_{\tau_c}^\alpha$  can be defined similarly by replacing the fine time steps with the coarse time steps. One can observe from formulas (4.10) that each calculation of  $\bar{\partial}_{\tau_c}^\alpha v(x, T^n)$  involves  $N_{\text{exp}}$  calculation and  $N_{\text{exp}}$  storage for  $n = 2, \dots, M_c$ . Note from (4.2) and (4.9) that  $N_{\text{exp}} \ll M_f$  and  $N_{\text{exp}}$  is independent of the final time  $T$  as  $T \rightarrow \infty$ . Comparing with the L1 scheme (2.2), the new time stepping scheme is much more efficient.

For the sake of conciseness, we will rewrite the time integrator as

$$\begin{aligned}
\bar{\partial}_{\tau_f}^\alpha v(x, t_{n+1}) &=: \frac{v(x, t_{n+1}) - v(x, t_n)}{\tau_f^\alpha c_\alpha} + \frac{1}{\Gamma(1 - \alpha)} \left( \frac{v(x, t_n)}{\tau_f^\alpha} - \frac{v(x, 0)}{t_{n+1}^\alpha} - \alpha \sum_{j=1}^{N_{\text{exp}}} \omega_j \psi_j[n] \right) \\
\psi_j[n+1] &= \mathcal{H}(\psi_j[n]; \tau_f, v(x, t_n), v(x, t_{n+1})) \text{ for } n \geq 0 \text{ and } \psi_j[0] = 0 \text{ for all } j = 1, \dots, N_{\text{exp}}.
\end{aligned} \tag{4.13}$$

Here, the propagation operator of the history information is

$$\mathcal{H}(\psi_j[n]; \tau_f, v(x, t_n), v(x, t_{n+1})) := e^{-\lambda_j \tau_f} \psi_j[n] + c_{1,j}^{\tau_f} v(x, t_n) + c_{2,j}^{\tau_f} v(x, t_{n+1}). \tag{4.14}$$

### 4.3 Multiscale-SOE-scheme

In this section, we formulate the numerical solution based on the time integrator (4.13) in the multiscale ansatz space  $V_{\text{ms},\ell}$ . The multiscale-SOE-scheme aims to seeking  $U_{\text{ms},\ell}^n \in V_{\text{ms},\ell}$  for  $n = 1, \dots, M_f$ , satisfying

$$\left\{ \begin{aligned}
&(U_{\text{ms},\ell}^n, v_{\text{ms}})_D + c_\alpha \tau_f^\alpha a(U_{\text{ms},\ell}^n, v_{\text{ms}}) = \alpha (U_{\text{ms},\ell}^{n-1}, v_{\text{ms}})_D + \alpha (1 - \alpha) \tau_f^\alpha \sum_{j=1}^{N_{\text{exp}}} \omega_j (\psi_j[n-1], v_{\text{ms}})_D \\
&\quad + \frac{1 - \alpha}{(n+1)^\alpha} (U_{\text{ms},\ell}^0, v_{\text{ms}})_D + c_\alpha \tau_f^\alpha (\mathcal{P}_\ell f(\cdot, t_n), v_{\text{ms}})_D \quad \text{for all } v_{\text{ms}} \in V_{\text{ms},\ell} \\
&\psi_j[n] = \mathcal{H}(\psi_j[n-1]; \tau_f, U_{\text{ms},\ell}^{n-1}, U_{\text{ms},\ell}^n) \text{ for } n \geq 0 \text{ and } \psi_j[0] = 0 \text{ for all } j = 1, \dots, N_{\text{exp}} \\
&U_{\text{ms},\ell}^0 = \mathcal{P}_\ell u_0.
\end{aligned} \right. \tag{4.15}$$

For the sake of the convergence analysis, we require a semidiscrete Galerkin scheme in the multiscale ansatz space  $V_{\text{ms},\ell}$ , defined by seeking  $u_{\text{ms},\ell}(t) \in V_{\text{ms},\ell}$ , satisfying

$$\begin{cases} (\partial_t^\alpha u_{\text{ms},\ell}, v_{\text{ms}})_D + a(u_{\text{ms},\ell}, v_{\text{ms}}) = (f, v_{\text{ms}})_D & \text{for all } v_{\text{ms}} \in V_{\text{ms},\ell} \\ u_{\text{ms}}(x, 0) = \mathcal{P}_\ell u_0. \end{cases} \quad (4.16)$$

## 5 Wavelet-based Edge Multiscale Parareal Algorithm

We construct in this section the Wavelet-based Edge Multiscale Parareal (WEMP) Algorithm. To this end, we first recap a few terminologies commonly appeared in parareal algorithm. The one step coarse solver on the coarse subinterval  $(T^n, T^{n+1})$  is

$$\begin{bmatrix} U_{\text{ms},\ell}^{n+1} \\ \Phi_{\text{ms},\ell}^{n+1} \end{bmatrix} = \mathcal{G}^{\text{ms},\ell}(T^n, U_{\text{ms},\ell}^n; \Phi_{\text{ms},\ell}^n) \quad (5.1)$$

which yields  $U_{\text{ms},\ell}^{n+1} \in V_{\text{ms},\ell}$  as a coarse approximation to  $u(\cdot, T^{n+1})$ , provided with an approximation  $U_{\text{ms},\ell}^n$  of  $u(\cdot, T^n)$ . Here,  $\Phi_{\text{ms},\ell}^n$  serves as the input parameter to calculate  $U_{\text{ms},\ell}^{n+1}$ , which represents the memory effect from the previous time steps.

In matrix form, it reads

$$\begin{cases} U_{\text{ms},\ell}^{n+1} = \left( \frac{\tau_c^{-\alpha}}{\Gamma(2-\alpha)} M_{\text{ms},\ell} + A_{\text{ms},\ell} \right)^{-1} M_{\text{ms},\ell} \left( \tau_c^{-\alpha} \frac{\alpha}{\Gamma(2-\alpha)} U_{\text{ms},\ell}^n + \frac{\alpha}{\Gamma(1-\alpha)} \Theta^T \Phi_{\text{ms},\ell}^n + \frac{1}{\Gamma(1-\alpha) T^{n+1}} U_{\text{ms},\ell}^0 + F_h^{n+1} \right) \\ \Phi_{\text{ms},\ell}^{n+1} = D^{\tau_c} \Phi_{\text{ms},\ell}^n + \mathbf{C}_1^{\tau_c} U_{\text{ms},\ell}^n + \mathbf{C}_2^{\tau_c} U_{\text{ms},\ell}^{n+1}. \end{cases}$$

Here, we use the following notations,

$$\begin{cases} \Phi_{\text{ms},\ell}^n := [\psi_1[n], \psi_2[n], \dots, \psi_{N_{\text{exp}}}[n]] \\ \Theta = [\omega_1, \omega_2, \dots, \omega_{N_{\text{exp}}}]^T \\ \mathbf{C}_i^\tau = [c_{i,1}^\tau, c_{i,2}^\tau, \dots, c_{i,N_{\text{exp}}}^\tau]^T & \text{for } i = 1, 2 \\ D^\tau = \text{diag}(e^{-\lambda_1 \tau}, e^{-\lambda_2 \tau}, \dots, e^{-\lambda_{N_{\text{exp}}} \tau}) \\ M_{\text{ms},\ell} = \Psi_{\text{ms},\ell}^T M_h \Psi_{\text{ms},\ell} \text{ and } A_{\text{ms},\ell} = \Psi_{\text{ms},\ell}^T A_h \Psi_{\text{ms},\ell}, \end{cases}$$

where  $M_h$  and  $A_h$  are the mass matrix and stiffness matrix corresponding to the discretization of the elliptic operator  $-\nabla \cdot (\kappa \nabla \cdot)$  in the finite element space  $V_h := \text{span}\{\phi_1, \dots, \phi_{\text{dof}_h}\}$ ,  $\text{dof}_h$  denotes the dimension of  $V_h$ .  $(F_h^{n+1})_i := \int_D f(\cdot, t_{n+1}) \phi_i \, dx$  for all  $i = 1, \dots, \text{dof}_h$ ,  $\Psi_{\text{ms},\ell}$  denotes a matrix with columns composed of the coefficients of multiscale basis functions in  $V_{\text{ms},\ell}$  in the finite element space  $V_h$ . Let  $\text{dof}_\ell$  be the dimension of the multiscale space  $V_{\text{ms},\ell}$ . Note that  $\Phi_{\text{ms},\ell}^n$  is vector-valued function with  $N_{\text{exp}}$  components, and the length of each component is the dimension of the multiscale ansatz space  $V_{\text{ms},\ell}$ .

The one step fine solver on the coarse subinterval  $(T^n, T^{n+1})$  is

$$\begin{bmatrix} u_{\text{ms},\ell}^{n+1} \\ \Phi_{\text{ms},\ell}^{n+1} \end{bmatrix} = \mathcal{F}_f^{\text{ms},\ell}(T^n, U_{\text{ms},\ell}^n; \Phi_{\text{ms},\ell}^n), \quad (5.2)$$



which is defined in analogous to the one step coarse solver (5.1), but with a uniform discrete time step  $\tau_f$ .

