

# A grid-overlay finite difference method for the fractional Laplacian on arbitrary bounded domains

Weizhang Huang\* and Jinye Shen†

A grid-overlay finite difference method is proposed for the numerical approximation of the fractional Laplacian on arbitrary bounded domains. The method uses an unstructured simplicial mesh and an overlay uniform grid for the underlying domain and constructs the approximation based on a uniform-grid finite difference approximation and a data transfer from the unstructured mesh to the uniform grid. The method takes full advantages of both uniform-grid finite difference approximation in efficient matrix-vector multiplication via the fast Fourier transform and unstructured meshes for complex geometries and mesh adaptation. It is shown that its stiffness matrix is similar to a symmetric and positive definite matrix and thus invertible if the data transfer has full column rank and positive column sums. Piecewise linear interpolation is studied as a special example for the data transfer. It is proved that the full column rank and positive column sums of linear interpolation is guaranteed if the spacing of the uniform grid is smaller than or equal to a positive bound proportional to the minimum element height of the unstructured mesh. Moreover, a sparse preconditioner is proposed for the iterative solution of the resulting linear system for the homogeneous Dirichlet problem of the fractional Laplacian. Numerical examples demonstrate that the new method has similar convergence behavior as existing finite difference and finite element methods and that the sparse preconditioning is effective. Furthermore, the new method can readily be incorporated with existing mesh adaptation strategies. Numerical results obtained by combining with the so-called MMPDE moving mesh method are also presented.

**AMS 2020 Mathematics Subject Classification.** 65N06, 35R11

**Key Words.** Fractional Laplacian, finite difference, arbitrary domain, mesh adaptation, overlay grid

---

\*Department of Mathematics, University of Kansas, Lawrence, Kansas, U.S.A. [whuang@ku.edu](mailto:whuang@ku.edu)

†Corresponding author. School of Mathematics, Southwestern University of Finance and Economics, Chengdu, Sichuan, China. [jyshen@swufe.edu.cn](mailto:jyshen@swufe.edu.cn)

# 1 Introduction

The fractional Laplacian is a fundamental non-local operator for modeling anomalous dynamics and its numerical approximation has attracted considerable attention recently; e.g. see [5, 24, 27] and references therein. A number of numerical methods have been developed along the lines of various representations of the fractional Laplacian, such as the Fourier/spectral representation, the singular integral (including Riemann-Liouville and Caputo) representation, the Grünwald-Letnikov representation, and the heat semi-group representation. For examples, methods based on the Fourier/spectral representation include finite difference (FD) methods [17, 23, 24, 25, 30, 31, 39], spectral element method [35], and sinc-based method [6]. Methods based on the singular integral representation include FD methods [14, 28, 36], finite element methods [1, 2, 3, 4, 8, 15, 37], discontinuous Galerkin methods [12, 13], and spectral method [26]; and FD methods [11, 32, 38] based on the Grünwald-Letnikov representation. Loosely speaking, most of the existing FD methods have been constructed on uniform grids, have the advantage of efficient matrix-vector multiplication via the fast Fourier transform (FFT), but do not work for domains with complex geometries and have difficulty to incorporate with mesh adaptation. On the other hand, finite element methods can work for arbitrary bounded domains and are easy to combine with mesh adaptation but suffer from slowness of matrix-vector multiplication because the stiffness matrix is a full matrix. A sparse approximation to the stiffness matrix and an efficient multigrid implementation have been proposed by Ainsworth and Glusa [4]. There exists special effort to apply FD and spectral methods to domains with complex geometries. For example, Song et al. [35] construct an approximation of the fractional Laplacian based on the spectral decomposition and the spectral element approximation of the Laplacian operator on arbitrary domains. Hao et al. [17] combine a uniform-grid FD method with the penalty method of [34] for non-rectangular domains.

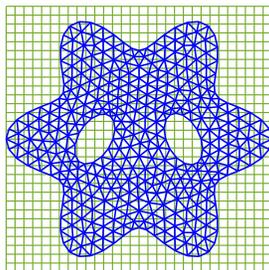


Figure 1: A sketch of an unstructured simplicial mesh  $\mathcal{T}_h$  (in color blue) overlaid by a uniform grid  $\mathcal{T}_{\text{FD}}$  (in color green) . Boundary value problem (1) is solved on  $\mathcal{T}_h$  that is not necessarily quasi-uniform.

The objective of this work is to present a simple FD method, called the grid-overlay FD method or GoFD, for the fractional Laplacian on arbitrary bounded domains. The method has the full advantage of uniform-grid FD methods in efficient matrix-vector multiplication via FFT. Specifically, we consider the homogeneous Dirichlet problem

$$\begin{cases} (-\Delta)^s u = f, & \text{in } \Omega \\ u = 0, & \text{in } \Omega^c \end{cases} \quad (1)$$

where  $(-\Delta)^s$  is the fractional Laplacian with the fractional order  $s \in (0, 1)$ ,  $\Omega$  is a bounded domain in  $\mathbb{R}^d$  ( $d \geq 1$ ),  $\Omega^c \equiv \mathbb{R}^d \setminus \Omega$ , and  $f$  is a given function. Given an unstructured simplicial mesh  $\mathcal{T}_h$  that fits or approximately fits  $\Omega$  and is not necessarily quasi-uniform, we want to approximate the solution of (1) on  $\mathcal{T}_h$ . A uniform grid  $\mathcal{T}_{\text{FD}}$  (with spacing  $h_{\text{FD}}$ ) that overlays  $\bar{\Omega}$  (see Fig. 1) is first created and a uniform-grid FD approximation  $h_{\text{FD}}^{-2s} A_{\text{FD}}$  for the fractional Laplacian is constructed thereon. Then, the GoFD approximation of the fractional Laplacian on  $\mathcal{T}_h$  is defined as  $h_{\text{FD}}^{-2s} A_h$ , where  $A_h$  is given by

