DISCRETE AND CONTINUOUS
DYNAMICAL SYSTEMS SERIES S

doi:10.3934/dcdss.2022016

HIGHLY ACCURATE OPERATOR FACTORIZATION METHODS FOR THE INTEGRAL FRACTIONAL LAPLACIAN AND ITS GENERALIZATION

Yixuan Wu and Yanzhi Zhang*

Department of Mathematics and Statistics
Missouri University of Science and Technology
Rolla, MO 65409, USA

Abstract. In this paper, we propose a new class of operator factorization methods to discretize the integral fractional Laplacian $(-\Delta)^{\frac{\alpha}{2}}$ for $\alpha \in (0,2)$. One main advantage is that our method can easily increase numerical accuracy by using high-degree Lagrange basis functions, but remain its scheme structure and computer implementation unchanged. Moreover, it results in a symmetric (multilevel) Toeplitz differentiation matrix, enabling efficient computation via the fast Fourier transforms. If constant or linear basis functions are used, our method has an accuracy of $\mathcal{O}(h^2)$, while $\mathcal{O}(h^4)$ for quadratic basis functions with h a small mesh size. This accuracy can be achieved for any $\alpha \in (0,2)$ and can be further increased if higher-degree basis functions are chosen. Numerical experiments are provided to approximate the fractional Laplacian and solve the fractional Poisson problems. It shows that if the solution of fractional Poisson problem satisfies $u \in C^{m,l}(\bar{\Omega})$ for $m \in \mathbb{N}$ and $0 < l < 1$, our method has an accuracy of $\mathcal{O}\big(h^{\min\{m+l,\,2\}}\big)$ for constant and linear basis functions, while $\mathcal{O}\big(h^{\min\{m+l,\,4\}}\big)$ for quadratic basis functions. Additionally, our method can be readily applied to approximate the generalized fractional Laplacians with symmetric kernel function, and numerical study on the tempered fractional Poisson problem demonstrates its efficiency.

1. **Introduction.** The fractional Laplacian $(-\Delta)^{\frac{\alpha}{2}}$, representing the infinitesimal generator of a symmetric α-stable Lévy process, is a nonlocal generalization of the classical Laplace operator $-\Delta = -\sum_{k=1}^{d} \partial^2_{x^{(k)}}$ for dimension $d \geq 1$ and $\mathbf{x} = \big(x^{(1)}, \cdots, x^{(d)}\big) \in \mathbb{R}^d$. Over the recent decade, the fractional Laplacian has been widely applied to study anomalous diffusion in many fields, and numerous theoretical studies have been reported on this nonlocal operator (see [6, 21, 9, 2, 5, 20] and references therein). Computationally, the nonlocal nature of the fractional Laplacian introduces significant challenges in its numerical discretization and efficient implementation. In this paper, we propose a new class of operator factorization methods to discretize the integral fractional Laplacian $(-\Delta)^{\frac{\alpha}{2}}$ for $\alpha \in (0,2)$. One main advantage of our method is its flexibility to increase numerical accuracy by using high-degree Lagrange basis functions.

2020 *Mathematics Subject Classification.* Primary: 65R20, 65N35, 45E05; Secondary: 47G10, 47G30.

Key words and phrases. Fractional Laplacian, operator factorization, Lagrange basis functions, fractional Poisson problems, tempered fractional Laplacian.

This work was partially supported by the US National Science Foundation under Grant Number DMS-1913293 and DMS-1953177.

* Corresponding author: Yanzhi Zhang.

Let $\Omega \subset \mathbb{R}^d$ be an open bounded domain, and denote $\Omega^c = \mathbb{R}^d \backslash \Omega$ as the complement of Ω. The fractional Poisson problem with extended Dirichlet boundary conditions takes the following form:

$$(-\Delta)^{\frac{\alpha}{2}} u(\mathbf{x}) = f(\mathbf{x}), \qquad \text{for} \quad \mathbf{x} \in \Omega, \tag{1}$$

$$u(\mathbf{x}) = g(\mathbf{x}), \qquad \text{for} \quad \mathbf{x} \in \Omega^c. \tag{2}$$

The fractional Laplacian $(-\Delta)^{\frac{\alpha}{2}}$ is defined in a hypersingular integral form [19, 23]:

$$(-\Delta)^{\frac{\alpha}{2}} u(\mathbf{x}) = c_{d,\alpha}\ \text{P.V.} \int_{\mathbb{R}^d} \frac{u(\mathbf{x}) - u(\mathbf{y})}{|\mathbf{x}-\mathbf{y}|^{d+\alpha}}\, d\mathbf{y}, \qquad \text{for} \quad \alpha \in (0,2), \tag{3}$$

where P.V. stands for the Cauchy principal value, $|\mathbf{x}-\mathbf{y}|$ denotes the Euclidean distance between points $\mathbf{x}$ and $\mathbf{y}$, and the normalization constant $c_{d,\alpha}$ is defined as

$$c_{d,\alpha} = \frac{2^{\alpha-1}\alpha\,\Gamma\big((d+\alpha)/2\big)}{\sqrt{\pi^d}\,\Gamma\big(1-\alpha/2\big)}, \qquad \text{for} \quad d = 1, 2, 3$$

with $\Gamma(\cdot)$ denoting the Gamma function. The operator (3) collapses to the identity operator as $\alpha \to 0$, while it converges to the classical Laplacian $-\Delta$ as $\alpha \to 2$. In the literature, the fractional Laplacian can be also defined via a pseudo-differential operator with symbol $|\mathbf{k}|^\alpha$ [19, 23], i.e.,

$$(-\Delta)^{\frac{\alpha}{2}} u(\mathbf{x}) = \mathcal{F}^{-1}\big[|\mathbf{k}|^\alpha \mathcal{F}[u]\big], \qquad \text{for} \quad \alpha > 0, \tag{4}$$

where $\mathcal{F}$ represents the Fourier transform, and $\mathcal{F}^{-1}$ denotes its inverse. It shows in [23, 18] that these two definitions are equivalent on the Schwartz space on $\mathbb{R}^d$. More discussion of the fractional Laplacian and its related nonlocal operators can be found in [9, 11] and references therein. Note that the fractional Laplacian $(-\Delta)^{\frac{\alpha}{2}}$ is rotational invariant, which is an important property in modeling isotropic anomalous diffusion, especially when $d \geq 2$ [16].

In the literature, numerical methods for the fractional Laplacian $(-\Delta)^{\frac{\alpha}{2}}$ can be mainly classified in three groups based on which definition (e.g. (4) or (3)) the method is developed on. The pseudo-differential definition in (4) allows one to utilize the Fourier transform for both analysis and simulations. Based on this definition, it is natural to introduce the Fourier pseudospectral methods to discretize the fractional Laplacian $(-\Delta)^{\frac{\alpha}{2}}$ on a bounded domain with periodic boundary conditions [17]. This method can be applied for both classical ($\alpha = 2$) and fractional ($\alpha < 2$) Laplacians. However, it is challenging to incorporate non-periodic boundary conditions into the pseudo-differential definition (4). For example, a meshfree pseudospectral method was proposed in [22] to discretize $(-\Delta)^{\frac{\alpha}{2}}$ with extended Dirichlet boundary conditions. In order to account Dirichlet boundary conditions, a large computational domain (much larger than physical domain Ω) as well as boundary approximations were introduced, and moreover numerical quadrature rules were required to approximate the Fourier integrals in definition (4). On the other hand, the integral definition in (3) provides a pointwise formulation and thus can easily work with different boundary conditions. Based on the integral definition, finite difference methods [15, 10, 14] and finite element methods [2, 1, 7, 4] have been developed to discretize the fractional Laplacian $(-\Delta)^{\frac{\alpha}{2}}$ with extended Dirichlet boundary conditions. In contrast to the pseudo-differential definition, the integral definition in (3) is valid only for $\alpha \in (0,2)$, so do the resulting numerical methods [15, 10, 14, 2, 1, 7, 4]. Moreover, the above methods based on the integral definition are mainly limited to the second order accuracy. Recently, a new class of methods using both definitions (4) and (3) of the fractional Laplacian $(-\Delta)^{\frac{\alpha}{2}}$ were proposed

in [8, 26, 27]. These methods, based on radial basis functions, are meshfree and work for both classical and fractional Laplacians. Note that similar spectral methods were recently introduced in [25, 24] to approximate the fractional Laplacian on unbounded domains.

In this paper, we propose a novel operator factorization method to discretize the integral fractional Laplacian in (3). Our method first factorizes its integrand into a product of central difference quotient $\Phi_{d,\gamma}(\mathbf{x},\boldsymbol{\xi})$ and power function $\mu_\gamma(\boldsymbol{\xi})$, and then approximates $\Phi_{d,\gamma}(\mathbf{x},\boldsymbol{\xi})$ with respect to $\boldsymbol{\xi}$ using Lagrange basis functions φ^p (for $p \in \mathbb{N}^0$). One main advantage of our method is that it can easily increase numerical accuracy by using high-degree Lagrange basis functions φ^p, but remain the scheme structure and computer implementation unchanged. Moreover, our method results in a symmetric (multilevel) Toeplitz matrix, and thus algorithms via the fast Fourier transforms can be developed for their efficient simulations. Numerical studies show that our method with constant basis φ^0 or linear basis φ^1 has an optimal accuracy of $\mathcal{O}(h^2)$ with h small mesh size, and this rate can be improved to $\mathcal{O}(h^4)$ if quadratic basis φ^2 is used. This accuracy can be further increased if high-degree basis functions are used. The performance of our method under different conditions are detailedly investigated and compared. If the solution of fractional Poisson problem satisfies $u \in C^{m,l}(\bar{\Omega})$ for $m \in \mathbb{N}$ and $0 < l < 1$, then our method has an accuracy of $\mathcal{O}\big(h^{\min\{m+l,\,2\}}\big)$ for constant and linear basis functions, while $\mathcal{O}\big(h^{\min\{m+l,\,4\}}\big)$ for quadratic basis functions. Furthermore, our method can be readily applied to study the generalized fractional Laplacian with symmetric kernel function, and numerical experiments on the tempered fractional Poisson problem are provided to demonstrate it.

The paper is organized as follows. In Section 2, we first introduce the operator factorization framework, and then derive the detailed scheme for one-, two-, and three-dimensional cases. In Section 3, we examine the performance of our method in approximating the fractional Laplacian $(-\Delta)^{\frac{\alpha}{2}}$ and in solving fractional Poisson problems. Moreover, we generalize our method to solve the tempered fractional Poisson problems. Finally, conclusions are drawn in Section 4.

2. Method of operator factorization.

Numerical methods for the integral fractional Laplacian in (3) still remain limited, and main challenges come from its nonlocality and strong singularity. So far, the existing finite difference/element methods are mostly the second-order accurate. In this section, we introduce a new class of methods based on operator factorization to discretize the integral fractional Laplacian $(-\Delta)^{\frac{\alpha}{2}}$ with extended Dirichlet boundary conditions. Our method has advantages of easily improving numerical accuracy without changing the scheme structure.

To introduce our method, we will first reformulate the integral fractional Laplacian (3) into the form of operator factorization. For points $\mathbf{x}, \mathbf{y} \in \mathbb{R}^d$, we define a vector $\boldsymbol{\xi} = \big(\xi^{(1)}, \xi^{(2)}, \cdots, \xi^{(d)}\big)$ with $\xi^{(i)} = |x^{(i)} - y^{(i)}|$ denoting the distance of $\mathbf{x}$ and $\mathbf{y}$ in the i-th direction, and then rewrite the fractional Laplacian as:

$$(-\Delta)^{\frac{\alpha}{2}} u(\mathbf{x}) = -c_{d,\alpha} \int_{\mathbb{R}^d_+} \left(\sum_{\mathbf{m}\in\varkappa_1} u(\mathbf{x} + (-1)^{\mathbf{m}} \circ \boldsymbol{\xi}) - 2^d u(\mathbf{x}) \right) \frac{d\boldsymbol{\xi}}{|\boldsymbol{\xi}|^{d+\alpha}}, \tag{5}$$

where we denote $\mathbb{R}^d_+ = [0,\infty)^d$, and $\mathbf{a} \circ \mathbf{b}$ represents the Hadamard product of $\mathbf{a}$ and $\mathbf{b}$. For $M \in \mathbb{N}$, the index set $\varkappa_M$ is defined as

$$\varkappa_M = \{(m_1, m_2, \cdots, m_d) \mid 0 \le m_i \le M, \text{ for } 1 \le i \le d\},$$

and the vector $(-1)^{\mathbf{m}} = \big((-1)^{m_1}, (-1)^{m_2}, \cdots, (-1)^{m_d}\big)$. Choosing a splitting parameter $\gamma \in (\alpha, 2]$, and introducing functions

$$\begin{aligned} \Phi_{d,\gamma}(\mathbf{x}, \boldsymbol{\xi}) &= \bigg(\sum_{\mathbf{m} \in \varkappa_1} u(\mathbf{x} + (-1)^{\mathbf{m}} \circ \boldsymbol{\xi}) - 2^d u(\mathbf{x}) \bigg) |\boldsymbol{\xi}|^{-\gamma}, \\ \mu_\gamma(\boldsymbol{\xi}) &= |\boldsymbol{\xi}|^{\gamma-(d+\alpha)}, \end{aligned} \tag{6}$$

we then further rewrite (5) into the following operator factorization form:

$$(-\Delta)^{\frac{\alpha}{2}} u(\mathbf{x}) = -c_{d,\alpha} \int_{\mathbb{R}^d_+} \Phi_{d,\gamma}(\mathbf{x}, \boldsymbol{\xi})\, \mu_\gamma(\boldsymbol{\xi}) d\boldsymbol{\xi}. \tag{7}$$

That is, we factorize the integrand in (5) as a product of central difference quotient $\Phi_{d,\gamma}(\mathbf{x}, \boldsymbol{\xi})$ and power function $\mu_\gamma(\boldsymbol{\xi})$; thus the fractional Laplacian in (7) can be viewed as a weighted integral of function $\Phi_{d,\gamma}(\mathbf{x}, \boldsymbol{\xi})$. The above operator factorization was introduced in [10, 13] to develop finite difference methods for the integral fractional Laplacian $(-\Delta)^{\frac{\alpha}{2}}$, and also applied to solve the fractional Schödinger equation in a box potential (with $\gamma = 1 + \alpha/2$) [12]. Note that even though both of them are based on the same operator factorization form, the method proposed in this work is significantly different from those in [10, 13]. The finite difference methods in [10, 13] approximate the integral in (7) by the weighted trapezoidal rules, while our proposed method interpolates function $\Phi_{d,\gamma}(\mathbf{x}, \boldsymbol{\xi})$ with Lagrange basis functions, making it easy to increase accuracy via high-degree basis functions. The parameter γ plays an important role in determining the numerical accuracy of our method, and the optimal splitting parameter is $\gamma = 2$ which leads to the smallest numerical errors and best accuracy rates. More discussion and illustrations can be found in Section 3.