Analogously, we can define the semidiscrete Galerkin approximation in  $V_{\text{ms},\ell}$

$$\begin{bmatrix} v_{\text{ms},\ell} \\ \Phi_1 \end{bmatrix} = \mathcal{F}^{\text{ms},\ell}(T^n, U_{\text{ms},\ell}^n; \Phi_{\text{ms},\ell}^n). \quad (5.3)$$

Time stepping schemes with  $\bar{\cdot}$  on top refers to the same scheme with a vanishing source term, for example,  $\bar{\mathcal{G}}^{\text{ms},\ell}(T^n, U_{\text{ms},\ell}^n; \Phi_0)$  denotes the one step coarse solver with  $f = 0$ .  $\bar{\mathcal{F}}^{\text{ms},\ell}(T^n, U_{\text{ms},\ell}^n; \Phi_{\text{ms},\ell}^n)$  and  $\bar{\mathcal{F}}_f^{\text{ms},\ell}(T^n, U_{\text{ms},\ell}^n; \Phi_{\text{ms},\ell}^n)$  are defined analogously.

In the parareal algorithm 1, the cheap coarse time stepping scheme  $\mathcal{G}^{\text{ms},\ell}$  is sequentially utilized over the global time interval  $I$  to provide a rough approximation to  $u(\cdot, T^{n+1})$ , while the expensive fine time stepping scheme  $\mathcal{F}_f^{\text{ms},\ell}$  is applied in each subinterval  $[T^n, T^{n+1}]$  for  $n = 0, 1, \dots, M_c - 1$  in parallel to provide a more accurate solution. They do not agree with each other, and the discrepancy between them is denoted as the correction operator,

$$\mathcal{S}(T^n, U_{\text{ms},\ell}^n; \Phi_{\text{ms},\ell}^n) := \mathcal{F}_f^{\text{ms},\ell}(T^n, U_{\text{ms},\ell}^n; \Phi_{\text{ms},\ell}^n)_1 - \mathcal{G}^{\text{ms},\ell}(T^n, U_{\text{ms},\ell}^n; \Phi_{\text{ms},\ell}^n)_1. \quad (5.4)$$

Here,  $\cdot_1$  denotes taking the first component.

In comparison with recently proposed parareal algorithm for time-fractional diffusion problems [38], our new proposed algorithm corrects only the solution  $u(\cdot, T^{n+1})$ . Its associated history component  $\Phi_{\text{ms},\ell}^{n+1}$  will be updated by (4.14) using the corrected solution  $u(\cdot, T^{n+1})$ . Extensive numerical tests shows that our proposed algorithm converges to the fine-scale solution within three iterations.

Next, we are ready to present our main algorithm, i.e., Algorithm 3. Its main objective is to obtain a good approximation to the exact solution  $u$  of Problem (1.1) at the discrete coarse time step  $T^n$  for  $n = 1, 2, \dots, M_c$ , albeit that the time stepping scheme is built upon a coarse discretization. This relies on a parallel correction operator defined by (5.4).

We can solve (4.15) using the coarse-scale time stepping scheme as the initial guess  $\begin{bmatrix} U_0^n \\ \Phi_0^n \end{bmatrix}$  for  $n = 0, \dots, M_c$ . Given the iteration parameter  $k$ , we apply the local coarse solver (5.1) in Step 4 to obtain  $U_k^{n+1}$ . Using the coarse solution  $U_k^n$  as the initial condition, the fine solver (5.2) subsequently is used to calculate the fine solution  $u_k^{n+1}$  in parallel on each local time subinterval  $[T^n, T^{n+1}]$ . Then we calculate the discrepancy between the coarse solution and the fine solution in Step 6 on each discrete coarse time point  $T^n$  for  $n = 1, 2, \dots, M_c$ , and denote it as  $\mathcal{S}(T^{n-1}, U_k^{n-1}; \Phi_{\text{ms},\ell}^{n-1})$ . Subsequently, this jump term is utilized in Step 7 to correct the coarse solution calculated by the global coarse solver (5.1). This process will be performed iteratively until certain tolerance on the jump terms is satisfied.

## 6 Error analysis

This section is concerned with the convergence analysis for the multiscale-SOE-scheme (4.15) and the WEMP Algorithm 3.

### 6.1 Error analysis for the multiscale-SOE-scheme (4.15)

To derive the error estimate for Problem (4.15), we start with a regularity property of the semidiscrete Galerkin approximation solution  $u_{\text{ms},\ell}$  in Lemma 6.1 that is standard in the numerical analysis, see, e.g., [20]. We then show in Lemma 6.2 that the local time stepping scheme based on summation of exponentials

$\bar{\partial}_t^\alpha$  is a small perturbation of the L1 scheme  $\tilde{\partial}_t^\alpha$ . Based upon this, we establish the stability estimate of Problem (4.15). Finally, the error estimate is presented in Theorem 6.1.

---

**Algorithm 3:** Wavelet-based Edge Multiscale Parareal (WEMP) Algorithm

---

**Data:** Multiscale space  $V_{\text{ms},\ell}$ ; initial data  $u_0$ ; source term  $f$ ; tolerance  $\delta$  and initial solution  $\begin{bmatrix} U_0^n \\ \Phi_0^n \end{bmatrix}$

for  $n = 0, \dots, M_c$ .

**Result:**  $U_k^n$  for  $n = 0, \dots, M_c$ .

1:  $k = 1$ .

2: **do**

3: Set the initial value:  $\begin{bmatrix} U_k^0 \\ \Phi_k^0 \end{bmatrix} = \begin{bmatrix} \mathcal{P}_\ell u_0 \\ 0 \end{bmatrix}$ .

4: Parallel compute  $U_k^n$  on each local time subinterval  $[T^{n-1}, T^n]$  for  $n = 1, \dots, M_c$ ,

$$U_k^n = \mathcal{G}^{\text{ms},\ell}(T^n, U_{k-1}^{n-1}; \Phi_{k-1}^{n-1})_1.$$

5: Parallel compute  $u_k^n$  on each local time subinterval  $[T^{n-1}, T^n]$  for  $n = 1, \dots, M_c$ ,

$$u_k^n = \mathcal{F}_f^{\text{ms},\ell}(T^{n-1}, U_{k-1}^{n-1}; \Phi_{k-1}^{n-1})_1.$$

6: Parallel compute the solution jumps for  $n = 1, \dots, M_c$ ,

$$\mathcal{S}(T^{n-1}, U_{k-1}^{n-1}; \Phi_{k-1}^{n-1}) := u_k^n - U_k^n.$$

7: Sequentially compute the corrected coarse solutions  $\begin{bmatrix} U_k^n \\ \Phi_k^n \end{bmatrix}$  for  $n = 1, \dots, M_c$ ,

$$U_k^n = \mathcal{S}(T^{n-1}, U_{k-1}^{n-1}; \Phi_{k-1}^{n-1}) + \mathcal{G}^{\text{ms},\ell}(T^{n-1}, U_k^{n-1}; \Phi_k^{n-1})_1$$

$$\Phi_k^n = \mathcal{H}(\Phi_k^{n-1}; \tau_c, U_k^{n-1}, U_k^n).$$

8: Estimate the error,

$$\text{err} := 1/M_c \sum_{n=1}^{M_c} \|U_k^n - U_{k-1}^n\|_{\ell_2}.$$

9:  $k \leftarrow k + 1$ .

10: **while**  $\text{err} > \delta$

---

We begin with establishing a certain regularity result of the semidiscrete Galerkin approximation solution  $u_{\text{ms},\ell}$  with respect to the temporal variable  $t$ , which is similar to [20, Assumption 3.7].

**Lemma 6.1.** *Let  $u_{\text{ms},\ell}$  be the solution to (4.16), then there holds*

$$\begin{aligned} \|u_{\text{ms},\ell}(\cdot, t)\|_D &\leq C_0(\|f\|_{W^{2,\infty}(I, L^2(D))} + \|u_0\|_D) \\ \|\partial_t^m u_{\text{ms},\ell}(\cdot, t)\|_D &\leq C_0 t^{\sigma\alpha - m} (\|f\|_{W^{2,\infty}(I, L^2(D))} + |u_0|_{2\sigma}) \text{ for } m \geq 1 \end{aligned} .$$

Here, and throughout this paper,  $C_0$  denotes a positive constant that is independent of  $H$ ,  $h$ ,  $\tau_c$  and  $\tau_f$  with varying value from context to context.

Next we present the truncation error estimate of our proposed time integrator  $\bar{\partial}_{\tau_f}^\alpha$ :

**Lemma 6.2.** *Let  $u_{ms,\ell}$  be the solution to (4.16), then there holds*

$$\left\| \partial_t^\alpha u_{ms,\ell}(\cdot, t_n) - \bar{\partial}_{\tau_f}^\alpha u_{ms,\ell}(\cdot, t_n) \right\|_D \leq C_0 \left( \min\{(n-1)^{-\alpha}, 1\} \tau_f^{-(1-\sigma)\alpha} + \epsilon \sqrt{t_{n-1}} \right) \times (|u_0|_{2\sigma} + \|f\|_{W^{2,\infty}(I, L^2(D))}).$$

*Proof.* An application of the triangle inequality implies

$$\left\| \partial_t^\alpha u_{ms,\ell}(\cdot, t_n) - \bar{\partial}_{\tau_f}^\alpha u_{ms,\ell}(\cdot, t_n) \right\|_D \leq \left\| \partial_t^\alpha u_{ms,\ell}(\cdot, t_n) - \tilde{\partial}_{\tau_f}^\alpha u_{ms,\ell}(\cdot, t_n) \right\|_D + \left\| \tilde{\partial}_{\tau_f}^\alpha u_{ms,\ell}(\cdot, t_n) - \bar{\partial}_{\tau_f}^\alpha u_{ms,\ell}(\cdot, t_n) \right\|_D.$$