$$A_h = D_h^{-1} (I_h^{\text{FD}})^T A_{\text{FD}} I_h^{\text{FD}}. \quad (2)$$

Here,  $I_h^{\text{FD}}$  is a transfer matrix from  $\mathcal{T}_h$  to  $\mathcal{T}_{\text{FD}}$  and  $D_h$  is the diagonal matrix formed by the column sums of  $I_h^{\text{FD}}$ . Notice that the multiplication of  $A_h$  with vectors can be performed efficiently since the multiplication of  $A_{\text{FD}}$  with vectors can be carried out using FFT and  $I_h^{\text{FD}}$  is sparse. Moreover,  $A_h$  is invertible if  $I_h^{\text{FD}}$  has full column rank and positive column sums (cf. Theorem 3.1). For a special choice of  $I_h^{\text{FD}}$ , piecewise linear interpolation from  $\mathcal{T}_h$  to  $\mathcal{T}_{\text{FD}}$ , Theorem 3.2 states that the full column rank and positive column sums of  $I_h^{\text{FD}}$  are guaranteed if  $h_{\text{FD}}$  is smaller than or equal to a positive bound proportional to the minimum element height of  $\mathcal{T}_h$  (cf. (40)). Stability and sparse preconditioning for the resulting linear system are studied. Furthermore, the use of unstructured meshes in GoFD allows easy incorporation with existing mesh adaptation strategies. As an example, the incorporation with the so-called MMPDE moving mesh method [19, 20, 21] is discussed. Numerical examples in 1D, 2D, and 3D are presented to demonstrate that GoFD has similar convergence behavior as existing FD and finite element methods and that the sparse preconditioning and mesh adaptation are effective.

An outline of the paper is as follows. The construction of uniform-grid FD approximation for the fractional Laplacian is presented in Section 2. Section 3 is devoted to the description of GoFD, studies of its properties, and construction of sparse preconditioning. The MMPDE moving mesh method and its combination with GoFD are discussed in Section 4. Numerical examples are presented in Section 5. Finally, conclusions are drawn and further comments are given in Section 6.

## 2 Uniform-grid FD approximation of the fractional Laplacian

In this section we briefly describe the FD approximation of the fractional Laplacian on a uniform grid through the Fourier transform. The reader is referred to, e.g., [17, 24, 30, 31], for detail. The properties of the approximation and the computation of the stiffness matrix and its multiplication with vectors through the fast Fourier transform (FFT) are also discussed. For notational simplicity and without loss of generality, we restrict our discussion in 2D.

### 2.1 FD approximation on a uniform grid

Consider an absolutely integrable function  $u$  in  $\mathbb{R}^2$ . Recall that its Fourier transform is defined as

$$\hat{u}(\xi, \eta) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} u(x, y) e^{-ix\xi} e^{-iy\eta} dx dy, \quad (3)$$

and the inverse Fourier transform is given by

$$u(x, y) = \frac{1}{(2\pi)^2} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \hat{u}(\xi, \eta) e^{ix\xi} e^{iy\eta} d\xi d\eta. \quad (4)$$

Applying the Laplacian operator to the above equation, we have

$$(-\Delta)u(x, y) = \frac{1}{(2\pi)^2} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} (\xi^2 + \eta^2) \hat{u}(\xi, \eta) e^{ix\xi} e^{iy\eta} d\xi d\eta.$$

This implies

$$\widehat{(-\Delta)u}(\xi, \eta) = (\xi^2 + \eta^2) \hat{u}(\xi, \eta).$$

Based on this, the Fourier transform of the fractional Laplacian can be defined as

$$\widehat{(-\Delta)^s u}(\xi, \eta) = (\xi^2 + \eta^2)^s \hat{u}(\xi, \eta). \quad (5)$$

Accordingly, the fractional Laplacian is given by

$$(-\Delta)^s u(x, y) = \frac{1}{(2\pi)^2} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \widehat{(-\Delta)^s u}(\xi, \eta) e^{ix\xi} e^{iy\eta} d\xi d\eta. \quad (6)$$

A uniform-grid FD approximation for the fractional Laplacian can be defined in a similar manner. Consider an infinite uniform grid (lattice)

$$(x_j, y_k) = (jh_{\text{FD}}, kh_{\text{FD}}), \quad j, k \in \mathbb{Z},$$

where  $h_{\text{FD}}$  is a given positive number. The discrete Fourier transform (DFT) on this grid is defined as

$$\check{u}(\xi, \eta) = \sum_{j=-\infty}^{\infty} \sum_{k=-\infty}^{\infty} u_{j,k} e^{-ix_j\xi} e^{-iy_k\eta}, \quad (7)$$

where  $u_{j,k} = u(x_j, y_k)$ . Here, we use  $\check{u}$  to denote the DFT of  $u$  to avoid confusion with the continuous Fourier transform (cf. (3)). The inverse DFT is given by

$$\begin{aligned} u(x_j, y_k) &= \frac{h_{\text{FD}}^2}{(2\pi)^2} \int_{-\frac{\pi}{h_{\text{FD}}}}^{\frac{\pi}{h_{\text{FD}}}} \int_{-\frac{\pi}{h_{\text{FD}}}}^{\frac{\pi}{h_{\text{FD}}}} \check{u}(\xi, \eta) e^{ix_j\xi} e^{iy_k\eta} d\xi d\eta \\ &= \frac{1}{(2\pi)^2} \int_{-\pi}^{\pi} \int_{-\pi}^{\pi} \check{u}\left(\frac{\xi}{h_{\text{FD}}}, \frac{\eta}{h_{\text{FD}}}\right) e^{ij\xi} e^{ik\eta} d\xi d\eta. \end{aligned} \quad (8)$$

Now we consider a central FD approximation to the Laplacian on  $\{(x_j, y_k)\}$ ,

$$(-\Delta_h)u(x_j, y_k) = \frac{1}{h_{\text{FD}}^2} (u_{j+1,k} - 2u_{j,k} + u_{j-1,k}) + \frac{1}{h_{\text{FD}}^2} (u_{j,k+1} - 2u_{j,k} + u_{j,k-1}). \quad (9)$$

Applying the DFT (7) to the above equation, we get

$$\widehat{(-\Delta_h)u}(\xi, \eta) = \frac{1}{h_{\text{FD}}^2} \left( 4 \sin^2\left(\frac{\xi h_{\text{FD}}}{2}\right) + 4 \sin^2\left(\frac{\eta h_{\text{FD}}}{2}\right) \right) \check{u}(\xi, \eta).$$