Denote the d-dimensional domain $\Omega = (a_1, b_1) \times (a_2, b_2) \cdots \times (a_d, b_d)$, and introduce

$$\Upsilon = [0, L]^d \quad \text{with} \quad L = \max_{1 \le i \le d} |b_i - a_i|; \qquad \Upsilon^c_+ = \mathbb{R}^d_+ \backslash \Upsilon.$$

Then the integral in (7) can be divided into two parts, i.e.

$$(-\Delta)^{\frac{\alpha}{2}} u(\mathbf{x}) = -c_{d,\alpha} \bigg(\int_{\Upsilon} \Phi_{d,\gamma}(\mathbf{x}, \boldsymbol{\xi}) \mu_\gamma(\boldsymbol{\xi})\, d\boldsymbol{\xi} + \int_{\Upsilon^c_+} \Phi_{d,\gamma}(\mathbf{x}, \boldsymbol{\xi}) \mu_\gamma(\boldsymbol{\xi})\, d\boldsymbol{\xi} \bigg), \tag{8}$$

for $\mathbf{x} \in \Omega$. Given the definition of Υ^c_+, it is easy to conclude that for any $\mathbf{x} \in \Omega$ and $\boldsymbol{\xi} \in \Upsilon^c_+$, the point $\big(\mathbf{x} + (-1)^{\mathbf{m}} \circ \boldsymbol{\xi}\big)$ for $\mathbf{m} \in \varkappa_1$ locates outside of domain Ω. Noticing the extended Dirichlet boundary conditions in (2), we immediately obtain

$$\begin{aligned} \mathcal{L}^{\alpha}_{\Upsilon^c_+} u(\mathbf{x}) &:= \int_{\Upsilon^c_+} \Phi_{d,\gamma}(\mathbf{x}, \boldsymbol{\xi}) \mu_\gamma(\boldsymbol{\xi})\, d\boldsymbol{\xi} \\ &= -2^d u(\mathbf{x}) \int_{\Upsilon^c_+} \frac{d\boldsymbol{\xi}}{|\boldsymbol{\xi}|^{d+\alpha}} + \sum_{\mathbf{m} \in \varkappa_1} \int_{\Upsilon^c_+} \frac{g\big(\mathbf{x} + (-1)^{\mathbf{m}} \circ \boldsymbol{\xi}\big)}{|\boldsymbol{\xi}|^{d+\alpha}}\, d\boldsymbol{\xi}, \end{aligned} \tag{9}$$

for $\mathbf{x} \in \Omega$. The above integrals are free of singularity, and thus can be accurately computed by traditional quadrature rules. Moreover, if homogeneous boundary conditions (i.e., $g(\mathbf{x}) = 0$) are considered, the second term in (9) vanishes. Next, we move to approximate the first integral in (8), i.e.,

$$\mathcal{L}^{\alpha}_{\Upsilon} u(\mathbf{x}) := \int_{\Upsilon} \Phi_{d,\gamma}(\mathbf{x}, \boldsymbol{\xi}) \mu_\gamma(\boldsymbol{\xi})\, d\boldsymbol{\xi}. \tag{10}$$

We will start with introducing our method for the one-dimensional cases, and then generalize it to two and three dimensions. Without loss of generality, we will derive the numerical scheme for parameter $\gamma \in (\alpha, 2]$.

2.1. **One-dimensional cases.** In one-dimensional ($d = 1$) cases, we let domain $\Omega = (a, b)$, and then region $\Upsilon = [0, L]$ and $\Upsilon_+^c = (L, \infty)$ with $L = b - a$. For notational simplicity, we will denote $\xi := \xi^{(1)}$ and $x := x^{(1)}$, and the function

$$\Phi_{1,\gamma}(x,\, \xi) = \frac{u(x+\xi) - 2u(x) + u(x-\xi)}{\xi^\gamma}, \qquad \text{for} \quad \gamma \in (\alpha, 2].$$

In the special case of $\gamma = 2$, function $\Phi_{1,2}(x, \xi)$ represents the central difference quotient for the classical Laplacian Δ at point x.

Next, we will focus on approximating the integral $\mathcal{L}_\Upsilon^\alpha u$. Define mesh size $h = (b-a)/N$ with a positive integer N, and denote grid points

$$x_i = a + ih, \qquad \xi_i = ih, \qquad \text{for} \quad i = 0, 1, \ldots, N.$$

It implies that $\xi_0 = 0$ and $\xi_N = L$. Assume the ansatz for function $\Phi_{1,\,\gamma}(x, \xi)$ as:

$$\Phi_{1,\,\gamma}(x, \xi) = \sum_{k=0}^{N} \Phi_{1,\,\gamma}(x, \xi_k)\, \varphi_k^p(\xi), \qquad \text{for} \quad \xi \in [0, L], \tag{11}$$

where $\varphi_k^p(\xi)$, representing the p-th degree Lagrange polynomial, is the basis function at point ξ_k. The main novelty of our method is to interpolate function $\Phi_{1,\,\gamma}(x, \xi)$ with respect to ξ, instead of x. Note that at point $\xi = \xi_0$, function $\Phi_{1,\,\gamma}(x, \xi_0)$ is defined in a limit sense as $\xi \to 0$. To see this, we will divide our discussion into two cases:

(i). If the splitting parameter $\gamma = 2$, we obtain

$$\Phi_{1,2}(x,\, \xi_0) = \lim_{\xi \to 0} \frac{u(x-\xi) - 2u(x) + u(x+\xi)}{\xi^2} = u''(x) \;\approx\; \Phi_{1,\,2}(x,\, \xi_1), \tag{12}$$

via the central difference approximation to $u''(x)$.

(ii). If $\gamma \in (\alpha, 2)$, there is

$$\begin{aligned} \Phi_{1,\,\gamma}(x, \xi_0) &= \lim_{\xi \to 0} \frac{u(x-\xi) + u(x+\xi) - 2u(x)}{\xi^2} \cdot \xi^{2-\gamma} \\ &= \Phi_{1,\,2}(x,\, \xi_0) \lim_{\xi \to 0} \xi^{2-\gamma} = 0. \end{aligned} \tag{13}$$

Consequently, the summation in (11) starts from $k = 1$ (instead of $k = 0$), if $\gamma \neq 2$.

Substituting the ansatz (11) into (10) with $d = 1$, we obtain the approximation

$$\mathcal{L}_{\Upsilon,h}^\alpha u(x) = \sum_{k=0}^{N} \Phi_{1,\,\gamma}(x, \xi_k) \int_0^L \varphi_k^p(\xi)\, \frac{d\xi}{\xi^{1+\alpha-\gamma}}, \qquad \text{for} \quad x \in \Omega. \tag{14}$$

For notational convenience, we will denote the *weight integral*

$$\omega_k^p := \int_0^L \varphi_k^p(\xi)\, \frac{d\xi}{\xi^{1+\alpha-\gamma}}, \qquad \text{for} \quad k = 0, 1, 2, \ldots, N. \tag{15}$$

Since Lagrange polynomials are considered, the integration in (15) can be reduced from $[0, L]$ to the support of function $\varphi_k^p(\xi)$. Moreover, the weight integral ω_k^p can be found analytically. For convenience of the readers, we will provide the values

of ω_k^p for $p = 0, 1, 2$ in Appendix A. Substituting $x = x_i$ in (14) and noticing the definition of $\Phi_{1,\gamma}(x, \xi)$, we obtain the approximation of $\mathcal{L}_\Upsilon^\alpha u(x_i)$ as:

$$\mathcal{L}_{\Upsilon,h}^\alpha u_i = \sum_{k=1}^{N} \omega_k^p \frac{u_{i-k} - 2u_i + u_{i+k}}{\xi_k^\gamma} + \zeta\, \omega_0^p \frac{u_{i-1} - 2u_i + u_{i+1}}{\xi_1^\gamma}, \tag{16}$$

for $1 \le i \le N-1$, where we denote $u_i = u(x_i)$, and $\zeta = \lfloor \gamma/2 \rfloor$ with $\lfloor \cdot \rfloor$ representing the floor function. The second term in (16) vanishes if $\gamma \neq 2$, due to (13). On the other hand, the integral in (9) at point $x = x_i$ becomes

$$\mathcal{L}_{\Upsilon_+^c}^\alpha u_i = -\frac{2}{\alpha L^\alpha} u_i + \int_L^\infty \big[g(x_i - \xi) + g(x_i + \xi)\big] \frac{d\xi}{\xi^{1+\alpha}}, \tag{17}$$

for $1 \le i \le N-1$,

Combining (16) and (17) and reorganizing the terms yield our numerical approximation to the one-dimensional integral fractional Laplacian $(-\Delta)^{\frac{\alpha}{2}}$ with extended Dirichlet boundary conditions as:

$$\begin{aligned}(-\Delta)_h^{\frac{\alpha}{2}} u_i &= -c_{1,\alpha}\bigg(a_0 u_i + \sum_{k=1}^{N} a_k \big(u_{i+k} + u_{i-k}\big)\bigg) \\ &= -c_{1,\alpha}\bigg(a_0 u_i + \sum_{j=i+1}^{N-1} a_{j-i}\, u_j + \sum_{j=1}^{i-1} a_{i-j} u_j + \bar{b}_i\bigg),\end{aligned} \tag{18}$$

for $1 \le i \le N-1$, where the coefficients

$$a_0 = -2\bigg(\sum_{j=1}^{N} a_j + \frac{1}{\alpha L^\alpha}\bigg), \quad \text{with} \quad a_j = \frac{1}{\xi_j^\gamma} \begin{cases} \zeta\, \omega_0^p + \omega_1^p, & \text{if } j = 1, \\ \omega_j^p, & \text{if } j \neq 0, 1. \end{cases}$$

The term $\bar{b}_i$ comes from boundary conditions and is defined by

$$\bar{b}_i = \sum_{j=N}^{N+i} a_{j-i}\, g(x_j) + \sum_{j=i-N}^{0} a_{i-j}\, g(x_j) + \int_L^\infty \big[g(x_i - \xi) + g(x_i + \xi)\big] \frac{d\xi}{\xi^{1+\alpha}}.$$

If homogeneous boundary conditions are considered, $\bar{b}_i \equiv 0$ for $1 \le i \le N-1$. The scheme in (18) provides a general structure of our operator factorization method, and the dependence of basis function φ_k^p is counted through the value of weight integral ω_k^p. In other words, if different basis functions are used, we only need to update the value of ω_k^p but keep the scheme and its computer implementation unchanged.

Let $\mathbf{u} = \big(u_1, u_2, \ldots, u_{N-1}\big)^T$ and $\mathbf{b} = -c_{1,\alpha}\big(\bar{b}_1, \bar{b}_2, \ldots, \bar{b}_{N-1}\big)^T$. The discretization of the one-dimensional fractional Laplacian in (18) can be written into a matrix-vector form, i.e., $(-\Delta)_h^{\frac{\alpha}{2}} \mathbf{u} = A^{(1)}\mathbf{u} + \mathbf{b}$, where

$$A^{(1)} = -c_{1,\alpha} \begin{pmatrix} a_0 & a_1 & \cdots & a_{N-3} & a_{N-2} \\ a_1 & a_0 & a_1 & \cdots & a_{N-3} \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ a_{N-3} & \cdots & a_1 & a_0 & a_1 \\ a_{N-2} & a_{N-3} & \cdots & a_1 & a_0 \end{pmatrix}.$$

It is clear that $A^{(1)}$ is a symmetric Toeplitz matrix, and thus the product $A^{(1)}\mathbf{u}$ can be computed efficiently by using the fast Fourier transform (FFT) with computational cost $\mathcal{O}\big((N-1)\log(N-1)\big)$.

Remark 1. Our method interpolating function $\Phi_{1,\gamma}(x,\xi)$ with respect to ξ is fundamentally different from that proposed in [15], although both use Lagrange basis functions φ^p (for $p \in \mathbb{N}^0$). The method in [15] has an α-dependent accuracy – $\mathcal{O}(h^{2-\alpha})$ for linear basis functions φ^1, while $\mathcal{O}(h^{3-\alpha})$ for quadratic basis functions φ^2. In contrast to it, our method with $\gamma = 2$ can achieve a uniform accuracy for any $\alpha \in (0,2)$. Specifically, the accuracy is $\mathcal{O}(h^2)$ for both constant basis φ^0 and linear basis φ^1, while $\mathcal{O}(h^4)$ for quadratic basis functions φ^2. Figure 1 presents the comparison of our method and that in [15], and more discussions and illustrations can be found in Section 3.1.