The first term can be estimated by [20, Lemma 3.9], and a direct calculation leads to the estimate of the second term.  $\square$

**Lemma 6.3** (Stability of Problem (4.15)). *For any given positive parameter  $\eta \in (0, 1)$ , let  $\epsilon$  be sufficiently small, s.t.,*

$$\epsilon \leq \min \left\{ \frac{\alpha M_f^\alpha}{T^{1+\alpha}} \eta, \frac{1}{2T^{1+\alpha}} \right\}. \quad (6.1)$$

*Then it holds*

$$\tau_f^{\alpha/2} |U_{ms,\ell}^n|_1 + \|U_{ms,\ell}^n\|_D \lesssim \|\mathcal{P}_\ell u_0\|_D + (1 + \eta) c_\alpha \tau_f^\alpha \sum_{k=0}^{n-1} \|\mathcal{P}_\ell f(\cdot, t_{k+1})\|_D.$$

*Moreover, there holds for any  $\sigma \in (0, 1/2)$  that*

$$\tau_f^{\sigma\alpha} |U_{ms,\ell}^n|_{2\sigma} \lesssim \|\mathcal{P}_\ell u_0\|_D + (1 + \eta) c_\alpha \tau_f^\alpha \sum_{k=0}^{n-1} \|\mathcal{P}_\ell f(\cdot, t_{k+1})\|_D.$$

*Proof.* On the one hand, we can derive by (4.1) and (4.10),

$$\bar{\partial}_{\tau_f}^\alpha U_{ms,\ell}^{n+1} = \tilde{\partial}_{\tau_f}^\alpha U_{ms,\ell}^{n+1} + \frac{1}{\tau_f^\alpha c_\alpha} \sum_{j=1}^n \left( a_{n+1,j} U_{ms,\ell}^{j-1} + b_{n+1,j} U_{ms,\ell}^j \right)$$

with

$$|a_{n+1,j}|, |b_{n+1,j}| \leq \frac{1}{2} \alpha (1 - \alpha) \tau_f^{1+\alpha} \epsilon.$$

Together with (2.2), this yields

$$\bar{\partial}_{\tau_f}^\alpha U_{ms,\ell}^{n+1} = \frac{1}{\tau_f^\alpha c_\alpha} \left( v_{n+1} - \sum_{j=1}^n (b_{j-1} - b_j - \xi_j) U_{ms,\ell}^{n+1-j} \right)$$

with

$$|\xi_j| \leq \alpha (1 - \alpha) \tau_f^{1+\alpha} \epsilon.$$

On the other hand, [20, Lemma 3.4] implies

$$\sum_{j=1}^n (b_{j-1} - b_j) (n+1-j)^{\alpha-1} \leq (n+1)^{\alpha-1},$$

which, combining with (6.1), yields

$$\begin{aligned}
\sum_{j=1}^n (b_{j-1} - b_j - \xi_j)(n+1-j)^{\alpha-1} &\leq (n+1)^{\alpha-1} + \tau_f^{1+\alpha} \epsilon \sum_{j=1}^n (n+1-j)^{\alpha-1} \\
&\leq (n+1)^{\alpha-1} + \tau_f^{1+\alpha} \epsilon \int_0^n (n+1-x)^{\alpha-1} dx \\
&= (n+1)^{\alpha-1} + \tau_f^{1+\alpha} \epsilon \alpha^{-1} (n+1)^\alpha \\
&\leq (1+\eta)(n+1)^{\alpha-1}.
\end{aligned}$$

Then the desired result follows from an energy argument by mathematical induction analogous to the proof to [20, Theorem 3.5].  $\square$

Note that the stability of Problem (4.15) using L2 time stepping scheme and summation of exponential approximation was derived in [39], which imposed a more restrictive constraint on  $\epsilon$  than our proposed requirement (6.1).

Then a combination of Lemma 6.2 and the approximation properties (3.7) and (3.9) implies

**Theorem 6.1** (Convergence rate of Problem (4.15)). *Assume that (6.1) holds. Let  $f \in W^{2,\infty}(I; L^2(D))$  and  $u_0 \in \dot{H}^{2\sigma}(D)$  with  $0 < \sigma \leq 1$ , then there holds*

$$\begin{aligned}
\|u(\cdot, t_n) - U_{ms,\ell}^n\|_D &\lesssim \left( \tau_f^{\sigma\alpha} + \eta(H, \ell) |\log \eta(H, \ell)| \right) \|f\|_{W^{2,\infty}(I; L^2(D))} \\
&\quad + \left( \tau_f^{\sigma\alpha} + \eta(H, \ell) |\log \eta(H, \ell)| t_n^{-\alpha(1-\sigma)} \right) |u_0|_{2\sigma}.
\end{aligned}$$

*Proof.* The proof is analogous to [20, Theorems 3.10 and 3.11].  $\square$

## 6.2 Convergence analysis for WEMP Algorithm 3

This section is concerned with the convergence of our main algorithm (Algorithm 3). We will first derive the approximation properties of the one step coarse time stepping scheme  $\mathcal{G}^{ms,\ell}$  in the first component and the jump operator  $\mathcal{S}$  in Lemma 6.4. The convergence of Algorithm 3 is derived by the mathematical induction on the solution itself, based upon which we can derive the convergence rate of the history information using (4.14). This will lead to the inequality (6.15), and then the proof follows from [24, Theorem 4.1].

To start with, we first introduce a new norm  $\|\!\|\!\| \cdot \|\!\|\!\|$  to measure the effect of the history component.

**Definition 6.1.** *Let  $\Psi := [\psi_1, \dots, \psi_{N_{exp}}]$  with each component  $\psi_j \in L^2(D)$ . Then we define its associated history contribution by  $v_{his} := \sum_{j=1}^{N_{exp}} \omega_j \psi_j$ . This vector  $\Psi$  is measured by the following norm*

$$\|\!\|\!\| \Psi \|\!\|\!\| := \tau_c^\alpha \|v_{his}\|_D.$$

We next prove the boundedness and Lipschitz continuity properties of the first component of the coarse solver  $\mathcal{G}^{ms,\ell}$  and the jump operator  $\mathcal{S}$  in the multiscale space  $V_{ms,\ell}$ :

**Lemma 6.4.** *Let  $\begin{bmatrix} U_i^0 \\ \Phi_i^0 \end{bmatrix} = \begin{bmatrix} v_0 \\ 0 \end{bmatrix}$  for  $i = 1, 2$  with  $v_0 \in V_{ms,\ell}$  being the initial data, and let  $v_i, \phi_{i,j} \in V_{ms,\ell}$  with  $i = 1, 2$  and  $j = 1, 2, \dots, N_{exp}$ . Denote  $\Phi_i := [\phi_{i,1}, \dots, \phi_{i,N_{exp}}]$  for  $i = 1, 2$  that represents the history information at the previous iteration. Then for all  $n \in \{1, \dots, M_c - 1\}$ , the following properties hold.*

1. The one step coarse solver  $\mathcal{G}^{\text{ms},\ell}$  is Lipschitz in  $V_{\text{ms},\ell}$ . There holds

$$\left\| \mathcal{G}^{\text{ms},\ell}(T^n, v_1; \Phi_1)_1 - \mathcal{G}^{\text{ms},\ell}(T^n, v_2; \Phi_2)_1 \right\|_D \leq \alpha \|v_1 - v_2\|_D + \alpha(1 - \alpha) \|\Phi_1 - \Phi_2\|.$$

2. The jump operator  $\mathcal{S}$  has the following approximation property. For any  $\sigma \in (0, 1)$ , it holds

$$\|\mathcal{S}(T^n, v_1; \Phi_1) - \mathcal{S}(T^n, v_2; \Phi_2)\|_D \leq C_0 \tau_c^{\sigma\alpha} \left( |v_1 - v_2|_{2\sigma} + \alpha \|\Phi_1 - \Phi_2\| \right). \quad (6.2)$$

*Proof.* 1. Let  $e_{\text{ms}}^{n+1} := \mathcal{G}^{\text{ms},\ell}(T^n, v_1; \Phi_1)_1 - \mathcal{G}^{\text{ms},\ell}(T^n, v_2; \Phi_2)_1$  and let  $v_{i,\text{his}} := \sum_{j=1}^{N_{\text{exp}}} \omega_j \phi_{i,j}$  for  $i = 1, 2$ , then it holds for all multiscale test function  $w_{\text{ms}} \in V_{\text{ms},\ell}$ ,

$$\begin{aligned} \int_D e_{\text{ms}}^{n+1} w_{\text{ms}} \, dx + c_\alpha \tau_c^\alpha \int_D \kappa \nabla e_{\text{ms}}^{n+1} \cdot \nabla w_{\text{ms}} \, dx &= \alpha \int_D (v_1 - v_2) w_{\text{ms}} \, dx \\ &+ \alpha(1 - \alpha) \tau_c^\alpha \int_D (v_{1,\text{his}} - v_{2,\text{his}}) w_{\text{ms}} \, dx. \end{aligned}$$

Choosing  $w_{\text{ms}} := e_{\text{ms}}^{n+1}$  leads to

$$\begin{aligned} \|e_{\text{ms}}^{n+1}\|_D^2 + c_\alpha \tau_c^\alpha \|e_{\text{ms}}^{n+1}\|_{H_k^1(D)}^2 &= \alpha \int_D (v_1 - v_2) e_{\text{ms}}^{n+1} \, dx \\ &+ \alpha(1 - \alpha) \tau_c^\alpha \int_D e_{\text{ms}}^{n+1} (v_{1,\text{his}} - v_{2,\text{his}}) \, dx. \end{aligned}$$

Then an application of Young's inequality implies

$$\|e_{\text{ms}}^{n+1}\|_D^2 \leq \alpha \|v_1 - v_2\|_D \|e_{\text{ms}}^{n+1}\|_D + \alpha(1 - \alpha) \tau_c^\alpha \|v_{1,\text{his}} - v_{2,\text{his}}\|_D \|e_{\text{ms}}^{n+1}\|_D.$$

The a direct calculation proves the first assertion.