From this, the DFT of the FD approximation of the fractional Laplacian can be defined as

$$\widehat{(-\Delta_h)^s u}(\xi, \eta) = \frac{1}{h_{\text{FD}}^{2s}} \left( 4 \sin^2\left(\frac{\xi h_{\text{FD}}}{2}\right) + 4 \sin^2\left(\frac{\eta h_{\text{FD}}}{2}\right) \right)^s \check{u}(\xi, \eta). \quad (10)$$

Then, the FD approximation of the fractional Laplacian reads as

$$\begin{aligned} (-\Delta_h)^s u(x_j, y_k) &= \frac{1}{(2\pi)^2} \int_{-\pi}^{\pi} \int_{-\pi}^{\pi} \overline{(-\Delta_h)^s u}\left(\frac{\xi}{h_{\text{FD}}}, \frac{\eta}{h_{\text{FD}}}\right) e^{ij\xi} e^{ik\eta} d\xi d\eta \\ &= \frac{1}{(2\pi)^2 h_{\text{FD}}^{2s}} \sum_{m=-\infty}^{\infty} \sum_{n=-\infty}^{\infty} u_{m,n} \int_{-\pi}^{\pi} \int_{-\pi}^{\pi} \psi(\xi, \eta) e^{i(j-m)\xi} e^{i(k-n)\eta} d\xi d\eta, \end{aligned} \quad (11)$$

where

$$\psi(\xi, \eta) = \left( 4 \sin^2\left(\frac{\xi}{2}\right) + 4 \sin^2\left(\frac{\eta}{2}\right) \right)^s. \quad (12)$$

If we define

$$T_{p,q} = \frac{1}{(2\pi)^2} \int_{-\pi}^{\pi} \int_{-\pi}^{\pi} \psi(\xi, \eta) e^{ip\xi} e^{iq\eta} d\xi d\eta, \quad (13)$$

$$A_{(j,k),(m,n)} = T_{j-m, k-n}, \quad (14)$$

we can rewrite (11) into

$$(-\Delta_h)^s u(x_j, y_k) = \frac{1}{h_{\text{FD}}^{2s}} \sum_{m=-\infty}^{\infty} \sum_{n=-\infty}^{\infty} A_{(j,k),(m,n)} u_{m,n}, \quad -\infty < j, k < \infty. \quad (15)$$

Notice that  $A_{(j,k),(m,n)}$  is the entry of an infinite matrix  $A$  at the  $(j, k)$ -th row and the  $(m, n)$ -th column with the understanding that the 2D indices  $(j, k)$  and  $(m, n)$  are converted into linear indices using a certain ordering (such as the natural ordering). Moreover,  $T_{p,q}$ 's are the coefficients of the Fourier series of  $\psi(\xi, \eta)$ . From (13), it is not difficult to show

$$T_{-p,-q} = T_{p,q}, \quad T_{-p,q} = T_{p,q}, \quad T_{p,-q} = T_{p,q}. \quad (16)$$

Furthermore, (14) implies that  $A$  is a Toeplitz matrix of infinite order in 1D and a block Toeplitz matrix of Toeplitz blocks in multi-dimensions.

**Lemma 2.1** (Parseval's equality). *If  $\sum_{j=-\infty}^{\infty} \sum_{k=-\infty}^{\infty} u_{j,k}^2 < \infty$ , then*

$$\frac{1}{(2\pi)^2} \int_{-\pi}^{\pi} \int_{-\pi}^{\pi} |\tilde{u}\left(\frac{\xi}{h_{\text{FD}}}, \frac{\eta}{h_{\text{FD}}}\right)|^2 d\xi d\eta = \sum_{j=-\infty}^{\infty} \sum_{k=-\infty}^{\infty} u_{j,k}^2. \quad (17)$$

*Proof.* (17) can be proved using (8) and direct calculation.  $\square$

Next, we consider functions with compact support in  $\Omega$  for solving the homogeneous Dirichlet problem (1). More specifically, we consider a square  $(-R, R) \times (-R, R)$ , where  $R$  is a positive number such that  $\Omega$  is covered by the square (cf. Fig. 1). For a given positive integer  $N$ , we choose  $h_{\text{FD}} = \frac{R}{N}$ . Let  $\mathcal{T}_{\text{FD}} = \{(x_j, y_k), -N \leq j, k \leq N\}$  (a finite uniform grid) and  $\vec{u}_{\text{FD}} = \{u_{j,k}, j, k = -N, \dots, N\}$ . Notice that the right-hand side of (17) becomes a finite double sum for the current situation since  $u_{j,k} = 0$  for any  $(x_j, y_k) \notin (-R, R) \times (-R, R)$ .

For notational convenience, we denote the restriction of the infinite matrix  $A$  on  $\mathcal{T}_{\text{FD}}$  by  $A_{\text{FD}}$ , i.e.,

$$A_{\text{FD}} = \left( A_{(j,k),(m,n)} \right)_{(2N+1)^2 \times (2N+1)^2}. \quad (18)$$

Notice that  $h_{\text{FD}}^{-2s} A_{\text{FD}}$  is an FD approximation matrix of the fractional Laplacian on  $\mathcal{T}_{\text{FD}}$ . Recall from (14) (now with  $-N \leq j, k, m, n \leq N$ ) that  $A_{\text{FD}}$  is a block Toeplitz matrix of Toeplitz blocks.

**Lemma 2.2** (The fractional Poincaré inequality). *For any function  $u$  with support in  $(-R, R) \times (-R, R)$ , there holds*

$$\begin{aligned} & \int_{-\pi}^{\pi} \int_{-\pi}^{\pi} (|\xi|^2 + |\eta|^2)^s |\tilde{u}(\frac{\xi}{h_{FD}}, \frac{\eta}{h_{FD}})|^2 d\xi d\eta \\ & \geq C h_{FD}^{2s} \int_{-\pi}^{\pi} \int_{-\pi}^{\pi} |\tilde{u}(\frac{\xi}{h_{FD}}, \frac{\eta}{h_{FD}})|^2 d\xi d\eta, \end{aligned} \quad (19)$$

where  $C$  is a positive constant independent of  $u$ ,  $h_{FD}$ , and  $N$ .

*Proof.* The proof of the continuous fractional Poincaré inequality can be found, for example, in [7, Proposition 1.55, p. 39]. (19) is different from the continuous one since it uses DFT instead of the continuous Fourier transform. Nevertheless, it can be proved by following the proof of the continuous version and using Lemma 2.1.  $\square$

We remark that the left-hand and right-hand sides of (19) can be viewed as the  $H^{2s}$  semi-norm and  $L^2$  norm of  $u$ , respectively.