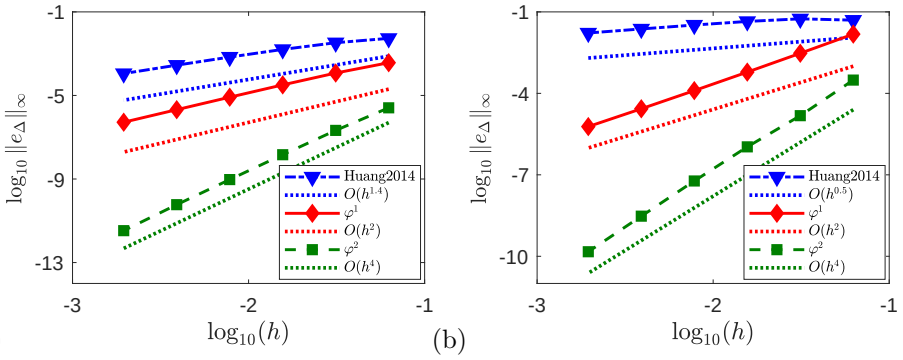

FIGURE 1. Comparison of our method with linear basis (i.e., φ^1) or quadratic basis (i.e., φ^2) and the method in [15] with linear basis (i.e., Huang2014) in approximating function $(-\Delta)^{\frac{\alpha}{2}}u(x)$, where $u(x) = (1-x^2)_+^6$ and the error function is defined in (27). (a) $\alpha = 0.6$; (b) $\alpha = 1.5$.

2.2. **Two-dimensional cases.** Our method uses Lagrange interpolation to the central difference quotient function and can be easily generalized to high dimensions with corresponding changes to ansatz in (11). We will show the two-dimensional scheme in this section. In two-dimensional ($d = 2$) cases, we have

$$\Phi_{2,\gamma}(\mathbf{x},\boldsymbol{\xi}) = \Big(\sum_{\mathbf{m}\in\varkappa_1} u\big(x^{(1)} + (-1)^{m_1}\xi^{(1)},\, x^{(2)} + (-1)^{m_2}\xi^{(2)}\big) - 4u(\mathbf{x}) \Big) \frac{1}{|\boldsymbol{\xi}|^\gamma}. \quad (19)$$

Define mesh size $h = L/N$ with a positive integer N, and introduce grid points

$$x_i^{(s)} = a_s + ih, \qquad \xi_i^{(s)} = ih, \qquad \text{for} \quad i = 0,1,\ldots,N, \quad s = 1,2.$$

Denote point $\boldsymbol{\xi}_{kl} = \big(\xi_k^{(1)}, \xi_l^{(2)}\big)$, for $0 \le k,l \le N$. We assume the two-dimensional ansatz:

$$\Phi_{2,\gamma}(\mathbf{x},\,\boldsymbol{\xi}) = \sum_{k=0}^{N}\sum_{l=0}^{N} \Phi_{2,\gamma}(\mathbf{x},\,\boldsymbol{\xi}_{kl})\,\varphi_k^p\big(\xi^{(1)}\big)\varphi_l^p\big(\xi^{(2)}\big), \qquad \text{for} \quad \boldsymbol{\xi} \in \Upsilon. \quad (20)$$

Similar to the one-dimensional cases, function $\Phi_{2,\gamma}(\mathbf{x},\boldsymbol{\xi}_{00})$ is defined in a limit sense as $\boldsymbol{\xi} \to (0,0)$. Specifically, we approximate it as

$$\begin{aligned} \Phi_{2,2}(\mathbf{x},\boldsymbol{\xi}_{00}) &= \lim_{\boldsymbol{\xi}\to(0,0)} \Phi_{2,2}(\mathbf{x},\boldsymbol{\xi}) \\ &\approx \Phi_{2,2}(\mathbf{x},\boldsymbol{\xi}_{10}) + \Phi_{2,2}(\mathbf{x},\boldsymbol{\xi}_{01}) - \Phi_{2,2}(\mathbf{x},\boldsymbol{\xi}_{11}), \quad \text{for} \quad \gamma = 2. \end{aligned}$$

For $\gamma \in (\alpha, 2)$, we get

$$\Phi_{2,\gamma}(\mathbf{x}, \boldsymbol{\xi}_{00}) = \lim_{\boldsymbol{\xi}\to(0,0)} \Phi_{2,\gamma}(\mathbf{x}, \boldsymbol{\xi}) = \Phi_{2,2}(\mathbf{x}, \boldsymbol{\xi}_{00}) \lim_{\boldsymbol{\xi}\to(0,0)} |\boldsymbol{\xi}|^{2-\gamma} = 0, \quad \text{for } \gamma \in (\alpha, 2).$$

Substituting ansatz (20) into $\mathcal{L}^{\alpha}_{\Upsilon} u$ with $d = 2$, we obtain

$$\mathcal{L}^{\alpha}_{\Upsilon,h} u(\mathbf{x}) = \sum_{k=0}^{N}\sum_{l=0}^{N} \Phi_{2,\gamma}(\mathbf{x}, \boldsymbol{\xi}_{kl}) \underbrace{\int_0^L \int_0^L \varphi_k^p(\xi^{(1)}) \varphi_l^p(\xi^{(2)}) \frac{d\boldsymbol{\xi}}{|\boldsymbol{\xi}|^{2+\alpha-\gamma}}}_{\omega_{kl}^p}, \tag{21}$$

for $\mathbf{x} \in \Omega$. The weight integral ω_{kl}^p is actually on the intersection region of the supports of two basis function φ_k^p and φ_l^p, instead of $[0, L]\times[0, L]$. Different from one-dimensional cases, it is challenging to obtain the analytical results of these integrals, and thus we will use numerical quadrature rules to compute them. Noticing the approximation of $\Phi_{2,\gamma}(\mathbf{x}, \boldsymbol{\xi}_{00})$, we can further formulate (21) as:

$$\begin{aligned}\mathcal{L}^{\alpha}_{\Upsilon,h} u(\mathbf{x}) = &\sum_{\substack{k=1\\k+l\neq 2}}^{N}\sum_{l=1}^{N} \omega_{kl}^p \, \Phi_{2,\gamma}(\mathbf{x}, \boldsymbol{\xi}_{kl}) + \sum_{k=2}^{N} \Big(\omega_{0k}^p \, \Phi_{2,\gamma}(\mathbf{x}, \boldsymbol{\xi}_{0k}) + \omega_{k0}^p \, \Phi_{2,\gamma}(\mathbf{x}, \boldsymbol{\xi}_{k0})\Big)\\ &+\big(\omega_{10} + \zeta\omega_{00}\big)\Phi_{2,\gamma}(\mathbf{x}, \boldsymbol{\xi}_{10}) + \big(\omega_{01} + \zeta\omega_{00}\big)\Phi_{2,\gamma}(\mathbf{x}, \boldsymbol{\xi}_{01})\\ &+\big(\omega_{11} - \zeta\omega_{00}\big)\Phi_{2,\gamma}(\mathbf{x}, \boldsymbol{\xi}_{11}).\end{aligned}$$

Without loss of generality, we assume that $N_1 = N$, and choose N_2 as the smallest integer such that $a_2 + N_2 h \geq b_2$. Denote $\mathbf{x}_{ij} = \big(x_i^{(1)}, x_j^{(2)}\big)$ and $u_{ij} = u(\mathbf{x}_{ij})$. Substituting $\mathbf{x} = \mathbf{x}_{ij}$ into $\mathcal{L}^{\alpha}_{\Upsilon,h} u(\mathbf{x})$ and $\mathcal{L}^{\alpha}_{\Upsilon_+^c} u(\mathbf{x})$, noticing the definition of $\Phi_{2,\gamma}(\mathbf{x}, \boldsymbol{\xi})$, we then obtain the numerical approximation the two-dimensional fractional Laplacian $(-\Delta)^{\frac{\alpha}{2}}$ with Dirichlet boundary conditions as follows:

$$\begin{aligned}\mathcal{L}^{\alpha}_{\Upsilon,h} u_{ij} = -c_{2,\alpha}\Bigg[&a_{00}\, u_{ij} + \sum_{k=0}^{i-1}\Bigg(\sum_{\substack{l=0\\k+l\neq 0}}^{j-1} a_{kl} u_{(i-k)(j-l)} + \sum_{l=1}^{N_2-(j+1)} a_{kl} u_{(i-k)(j+l)}\Bigg)\\ &+ \sum_{k=1}^{N_1-(i+1)}\Bigg(\sum_{l=0}^{j-1} a_{kl} u_{(i+k)(j-l)} + \sum_{l=1}^{N_2-(j+1)} a_{kl} u_{(i+k)(j+l)}\Bigg) + \bar{b}_{ij}\Bigg],\end{aligned} \tag{22}$$

for $1 \leq i \leq N_1 - 1$ and $1 \leq j \leq N_2 - 1$, where the coefficients

$$a_{kl} = \frac{1}{|\boldsymbol{\xi}_{kl}|^{\gamma}} \begin{cases} 2\omega_{kl}^p + \zeta\omega_{00}^p, & \text{if } (k,l) = (0,1) \text{ or } (1,0),\\ \omega_{kl}^p - \zeta\omega_{00}^p, & \text{if } k = l = 1,\\ 2\omega_{kl}^p, & \text{if } k = 0 \text{ or } l = 0, \text{ and } k + l \geq 2,\\ \omega_{kl}^p, & \text{otherwise,}\end{cases}$$

$$a_{00} = -2\sum_{k=1}^{N}\big(a_{k0} + a_{0k}\big) - 4\sum_{i=1}^{N}\sum_{j=1}^{N} a_{ij} - 4\int_{\Upsilon_+^c} \frac{d\boldsymbol{\xi}}{|\boldsymbol{\xi}|^{2+\alpha}}.$$

The term $\bar{b}_{ij}$ is defined as

$$\begin{aligned}\bar{b}_{ij} = &\sum_{l=-N}^{N}\Bigg(\sum_{k=i}^{N} a_{k|l|}\, g\big(\mathbf{x}_{(i-k)(j+l)}\big) + \sum_{k=N_1-i}^{N} a_{k|l|}\, g\big(\mathbf{x}_{(i+k)(j+l)}\big)\Bigg)\\ &+\sum_{k=0}^{i-1}\Bigg(\sum_{l=j}^{N} a_{kl}\, g\big(\mathbf{x}_{(i-k)(j-l)}\big) + \sum_{l=N_2-j}^{N} a_{kl}\, g\big(\mathbf{x}_{(i-k)(j+l)}\big)\Bigg)\end{aligned}$$

$$+\sum_{k=1}^{N_1-(i+1)}\Big(\sum_{l=j}^{N}a_{kl}\,g\big(\mathbf{x}_{(i-k)(j-l)}\big)+\sum_{l=N_2-j}^{N}a_{kl}\,g\big(\mathbf{x}_{(i-k)(j+l)}\big)\Big)$$
$$+\sum_{\mathbf{m}\in\varkappa_1}\int_{\Upsilon_+^c}g\big(\mathbf{x}_{ij}+(-1)^{\mathbf{m}}\circ\boldsymbol{\xi}\big)\frac{d\boldsymbol{\xi}}{|\boldsymbol{\xi}|^{2+\alpha}},$$

for $1\le i\le N_1-1$ and $1\le j\le N_2-1$.

We can further write the scheme in (22) into matrix-vector form. For $1\le j\le N_2-1$, denote the vector $\mathbf{u}_j^{(1)}=\big(u_{1j},u_{2j},\dots,u_{(N_1-1)j}\big)$, and let the block vector $\mathbf{u}=\big(\mathbf{u}_1^{(1)},\mathbf{u}_2^{(1)},\dots,\mathbf{u}_{N_2-1}^{(1)}\big)^T$, and the block vector $\mathbf{b}$ is defined in the same manner as $\mathbf{u}$ with entries $-c_{2,\alpha}b_{ij}$. Then the matrix-vector form of the scheme (22) is given by $(-\Delta)_h^{\frac{\alpha}{2}}\mathbf{u}=A^{(2)}\mathbf{u}+\mathbf{b}$, where $A^{(2)}$ is a block-Toeplitz–Toeplitz-block matrix defined as

$$A^{(2)}=\begin{pmatrix}A_0 & A_1 & \dots & A_{N_2-3} & A_{N_2-2}\\ A_1 & A_0 & A_1 & \cdots & A_{N_2-3}\\ \vdots & \ddots & \ddots & \ddots & \vdots\\ A_{N_2-3} & \dots & A_1 & A_0 & A_1\\ A_{N_2-2} & A_{N_2-3} & \dots & A_1 & A_0\end{pmatrix}$$

with each block A_j a symmetricToeplitz matrix

$$\mathrm{A}_j=\begin{pmatrix}a_{0j} & a_{1j} & \dots & a_{(N_1-3)j} & a_{(N_1-2)j}\\ a_{1j} & a_{0j} & a_{1j} & \cdots & a_{(N_1-3)j}\\ \vdots & \ddots & \ddots & \ddots & \vdots\\ a_{(N_1-3)j} & \dots & a_{1j} & a_{0j} & a_{1j}\\ a_{(N_1-2)j} & a_{(N_1-3)j} & \dots & a_{1j} & a_{0j}\end{pmatrix},$$

for $j=0,1,\dots,N_2-2$. It is easy to see that $A^{(2)}$ is a full positive definite matrix, which usually requires huge memory and computational costs for computing matrix-vector products. However, the multilevel Toeplitz structure of $A^{(2)}$ enables us to develop fast algorithms via fast Fourier transform and thus efficiently compute matrix-vector multiplication.