2. To prove the second assertion, let

$$\begin{aligned} e^{n+1} &:= \mathcal{S}(T^n, v_1; \Phi_1) - \mathcal{S}(T^n, v_2; \Phi_2) \\ &= \left( \mathcal{F}_f^{\text{ms},\ell}(T^n, v_1; \Phi_1) - \mathcal{F}_f^{\text{ms},\ell}(T^n, v_2; \Phi_2) \right) - \left( \mathcal{G}^{\text{ms},\ell}(T^n, v_1; \Phi_1) - \mathcal{G}^{\text{ms},\ell}(T^n, v_2; \Phi_2) \right) \\ &= \bar{\mathcal{F}}_f^{\text{ms},\ell}(T^n, v_1 - v_2; \Phi_1 - \Phi_2) - \bar{\mathcal{G}}^{\text{ms},\ell}(T^n, v_1 - v_2; \Phi_1 - \Phi_2) \\ &= \left( \bar{\mathcal{F}}_f^{\text{ms},\ell}(T^n, v_1 - v_2; \Phi_1 - \Phi_2) - \bar{\mathcal{F}}^{\text{ms},\ell}(T^n, v_1 - v_2; \Phi_1 - \Phi_2) \right) \\ &\quad - \left( \bar{\mathcal{G}}^{\text{ms},\ell}(T^n, v_1 - v_2; \Phi_1 - \Phi_2) - \bar{\mathcal{F}}^{\text{ms},\ell}(T^n, v_1 - v_2; \Phi_1 - \Phi_2) \right) \\ &=: e_f^{n+1} - e_c^{n+1}. \end{aligned}$$

To estimate  $e^{n+1}$ , we only need to derive the estimate for  $e_f^{n+1}$  and  $e_c^{n+1}$ , separately.

Firstly, we introduce the following semidiscrete Galerkin approximation problem in the spatial domain  $D$ : let  $v := \bar{\mathcal{F}}^{\text{ms},\ell}(T^n, v_1 - v_2; \Phi_1 - \Phi_2)_1 \in L^2([T^n, T^{n+1}]; V_{\text{ms},\ell})$  satisfy

$$\begin{aligned} \forall w_{\text{ms}} \in V_{\text{ms},\ell} : \int_D \partial_N^\alpha v w_{\text{ms}} \, dx + \int_D \kappa \nabla v \cdot \nabla w_{\text{ms}} \, dx &= -\frac{1}{\tau_c^\alpha \Gamma(1 - \alpha)} \int_D (v_1 - v_2) w_{\text{ms}} \, dx \\ &+ \frac{\alpha}{\Gamma(1 - \alpha)} \int_D (v_{1,\text{his}} - v_{2,\text{his}}) w_{\text{ms}} \, dx \\ v(\cdot, T^n) &= v_1 - v_2. \end{aligned} \quad (6.3)$$

Note that the error equation for  $e_c^{n+1}$  derived from a fully discretization problem with L1 scheme in the temporal domain is

$$\forall w_{\text{ms}} \in V_{\text{ms},\ell} : \int_D \bar{\partial}_{\tau_c, \mathcal{N}}^\alpha e_c^{n+1} w_{\text{ms}} dx + \int_D \kappa \nabla e_c^{n+1} \cdot \nabla w_{\text{ms}} dx = \int_D w_0 \cdot w_{\text{ms}} dx \quad (6.4)$$

$$e_c^n = 0.$$

with  $w_0$  being the approximation error from L1 scheme,

$$w_0 := -\partial_{\mathcal{N}}^\alpha v(\cdot, T^{n+1}) + \bar{\partial}_{\tau_c, \mathcal{N}}^\alpha v(\cdot, T^{n+1}),$$

which can be estimated by Lemma 6.2,

$$\|w_0\|_D \leq C_0 \tau_c^{-(1-\sigma)\alpha} \left( |v_1 - v_2|_{2\sigma} + \alpha \|\Phi_1 - \Phi_2\| \right).$$

The stability estimate [20, Theorem 3.5] implies

$$\|e_c^{n+1}\|_D \leq C_0 \tau_c^{\sigma\alpha} \left( |v_1 - v_2|_{2\sigma} + \alpha \|\Phi_1 - \Phi_2\| \right).$$

Analogously, we can obtain the estimate for  $e_f^{n+1}$ , which reads

$$\|e_f^{n+1}\|_D \leq C_0 \tau_f^\alpha \tau_c^{\sigma\alpha - \alpha} \left( |v_1 - v_2|_{2\sigma} + \alpha \tau_f^\alpha \tau_c^{-\alpha} \|\Phi_1 - \Phi_2\| \right).$$

Note that  $\tau_f \ll \tau_c$ , then a combination of the two estimates above with the triangle inequality, shows the second assertion.  $\square$

**Remark 6.1.** *The key idea to prove the approximation property of the jump operator  $\mathcal{S}$  lies in the auxiliary problem (6.3), which is a time-fractional problem defined in the subdomain  $[T^n, T^{n+1}]$ . Together with the history information on  $[0, T^n]$  from the terms  $\Phi_1$  and  $\Phi_2$ , this results in a time-fraction problem on  $[T^n, T^{n+1}]$ , with both  $\bar{\mathcal{F}}_f^{ms,\ell}(T^n, v_1 - v_2; \Phi_1 - \Phi_2)$  and  $\bar{\mathcal{G}}^{ms,\ell}(T^n, v_1 - v_2; \Phi_1 - \Phi_2)$  as its temporal discretization under time step size of  $\tau_f$  and  $\tau_c$ , respectively.*

We present in the following a smoothing property of the SOE-scheme (4.13) that plays a crucial role in the convergence analysis with rough initial data. This property has been utilized without proof in [24, Assumption 4.1]. This property is proved by first establishing the global fully discretized scheme for  $U_k^n$ , and then derive the global error equation between the multiscale SOE scheme (4.15) and Algorithm 3 to eliminate the source term  $f$ .

**Lemma 6.5** (Smoothing property of the SOE-scheme (4.13)). *Let Assumption 3.1 hold. Assume that the source term  $f \in W^{2,\infty}(I; L^2(D))$  and initial data  $u_0 \in \dot{H}^{2\sigma}(D)$  for some  $0 < \sigma \leq 1$ . Let  $\ell \in \mathbb{N}_+$  be the level parameter. The coarse time step size and fine time step size are  $\tau_c$  and  $\tau_f$ . Let  $U_{\text{ms},\ell}^m \in V_{\text{ms},\ell}$  be the solution to Problem (4.15) and let  $U_k^n$  be the solution from Algorithm 3 with iteration  $k \in \mathbb{N}_+$ . Then for any  $\sigma \in (0, 1)$ , it holds*

$$\|U_{\text{ms},\ell}^m - U_k^n\|_{2\sigma} \lesssim (T^n)^{-\alpha\sigma} \left\| u_{\text{ms},\ell}^{m'} - U_k^{n-1} \right\|_D. \quad (6.5)$$

*Proof.* Note that Algorithm 3 is equivalent to seeking  $U_k^n \in V_{\text{ms},\ell}$  for  $n = 1, \dots, M_c$ , satisfying

$$\begin{cases} \bar{\partial}_{\tau_c}^\alpha (U_k^n, v_{\text{ms}})_D + a(U_k^n, v_{\text{ms}}) = (\mathcal{P}_\ell f(\cdot, T^n), v_{\text{ms}})_D + \bar{\partial}_{\tau_c}^\alpha (S_{k-1}^n, v_{\text{ms}})_D + a(S_{k-1}^n, v_{\text{ms}}) & \text{for all } v_{\text{ms}} \in V_{\text{ms},\ell} \\ \Phi_{k,j}^n = \mathcal{H}(\Phi_{k,j}^{n-1}; \tau_c, U_k^{n-1}, U_k^n) & \text{for } n \geq 0 \text{ and } \Phi_{k,j}^0 = 0 \text{ for all } j = 1, \dots, N_{\text{exp}} \\ U_k^0 = \mathcal{P}_\ell u_0. \end{cases} \quad (6.6)$$

Here, the local correction  $S_{k-1}^n := \mathcal{S}(T^{n-1}, U_{k-1}^{n-1}; \Phi_{k-1}^{n-1})$ . Note that for all  $v_{\text{ms}} \in V_{\text{ms},\ell}$ , it holds

$$(\bar{\partial}_{\tau_c}^\alpha S_{k-1}^n, v_{\text{ms}})_D + a(S_{k-1}^n, v_{\text{ms}}) = (\bar{\partial}_{\tau_c}^\alpha - \bar{\partial}_{\tau_f}^\alpha) \left( \mathcal{F}_f^{\text{ms},\ell}(T^{n-1}, U_{k-1}^{n-1}; \Phi_{k-1}^{n-1})_1, v_{\text{ms}} \right)_D.$$