**Proposition 2.1.** *The matrix  $A_{FD} = (A_{(j,k),(m,n)})$  is symmetric and positive definite. Particularly,*

$$\vec{u}_{FD}^T A_{FD} \vec{u}_{FD} \geq C h_{FD}^{2s} \vec{u}_{FD}^T \vec{u}_{FD}, \quad \forall \vec{u}_{FD} \in \mathbb{R}^{(2N+1)^2} \quad (20)$$

where  $C$  is a positive constant.

*Proof.* From (14) and (16), we can see that  $A_{FD}$  is symmetric. Moreover, from (13) and (14) we have

$$\begin{aligned} \vec{u}_{FD}^T A_{FD} \vec{u}_{FD} &= \sum_{j=-N}^N \sum_{k=-N}^N \sum_{m=-N}^N \sum_{n=-N}^N A_{(j,k),(m,n)} u_{j,k} u_{m,n} \\ &= \frac{1}{(2\pi)^2} \int_{-\pi}^{\pi} \int_{-\pi}^{\pi} \psi(\xi, \eta) \cdot \sum_{j=-N}^N \sum_{k=-N}^N u_{j,k} e^{ij\xi + ik\eta} \cdot \sum_{m=-N}^N \sum_{n=-N}^N u_{m,n} e^{-im\xi - in\eta} d\xi d\eta \\ &= \frac{1}{(2\pi)^2} \int_{-\pi}^{\pi} \int_{-\pi}^{\pi} \psi(\xi, \eta) |\tilde{u}(\frac{\xi}{h_{FD}}, \frac{\eta}{h_{FD}})|^2 d\xi d\eta. \end{aligned}$$

Since

$$\psi(\xi, \eta) = \left( 4 \sin^2\left(\frac{\xi}{2}\right) + 4 \sin^2\left(\frac{\eta}{2}\right) \right)^s \geq \left(\frac{2}{\pi}\right)^2 (|\xi|^2 + |\eta|^2)^s, \quad \forall \xi, \eta \in (-\pi, \pi)$$

we get

$$\vec{u}_{FD}^T A_{FD} \vec{u}_{FD} \geq \frac{1}{(2\pi)^2} \left(\frac{2}{\pi}\right)^2 \int_{-\pi}^{\pi} \int_{-\pi}^{\pi} (|\xi|^2 + |\eta|^2)^s |\tilde{u}(\frac{\xi}{h_{FD}}, \frac{\eta}{h_{FD}})|^2 d\xi d\eta.$$

Combining this with Lemmas 2.1 and 2.2 we obtain (20), which implies that  $A_{FD}$  is positive definite.  $\square$

When  $\Omega = (-R, R) \times (-R, R)$ , an FD approximation of the homogeneous Dirichlet problem (1) on  $\mathcal{T}_{FD}$  is given by

$$\frac{1}{h_{FD}^{2s}} A_{FD} \vec{u}_{FD} = \vec{f}_{FD}, \quad (21)$$

where  $\vec{f}_{FD} = \{f(x_j, y_k), j, k = -N, \dots, N\}$ .

**Proposition 2.2** (Stability). *The solution of (21) satisfies*

$$\vec{u}_{\text{FD}}^T A_{\text{FD}} \vec{u}_{\text{FD}} \leq C_1 h_{\text{FD}}^{2s} \vec{f}_{\text{FD}}^T \vec{f}_{\text{FD}}, \quad (22)$$

$$\vec{u}_{\text{FD}}^T \vec{u}_{\text{FD}} \leq C_2 \vec{f}_{\text{FD}}^T \vec{f}_{\text{FD}}, \quad (23)$$

where  $C_1$  and  $C_2$  are constants.

*Proof.* Multiplying  $\vec{u}_{\text{FD}}^T$  with (21) from left and using the Cauchy-Schwarz inequality, we have

$$\begin{aligned} \vec{u}_{\text{FD}}^T A_{\text{FD}} \vec{u}_{\text{FD}} &= h_{\text{FD}}^{2s} \vec{u}_{\text{FD}}^T \vec{f}_{\text{FD}} = h_{\text{FD}}^{2s} (A_{\text{FD}}^{\frac{1}{2}} \vec{u}_{\text{FD}})^T (A_{\text{FD}}^{-\frac{1}{2}} \vec{f}_{\text{FD}}) \\ &\leq h_{\text{FD}}^{2s} (\vec{u}_{\text{FD}}^T A_{\text{FD}} \vec{u}_{\text{FD}})^{\frac{1}{2}} (\vec{f}_{\text{FD}}^T A_{\text{FD}}^{-1} \vec{f}_{\text{FD}})^{\frac{1}{2}}. \end{aligned}$$

Thus, we have

$$\vec{u}_{\text{FD}}^T A_{\text{FD}} \vec{u}_{\text{FD}} \leq h_{\text{FD}}^{4s} \vec{f}_{\text{FD}}^T A_{\text{FD}}^{-1} \vec{f}_{\text{FD}}.$$

Notice that Proposition 2.1 implies  $\|A_{\text{FD}}^{-1}\| \leq Ch_{\text{FD}}^{-2s}$ . Combining these results gives (22).

The inequality (23) follows from (22) and Proposition 2.1.  $\square$

**Remark 2.1.** Error estimates and convergence order for the FD approximation of the fractional Laplacian have been established for sufficiently smooth solutions by a number of researchers; e.g., see [14, 17, 24]. Unfortunately, the existing analysis does not apply to solutions of (1) with the optimal regularity  $H^{s+1/2-\epsilon}(\Omega)$  for any  $\epsilon > 0$  [9]. Nevertheless, for those solutions it has been observed numerically (e.g., see [14, 17]) that the FD approximation converges at  $\mathcal{O}(h_{\text{FD}}^s)$  in  $L^\infty$  norm. This is confirmed by our numerical results (cf. Example 5.1). Our results also suggest that the error is  $\mathcal{O}(h_{\text{FD}}^{\min(1, s+1/2)})$  in  $L^2$  norm.  $\square$