2.3. **Three-dimensional cases.** In this section, we generalize our method to three dimensions. In three-dimensional ($d=3$) cases, the function

$$\Phi_{3,\gamma}(\mathbf{x},\boldsymbol{\xi})=\frac{1}{|\boldsymbol{\xi}|^{\gamma}}\Big(-8u(\mathbf{x})$$
$$+\sum_{\mathbf{m}\in\varkappa_1}u\big(x^{(1)}+(-1)^{m_1}\xi^{(1)},\,x^{(2)}+(-1)^{m_2}\xi^{(2)},\,x^{(3)}+(-1)^{m_3}\xi^{(3)}\big)\Big).$$

Define mesh size $h=L/N$, introduce grid points

$$x_i^{(m)}=a_m+ih,\qquad \xi_i^{(m)}=ih,\qquad \text{for } i=0,1,\dots,N,\quad m=1,2,3,$$

and then denote point $\boldsymbol{\xi}_{lns}=\big(\xi_l^{(1)},\xi_n^{(2)},\xi_s^{(3)}\big)$, for $0\le l,n,s\le N$. Assume the three-dimensional ansatz:

$$\Phi_{3,\gamma}(\mathbf{x},\boldsymbol{\xi})=\sum_{l,\,n,\,s=0}^{N}\Phi_{3,\gamma}(\mathbf{x},\boldsymbol{\xi}_{lns})\,\varphi_l^p\big(\xi^{(1)}\big)\varphi_n^p\big(\xi^{(2)}\big)\varphi_s^p\big(\xi^{(3)}\big),\quad \text{for } \boldsymbol{\xi}\in\Upsilon,\qquad(23)$$

where $\Phi_{3,\gamma}(\mathbf{x},\boldsymbol{\xi}_{000})$ is defined in a limit sense as $\boldsymbol{\xi}\to(0,0,0)$. Following the same lines as in Section 2.1 , we obtain

$$\Phi_{3,2}(\mathbf{x},\boldsymbol{\xi}_{000})\approx\frac{5}{3}\sum_{\substack{l,\,n,\,s=0,1\\ l+n+s=1}}\Phi_{3,2}(\mathbf{x},\boldsymbol{\xi}_{lns})-\sum_{\substack{l,\,n,\,s=0,1\\ l+n+s>1}}\Phi_{3,2}(\mathbf{x},\boldsymbol{\xi}_{lns}),\quad \text{for}\ \ \gamma=2,$$

while for $\gamma\in(\alpha,2)$, we get

$$\Phi_{3,\gamma}(\mathbf{x},\boldsymbol{\xi}_{000})=\lim_{\boldsymbol{\xi}\to\mathbf{0}}\Big(\Phi_{3,2}(\mathbf{x},\,\boldsymbol{\xi})|\boldsymbol{\xi}|^{2-\gamma}\Big)=0,\quad \text{for}\ \ \gamma\in(\alpha,2).$$

Substituting ansatz (23) into $\mathcal{L}^{\alpha}_{\Upsilon}u$ with $d=3$, we obtain

$$\mathcal{L}^{\alpha}_{\Upsilon,h}u(\mathbf{x})=\sum_{l,\,n,\,s=0}^{N}\Phi_{3,\gamma}(\mathbf{x},\boldsymbol{\xi}_{lns})\,\omega^{p}_{lns},\tag{24}$$

where

$$\omega^{p}_{lns}=\int_0^L\int_0^L\int_0^L\varphi^{p}_{l}\big(\xi^{(1)}\big)\varphi^{p}_{n}\big(\xi^{(2)}\big)\varphi^{p}_{s}\big(\xi^{(3)}\big)\frac{d\boldsymbol{\xi}}{|\boldsymbol{\xi}|^{3+\alpha-\gamma}}.$$

Similar to the two-dimensional cases, the weight integral ω^{p}_{lns} is actually on the intersection region of the supports of three basis functions φ^{p}_{l}, φ^{p}_{n} and φ^{p}_{s}, instead of $[0,L]^3$. We then can reformulate the scheme in (24) as:

$$\begin{aligned}\mathcal{L}^{\alpha}_{\Upsilon,h}u(\mathbf{x})=&\sum_{l=2}^{N}\Big[\omega^{p}_{l00}\Phi_{3,\gamma}(\mathbf{x},\boldsymbol{\xi}_{l00})+\omega^{p}_{0l0}\Phi_{3,\gamma}(\mathbf{x},\boldsymbol{\xi}_{0l0})+\omega^{p}_{00l}\Phi_{3,\gamma}(\mathbf{x},\boldsymbol{\xi}_{00l})\Big]\\&+\sum_{\substack{l,\,n=1\\ l+n\neq2}}^{N}\Big[\omega^{p}_{0ln}\Phi_{3,\gamma}(\mathbf{x},\boldsymbol{\xi}_{0ln})+\omega^{p}_{l0n}\Phi_{3,\gamma}(\mathbf{x},\boldsymbol{\xi}_{l0n})+\omega^{p}_{ln0}\Phi_{3,\gamma}(\mathbf{x},\boldsymbol{\xi}_{ln0})\Big]\\&+\sum_{\substack{l,\,n,\,s=1\\ l+n+s\neq3}}^{N}\omega^{p}_{lns}\Phi_{3,\gamma}(\mathbf{x},\boldsymbol{\xi}_{lns})+\sum_{\substack{l,\,n,\,s=0,1\\ l+n+s=1}}^{N}\Big(\omega^{p}_{lns}+\frac{5}{3}\zeta\omega^{p}_{000}\Big)\Phi_{3,\gamma}(\mathbf{x},\boldsymbol{\xi}_{lns})\\&+\sum_{\substack{l,\,n,\,s=0,1\\ l+n+s>1}}^{N}\Big(\omega^{p}_{lns}-\zeta\omega^{p}_{000}\Big)\Phi_{3,\gamma}(\mathbf{x},\boldsymbol{\xi}_{lns}).\end{aligned}$$

Assume that $N_1=N$, and choose N_2, N_3 as the smallest integers such that $a_2+N_2h\geq b_2$ and $a_3+N_3h\geq b_3$. Denote $\mathbf{x}_{ij}=\big(x^{(1)}_i,x^{(2)}_j,x^{(3)}_k\big)$ and $u_{ijk}=u(\mathbf{x}_{ijk})$. Substituting $\mathbf{x}=\mathbf{x}_{ijk}$ into $\mathcal{L}^{\alpha}_{\Upsilon,h}u(\mathbf{x})$ and $\mathcal{L}^{\alpha}_{\Upsilon^{c}_{+}}u(\mathbf{x})$, noticing the definition of $\Phi_{3,\gamma}(\mathbf{x},\boldsymbol{\xi})$, we then obtain the numerical approximation of the three-dimensional fractional Laplacian $(-\Delta)^{\frac{\alpha}{2}}$ with Dirichlet boundary conditions as follows:

$$\begin{aligned}&(-\Delta)^{\frac{\alpha}{2}}_{h,\gamma}u_{ijk}\\&=-c_{3,\alpha}\Big[a_{000}\,u_{ijk}+\sum_{p=0,1}\Big(\sum_{l\in S^{p}_{i}}a_{l00}\,u_{[i+(-1)^{p}l]jk}+\sum_{n\in S^{p}_{j}}a_{0n0}u_{i[j+(-1)^{p}n]k}\\&\quad+\sum_{s\in S^{p}_{k}}a_{00s}u_{ij[k+(-1)^{p}s]}\Big)+\sum_{p,q=0,1}\Big(\sum_{s\in S^{q}_{k}}\sum_{n\in S^{p}_{j}}a_{0ns}u_{i[j+(-1)^{p}n][k+(-1)^{q}s]}\end{aligned}$$

$$+\sum_{s\in S_k^q}\sum_{l\in S_i^p} a_{l0s}u_{[i+(-1)^p l]j[k+(-1)^q s]}+\sum_{n\in S_j^q}\sum_{l\in S_i^p} a_{ln0}u_{[i+(-1)^p l][j+(-1)^q n]\,k}\Big)$$
$$+\sum_{p,q,r=0,1}\sum_{s\in S_k^r}\sum_{n\in S_j^q}\sum_{l\in S_i^p} a_{lns}u_{[i+(-1)^p l][j+(-1)^q n][k+(-1)^r s]}+\bar{b}_{ijk}\Big], \tag{25}$$

for $1\le i\le N_1-1$, $1\le j\le N_2-1$, and $1\le k\le N_3-1$, where the index sets

$$S_i^p=\big\{l\,|\,l\in\mathbb{N},\ 1\le i+(-1)^p l\le N_1-1\big\},$$
$$S_j^p=\big\{l\,|\,l\in\mathbb{N},\ 1\le j+(-1)^p l\le N_2-1\big\},$$
$$S_k^p=\big\{l\,|\,l\in\mathbb{N},\ 1\le k+(-1)^p l\le N_3-1\big\},\qquad p=0,\text{ or }1.$$

The term $\bar{b}_{ijk}$ can be obtained similarly to the two-dimensional cases, and the coefficients a_{lns} are given by

$$a_{lns}=\frac{1}{|\boldsymbol{\xi}_{lns}|^\gamma}\begin{cases}4\big(\omega_{lns}^p+\dfrac{5\zeta}{3}\omega_{000}^p\big), & \text{if two of } l,n,s=0,\ \text{and}\ \ l+n+s=1,\\ 2\big(\omega_{lns}^p-\zeta\omega_{000}^p\big), & \text{if one of } l,n,s=0,\ \text{and}\ \ l+n+s=2,\\ \omega_{lns}^p-\zeta\omega_{000}^p, & \text{if } l=n=s=1,\\ 4\,\omega_{lns}^p, & \text{if two of } l,n,s=0,\ \text{and}\ \ l+n+s\ge 2,\\ 2\,\omega_{lns}^p, & \text{if one of } l,n,s=0,\ \text{and}\ \ l+n+s>2,\\ \omega_{lns}^p, & \text{otherwise,}\end{cases}$$

$$a_{000}=-8\sum_{l,n,s=1}^{N}a_{lns}-4\sum_{l,n=1}^{N}\big(a_{ln0}+a_{l0n}+a_{0ln}\big)-2\sum_{l=1}^{N}\big(a_{l00}+a_{0l0}+a_{00l}\big)$$
$$-8\int_{\Upsilon_+^c}\frac{d\boldsymbol{\xi}}{|\boldsymbol{\xi}|^{3+\alpha}}.$$

We can further write the scheme in (25) into matrix-vector form. Denote vector $\mathbf{u}=\big(\mathbf{u}_1^{(2)},\mathbf{u}_2^{(2)},\ldots,\mathbf{u}_{N_3-1}^{(2)}\big)^T$ with $\mathbf{u}_k^{(2)}=\big(\mathbf{u}_{1k}^{(1)},\mathbf{u}_{2k}^{(1)},\ldots,\mathbf{u}_{(N_2-1)k}^{(1)}\big)$, and each block $\mathbf{u}_{jk}^{(1)}=\big(u_{1jk},u_{2jk},\ldots,u_{(N_1-1)jk}\big)$. Let the block vector $\mathbf{b}$ be defined in the same manner as $\mathbf{u}$ with entries $-c_{3,\alpha}\bar{b}_{ijk}$. Then, the matrix-vector form of (25) is given by $(-\Delta)^{\frac{\alpha}{2}}\mathbf{u}=A^{(3)}\mathbf{u}+\mathbf{b}$. Here, A_3 is the matrix representation of the 3D fractional Laplacian, i.e.,

$$A^{(3)}=\begin{pmatrix}A_0^{xy} & A_1^{xy} & \cdots & A_{N_3-3}^{xy} & A_{N_3-2}^{xy}\\ A_1^{xy} & A_0^{xy} & A_1^{xy} & \cdots & A_{N_3-3}^{xy}\\ \vdots & \ddots & \ddots & \ddots & \vdots\\ A_{N_3-3}^{xy} & \cdots & A_1^{xy} & A_0^{xy} & A_1^{xy}\\ A_{N_3-2}^{xy} & A_{N_3-3}^{xy} & \cdots & A_1^{xy} & A_0^{xy}\end{pmatrix},$$

where for $k = 0, 1, \ldots, N_3 - 2$, the block matrix

$$A_k^{xy} = \begin{pmatrix} A_{0k}^x & A_{1k}^x & \cdots & A_{(N_2-3)k}^x & A_{(N_2-2)k}^x \\ A_{1k}^x & A_{0k}^x & A_{1k}^x & \cdots & A_{(N_2-3)k}^x \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ A_{(N_2-3)k}^x & \cdots & A_{1k}^x & A_{0k}^x & A_{1k}^x \\ A_{(N_2-2)k}^x & A_{(N_2-3)k}^x & \cdots & A_{1k}^x & A_{0k}^x \end{pmatrix},$$

with

$$A_{jk}^x = \begin{pmatrix} a_{0jk} & a_{1jk} & \cdots & a_{(N_1-3)jk} & a_{(N_1-2)jk} \\ a_{1jk} & a_{0jk} & a_{1jk} & \cdots & a_{(N_1-3)jk} \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ a_{(N_1-3)jk} & \cdots & a_{1jk} & a_{0jk} & a_{1jk} \\ a_{(N_1-2)jk} & a_{(N_1-3)jk} & \cdots & a_{1jk} & a_{0jk} \end{pmatrix},$$

for $j = 0, 1, \ldots, N_2 - 2$, and $k = 0, 1, \ldots, N_3 - 2$. Similar to the 2D case, the matrix-vector product can be efficiently computed via the 3D FFT, and the computational cost is of $\mathcal{O}(M \log M)$, and the memory requirement is $\mathcal{O}(M)$ with $M = (N_1 - 1)(N_2 - 1)(N_3 - 1)$.

Remark 2 (Extension to other nonlocal operators)**.** Our method provides a general framework of operator factorization, and can be easily generalized to solve the nonlocal operators of the form:

$$\mathcal{L}u(\mathbf{x}) = \bar{c}_{d,\alpha} \int_{\mathbb{R}^d} \frac{u(\mathbf{x}) - u(\mathbf{y})}{|\mathbf{x} - \mathbf{y}|^{d+\alpha}} K(|\mathbf{x} - \mathbf{y}|) d\mathbf{y}, \tag{26}$$

where $K(r)$ denotes a kernel function, e.g., $K(r) = \exp(-\lambda r)$ in the tempered fractional Laplacian [14]. In this case, we can formulate the operator (26) as

$$\mathcal{L}u(\mathbf{x}) = -\bar{c}_{d,\alpha} \int_{\mathbb{R}_+^d} \Phi_{d,\gamma}(\mathbf{x}, \boldsymbol{\xi}) \frac{K(|\boldsymbol{\xi}|)}{|\boldsymbol{\xi}|^{d+\alpha-\gamma}} d\boldsymbol{\xi}$$

with $\Phi_{d,\gamma}(\mathbf{x}, \xi)$ defined in (6). We then follow the same procedure as for (7) to approximate it. Note that the extra kernel function changes weight $\mu_\gamma(\boldsymbol{\xi})$ and thus affects the values of ω^p, but the scheme structure remains the same as that of fractional Laplacian $(-\Delta)^{\frac{\alpha}{2}}$.