Let  $e_k^n := U_k^n - U_{\text{ms},\ell}^n$ , then it satisfies

$$\begin{cases} \bar{\partial}_{\tau_c}^\alpha (e_k^n, v_{\text{ms}})_D + a(e_k^n, v_{\text{ms}}) = (R_n, v_{\text{ms}})_D & \text{for all } v_{\text{ms}} \in V_{\text{ms},\ell} \\ f_{j,k}[n] = \mathcal{H}(f_{j,k}[n-1]; \tau_c, e_k^{n-1}, e_k^n) & \text{for } n \geq 0 \text{ and } f_{j,k}[0] = 0 \text{ for all } j = 1, \dots, N_{\text{exp}} \\ e_k^0 = 0 \end{cases} \quad (6.7)$$

with the source term  $R_n$  being

$$\begin{aligned} R_n &:= -(\bar{\partial}_{\tau_c}^\alpha - \bar{\partial}_{\tau_f}^\alpha) U_{\text{ms},\ell}^n + (\bar{\partial}_{\tau_c}^\alpha - \bar{\partial}_{\tau_f}^\alpha) \mathcal{F}_f^{\text{ms},\ell}(T^{n-1}, U_{k-1}^{n-1}; \Phi_{k-1}^{n-1})_1 \\ &= -(\bar{\partial}_{\tau_c}^\alpha - \bar{\partial}_{\tau_f}^\alpha) \left( U_{\text{ms},\ell}^n - \mathcal{F}_f^{\text{ms},\ell}(T^{n-1}, U_{k-1}^{n-1}; \Phi_{k-1}^{n-1})_1 \right) \\ &= -(\bar{\partial}_{\tau_c}^\alpha - \bar{\partial}_{\tau_f}^\alpha) \mathcal{F}_f^{\text{ms},\ell}(T^{n-1}, U_{\text{ms},\ell}^{n-1} - U_{k-1}^{n-1}; \Phi_{\text{ms},\ell}^{n-1} - \Phi_{k-1}^{n-1})_1. \end{aligned}$$

Note that  $e_{k-1}^{n-1} = U_{\text{ms},\ell}^{n-1} - U_{k-1}^{n-1}$ , Lemma 6.2 implies

$$\|R_n\|_D \leq C_0(n-1)^{-\alpha} \tau_c^{-\alpha} \left( \|e_{k-1}^{n-1}\|_D + \alpha \left\| \Phi_{\text{ms},\ell}^{n-1} - \Phi_{k-1}^{n-1} \right\| \right).$$

By an application of Lemma 6.3 to the error equation (6.7), we obtain

$$\begin{aligned} \tau_c^{\alpha\sigma} |e_k^n|_{2\sigma} &\leq C_\alpha \tau_c^\alpha \sum_{m=1}^n \|R_m\|_D \\ &\leq C_0 n^{1-\alpha} \left( \frac{n}{n-1} \right)^\alpha \left( \|e_{k-1}^{n-1}\|_D + \alpha \left\| \Phi_{\text{ms},\ell}^{n-1} - \Phi_{k-1}^{n-1} \right\| \right) \\ &\leq C_0 n^{1-\alpha} \left( \|e_{k-1}^{n-1}\|_D + \alpha \left\| \Phi_{\text{ms},\ell}^{n-1} - \Phi_{k-1}^{n-1} \right\| \right). \end{aligned}$$

Consequently, we obtain

$$|e_k^n|_{2\sigma} \leq C_0 n^{1-\alpha} \times \tau_c^{-\alpha\sigma} \left( \|e_{k-1}^{n-1}\|_D + \alpha \left\| \Phi_{\text{ms},\ell}^{n-1} - \Phi_{k-1}^{n-1} \right\| \right).$$

Then one can follow the proof in [37, Lemma 7.3] and [33, Theorem 2.2] the desired result, under the condition that  $\tau_c$  is of reasonable size.  $\square$

We present in the next theorem the convergence rate of Algorithm 3 to Problems (1.1) in pointwise-in-time in  $L^2(D)$ -norm. To derive it, we first decompose the error from Algorithm 3 as a summation of the error from Problem (4.15) and the error from parareal algorithm. The former part is estimated by Theorem 6.1.

**Theorem 6.2** (Error estimate of Algorithm 3 to Problem (1.1) in  $L^2(D)$ -norm). *Let Assumption 3.1 hold. Assume that the source term  $f \in W^{2,\infty}(I; L^2(D))$  and initial data  $u_0 \in L^2(D)$ . Let  $\ell \in \mathbb{N}_+$  be the level parameter. The coarse time step size and fine time step size are  $\tau_c$  and  $\tau_f$ . Assume that  $\tau_f$  is sufficiently small, s.t.,*

$$e^{-\tau_c \gamma} ((1 - \alpha)^{-1} + \epsilon \tau_c^{1+\alpha} / 2) < \eta(\alpha, \theta) := \frac{\alpha}{1 + \alpha} (\theta - \alpha) \text{ for given } \theta \in [\alpha, 1). \quad (6.8)$$

Let  $u(\cdot, T^n) \in V$  be the solution to Problem (1.1) and let  $U_k^n$  be the solution from Algorithm 3 with iteration  $k \in \mathbb{N}_+$ . Assume that the iteration number  $k$  is not large. Then for any  $\sigma \in (0, 1)$ , it holds

$$\begin{aligned} \|u(\cdot, T^n) - U_k^n\|_D &\lesssim (\tau_f^{\sigma\alpha} + |\eta(H, \ell) \log \eta(H, \ell)|^2) \|f\|_{W^{2,\infty}(0,T;L^2(D))} \\ &\quad + \left( \tau_f^{\sigma\alpha} + \eta(H, \ell) |\log \eta(H, \ell)| (T^n)^{-\alpha} + \prod_{j=0}^k \left( \frac{\tau_c}{T^{n-j}} \right)^{\alpha\sigma} \right) |u_0|_{2\sigma}. \end{aligned} \quad (6.9)$$

*Proof.* We first split the error as the summation of  $\|u(\cdot, T^n) - U_{\text{ms},\ell}^m\|_D$  and  $\|U_{\text{ms},\ell}^m - U_k^n\|_D$  for  $m := \tau_c / \tau_f \times n$  with  $T^n = t_m$ . Similarly, let  $m' := \tau_c / \tau_f \times (n - 1)$ , then it holds  $t_{m'} = T^{n-1}$ . Here,  $U_{\text{ms},\ell}^m$  is the solution to Problem (4.15) and the error estimate is presented in Theorem 6.1.

We next estimate the error induced by parareal algorithm in the multiscale method, i.e., the second term  $\|U_{\text{ms},\ell}^m - U_k^n\|_D$ . We will prove:

$$e_k^n := \|U_{\text{ms},\ell}^m - U_k^n\|_D \leq C_0 \prod_{j=0}^k \left( \frac{\tau_c}{T^{n-j}} \right)^{1-\alpha} |u_0|_{2\sigma}. \quad (6.10)$$

Obviously, the inequality (6.10) holds when  $k = 0$ . Assume that it holds for iteration  $k$  for some  $k \in \mathbb{N}_+$ . We will show that it holds for the next iteration  $k + 1$ . Before that, we first derive the error estimate for the history information. Let  $\theta \in [\alpha, 1)$  be a positive parameter, then Lemma 4.2 implies

$$d_k^n := \|\Phi_{\text{ms},\ell}^m - \Phi_k^n\| \leq e^{-\tau_c \gamma} \|\Phi_{\text{ms},\ell}^{m'} - \Phi_k^{n-1}\| \quad (6.11)$$

$$\begin{aligned} &+ e^{-\tau_c \gamma} ((1 - \alpha)^{-1} + \epsilon \tau_c^{1+\alpha} / 2) \left( \|u_{\text{ms},\ell}^{m'} - U_k^{n-1}\|_D + \|u_{\text{ms},\ell}^m - U_k^n\|_D \right) \\ &\leq \eta(\alpha, \theta) (d_k^{n-1} + e_k^{n-1} + e_k^n). \end{aligned} \quad (6.12)$$

Here,  $\gamma$  is defined in (4.7) and we have used (6.8) in the last inequality.

Apply this inequality iteratively, we obtain

$$d_k^n \leq (1 + \eta(\alpha, \theta)) \sum_{i=1}^{n-k} \eta(\alpha, \theta)^{i-1} e_k^{n-i} + \eta(\alpha, \theta) e_k^n. \quad (6.13)$$

By (6.10), we derive

$$d_k^n \leq \frac{2C_0}{1 - \eta(\alpha, \theta)} \prod_{j=0}^k \left( \frac{\tau_c}{T^{n-j}} \right)^{\alpha\sigma} |u_0|_{2\sigma}. \quad (6.14)$$

Next we estimate  $e_{k+1}^n$ . We can obtain from Algorithm 3:

$$\begin{aligned} U_{\text{ms},\ell}^m - U_{k+1}^n &= \mathcal{S}(T^{n-1}, U_{\text{ms},\ell}^{m'}; \Phi_{\text{ms},\ell}^{m'}) - \mathcal{S}(T^{n-1}, U_k^{n-1}; \Phi_k^{n-1}) \\ &\quad + \mathcal{G}^{\text{ms},\ell}(T^{n-1}, U_{\text{ms},\ell}^{m'}; \Phi_{\text{ms},\ell}^{m'}) - \mathcal{G}^{\text{ms},\ell}(T^{n-1}, U_{k+1}^{n-1}; \Phi_{k+1}^{n-1}). \end{aligned}$$