## 2.2 Computation of the multiplication of $A_{\text{FD}}$ with vectors using FFT

For the moment, we assume that  $T_{p,q}$ 's have been computed. The process of computing the multiplication of  $A_{\text{FD}}$  with vectors starts with computing the DFT of  $T_{p,q}$ 's, i.e.,

$$\check{T}_{m,n} = \sum_{p=-2N}^{2N-1} \sum_{q=-2N}^{2N-1} T_{p,q} e^{-\frac{i2\pi(m+2N)(p+2N)}{4N}} e^{-\frac{i2\pi(n+2N)(q+2N)}{4N}},$$

for  $m, n = -2N, \dots, 2N-1$ . The inverse DFT is

$$T_{p,q} = \frac{1}{(4N)^2} \sum_{m=-2N}^{2N-1} \sum_{n=-2N}^{2N-1} \check{T}_{m,n} e^{\frac{i2\pi(m+2N)(p+2N)}{4N}} e^{\frac{i2\pi(n+2N)(q+2N)}{4N}},$$

for  $p, q = -2N, \dots, 2N-1$ . Then, from (14) we have

$$\begin{aligned} (A_{\text{FD}} \vec{u}_{\text{FD}})_{(j,k)} &= \sum_{m=-N}^N \sum_{n=-N}^N T_{j-m, k-n} u_{m,n} \\ &= \sum_{m=-N}^N \sum_{n=-N}^N u_{m,n} \frac{1}{(4N)^2} \sum_{p=-2N}^{2N-1} \sum_{q=-2N}^{2N-1} \check{T}_{p,q} e^{\frac{i2\pi(p+2N)(j-m+2N)}{4N}} e^{\frac{i2\pi(q+2N)(k-n+2N)}{4N}} \end{aligned}$$

$$\begin{aligned}
&= \frac{1}{(4N)^2} \sum_{p=-2N}^{2N-1} \sum_{q=-2N}^{2N-1} \check{T}_{p,q} (-1)^{p+2N} (-1)^{q+2N} e^{\frac{i2\pi(p+2N)(j+N)}{4N}} e^{\frac{i2\pi(q+2N)(k+N)}{4N}} \\
&\quad \cdot \sum_{m=-N}^N \sum_{n=-N}^N u_{m,n} e^{-\frac{i2\pi(p+2N)(m+N)}{4N}} e^{-\frac{i2\pi(q+2N)(n+N)}{4N}}.
\end{aligned} \tag{24}$$

If we expand  $u$  into

$$\tilde{u}_{m,n} = \begin{cases} u_{m,n}, & \text{for } -N \leq m, n \leq N \\ 0, & \text{otherwise} \end{cases}$$

we can rewrite the inner double sum in (24) into

$$\begin{aligned}
&\sum_{m=-N}^N \sum_{n=-N}^N u_{m,n} e^{-\frac{i2\pi(p+2N)(m+N)}{4N}} e^{-\frac{i2\pi(q+2N)(n+N)}{4N}} \\
&= \sum_{m=-N}^{3N-1} \sum_{n=-N}^{3N-1} \tilde{u}_{m,n} e^{-\frac{i2\pi(p+2N)(m+N)}{4N}} e^{-\frac{i2\pi(q+2N)(n+N)}{4N}},
\end{aligned}$$

which is the DFT of  $\tilde{u}$  (denoted by  $\check{\tilde{u}}$ ). Then,

$$(A_{\text{FD}} \vec{u}_{\text{FD}})_{(j,k)} = \frac{1}{(4N)^2} \sum_{p=-2N}^{2N-1} \sum_{q=-2N}^{2N-1} \check{T}_{p,q} \check{\tilde{u}}_{p,q} (-1)^{p+2N} (-1)^{q+2N} e^{\frac{i2\pi(p+2N)(j+N)}{4N}} e^{\frac{i2\pi(q+2N)(k+N)}{4N}}.$$

Thus,  $A_{\text{FD}} \vec{u}_{\text{FD}}$  can be obtained as the inverse DFT of  $\check{T}_{p,q} \check{\tilde{u}}_{p,q} (-1)^{p+2N} (-1)^{q+2N}$ .

In summary,  $A_{\text{FD}} \vec{u}_{\text{FD}}$  can be carried out by performing three FFTs. From the complexity of FFT, we can estimate the cost of computing  $A_{\text{FD}} \vec{u}_{\text{FD}}$  (in  $d$ -dimensions) as  $\mathcal{O}(N^d \log(N^d))$ . It is worth mentioning that the number of grid points of  $\mathcal{T}_{\text{FD}}$  and length of vector  $\vec{u}_{\text{FD}}$  are  $\mathcal{O}(N^d)$ . Thus, the cost is almost linear about the number of grid points.

### 2.3 Computation of matrix $T$

We rewrite (13) into

$$T_{p,q} = (-1)^{p+q} \int_0^1 \int_0^1 \tilde{\psi}(\xi, \eta) e^{i2\pi p\xi} e^{i2\pi q\eta} d\xi d\eta, \tag{25}$$

where

$$\tilde{\psi}(\xi, \eta) = (4 \cos^2(\pi\xi) + 4 \cos^2(\pi\eta))^s. \tag{26}$$

Recall from (16) that we only need to compute  $T_{p,q}$  for  $0 \leq p, q \leq 2N$ . We first consider the composite trapezoidal rule (2rd-order). For any given integer  $M \geq 2N + 1$ , let

$$\xi_j = \frac{j}{M}, \quad \eta_k = \frac{k}{M}, \quad j, k = 0, 1, \dots, M.$$

Then,

$$T_{p,q} = (-1)^{p+q} \sum_{j=0}^{M-1} \sum_{k=0}^{M-1} \int_{\xi_j}^{\xi_{j+1}} \int_{\eta_k}^{\eta_{k+1}} \tilde{\psi}(\xi, \eta) \cdot e^{i2\pi p\xi} e^{i2\pi q\eta} d\xi d\eta$$

$$\approx \frac{(-1)^{p+q}}{M^2} \sum_{j=0}^{M-1} \sum_{k=0}^{M-1} \tilde{\psi}(\xi_j, \eta_k) \cdot e^{\frac{i2\pi pj}{M}} e^{\frac{i2\pi qk}{M}}, \quad 0 \leq p, q \leq 2N. \quad (27)$$

Thus,  $T_{p,q}$ 's can be obtained with the inverse FFT.