3. Numerical experiments.

In this section, we test the performance of our method in discretizing the fractional Laplacian $(-\Delta)^{\frac{\alpha}{2}}$ and in solving fractional Poisson equations. Numerical results from constant basis (φ^0), linear basis (φ^1), and quadratic basis (φ^2) functions are compared and discussed under different conditions of function u and power α. Moreover, nonhomogeneous Dirichlet boundary conditions will be considered, which have been rarely studied in the literature [8, 26, 3]. Unless otherwise stated, we will choose splitting parameter $\gamma = 2$ in our simulations. More discussion and comparisons of different splitting parameters can be found in Example 3.1.3.

3.1. Estimation of the fractional Laplacian.

First, we study the accuracy of our method in approximating the fractional Laplacian $(-\Delta)^{\frac{\alpha}{2}}$ on a bounded domain with extended (homogeneous or nonhomogeneous) Dirichlet boundary conditions. Here, we consider the one-dimensional cases with $\Omega = (-1, 1)$, and thus Dirichlet

boundary conditions are imposed on $\Omega^c = (-\infty, -1] \cup [1, \infty)$. Denote the error function as

$$e_\Delta(x) = (-\Delta)_h^{\frac{\alpha}{2}} u(x) - (-\Delta)^{\frac{\alpha}{2}} u(x), \qquad \text{for } \ x \in \Omega, \tag{27}$$

where $(-\Delta)_h^{\frac{\alpha}{2}}$ represents a numerical approximation of the fractional Laplacian $(-\Delta)^{\frac{\alpha}{2}}$. Then numerical accuracy for different function u will be studied in the following examples. In the first two examples, we choose function $u(x) = \left(1-x^2\right)_+^s$ with constant $s \in \mathbb{R}_+$ and then have the analytical result

$$(-\Delta)^{\frac{\alpha}{2}}(1-x^2)_+^s = \frac{2^\alpha \Gamma(s+1)\Gamma((1+\alpha)/2)}{\sqrt{\pi}\Gamma(s+1-\alpha/2)}\, {}_2F_1\Big(\frac{1+\alpha}{2}, -s+\frac{\alpha}{2}; \frac{1}{2}; |x|^2\Big),$$

for $x \in (-1, 1)$.

Example 3.1.1. Here, we are interested in understanding the minimum conditions for our method to be consistent. To this end, we consider function $u(x) = (1-x^2)_+^{1+\lfloor\alpha\rfloor}$, satisfying that $u \in C^{\lfloor\alpha\rfloor,1}(\bar{\Omega})$. That is, function $u(x) = (1-x^2)_+$ for $\alpha < 1$, while $u(x) = (1-x^2)_+^2$ for $1 \le \alpha < 2$. Table 1 shows numerical errors $\|e_\Delta\|_\infty$ and convergence rates for different basis function φ^p and power α.

h	1/16	1/32	1/64	1/128	1/256	1/512
$\alpha = 0.5$						
φ^0	7.5879e-3	5.3220e-3	3.7499e-3	2.6475e-3	1.8707e-3	1.3224e-3
	c.r.	0.5117	0.5051	0.5023	0.5010	0.5005
φ^1	7.8596e-3	5.4986e-3	3.8696e-3	2.7304e-3	1.9287e-3	1.3632e-3
	c.r.	0.5154	0.5069	0.5031	0.5014	0.5007
φ^2	1.3338e-2	9.3477e-3	6.5851e-3	4.6488e-3	3.2848e-3	2.3219e-3
	c.r.	0.5129	0.5054	0.5024	0.5011	0.5005
$\alpha = 1$						
φ^0	8.1722e-4	3.8342e-4	1.8911e-4	9.4360e-5	4.7189e-5	2.3604e-5
	c.r.	1.0918	1.0197	1.0029	0.9997	0.9994
φ^1	8.1722e-4	3.8342e-4	1.8911e-4	9.4360e-5	4.7189e-5	2.3604e-5
	c.r.	1.0918	1.0197	1.0029	0.9997	0.9994
φ^2	4.7698e-3	2.3572e-3	1.1717e-3	5.8413e-4	2.9164e-4	1.4571e-4
	c.r.	1.0169	1.0085	1.0042	1.0021	1.0011
$\alpha = 1.7$						
φ^0	3.6356e-3	1.8041e-3	1.2777e-3	1.0195e-3	8.3276e-4	6.8083e-4
	c.r.	1.0109	0.4977	0.3257	0.2919	0.2906
φ^1	2.5288e-3	5.4873e-4	5.8948e-4	5.0041e-4	4.0137e-4	3.2097e-4
	c.r.	2.2043	-0.1033	0.2363	0.3182	0.3225
φ^2	9.9878e-2	7.8950e-2	6.3253e-2	5.1024e-2	4.1302e-2	3.3489e-2
	c.r.	0.3392	0.3198	0.3010	0.3050	0.3025

TABLE 1. Numerical errors $\|e_\Delta\|_\infty$ and convergence rates (c.r.) in approximating function $(-\Delta)^{\frac{\alpha}{2}} u$ with $u = (1-x^2)_+^{1+\lfloor\alpha\rfloor}$ and basis function φ^p (for $p = 0, 1, 2$).

It shows that as mesh size h reduces, numerical errors decrease with a rate depending on power α. Specifically, all three basis functions φ^p (for $p = 0, 1, 2$) lead to the same convergence rate – $\mathcal{O}(h^{1-\alpha})$ for $\alpha < 1$, while $\mathcal{O}(h^{2-\alpha})$ for $\alpha \ge 1$.

Numerical errors are maximized around the boundary points $x = \pm 1$. In this case, the function u has limited smoothness at two points $x = \pm 1$, which creates a bottleneck for numerical accuracy (see similar observations of finite difference methods in [10, 13]). But, our method can achieve the optimal accuracy – $\mathcal{O}(h^2)$

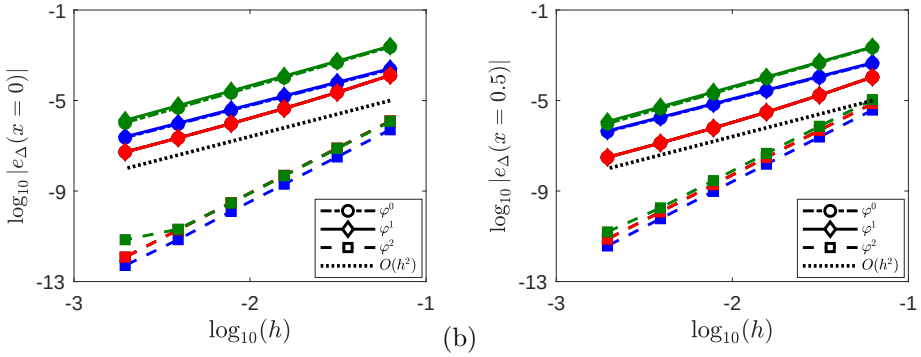

FIGURE 2. Numerical error $|e_\Delta(x)|$ at point $x = 0$ (left) and $x = 0.5$ (right) in approximating function $(-\Delta)^{\frac{\alpha}{2}} u$ with $u(x) = (1 - x^2)_+^{1+\lfloor\alpha\rfloor}$, where $\alpha = 0.5$ (blue), 1 (red), and 1.7 (green).

for constant basis φ^0 and linear basis φ^1, while $\mathcal{O}(h^4)$ for quadratic basis φ^2 – at inner points (see more discussion in Remark 4). To illustrate this, Figure 2 presents numerical errors at point $x = 0$ and $x = 0.5$, where an order line (i.e., dotted line) is included for easy comparison. We find that numerical errors from constant basis and linear basis are almost the same for different α, and they are identical if $\alpha = 1$. Numerical errors of quadratic basis φ^2 at inner points (e.g. $x = 0, 0.5$) are much smaller than those from constant and linear bases, suggesting the benefits of using high-degree basis functions. Moreover, comparing Figure 2 (a) and (b) shows that numerical error $|e_\Delta(x)|$ increases as x approaches the domain boundary, consistent with our earlier observations of maximum errors around boundary points.

The above results and our extensive studies suggest that the minimum consistency condition of our method in approximating the Dirichlet fractional Laplacian $(-\Delta)^{\frac{\alpha}{2}}$ depends on power α. For $m \in \mathbb{N}$ and $l \in (0, 1]$, we denote

$$C^{m,\,l}(\bar{\Omega}) = \left\{ u \in C^m(\bar{\Omega}) \;\middle|\; \sup_{\substack{x,\,y\in\bar{\Omega}\\ x\neq y}} \frac{|u^{(k)}(x) - u^{(k)}(y)|}{|x-y|^l} < \infty, \quad \text{for } k \in \mathbb{N} \text{ and } k \le m \right\}.$$

Then we can summarize the consistency results as follows.

Remark 3 (Consistency conditions). Let $(-\Delta)_h^{\frac{\alpha}{2}}$ be a numerical approximation to the Dirichlet fractional Laplacian $(-\Delta)^{\frac{\alpha}{2}}$ with small mesh size h. Our method with basis function φ^p (for $p = 0, 1$, or 2) is consistent if function $u \in C^{\lfloor\alpha\rfloor,\,\alpha-\lfloor\alpha\rfloor+\varepsilon}(\bar{\Omega})$ with small number $\varepsilon > 0$. Moreover, their local truncation errors satisfy

$$\|e_\Delta\|_\infty \le Ch^\varepsilon, \qquad \text{for } \alpha \in (0, 2)$$

with C a positive constant independent of h.

Our extensive studies show that the consistency result in Remark 3 is independent of splitting parameter γ, i.e., it holds for any $\gamma \in (\alpha, 2]$.

Example 3.1.2. In this example, we continue our study to test the accuracy of our method for functions that are smoother around boundary points. We are interested in understanding the optimal accuracy of our method with different basis functions and also the minimum conditions to achieve such an optimal accuracy.

To this end, let's first consider function $u = (1-x^2)_+^{2.1+\alpha}$. Table 2 presents numerical errors $\|e_\Delta\|_\infty$ and convergence rates for different basis functions φ^p. Compared to Table 1, numerical errors in this case are much smaller since function u is smoother over $\bar{\Omega}$. But, maximum errors are still found around two boundary points $x = \pm 1$. Both constant basis φ^0 and linear basis φ^1 have accuracy rate of $\mathcal{O}(h^2)$, while the quadratic basis φ^2 leads to a higher rate, i.e., $\mathcal{O}(h^{2.1})$. Moreover, numerical errors from quadratic basis are much smaller than those from constant and linear bases.

h	1/16	1/32	1/64	1/128	1/256	1/512
$\alpha = 0.5$						
φ^0	9.7624e-5	2.8827e-5	7.6872e-6	1.9687e-6	4.9667e-7	1.2457e-7
	c.r.	1.7598	1.9069	1.9653	1.9868	1.9953
φ^1	2.1391e-4	5.9663e-5	1.5540e-5	3.9426e-6	9.9077e-7	2.4817e-7
	c.r.	1.8421	1.9409	1.9787	1.9925	1.9972
φ^2	1.0716e-4	2.4168e-5	5.5431e-6	1.2823e-6	2.9789e-7	6.9347e-8
	c.r.	2.1486	2.1243	2.1120	2.1059	2.1029
$\alpha = 1$						
φ^0	5.9137e-4	7.5126e-5	9.4842e-6	2.0487e-6	6.4898e-7	1.7163e-7
	c.r.	2.9767	2.9857	2.2109	1.6584	1.9189
φ^1	5.9137e-4	7.5126e-5	9.4842e-6	2.0487e-6	6.4898e-7	1.7163e-7
	c.r.	2.9767	2.9857	2.2109	1.6584	1.9189
φ^2	2.4438e-4	5.4962e-5	1.2583e-5	2.9079e-6	6.7516e-7	1.5713e-7
	c.r.	2.1526	2.1270	2.1134	2.1067	2.1033
$\alpha = 1.7$						
φ^0	1.4385e-2	5.3211e-3	1.5476e-3	4.0205e-4	9.9477e-5	2.4066e-5
	c.r.	1.4348	1.7817	1.9446	2.0150	2.0474
φ^1	1.5206e-2	5.5501e-3	1.6318e-3	4.2867e-4	1.0729e-4	2.6267e-5
	c.r.	1.4540	1.7660	1.9285	1.9983	2.0302
φ^2	8.0506e-4	1.0070e-4	1.6011e-5	3.1025e-6	6.6674e-7	1.4996e-7
	c.r.	2.9991	2.6529	2.3675	2.2182	2.1526

TABLE 2. Numerical errors $\|e_\Delta\|_\infty$ and convergence rate (c.r.) in approximating function $(-\Delta)^{\frac{\alpha}{2}}u$ with $u = (1-x^2)_+^{2.1+\alpha}$ and basis function φ^p (for $p = 0, 1, 2$).