Consequently, an application of Lemma 6.4 and Lemma 6.5 leads to

$$\begin{aligned}
e_{k+1}^n &= \|U_{\text{ms},\ell}^m - U_{k+1}^n\|_D \\
&\leq \left\| \mathcal{S}(T^{n-1}, U_{\text{ms},\ell}^{m'}; \Phi_{\text{ms},\ell}^{m'}) - \mathcal{S}(T^{n-1}, U_k^{n-1}; \Phi_k^{n-1}) \right\|_D + \left\| \mathcal{G}^{\text{ms},\ell}(T^{n-1}, u_{\text{ms},\ell}^{m'}; \Phi_{\text{ms},\ell}^{m'}) - \mathcal{G}^{\text{ms},\ell}(T^{n-1}, U_{k+1}^{n-1}; \Phi_{k+1}^{n-1}) \right\|_D \\
&\leq C_0 \left( \frac{1}{T^n} \right)^{\alpha\sigma} \tau_c^{\alpha\sigma} \left( \|U_{\text{ms},\ell}^{m'} - U_k^{n-1}\|_D + \alpha \|\Phi_{\text{ms},\ell}^{m'} - \Phi_k^{n-1}\| \right) + \alpha \left( \|U_{\text{ms},\ell}^{m'} - U_{k+1}^{n-1}\|_D + (1-\alpha) \|\Phi_{\text{ms},\ell}^{m'} - \Phi_{k+1}^{n-1}\| \right)
\end{aligned}$$

for any  $\sigma \in (0, 1)$ . That is,

$$e_{k+1}^n \leq C_0 \left( \frac{\tau_c}{T^n} \right)^{\alpha\sigma} \left( e_k^{n-1} + \alpha d_k^{m-1} \right) + \alpha \left( e_{k+1}^{n-1} + (1-\alpha) d_{k+1}^{m-1} \right).$$

Let  $\beta := \frac{1}{1+\alpha}$  be a positive parameter, then together with the inequality (6.12), we obtain

$$\begin{aligned}
(1-\beta)e_{k+1}^n + \beta d_{k+1}^m &\leq C_0(1-\beta + \beta\eta(\alpha, \theta)(1+\alpha)) \left( \frac{\tau_c}{T^n} \right)^{\alpha\sigma} \left( e_k^{n-1} + \alpha d_k^{m-1} \right) \\
&\quad + ((1-\beta)\alpha + \beta\eta(\alpha, \theta)(1+\alpha)) e_{k+1}^{n-1} \\
&\quad + ((1-\beta)\alpha(1-\alpha) + \beta\eta(\alpha, \theta)(1+\alpha - \alpha^2)) d_{k+1}^{m-1}.
\end{aligned}$$

Notice that

$$(1-\beta)\alpha + \beta\eta(\alpha, \theta)(1+\alpha) = \theta(1-\beta),$$

and

$$(1-\beta)\alpha(1-\alpha) + \beta\eta(\alpha, \theta)(1+\alpha - \alpha^2) \leq \theta\beta.$$

These, together with inequalities (6.10) and (6.14), lead to

$$\begin{aligned}
(1-\beta)e_{k+1}^n + \beta d_{k+1}^m &\leq C_0^2 \left( 1 + \frac{2\alpha}{1-\eta(\alpha, \theta)} \right) \prod_{j=0}^{k+1} \left( \frac{\tau_c}{T^{n-j}} \right)^{\alpha\sigma} |u_0|_{2\sigma} \\
&\quad + \theta \left( (1-\beta)e_{k+1}^{n-1} + \beta d_{k+1}^{m-1} \right).
\end{aligned}$$

Consequently, we derive

$$(1-\beta)e_{k+1}^n + \beta d_{k+1}^m \leq C_0^2 \left( 1 + \frac{2\alpha}{1-\eta(\alpha, \theta)} \right) \prod_{j=0}^{k+1} \left( \frac{\tau_c}{T^{n-j}} \right)^{\alpha\sigma} |u_0|_{2\sigma} + \theta \left( (1-\beta)e_{k+1}^{n-1} + \beta d_{k+1}^{m-1} \right). \quad (6.15)$$

Repeat this recurrence relation, we obtain

$$(1-\beta)e_{k+1}^n + \beta d_{k+1}^m \leq \frac{1}{1-\theta} C_0^2 \left( 1 + \frac{2\alpha}{1-\eta(\alpha, \theta)} \right) \prod_{j=0}^{k+1} \left( \frac{\tau_c}{T^{n-j}} \right)^{\alpha\sigma} |u_0|_{2\sigma} \times A(n, k),$$

where

$$\begin{aligned}
A(n, k) &:= \sum_{j=1}^{n-(k+1)} \theta^{j-1} \left( \prod_{i=0}^{k+1} \frac{T^{n-i}}{T^{n-j-i}} \right)^{\alpha\sigma} \\
&= \sum_{m=k+1}^{n-1} \theta^{n-1-m} \left( \prod_{i=0}^{k+1} \frac{n-i}{m-i} \right)^{\alpha\sigma} \\
&\leq \sum_{m=k+1}^{n-1} \theta^{n-1-m} \left( \prod_{i=0}^{k+1} \frac{n-i}{m-i} \right).
\end{aligned}$$

The last inequality above holds because  $\alpha\sigma < 1$ . Notice that we have proved in [24, Theorem 4.1] that  $A(n, k)$  is uniformly bounded with respect to  $n$  when  $k$  is small. Consequently, this proves the case for  $k + 1$ , and completes the proof.  $\square$

## 7 Numerical experiment

In this section, we perform a series of numerical experiments to demonstrate the performance of the proposed WEMP Algorithm. Let the spatial domain be  $D = [0, 1]^2$ . The coarse-scale and fine-scale mesh sizes are  $H = \frac{1}{10}$  and  $h = \frac{1}{200}$ . The coarse-scale and fine-scale time step sizes are  $\tau_c = 0.1$  and  $\tau_f = 10^{-4}$ . For all the numerical tests, we take the level parameter  $\ell = 2$ . A larger  $\ell$  can significantly reduce the spatial error. We take a smooth initial data,

$$u_0(x, y) = x(1-x)y(1-y).$$

The error is measured in  $L^2(D)$  relative error and  $H^1(D)$  relative error, defined by

$$\frac{\|u_h^n - \hat{u}_h^n\|_D}{\|u_h^n\|_D} \times 100 \quad \frac{\|u_h^n - \hat{u}_h^n\|_{H_\kappa^1(D)}}{\|u_h^n\|_{H_\kappa^1(D)}} \times 100,$$

where  $u_h^n$  and  $\hat{u}_h^n$  denote the reference solution and its approximation at  $t_n$ , respectively.

### 7.1 The influence of $\epsilon$ and $\alpha$

We study in this section the influence of  $\epsilon$  and  $\alpha$  to the accuracy of (2.3) when we replace the L1 scheme  $\tilde{\partial}_{\tau_f}^\alpha$  with the summation of exponential approximation  $\bar{\partial}_{\tau_f}^\alpha$ , i.e., we solve for  $U_h^n \in V_h$  for  $n = 1, \dots, M_f$  satisfying

$$\left\{ \begin{array}{l} \left( \frac{1}{\tau_f^\alpha c_\alpha} + A_h \right) U_h^n = \frac{\alpha}{\tau_f^\alpha c_\alpha} U_h^{n-1} + \frac{\alpha}{\Gamma(1-\alpha)} \sum_{j=1}^{N_{\text{exp}}} \omega_j \psi_{h,j}^{n-1} + \frac{1}{\Gamma(1-\alpha) t_n^\alpha} U_h^0 + F_h^n \\ \psi_{h,j}^n = \mathcal{H}(\psi_{h,j}^{n-1}; \tau_f, U_h^{n-1}, U_h^n) \text{ for } n \geq 0 \text{ and } \psi_{h,j}^0 = 0 \text{ for all } j = 1, \dots, N_{\text{exp}} \\ U_h^0 = I_h u_0. \end{array} \right.$$

Let  $T := 1$ , then (6.1) implies the minimum requirement of  $\epsilon$  is  $\epsilon \leq \epsilon^* := 0.5$ .

### 7.1.1 Smooth source term

The first series of numerical tests are presented in Figures 2 and 3, with a smooth source data, different numbers of terms in the SOE approximation and different values of  $\alpha = 0.1$  and  $\alpha = 0.9$ . The source term is

$$f(x, y, t) = xyt.$$

One can observe from both cases that a more economic numerical scheme  $\bar{\partial}_{\tau f}^{\alpha}$  yields almost the same accuracy even with a much larger tolerance  $\epsilon \gg \epsilon^*$ . For example, using a number of  $N_{\text{exp}} = 19$  summations with  $\epsilon = 24823$ , both  $L^2(D)$  relative error and  $H^1(D)$  relative error are below 0.15% for  $\alpha = 0.9$  in Figure 3. The performance for a smaller  $\alpha = 0.1$  is better as it is shown in Figure 2 that both  $L^2(D)$  relative error and  $H^1(D)$  relative error are below 0.0003% in this case.