It should be pointed out that  $M$  should be chosen much larger than  $2N + 1$  in the above procedure for accurate computation of  $T$  due to the highly oscillatory nature of the factor  $e^{i2\pi p\xi} e^{i2\pi q\eta}$  in (25). As a result, the computation of  $T$  can be very expensive in terms of CPU time and memory for large  $N$ . To avoid this difficulty, we use Filon's approach [16] designed for highly oscillatory integrals including (25). We explain this in 1D,

$$T_p = (-1)^p \int_0^1 \tilde{\psi}(\xi) e^{i2\pi p\xi} d\xi = (-1)^p \sum_{j=0}^{M-1} \int_{\xi_j}^{\xi_{j+1}} \tilde{\psi}(\xi) e^{i2\pi p\xi} d\xi.$$

The key idea of Filon's approach is to approximate  $\tilde{\psi}(\xi)$  with a polynomial on each subinterval and then carry out the resulting integrals analytically. Here, we approximate  $\tilde{\psi}(\xi)$  with a linear polynomial on each subinterval. In this situation, we can first perform integration by parts and then do the approximation, i.e.,

$$T_p = (-1)^p \sum_{j=0}^{M-1} \left( \frac{1}{i2\pi p} \tilde{\psi}(\xi) e^{i2\pi p\xi} \Big|_{\xi_j}^{\xi_{j+1}} - \frac{1}{i2\pi p} \int_{\xi_j}^{\xi_{j+1}} \tilde{\psi}'(\xi) e^{i2\pi p\xi} d\xi \right).$$

The sum of the first term in the bracket vanishes since  $\tilde{\psi}(\xi) e^{i2\pi p\xi}$  is periodic. Then,

$$\begin{aligned} T_p &= \frac{(-1)^{p+1}}{i2\pi p} \sum_{j=0}^{M-1} \int_{\xi_j}^{\xi_{j+1}} \tilde{\psi}'(\xi) e^{i2\pi p\xi} d\xi \\ &\approx \frac{(-1)^{p+1}}{i2\pi p} \sum_{j=0}^{M-1} \int_{\xi_j}^{\xi_{j+1}} \frac{\tilde{\psi}(\xi_{j+1}) - \tilde{\psi}(\xi_j)}{1/M} e^{i2\pi p\xi} d\xi \\ &= \frac{(-1)^{p+1}}{(i2\pi p)^2} \sum_{j=0}^{M-1} \frac{\tilde{\psi}(\xi_{j+1}) - \tilde{\psi}(\xi_j)}{1/M} \left( e^{i2\pi p\xi_{j+1}} - e^{i2\pi p\xi_j} \right). \end{aligned}$$

Thus,

$$T_p \approx \frac{(-1)^{p+1}}{(2\pi p)^2} \sum_{j=0}^{M-1} \frac{\tilde{\psi}(\xi_{j+1}) - 2\tilde{\psi}(\xi_j) + \tilde{\psi}(\xi_{j-1}))}{1/M} e^{\frac{i2\pi pj}{M}}, \quad p = 0, \dots, 2N. \quad (28)$$

Notice that (28) can be computed using FFT. Moreover, the error of the quadrature is  $\mathcal{O}(\frac{1}{M^2})$ , independent of  $N$ .

In our computation, we combine the Filon approach with Richardson's extrapolation. We use  $M = 2^{14}$  and  $2^{11}$  for the finest level of Richardson's extrapolation for 2D and 3D computation, respectively. For 1D computation, we use the analytical formula (e.g., see [31]),

$$T_p = \frac{(-1)^p \Gamma(2s+1)}{\Gamma(p+s+1) \Gamma(s-p+1)}, \quad p = 0, \dots, 2N \quad (29)$$

where  $\Gamma(\cdot)$  is the  $\Gamma$ -function.

It is interesting to point out that the cost of computing the matrix  $T$  via FFT is  $\mathcal{O}(M^d \log(M^d))$ . Assuming that a computer can perform  $10^9$  floating-point operations per second, computing the matrix  $T$  takes about  $(M^d \log(M^d))/10^9 = 5.2$  seconds in 2D (with  $M = 2^{14}$ ) and 196 seconds in 3D (with  $M = 2^{11}$ ).

### 3 The grid-overlay FD method

In this section we describe GoFD for solving (1) in  $d$ -dimensions ( $d \geq 1$ ) on arbitrary bounded domain  $\Omega$ . We also study the choice of  $h_{\text{FD}}$  that guarantees the column full rank and positive column sums of the transfer matrix and therefore the solvability of the linear system resulting from the GoFD discretization of (1). Furthermore, the iterative solution and sparse preconditioning for the linear system are discussed.

#### 3.1 GoFD for arbitrary bounded domains

For a given bounded domain  $\Omega \in \mathbb{R}^d$ , we assume that an unstructured simplicial mesh  $\mathcal{T}_h$  has been given that fits or approximately fits  $\partial\Omega$ . Then, we take a  $d$ -dimensional cube,  $(-R, R)^d$ , such that it covers  $\Omega$  and  $\mathcal{T}_h$ . An overlaying uniform grid (denoted by  $\mathcal{T}_{\text{FD}}$ ) is created with  $2N + 1$  nodes in each axial direction for some positive integer  $N$  and the spacing is given by

$$h_{\text{FD}} = \frac{R}{N}.$$

See Fig. 1 for a sketch of  $\mathcal{T}_{\text{FD}}$  and  $\mathcal{T}_h$ .

Then, a uniform-grid FD approximation  $h_{\text{FD}}^{-2s} A_{\text{FD}}$  (of size  $(2N + 1)^d \times (2N + 1)^d$ ) of the fractional Laplacian can be obtained on  $\mathcal{T}_{\text{FD}}$  as described in the previous section. We define  $h_{\text{FD}}^{-2s} A_h$  as the GoFD approximation of the fractional Laplacian on  $\Omega$ , where  $A_h$  is defined in (2).