To further our understanding, we increase the smoothness of functions around boundary points. Figure 3 shows numerical errors for functions $u(x) = (1-x^2)_+^{3.1+\alpha}$ and $u = (1-x^2)_+^{4.1+\alpha}$. For both functions, the constant basis φ^0 and linear basis φ^1 remain the second-order accuracy, suggesting the best accuracy of these two basis functions is $\mathcal{O}(h^2)$. In contrast, the accuracy of quadratic basis φ^2 increases as function u becomes smoother, and the best accuracy is $\mathcal{O}(h^4)$ (see Figure 3 (b)). Figure 3 additionally suggests that the smaller the power α, the less the numerical

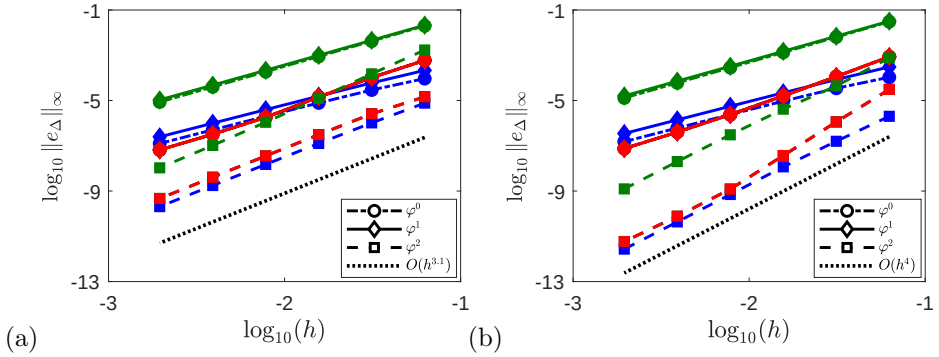

FIGURE 3. Numerical errors $\|e_\Delta\|_\infty$ in approximating function $(-\Delta)^{\frac{\alpha}{2}}u$, where $\alpha = 0.5$ (blue), 1 (red), and 1.7 (green). (a) $u = (1-x^2)_+^{3.1+\alpha}$; (b) $u = (1-x^2)_+^{4.1+\alpha}$.

errors. In Figure 4, we further compare our method with the finite difference method in the literature [10]. It shows that our method with linear basis φ^1 has similar

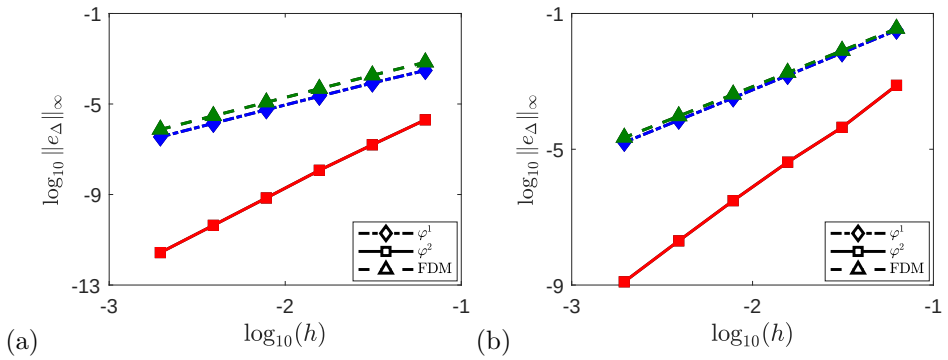

FIGURE 4. Comparison of our method with linear basis (i.e., φ^1) or quadratic basis (i.e., φ^2) and the finite difference method in [10] (i.e., FDM) in approximating function $(-\Delta)^{\frac{\alpha}{2}}u(x)$, where $u(x) = (1-x^2)_+^{4.1+\alpha}$, and $\alpha = 0.5$ (a) or 1.7 (b).

accuracy to the finite difference method, but our accuracy can be further increased if quadratic basis φ^2 is used.

From the above observations and our extensive studies, we summarize the optimal accuracy of our methods in the following remark.

Remark 4 (Optimal accuracy). Let $(-\Delta)_{h,2}^{\frac{\alpha}{2}}$ be a numerical approximation to the Dirichlet fractional Laplacian $(-\Delta)^{\frac{\alpha}{2}}$ with splitting parameter $\gamma = 2$ and small mesh size h.

(i) If function $u \in C^{2+\lfloor\alpha\rfloor,\,\alpha-\lfloor\alpha\rfloor+\varepsilon}(\bar{\Omega})$, the error function from basis function φ^p satisfies

$$\|e_\Delta\|_\infty \le Ch^2, \qquad \text{for} \quad \alpha \in (0,2),$$

for $p = 0$ or 1.

(ii) If function $u \in C^{4+\lfloor\alpha\rfloor,\,\alpha-\lfloor\alpha\rfloor+\varepsilon}(\bar{\Omega})$, the error function from quadratic basis φ^2 satisfies

$$\|e_\Delta\|_\infty \le Ch^4, \qquad \text{for} \quad \alpha \in (0,2).$$

Here, C is a positive constant independent of mesh size h, and $\varepsilon > 0$ is a small number.

Our studies show that the optimal accuracy can be achieved only when splitting parameter $\gamma = 2$. In contrast, other splitting parameters $\gamma \in (\alpha, 2)$ lead to larger numerical errors and lower accuracy rates. More comparisons and discussions of different splitting parameters can be found in next example.

Example 3.1.3. In this example, we consider infinitely smooth function $u = 1/(1+x^2)$ for $x \in \mathbb{R}$, and approximate $(-\Delta)^{\frac{\alpha}{2}}u(x)$ for $x \in (-1,1)$. It is equivalent to approximate the fractional Laplacian with extended nonhomogeneous boundary conditions $u = 1/(1+x^2)$ for $x \in (-1,1)^c$. Note that the exact solution is given by [26]

$$(-\Delta)^{\frac{\alpha}{2}}\left(\frac{1}{1+x^2}\right) = \Gamma(1+\alpha)\frac{\cos\left[(1+\alpha)\arctan(x)\right]}{(1+x^2)^{\frac{1+\alpha}{2}}}, \qquad \text{for} \quad x \in \mathbb{R}.$$

In addition to numerical accuracy, we will also study the effects of splitting parameter γ.

Table 3 shows that quadratic basis functions yield significantly smaller numerical errors than constant and linear bases. This is one main advantage of our method – when function u is smooth enough, our method enables us to increase the accuracy by using high-degree basis functions (e.g., φ^2). In terms of computer implementation, different basis functions only change the predefined values of ω_k^p, but do not affect the structures of main programs. Additionally, numerical errors in this example are maximized at point $x = 0$, different from those in Examples 3.1.1 and 3.1.2.

In Figure 5, we study the effects of splitting parameter γ on numerical accuracy for different basis function φ^p, where parameter $\gamma = 2,\ 1,\ 1+\alpha/2$, and α are considered. It shows that the splitting parameter $\gamma = 2$ leads to the smallest errors for each basis function. Moreover, when $\gamma = 2$ our method can achieve the optimal convergence rates, i.e., $\mathcal{O}(h^2)$ for φ^0 and φ^1, while $\mathcal{O}(h^4)$ for φ^2. In contrast, an α-dependent rate is observed if splitting parameter $\gamma \neq 2$. For instance, when $\gamma = 1+\alpha/2$ we find that the accuracy is $\mathcal{O}(h^{2-\alpha})$ for constant basis φ^0 and linear basis φ^1, while $\mathcal{O}(h^{3-\alpha})$ for quadratic basis φ^2. Generally, if $\gamma \in (\alpha, 2)$ the accuracy rate depends on power α, which is lower than that from $\gamma = 2$. Hence, we suggest to choose $\gamma = 2$ in practice.

3.2. **Solution of Poisson problems.** In this section, we further explore the performance of our method in solving the fractional Poisson equation and its generalization. Denote the error function in solution as

$$e_u(\mathbf{x}) = u_h(\mathbf{x}) - u(\mathbf{x}), \qquad \text{for} \quad \mathbf{x} \in \Omega,$$

where u_h and u represent numerical and exact solutions, respectively. As seen in Example 3.1.3, the splitting parameter $\gamma = 2$ outperforms other parameter $\gamma \in (\alpha, 2)$, and thus we will always take $\gamma = 2$ in the following examples.

Example 3.2.1 (1D fractional Poisson equation). Let the one-dimensional domain $\Omega = (-1,1)$. First, we consider the benchmark fractional Poisson problem,

h	1/16	1/32	1/64	1/128	1/256
$\alpha = 0.5$					
φ^0	2.0023e-5	5.4222e-6	1.3919e-6	3.5121e-7	8.8084e-8
	c.r.	1.8847	1.9618	1.9866	1.9954
φ^1	3.0000e-5	8.3090e-6	2.1718e-6	5.5132e-7	1.3857e-7
	c.r.	1.8522	1.9357	1.9780	1.9923
φ^2	1.7384e-7	1.2148e-8	7.8796e-10	4.9569e-11	2.6986e-12
	c.r.	3.8391	3.9464	3.9906	4.1991
$\alpha = 1$					
φ^0	1.1056e-4	1.7994e-5	3.2863e-6	6.6985e-7	1.4849e-7
	c.r.	2.6193	2.4530	2.2945	2.1735
φ^1	1.1056e-4	1.7994e-5	3.2863e-6	6.6985e-7	1.4849e-7
	c.r.	2.6193	2.4530	2.2945	2.1735
φ^2	8.0637e-7	2.5951e-8	8.3813e-10	2.7735e-11	9.3070e-13
	c.r.	4.9576	4.9525	4.9174	4.8972
$\alpha = 1.7$					
φ^0	2.2284e-3	4.6838e-4	9.8834e-5	2.0988e-5	4.4908e-6
	c.r.	2.2503	2.2446	2.2355	2.2245
φ^1	2.3784e-3	5.1283e-4	1.1135e-4	2.4403e-5	5.4026e-6
	c.r.	2.2135	2.2033	2.1900	2.1753
φ^2	3.1706e-5	1.6865e-6	8.9337e-8	4.7546e-9	2.6046e-10
	c.r.	4.2326	4.2386	4.2319	4.1902

TABLE 3. Numerical errors $\|e_\Delta\|_\infty$ and convergence rate (c.r.) in approximating function $(-\Delta)^{\frac{\alpha}{2}}u$ on (-1, 1) with $u(x) = 1/(1+x^2)$ and basis function φ^p (for $p = 0, 1, 2$).

i.e., choosing $f(x) = 1$ in (1) and $g(x) \equiv 0$ in (2). Its exact solution is given by

$$u(x) = -\frac{1}{\Gamma(1+\alpha)}(1-x^2)_+^{\frac{\alpha}{2}}, \qquad \text{for} \quad x \in \mathbb{R},$$

i.e., $u \in C^{0,\frac{\alpha}{2}}(\bar{\Omega})$. Table 4 presents numerical errors $\|e_u\|_\infty$ and convergence rates for different basis function φ^p and power α.

It shows that our method has an accuracy rate of $\mathcal{O}(h^{\frac{\alpha}{2}})$ for any $\alpha \in (0, 2)$, independent of basis function φ^p. In this case, the regularity of solution at boundary creates a bottleneck for numerical methods, and consequently using high-degree basis function does not necessarily benefit the accuracy. Note that the same convergence rate is observed for finite difference methods [10, 13, 14]. Generally, the larger the power α, the smoother the solution at boundary, the smaller the numerical errors. Similar to the observations in Section 3.1, the constant basis φ^0 and linear basis φ^1 have the same numerical errors if $\alpha = 1$, since their differentiation matrices for the fractional Laplacian $(-\Delta)^{\frac{1}{2}}$ are identical. The maximum numerical errors in Table 4 are found around two boundary points $x = \pm 1$. But, numerical errors at points far away from the boundary are much smaller, and moreover an accuracy rate of $\mathcal{O}(h)$ is observed at these points; see Figure 6 for errors at $x = 0$.

Next, we generalize our study and consider the Poisson problem (1)–(2) with extended homogeneous boundary conditions $g(x) \equiv 0$ and

$$f(x) = \frac{2^\alpha \Gamma(\frac{\alpha+1}{2})\Gamma(s+1)}{\sqrt{\pi}\,\Gamma(s+1-\frac{\alpha}{2})}\,{}_2F_1\Big(\frac{\alpha+1}{2}, -s+\frac{\alpha}{2}; \frac{1}{2}; x^2\Big), \quad \text{for} \quad x \in (-1,1), \qquad (28)$$

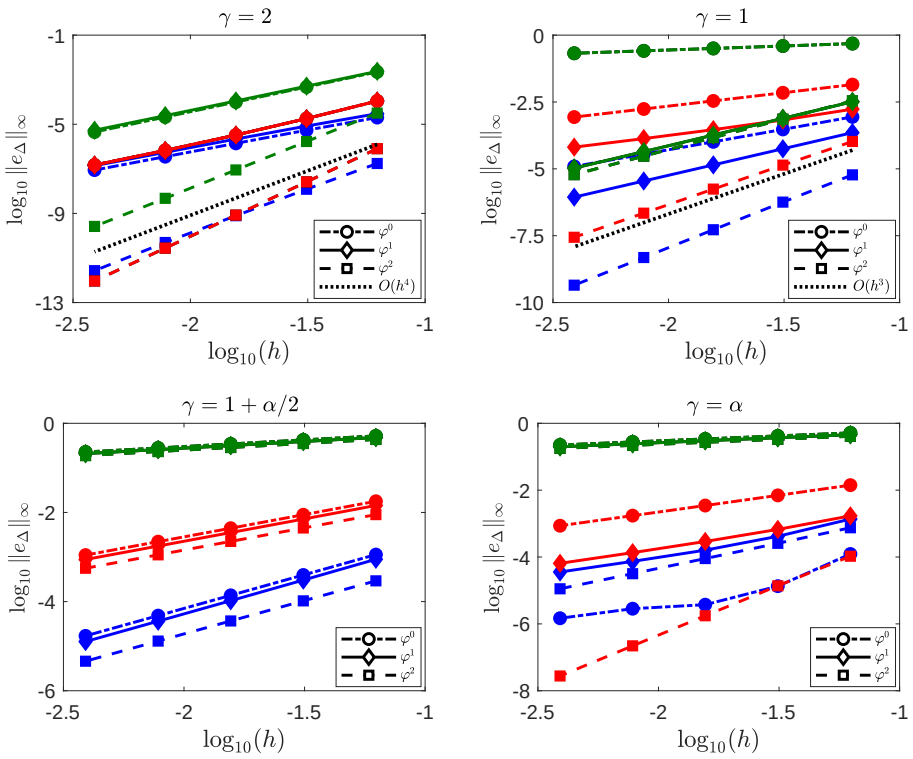

FIGURE 5. Effects of splitting parameter γ in approximating function $(-\Delta)^{\frac{\alpha}{2}}u$ on $(-1,1)$ with $u(x) = 1/(1+x^2)$, where $\alpha = 0.5$ (blue), 1 (red), and 1.7 (green).