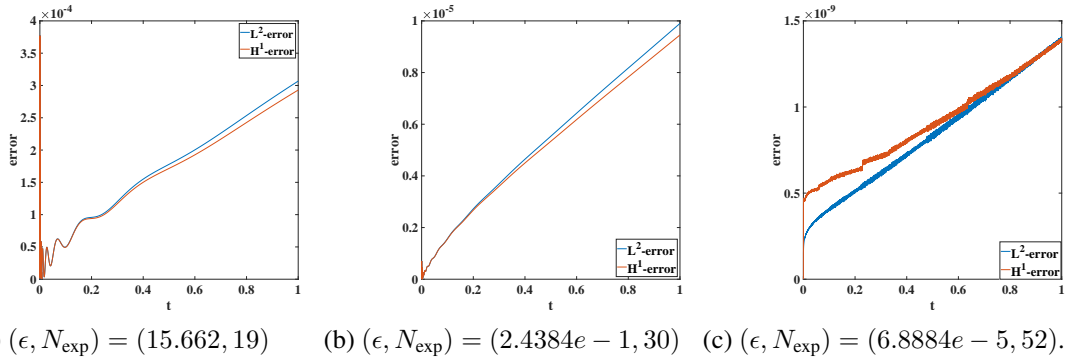


Figure 2:  $\alpha = 0.1, \epsilon^* = 0.5$ .

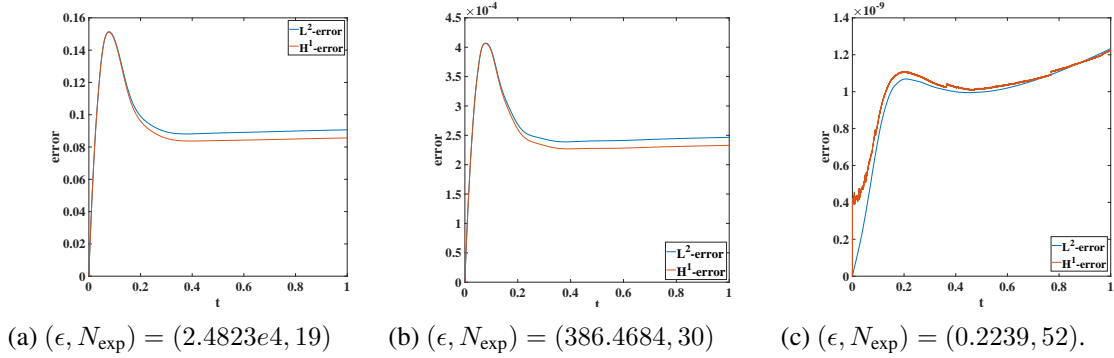


Figure 3:  $\alpha = 0.9, \epsilon^* = 0.5$ .

### 7.1.2 Rough source term

The next series of numerical tests are presented in Figure 4, with a rough source term and different numbers of terms in the SOE approximation. The rough source term has jump along the temporal variable, defined by

$$f(x, y, t) = \text{sgn}(\cos(2\pi t))xy.$$

Compared with Figure 3, the influence of rough source term is negligible. We observe similar behavior for varying  $\alpha$ .

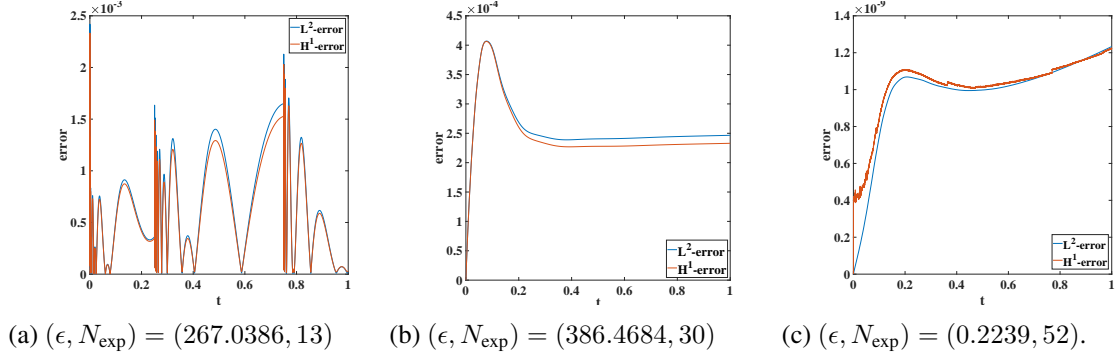


Figure 4:  $\alpha = 0.9, \epsilon^* = 0.5$ .

## 7.2 The performance of Algorithm 3

Next, we present the performance of Algorithm 3 with different  $\alpha = 0.1, 0.5, 0.9$ ,  $\epsilon$  and iteration number  $k = 1, 2, 3$ . The convergence history is presented in Figure 5, Figure 6 and Figure 7. For all tests, Algorithm 3 converges within 3 iterations. We observe that the relative  $L^2(D)$  error is less than 5% in the 3 iteration. We can not reduce further this error due to the coarse scale mesh size  $H = 1/10$ . To reduce this error further, we need to decrease  $H$ . Slightly more iteration numbers are needed for larger parameter  $\alpha$ .

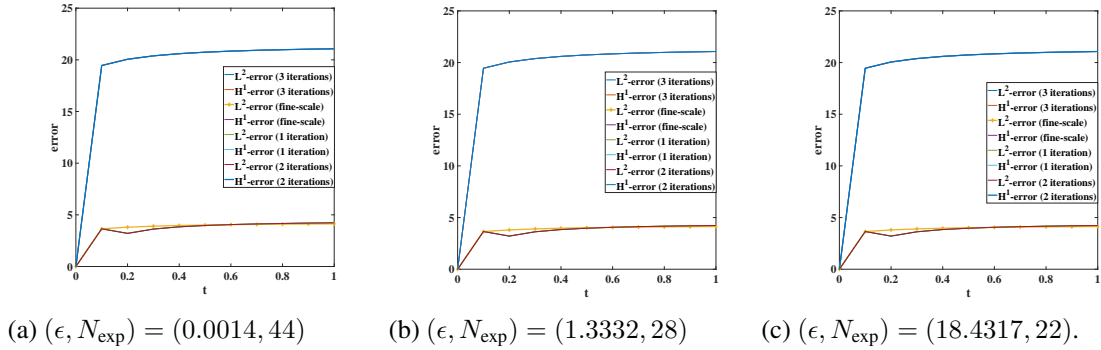
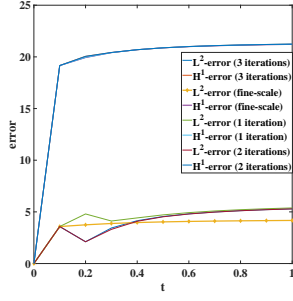


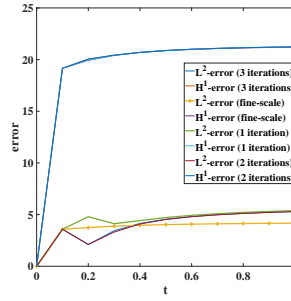
Figure 5:  $\alpha = 0.1, \epsilon^* = 0.5$ .

## 7.3 The performance of Algorithm 3 for long time

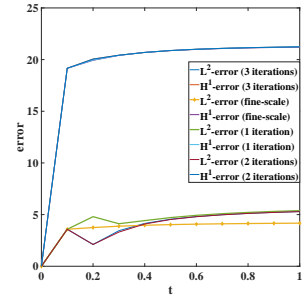
We test our proposed algorithm for longer time in this section with  $T = 10$ . The parameters of  $\epsilon$  and  $N_{\text{exp}}$  are determined by Lemma 6.3 and Corollary 4.1. Similar convergence behavior is observed as in the previous section, which is shown in Figure 8. For example, the  $L^2(D)$  relative error is below 5% for  $\alpha = 0.1$  and  $\alpha = 0.5$  after 3 iteration. However, the performance for  $\alpha = 0.9$  is slightly worse with relative  $L^2(D)$  error below 10% after 3 iteration. In this case, the performance of multiscale-SOE-scheme has a slightly larger error as observed from Figure 8c.



(a)  $(\epsilon, N_{\text{exp}}) = (0.0014, 55)$

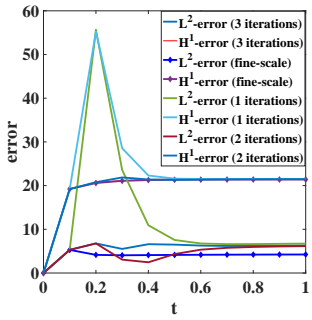


(b)  $(\epsilon, N_{\text{exp}}) = (1.6801, 38)$

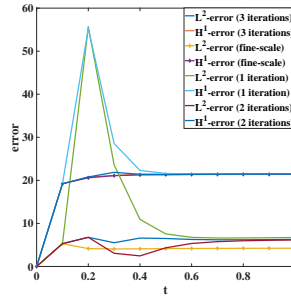


(c)  $(\epsilon, N_{\text{exp}}) = (19.629, 31)$ .

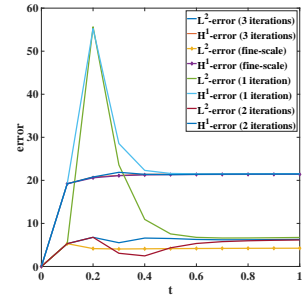
Figure 6:  $\alpha = 0.5, \epsilon^* = 0.5$ .



(a)  $(\epsilon, N_{\text{exp}}) = (0.0025, 64)$

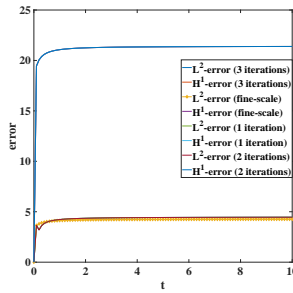


(b)  $(\epsilon, N_{\text{exp}}) = (2.6387, 47)$

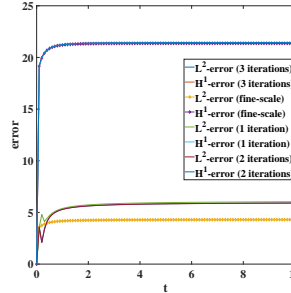


(c)  $(\epsilon, N_{\text{exp}}) = (41.9406, 38)$ .