**Remark 3.1.** The matrix  $D_h^{-1}(I_h^{\text{FD}})^T$  represents a data transfer from grid  $\mathcal{T}_{\text{FD}}$  to mesh  $\mathcal{T}_h$ . It is taken as a transpose of  $I_h^{\text{FD}}$  so that  $A_h$  is similar to a symmetric and positive definite matrix (cf. Theorem 3.1 below). Moreover,  $D_h^{-1}$  is included in the definition so that the row sums of  $D_h^{-1}(I_h^{\text{FD}})^T$  are equal to one and, thus, the transfer preserves constant functions. This inclusion is necessary for the data transfer  $D_h^{-1}(I_h^{\text{FD}})^T$  to be consistent.  $\square$

**Theorem 3.1.** *If  $I_h^{\text{FD}}$  has full column rank and positive column sums, then  $A_h$  defined in (2) is similar to a symmetric and positive definite matrix, i.e.,*

$$A_h = D_h^{-\frac{1}{2}} \cdot (I_h^{\text{FD}} D_h^{-\frac{1}{2}})^T A_{\text{FD}} (I_h^{\text{FD}} D_h^{-\frac{1}{2}}) \cdot D_h^{\frac{1}{2}}. \quad (30)$$

As a consequence,  $A_h$  is invertible.

*Proof.* When  $I_h^{\text{FD}}$  has positive column sums,  $D_h$  is invertible and thus, the definition (2) is meaningful. Moreover, it is straightforward to rewrite (2) into (30), indicating that  $A_h$  is similar to

$$\tilde{A}_h = (I_h^{\text{FD}} D_h^{-\frac{1}{2}})^T A_{\text{FD}} (I_h^{\text{FD}} D_h^{-\frac{1}{2}}). \quad (31)$$

It is obvious that  $\tilde{A}_h$  is symmetric and positive semi-definite. If we can show it to be nonsingular, then  $\tilde{A}_h$  is positive definite. Assume that there is a vector  $\vec{u}_h$  such that  $\vec{u}_h^T \tilde{A}_h \vec{u}_h = 0$ . Since  $A_{\text{FD}}$  is symmetric and positive definite (Proposition 2.1), this implies  $I_h^{\text{FD}} D_h^{-\frac{1}{2}} \vec{u}_h = 0$ . The full column rank assumption of  $I_h^{\text{FD}}$  and  $D_h$  being diagonal mean  $\vec{u}_h = 0$ . Thus,  $\tilde{A}_h$  is nonsingular and, therefore,  $\tilde{A}_h$  is symmetric and positive definite.  $\square$

**Remark 3.2.** The full column rank assumption of  $I_h^{\text{FD}}$  in Theorem 3.1 implies that  $I_h^{\text{FD}}$  has at least as many rows as columns, i.e.  $(2N+1)^d \geq N_v$ . In other words,  $\mathcal{T}_{\text{FD}}$  should have as many vertices as  $\mathcal{T}_h$  does.  $\square$

We now consider a special choice of  $I_h^{\text{FD}}$ : linear interpolation. Denote the vertices of  $\mathcal{T}_h$  by  $\mathbf{x}_j$ ,  $j = 1, \dots, N_v$ , and the vertices of  $\mathcal{T}_{\text{FD}}$  by  $\mathbf{x}_k^{\text{FD}}$ ,  $k = 1, \dots, N_v^{\text{FD}}$ . Consider the piecewise linear interpolation on  $\mathcal{T}_h$ ,

$$I_h u(\mathbf{x}) = \sum_{j=1}^{N_v} u_j \phi_j(\mathbf{x}), \quad (32)$$

where  $u_j = u(\mathbf{x}_j)$  and  $\phi_j$  is the Lagrange-type linear basis function associated with vertex  $\mathbf{x}_j$ . Here we assume that all of the basis functions vanish outside  $\bar{\Omega}$ . Recall that

$$0 \leq \phi_j(\mathbf{x}) \leq 1; \quad \phi_j(\mathbf{x}) = 0, \quad \forall \mathbf{x} \notin \bar{\omega}_j; \quad \sum_{j=1}^{N_v} \phi_j(\mathbf{x}) = 1, \quad \forall \mathbf{x} \in \bar{\Omega}$$

where  $\omega_j$  is the patch of elements that have  $\mathbf{x}_j$  as one of their vertices. When restricted on an element  $K$  of  $\mathcal{T}_h$ , the linear interpolation can be expressed as

$$I_h u|_K(\mathbf{x}) = \sum_{j=0}^d u_j^K \phi_j^K(\mathbf{x}), \quad \forall \mathbf{x} \in K. \quad (33)$$

Let  $\vec{u}_h = \{u(\mathbf{x}_j), j = 1, \dots, N_v\}$ . Then, for  $k = 1, \dots, N_v^{\text{FD}}$ ,

$$(I_h^{\text{FD}} \vec{u}_h)_k = I_h u(\mathbf{x}_k^{\text{FD}}) = \sum_{j=1}^{N_v} u_j \phi_j(\mathbf{x}_k^{\text{FD}}). \quad (34)$$

This gives

$$(I_h^{\text{FD}})_{k,j} = \phi_j(\mathbf{x}_k^{\text{FD}}), \quad k = 1, \dots, N_v^{\text{FD}}, \quad j = 1, \dots, N_v. \quad (35)$$

It is worth pointing out that  $I_h$  conforms with the homogeneous Dirichlet boundary condition. Consider an arbitrary vertex  $\mathbf{x}_k^{\text{FD}} \in \mathcal{T}_{\text{FD}}$ . When  $\mathbf{x}_k^{\text{FD}} \notin \Omega$ , it is obvious that  $I_h u(\mathbf{x}_k^{\text{FD}}) = 0$ . When  $\mathbf{x}_k^{\text{FD}} \in \Omega$ , on the other hand, there exists a mesh element  $K \in \mathcal{T}_h$  containing  $\mathbf{x}_k^{\text{FD}}$  since  $\mathcal{T}_h$  is a boundary-fitted mesh for  $\Omega$ . In this case,  $I_h u(\mathbf{x}_k^{\text{FD}})$  is determined by the values of  $u$  at the vertices of  $K$  (cf. (33)). Particularly,  $I_h u(\mathbf{x}_k^{\text{FD}})$  is affected by the values of  $u$  on the boundary of  $\Omega$  when  $K$  is a boundary element. In this sense,  $I_h$  and thus  $I_h^{\text{FD}}$  (cf. (34)) and  $A_h$  (cf. (2)) feel the homogeneous Dirichlet boundary condition.