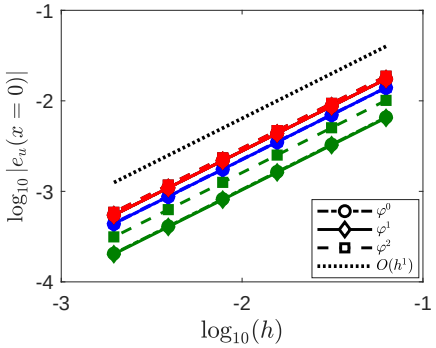

FIGURE 6. Numerical errors $|e_u(x)|$ at point $x = 0$ in solving the 1D fractional Poisson problem (1)–(2) with $f(x) = 1$ and $g(x) = 0$, where $\alpha = 0.5$ (blue), 1 (red), or 1.7 (green).

for $s > 0$, where ${}_2F_1$ denotes the hypergeometric Gauss function. The exact solution in this case is given by $u(x) = (1-x^2)_+^s$. It is clear that (28) is a general case of the benchmark problem (e.g., $s = \alpha/2$), and the regularity of solutions can be controlled by parameter s – the larger the value of s, the smoother the solution on $\bar{\Omega}$. Figure

h	1/16	1/32	1/64	1/128	1/256	1/512
$\alpha = 0.6$						
φ^0	7.4494e-2	5.9980e-2	4.8507e-2	3.9314e-2	3.1898e-2	2.5895e-2
	c.r.	0.3126	0.3063	0.3031	0.3016	0.3008
φ^1	7.5493e-2	6.0790e-2	4.9164e-2	3.9847e-2	3.2331e-2	2.6247e-2
	c.r.	0.3125	0.3062	0.3031	0.3016	0.3008
φ^2	8.4532e-2	6.8106e-2	5.5102e-2	4.4671e-2	3.6249e-2	2.9429e-2
	c.r.	0.3117	0.3057	0.3028	0.3014	0.3007
$\alpha = 1$						
φ^0	4.9166e-2	3.4508e-2	2.4310e-2	1.7158e-2	1.2121e-2	8.5671e-3
	c.r.	0.5107	0.5054	0.5027	0.5013	0.5007
φ^1	4.9166e-2	3.4508e-2	2.4310e-2	1.7158e-2	1.2121e-2	8.5671e-3
	c.r.	0.5107	0.5054	0.5027	0.5013	0.5007
φ^2	5.7935e-2	4.0695e-2	2.8682e-2	2.0248e-2	1.4306e-2	1.0112e-2
	c.r.	0.5096	0.5047	0.5023	0.5012	0.5006
$\alpha = 1.5$						
φ^0	1.6161e-2	9.5429e-3	5.6545e-3	3.3563e-3	1.9939e-3	1.1851e-3
	c.r.	0.7600	0.7550	0.7525	0.7513	0.7506
φ^1	1.5976e-2	9.4344e-3	5.5905e-3	3.3184e-3	1.9714e-3	1.1717e-3
	c.r.	0.7599	0.7550	0.7525	0.7513	0.7506
φ^2	2.2627e-2	1.3365e-2	7.9205e-3	4.7018e-3	2.7934e-3	1.6603e-3
	c.r.	0.7596	0.7548	0.7524	0.7512	0.7506

TABLE 4. Numerical errors $\|e_u\|_\infty$ and convergence rate (c.r.) in solving the 1D Poisson problem on $\Omega = (-1, 1)$, where $f(x) = 1$ in (1) and $g(x) = 0$ in (2).

7 shows numerical errors for different parameters $s = \alpha, 2, 3$, and 4, while Figure 8 further compares our methods with the finite difference method in [10]. From them, we find that

i). If $s \leq 2$, all three basis functions lead to an s-dependent accuracy, i.e., $\|e_u\|_\infty \sim \mathcal{O}(h^s)$. In this case, the regularity of solution plays a dominant role in determining the accuracy of numerical methods (see similar observations for finite difference method in [10, Table 3].)

ii). If $s > 2$, the solution $u \in C^2(\bar{\Omega})$. In this case, both constant basis φ^0 and linear basis φ^1 functions achieve the optimal accuracy of $\|e_u\|_\infty \sim \mathcal{O}(h^2)$. Moreover, their numerical errors are almost the same (see Figure 7 (b)–(d)). In contrast, the quadratic basis φ^2 function can achieve a higher accuracy rate $\mathcal{O}(h^s)$ for $2 < s < 4$ (see Figure 7 (c)–(d)).

iii). If $s \geq 4$, the quadratic basis φ^2 has much smaller numerical errors than those two basis functions and reaches the optimal accuracy rate $\|e_u\|_\infty \sim \mathcal{O}(h^4)$.

The above studies suggest when choosing basis functions, one should take the regularity of solution into account so as to achieve the best performance of our method.

Example 3.2.2 (1D tempered fractional Poisson equation). Remark 2 shows that our method can be easily generalized to study other nonlocal operators of form in (26). To illustrate this, we will apply our method to study the Poisson equation with tempered fractional Laplacian, i.e. choosing kernel function $K(|x - y|) = e^{-\lambda|x-y|}$ in (26). The tempered fractional Laplacian is used to study the coexistence

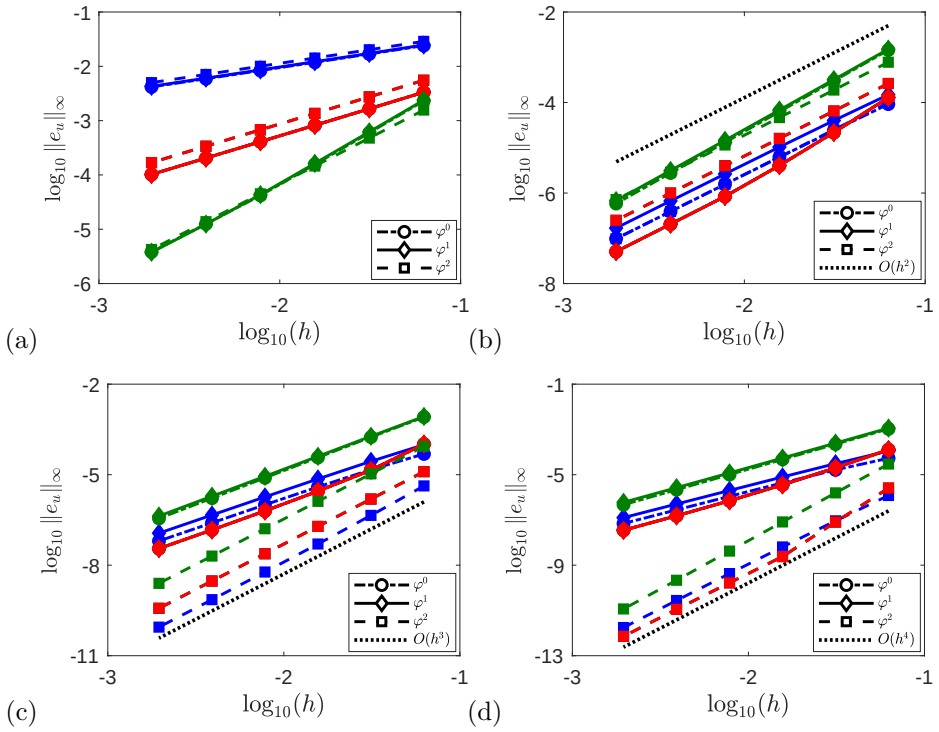

FIGURE 7. Numerical errors $\|e_u\|_\infty$ in solving the 1D Poisson problem (1)–(2) with $g(x) = 0$ and $f(x)$ in (28), where the exact solution is $u(x) = (1-x^2)_+^s$. From (a) to (d): $s = \alpha$, 2, 3, 4, where $\alpha = 0.5$ (blue), 1 (red), or 1.7 (green).

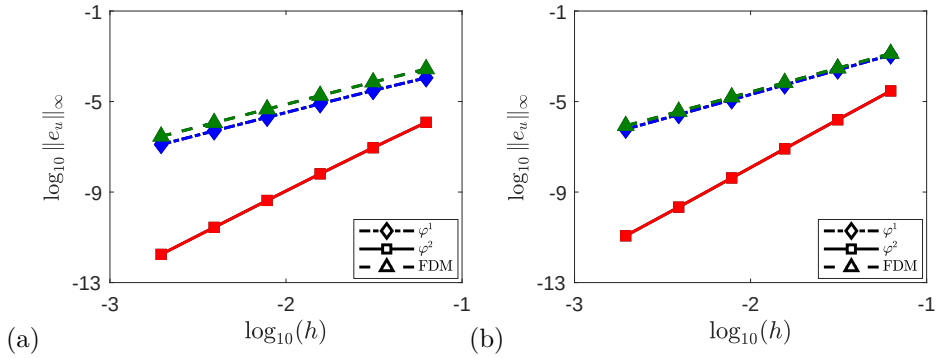

FIGURE 8. Comparison of our method with linear basis (i.e., φ^1) or quadratic basis (i.e., φ^2) and the finite difference method in [10] (i.e., FDM) in solving the fractional Poisson equation with exact solution $u(x) = (1-x^2)_+^4$, where $\alpha = 0.5$ (a) or 1.7 (b).

of normal and anomalous diffusion in many complex systems, where $\lambda \geq 0$ is a model parameter that mediates these two diffusion (see [14] and references therein for more

discussion). In our example, the right hand side function $f(x)$ is chosen such that the exact solution is given by $u(x) = (1-x^2)_+^2$ for $x \in \mathbb{R}$.

Figure 9 shows numerical errors for $\lambda = 0.5$ and $\lambda = 1$. Note that when $\lambda = 0$, it reduces to the fractional Poisson problem studied in Figure 7 (b). It shows that

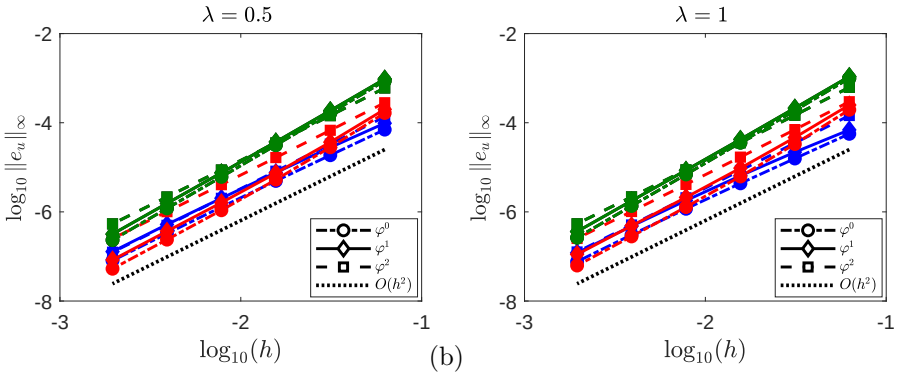

FIGURE 9. Numerical errors $\|e_u\|_\infty$ in solving the 1D tempered fractional Poisson problem with exact solution $u(x) = (1-x^2)_+^2$, where $\alpha = 0.6$ (blue), 1 (red), or 1.5 (green).

the uniform second-order accuracy is achieved for both $\lambda = 0.5$ and $\lambda = 1$, same rate as in Figure 7 (b) for $\lambda = 0$. Moreover, numerical errors are insensitive to parameter λ. The computer implementation of this problem is the same as that for the fractional Poisson problem in Example 3.2.1, except the values of ω_k^p. Due to the tempered term $e^{-\lambda|x-y|}$, the integrals of ω_k^p in (15) cannot be analytically obtained, but they can be pre-calculated with numerical quadrature rules.

Example 3.2.3 (2D fractional Poisson equation). In this example, we consider the two-dimensional Poisson problem (1)–(2) on a square domain $\Omega = (-1,1)^2$ with extended nonhomogeneous boundary conditions:

$$g(\mathbf{x}) = e^{-|\mathbf{x}|^2}, \qquad \text{for } \mathbf{x} \in \Omega^c. \tag{29}$$

The right hand function in (1) is chosen as

$$f(\mathbf{x}) = 2^\alpha \Gamma\Big(1+\frac{\alpha}{2}\Big)\, {}_1F_1\Big(1+\frac{\alpha}{2}; 1; -|\mathbf{x}|^2\Big), \qquad \text{for } \mathbf{x} \in \Omega, \tag{30}$$

where ${}_1F_1$ denotes the confluent hypergeometric function. Compared to homogeneous cases, nonhomogeneous boundary conditions require extra computation of integrals involving nonzero $g(\mathbf{x})$, e.g., the second term in $\mathcal{L}^\alpha_{\Upsilon_+^c}$, which can be accurately computed with traditional quadrature rules. Since the discretization matrix has multilevel Toeplitz structure, the resulting linear system can be solved with fast algorithms based on the fast Fourier transforms.