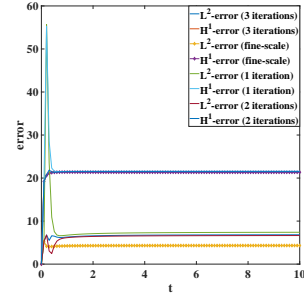
Figure 7:  $\alpha = 0.9, \epsilon^* = 0.5$ .



(a)  $\alpha = 0.1$



(b)  $\alpha = 0.5$



(c)  $\alpha = 0.9$

Figure 8:  $T = 10$ .

## 8 Conclusion

We propose in this paper a new efficient algorithm for time-fractional diffusion equations with heterogeneous coefficients. This algorithm is named as the Wavelet-based Edge Multiscale Parareal (WEMP) Algorithm, which can reduce computational cost and storage cost significantly. Moreover, it is suitable for long time

simulation when multiple processors are available. We want to remark that the total number of terms in the exponential sums can be reduced hugely for a given accuracy requirement (4.2) if an adaptive representation instead of a uniform representation is utilized. Consequently, this will lead to a more efficient algorithm for more challenging applications, for example, for much longer time simulation.

## References

- [1] R. Altmann, P. Henning, and D. Peterseim. Numerical homogenization beyond scale separation. *Acta Numerica*, 30:1–86, 2021.
- [2] G. Ariel, S. J. Kim, and R. Tsai. Parareal multiscale methods for highly oscillatory dynamical systems. *SIAM J. Sci. Comput.*, 38(6):A3540–A3564, 2016.
- [3] G. Bal. On the convergence and the stability of the parareal algorithm to solve partial differential equations. In *Domain decomposition methods in science and engineering*, pages 425–432. Springer, 2005.
- [4] G. Bal and Y. Maday. A “parareal” time discretization for non-linear PDE’s with application to the pricing of an American put. In *Recent developments in domain decomposition methods (Zürich, 2001)*, volume 23 of *Lect. Notes Comput. Sci. Eng.*, pages 189–202. Springer, Berlin, 2002.
- [5] L. Berlyand and H. Owhadi. Flux norm approach to finite dimensional homogenization approximations with non-separated scales and high contrast. *Arch. Ration. Mech. Anal.*, 198(2):677–721, 2010.
- [6] A. Blouza, L. Boudin, and S. M. Kaber. Parallel in time algorithms with reduction methods for solving chemical kinetics. *Commun. Appl. Math. Comput. Sci.*, 5(2):241–263, 2010.
- [7] D. Braess and W. Hackbusch. Approximation of  $1/x$  by exponential sums in  $[1, \infty)$ . *IMA journal of numerical analysis*, 25(4):685–697, 2005.
- [8] A. Christlieb, C. Macdonald, and B. Ong. Parallel high-order integrators. *SIAM J. Sci. Comput.*, 32(2):818–835, 2010.
- [9] W. E and B. Engquist. The heterogeneous multiscale methods. *Commun. Math. Sci.*, 1(1):87–132, 2003.
- [10] Y. Efendiev, J. Galvis, and T. Hou. Generalized multiscale finite element methods. *J. Comput. Phys.*, 251:116–135, 2013.
- [11] R. Falgout, S. Friedhoff, T. Kolev, S. MacLachlan, and J. Schroder. Parallel time integration with multigrid. *SIAM J. Sci. Comput.*, 36(6):C635–C661, 2014.
- [12] C. Farhat and M. Chandesis. Time-decomposed parallel time-integrators: theory and feasibility studies for fluid, structure, and fluid-structure applications. *Internat. J. Numer. Methods Engrg.*, 58(9):1397–1434, 2003.
- [13] C. Farhat, J. Cortial, C. Dastillung, and H. Bavestrello. Time-parallel implicit integrators for the near-real-time prediction of linear structural dynamic responses. *Internat. J. Numer. Methods Engrg.*, 67(5):697–724, 2006.

- [14] P. Fischer, F. Hecht, and Y. Maday. A parareal in time semi-implicit approximation of the Navier-Stokes equations. In *Domain decomposition methods in science and engineering*, volume 40 of *Lect. Notes Comput. Sci. Eng.*, pages 433–440. Springer, Berlin, 2005.
- [15] S. Fu, E. Chung, and G. Li. Edge multiscale methods for elliptic problems with heterogeneous coefficients. *J. Comput. Phys.*, 369(1):228–242, 2019.
- [16] S. Fu, G. Li, R. Craster, and S. Guenneau. Wavelet-based edge multiscale finite element method for helmholtz problems in perforated domains. *To appear in Multiscale Model. Simul.*, 2021.
- [17] T. Hou and X.-H. Wu. A multiscale finite element method for elliptic problems in composite materials and porous media. *J. Comput. Phys.*, 134(1):169–189, 1997.
- [18] T. Hughes, G. Feijóo, L. Mazzei, and J.-B. Quinicy. The variational multiscale method—a paradigm for computational mechanics. *Comput. Methods Appl. Mech. Engrg.*, 166(1-2):3–24, 1998.
- [19] S. Jiang, J. Zhang, Q. Zhang, and Z. Zhang. Fast evaluation of the caputo fractional derivative and its applications to fractional diffusion equations. *Communications in Computational Physics*, 21(3):650–678, 2017.
- [20] B. Jin and Z. Zhou. An analysis of Galerkin proper orthogonal decomposition for subdiffusion. *ESAIM Math. Model. Numer. Anal.*, 51(1):89–113, 2017.
- [21] A. Kilbas, H. Srivastava, and J. Trujillo. *Theory and Applications of Fractional Differential Equations*, volume 204 of *North-Holland Mathematics Studies*. Elsevier Science B.V., Amsterdam, 2006.
- [22] F. Legoll, T. Lelièvre, and G. Samaey. A micro-macro parareal algorithm: application to singularly perturbed ordinary differential equations. *SIAM J. Sci. Comput.*, 35(4):A1951–A1986, 2013.
- [23] G. Li. On the convergence rates of gmsfems for heterogeneous elliptic problems without oversampling techniques. *Multiscale Model. Simul.*, 17(2):593–619, 2019.
- [24] G. Li and J. Hu. Wavelet-based edge multiscale parareal algorithm for parabolic equations with heterogeneous coefficients. *J. Comput. Phys.*, 444, 2021.
- [25] G. Li, D. Peterseim, and M. Schedensack. Error analysis of a variational multiscale stabilization for convection-dominated diffusion equations in two dimensions. *IMA J. Numer. Anal.*, 38(3):1229–1253, 2018.
- [26] Y. Lin and C. Xu. Finite difference/spectral approximations for the time-fractional diffusion equation. *Journal of Computational Physics*, 225(2):1533–1552, 2007.
- [27] J.-L. Lions, Y. Maday, and G. Turinici. Résolution d’EDP par un schéma en temps “pararéel”. *C. R. Acad. Sci. Paris Sér. I Math.*, 332(7):661–668, 2001.
- [28] Y. Liu and J. Hu. Modified propagators of parareal in time algorithm and application to Princeton ocean model. *Internat. J. Numer. Methods Fluids*, 57(12):1793–1804, 2008.
- [29] Y. Maday. Parareal in time algorithm for kinetic systems based on model reduction. In *High-dimensional partial differential equations in science and engineering*, volume 41 of *CRM Proc. Lecture Notes*, pages 183–194. Amer. Math. Soc., Providence, RI, 2007.

- [30] Y. Maday, J. Salomon, and G. Turinici. Monotonic parareal control for quantum systems. *SIAM J. Numer. Anal.*, 45(6):2468–2482, 2007.
- [31] A. Målqvist and D. Peterseim. Localization of elliptic multiscale problems. *Math. Comp.*, 83(290):2583–2603, 2014.
- [32] I. Podlubny. *Fractional Differential Equations*, volume 198 of *Mathematics in Science and Engineering*. Academic Press, Inc., San Diego, CA, 1999. An introduction to fractional derivatives, fractional differential equations, to methods of their solution and some of their applications.
- [33] K. Sakamoto and M. Yamamoto. Initial value/boundary value problems for fractional diffusion-wave equations and applications to some inverse problems. *J. Math. Anal. Appl.*, 382(1):426–447, 2011.
- [34] F. Stenger. *Numerical methods based on sinc and analytic functions*, volume 20 of *Springer Series in Computational Mathematics*. Springer-Verlag, New York, 1993.
- [35] T. Strouboulis, I. Babuška, and K. Copps. The design and analysis of the generalized finite element method. *Comput. Methods Appl. Mech. Engrg.*, 181:43–69, 2000.
- [36] Z. Sun and X. Wu. A fully discrete difference scheme for a diffusion-wave system. *Applied Numerical Mathematics*, 56(2):193–209, 2006.
- [37] V. Thomée. *Galerkin finite element methods for parabolic problems*, volume 1054. Springer, 2006.
- [38] S.-L. Wu and T. Zhou. Parareal algorithms with local time-integrators for time fractional differential equations. *Journal of Computational Physics*, 358:135–149, 2018.
- [39] H. Zhu and C. Xu. A fast high order method for the time-fractional diffusion equation. *SIAM Journal on Numerical Analysis*, 57(6):2829–2849, 2019.