Figure 10 shows numerical errors in solution for $\alpha = 0.7$ and 1.9 with linear basis function φ^1 and mesh size $h = 1/64$. It is clear that numerical errors reduce as $|\mathbf{x}|$ increases, and the maximum errors are obtained at point $\mathbf{x} = (0,0)$. Moreover, it shows that the errors of $\alpha = 1.9$ are smaller than those of $\alpha = 0.7$. In Table 5, we further explore numerical errors $\|e_u\|_\infty$ and convergence rates of our method for different α. It shows that our method with linear basis function has the second order of accuracy for any $\alpha \in (0,2)$, but the smaller the power α, the less

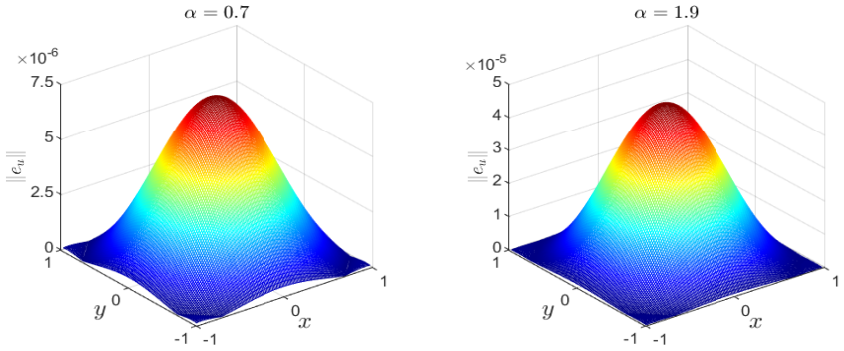

FIGURE 10. Numerical errors in the solution of the 2D fractional Poisson problems with basis function φ^1 and mesh size $h = 1/64$.

h	1/4	1/8	1/16	1/32	1/64	1/128
$\alpha = 0.2$	5.3181e-4	1.1946e-4	2.8883e-5	7.1505e-6	1.7827e-6	4.4531e-7
	c.r.	2.1544	2.0482	2.0141	2.0040	2.0011
$\alpha = 0.7$	2.3855e-3	5.1805e-4	1.2151e-4	2.9565e-5	7.3092e-6	1.8190e-6
	c.r.	2.2031	2.0921	2.0391	2.0161	2.0065
$\alpha = 1$	3.9406e-3	8.3747e-4	1.9083e-4	4.5384e-5	1.1056e-5	2.7276e-6
	c.r.	2.2343	2.1338	2.0720	2.0374	2.0191
$\alpha = 1.4$	6.9983e-3	1.4880e-3	3.2910e-4	7.5175e-5	1.7618e-5	4.2102e-6
	c.r.	2.2337	2.1767	2.1302	2.0932	2.0651
$\alpha = 1.9$	1.4264e-2	3.3824e-3	8.0943e-4	1.9424e-4	4.6676e-5	1.1230e-5
	c.r.	2.0762	2.0631	2.0591	2.0571	2.0554

TABLE 5. Numerical errors $\|e_u\|_\infty$ and convergence rate (c.r.) in solving the 2D Poisson problem on $\Omega = (-1,1)^2$ with f and g defined in (29)–(30), where linear basis φ^1 is used.

the numerical errors. Extensive studies show that the performance of our method in two-dimensional cases are similar to those in one dimension, and we will omit showing details for brevity.

4. **Conclusions and discussion.** We proposed a new class of operator factorization methods for discretizing the integral fractional Laplacian $(-\Delta)^{\frac{\alpha}{2}}$ in (3). The performance of our method in approximating the fractional Laplacian and solving the fractional Poisson problems was detailedly examined. The differentiation matrix resulting from our method is of symmetric (multilevel) Toeplitz structure, which enables us to save memory cost and design efficient algorithms via the fast Fourier transforms. Compared to other existing methods, our method has flexibility to increase numerical accuracy by using high-degree basis functions. Moreover, the application of different basis functions only changes the entries of differentiation matrix, but does not affect its (multilevel) Toeplitz structure.

Numerical studies show that for sufficiently smooth functions, our method with constant basis φ^0 and linear basis φ^1 has an optimal accuracy of $\mathcal{O}(h^2)$, and this rate can be improved to $\mathcal{O}(h^4)$ if quadratic basis φ^2 is used. This accuracy can

be further increased if higher degree basis function φ^p is used. When approximating operator $(-\Delta)^{\frac{\alpha}{2}}$, the minimum condition to achieve the above optimal accuracy is $u \in C^{2+\lfloor\alpha\rfloor,\,\alpha-\lfloor\alpha\rfloor+\varepsilon}(\bar{\Omega})$ for constant and linear basis functions, while $u \in C^{4+\lfloor\alpha\rfloor,\,\alpha-\lfloor\alpha\rfloor+\varepsilon}(\bar{\Omega})$ for quadratic basis with $\varepsilon > 0$ a small constant. While solving fractional Poisson problems with solution $u \in C^{m,l}(\bar{\Omega})$ for $m \in \mathbb{N}$ and $0 < l < 1$, our method has an accuracy of $\mathcal{O}\big(h^{\min\{m+l,\,2\}}\big)$ for constant and linear basis functions, while $\mathcal{O}\big(h^{\min\{m+l,\,4\}}\big)$ for quadratic basis functions. Numerical experiments showed that choosing splitting parameter $\gamma = 2$ leads to the best performance of our method, and thus this optimal parameter $\gamma = 2$ should be always taken in our method. Our additional study on the tempered fractional Poisson problem confirmed that our method can be readily applied to a broader class of nonlocal operators.

Appendix A. **Calculation of weight functions ω^p.** In one-dimensional cases, the weight integral in (15) can be found analytically. For convenience of the readers, we will summarize their analytical results of ω_k^p for $p = 0, 1, 2$ as below.

For $p = 0$, the weight integral ω_k^0 in (15) is calculated as

$$\omega_k^0 = \frac{1}{\sigma_0}\begin{cases} \xi_{1/2}^{\sigma_0}, & \text{if } \ k = 0,\\ \xi_{N}^{\sigma_0} - \xi_{N-1/2}^{\sigma_0}, & \text{if } \ k = N,\\ \xi_{k+1/2}^{\sigma_0} - \xi_{k-1/2}^{\sigma_0}, & \text{otherwise,}\end{cases}$$

where we denote $\sigma_0 = \gamma - \alpha$. For $p = 1$, the weight integral ω_k^1 in (15) is calculated as

$$\omega_k^1 = \frac{h^{-1}}{\sigma_1\sigma_0}\begin{cases} h^{\sigma_1}, & \text{if } \ k = 0,\\ \xi_{N-1}^{\sigma_1} - \xi_{N}^{\sigma_1} + \sigma_1 h\,\xi_N^{\sigma_0}, & \text{if } \ k = N,\\ \xi_{k+1}^{\sigma_1} + \xi_{k-1}^{\sigma_1} - 2\xi_k^{\sigma_1}, & \text{otherwise,}\end{cases}$$

where we denote $\sigma_1 = \sigma_0 + 1$. While $p = 2$, the weight integral ω_k^2 in (15) is calculated as

$$\omega_k^2 = \frac{1}{2h^2}\begin{cases} \dfrac{\xi_2^{\sigma_2}}{\sigma_2} - \xi_3\dfrac{\xi_2^{\sigma_1}}{\sigma_1} + 2h^2\dfrac{\xi_2^{\sigma_0}}{\sigma_0}, & \text{if } \ k = 0,\\ \dfrac{\xi_N^{\sigma_2} - \xi_{N-2}^{\sigma_2}}{\sigma_2} - 2\xi_{N-\frac{3}{4}}\dfrac{\xi_N^{\sigma_1} - \xi_{N-2}^{\sigma_1}}{\sigma_1} + \xi_{N-1}\xi_{N-2}\dfrac{\xi_N^{\sigma_0} - \xi_{N-2}^{\sigma_0}}{\sigma_0}, & \text{if } \ k = N,\\ -2\dfrac{\xi_{k+1}^{\sigma_2} - \xi_{k-1}^{\sigma_2}}{\sigma_2} + 4\xi_k\dfrac{\xi_{k+1}^{\sigma_1} - \xi_{k-1}^{\sigma_1}}{\sigma_1} - 2\xi_{k+1}\xi_{k-1}\dfrac{\xi_{k+1}^{\sigma_0} - \xi_{k-1}^{\sigma_0}}{\sigma_0}, & \text{if } \ 0 < k < N, \ \text{odd},\\ \dfrac{\xi_{k+2}^{\sigma_2} - \xi_{k-2}^{\sigma_2}}{\sigma_2} - \dfrac{2}{\sigma_1}\big[\xi_{k-\frac{1}{2}}\big(\xi_k^{\sigma_1} - \xi_{k-2}^{\sigma_1}\big) + \xi_{k+\frac{1}{2}}\big(\xi_{k+2}^{\sigma_1} - \xi_k^{\sigma_1}\big)\big] & \\ \quad + \dfrac{1}{\sigma_0}\big[\xi_{k-2}\xi_{k-1}\big(\xi_k^{\sigma_0} - \xi_{k-2}^{\sigma_0}\big) + \xi_{k+2}\xi_{k+1}\big(\xi_{k+2}^{\sigma_0} - \xi_k^{\sigma_0}\big)\big], & \text{if } \ 0 < k < N, \ \text{even},\end{cases}$$

where $\sigma_2 = \sigma_0 + 2$.

In high dimensions, the weight integral can not be calculated analytically, and thus numerical quadrature rules will be used to compute these integrals.

Acknowledgments. The authors thank Dr. Siwei Duo for her discussion and contributions at the initial stage of this project. We appreciate two anonymous reviewers for their valuable comments that greatly help to improve this manuscript.

REFERENCES

[1] G. Acosta, F. M. Bersetche and J. P. Borthagaray, A short FE implementation for a 2d homogeneous Dirichlet problem of a fractional Laplacian, *Comput. Math. Appl.*, **74** (2017), 784–816.

[2] G. Acosta and J. P. Borthagaray, A fractional Laplace equation: Regularity of solutions and finite element approximations, *SIAM J. Numer. Anal.*, **55** (2017), 472–495.

[3] G. Acosta, J. P. Borthagaray and N. Heuer, Finite element approximations of the nonhomogeneous fractional Dirichlet problem, *IMA J. Numer. Anal.*, **39** (2019), 1471–1501.

[4] M. Ainsworth and C. Glusa, Towards an efficient finite element method for the integral fractional Laplacian on polygonal domains, *Contemporary Computational Mathematics—a Celebration of the 80th Birthday of Ian Sloan*, Vol. 1, 2, 17–57, Springer, Cham, 2018.

[5] U. Biccari, M. Warma and E. Zuazua, Local elliptic regularity for the Dirichlet fractional Laplacian, *Adv. Nonlinear Stud.*, **17** (2017), 387–409.

[6] K. Bogdan, The boundary Harnack principle for the fractional Laplacian, *Studia Math.*, **123** (1997), 43–80.

[7] A. Bonito and J. E. Pasciak, Numerical approximation of fractional powers of regularly accretive operators, *IMA J. Numer. Anal.*, **37** (2017), 1245–1273.

[8] J. Burkardt, Y. Wu and Y. Zhang, A unified meshfree pseudospectral method for solving both classical and fractional PDEs, *SIAM J. Sci. Comput.*, **43** (2021), A1389–A1411.

[9] Q. Du, M. Gunzburger, R. B. Lehoucq and K. Zhou, Analysis and approximation of nonlocal diffusion problems with volume constraints, *SIAM Rev.*, **54** (2012), 667–696.

[10] S. Duo, H. W. van Wyk and Y. Zhang, A novel and accurate finite difference method for the fractional Laplacian and the fractional Poisson problem, *J. Comput. Phys.*, **355** (2018), 233–252.

[11] S. Duo, H. Wang and Y. Zhang, A comparative study on nonlocal diffusion operators related to the fractional Laplacian, *Discrete Contin. Dyn. Syst. Ser. B*, **24** (2019), 231–256.

[12] S. Duo and Y. Zhang, Computing the ground and first excited states of the fractional Schrödinger equation in an infinite potential well, *Commun. Comput. Phys.*, **18** (2015), 321–350.

[13] S. Duo and Y. Zhang, Accurate numerical methods for two and three dimensional integral fractional Laplacian with applications, *Comput. Methods. Appl. Mech. Eng.*, **355** (2019), 639–662.

[14] S. Duo and Y. Zhang, Numerical approximations for the tempered fractional Laplacian: Error analysis and applications, *J. Sci. Comput.*, **81** (2019), 569–593.

[15] Y. Huang and A. Oberman, Numerical methods for the fractional Laplacian: A finite difference–quadrature approach, *SIAM J. Numer. Anal.*, **52** (2014), 3056–3084.

[16] F. Izsák and B. J. Szekeres, Models of space-fractional diffusion: A critical review, *Appl. Math. Lett.*, **71** (2017), 38–43.

[17] K. Kirkpatrick and Y. Zhang, Fractional Schrödinger dynamics and decoherence, *Phys. D*, **332** (2016), 41–54.

[18] M. Kwaśnicki, Ten equivalent definitions of the fractional Laplace operator, *Fract. Calc. Appl. Anal.*, **20** (2017), 7–51.

[19] N. S. Landkof, *Foundations of Modern Potential Theory*, Springer-Verlag, New York-Heidelberg, 1972.

[20] A. Lischke, G. Pang, M. Gulian, F. Song, C. Glusa, X. Zheng, Z. Mao, W. Cai, M. M. Meerschaert, M. Ainsworth and G. E. Karniadakis, What is the fractional Laplacian? A comparative review with new results, *J. Comput. Phys.*, **404** (2020), 109009.

[21] X. Ros-Oton and J. Serra, Fractional Laplacian: Pohozaev identity and nonexistence results, *C. R. Math. Acad. Sci. Paris*, **350** (2012), 505–508.

[22] J. A. Rosenfeld, S. A. Rosenfeld and W. E. Dixon, A mesh-free pseudospectral approach to estimating the fractional Laplacian via radial basis functions, *J. Comput. Phys.*, **390** (2019), 306–322.

[23] S. G. Samko, A. A. Kilbas and O. I. Marichev, *Fractional Integrals and Derivatives*, Gordon and Breach Science Publishers, Yverdon, 1993.

[24] C. Sheng, J. Shen, T. Tang, L.-L. Wang and H. Yuan, Fast Fourier-like mapped Chebyshev spectral-Galerkin methods for PDEs with integral fractional Laplacian in unbounded domains, *SIAM J. Numer. Anal.*, **58** (2020), 2435–2464.

[25] T. Tang, L.-L. Wang, H. Yuan and T. Zhou, Rational spectral methods for PDEs involving fractional Laplacian in unbounded domains, *SIAM J. Sci. Comput.*, **42** (2020), A585–A611.
[26] Y. Wu and Y. Zhang, A universal solution scheme for fractional and classical PDEs, arXiv:2102.00113, 2021.
[27] Y. Wu and Y. Zhang, Variable-order Laplacian and its computations with meshfree methods, preprint, 2021.

Received March 2021; revised December 2021; early access February 2022.

E-mail address: ywx7c@mst.edu
E-mail address: zhangyanz@mst.edu