In the following analysis, we need the following quantities,

$$N_{\text{val}} = \max_{j=1, \dots, N_v} \#\omega_j, \quad N_{\text{FD}}^h = \max_{K \in \mathcal{T}_h} \#\{\mathbf{x}_k^{\text{FD}} \in \bar{K}\},$$

where  $\#$  stands for the number of members in a set.  $N_{val}$  is usually referred to as the valence of  $\mathcal{T}_h$ .  $N_{FD}^h$  can be estimated as

$$N_{FD}^h \approx \frac{|K|_{max}}{h_{FD}^d},$$

where  $|K|_{max}$  denotes the volume of the largest element of  $\mathcal{T}_h$ . We assume that both  $N_{val}$  and  $N_{FD}^h$  are finite and small.

**Lemma 3.1.** *The transfer matrix  $I_h^{FD}$  associated with piecewise linear interpolation has the following properties.*

- (i) *Nonnegativity and boundedness:*  $0 \leq (I_h^{FD})_{k,j} \leq 1$  for all  $k, j$ .
- (ii) *Sparsity:*  $(I_h^{FD})_{k,j} = 0$  when  $\mathbf{x}_k^{FD} \notin \omega_j$ .
- (iii) *The row sums of  $I_h^{FD}$  are either 0 or 1, i.e.,*

$$\sum_{j=1}^{N_v} (I_h^{FD})_{k,j} = \begin{cases} 1, & \text{for } \mathbf{x}_k^{FD} \in \bar{\Omega} \\ 0, & \text{otherwise} \end{cases} \quad \forall k = 1, \dots, N_v^{FD}.$$

- (iv) *The column sums,  $D_{h,j} = \sum_{k=1}^{N_v^{FD}} (I_h^{FD})_{k,j}$ ,  $j = 1, \dots, N_v$ , are bounded by*

$$\max_{\mathbf{x}_k^{FD} \in \omega_j} \phi_j(\mathbf{x}_k^{FD}) \leq D_{h,j} \leq N_{val} N_{FD}^h. \quad (36)$$

- (v)

$$\lambda_{max}((I_h^{FD})^T I_h^{FD}) \leq N_{val} N_{FD}^h, \quad \lambda_{max}(I_h^{FD} (I_h^{FD})^T) \leq N_{val} N_{FD}^h. \quad (37)$$

*Proof.* (i), (ii), and (iii) follow from (35) and the properties of linear basis functions. For (iv), we have

$$D_{h,j} = \sum_{k=1}^{N_v^{FD}} \phi_j(\mathbf{x}_k^{FD}) = \sum_{\mathbf{x}_k^{FD} \in \omega_j} \phi_j(\mathbf{x}_k^{FD}) = \sum_{K \in \omega_j} \sum_{\mathbf{x}_k^{FD} \in K} \phi_j(\mathbf{x}_k^{FD}).$$

Then, (36) follows from the above equation, the definition of  $N_{val}$  and  $N_{FD}^h$ , and the nonnegativity of the basis functions.

For (v), we notice that both  $(I_h^{FD})^T I_h^{FD}$  and  $I_h^{FD} (I_h^{FD})^T$  are nonnegative matrices. Using (iii) and (iv), we can show that their row sums are bounded above by  $N_{val} N_{FD}^h$ . Then, (37) follows from [18, Theorem 8.1.22] about the spectral radius of nonnegative matrices.  $\square$

Next we study how small  $h_{FD}$  should be (or, equivalently, how large  $N$  should be) to guarantee that  $D_h$  is invertible and  $I_h^{FD}$  has full column rank. To this end, we need a few properties of simplexes in  $\mathbb{R}^d$ . For a simplex  $K$ , we denote the facet formed by all of its vertices except  $\mathbf{x}_j^K$  by  $S_j$  and the distance (called the height or altitude) from  $\mathbf{x}_j^K$  to  $S_j$  by  $a_j$ . The minimum height of  $K$  is denoted by  $a_K$ , i.e.,  $a_K = \min_j a_j$  and the minimum element height of  $\mathcal{T}_h$  is denoted by  $a_h$ , i.e.,  $a_h = \min_K a_K$ .

**Lemma 3.2.** *Any simplex  $K \in \mathbb{R}^d$  contains a cube of side length at least  $\frac{2a_K}{(d+1)\sqrt{d}}$ .*

*Proof.* It is known from geometry (e.g., see [29, Theorem 1]) that the radius of the largest ball inscribed in any simplex  $K$  is related to the heights of  $K$  as

$$\frac{1}{r_{in}} = \sum_{j=1}^{d+1} \frac{1}{a_j}.$$

From this, we have

$$r_{in} \geq \frac{a_K}{d+1}.$$

Since the length of the diagonals of the largest cube inscribed in the ball is equal to the diameter of the ball, i.e.,  $\sqrt{d}a = 2r_{in}$ , where  $a$  is the side length of the cube, we get

$$a = \frac{2r_{in}}{\sqrt{d}} \geq \frac{2a_K}{(d+1)\sqrt{d}}.$$

□

**Lemma 3.3.** *The  $j$ -th barycentric coordinate of an arbitrary point  $\mathbf{x}$  on  $K$ ,  $\phi_j^K(\mathbf{x})$ , is equal to the ratio of the distance from  $\mathbf{x}$  to facet  $S_j$ , to the height  $a_j$ .*

*Proof.* The conclusion follows from  $\phi_j^K(\mathbf{x}_j^K) = 1$ ,  $\phi_j^K|_{S_j} = 0$ , and the linearity of  $\phi_j^K$ . □

**Lemma 3.4.** *Consider a simplex  $\tilde{K} \subset K$  with vertices  $\mathbf{y}_k$ ,  $k = 0, \dots, d$ , and define*

$$v_k = I_h u|_K(\mathbf{y}_k) = \sum_{j=0}^d u_j^K \phi_j^K(\mathbf{y}_k), \quad k = 0, \dots, d.$$

Then,

$$\frac{|\tilde{K}|^2}{(d+1)^d |K|^2} \sum_{j=0}^d (u_j^K)^2 \leq \sum_{j=0}^d v_j^2 \leq (d+1) \sum_{j=0}^d (u_j^K)^2, \quad (38)$$

where  $|K|$  and  $|\tilde{K}|$  denote the volume of  $K$  and  $\tilde{K}$ , respectively.

*Proof.* Recalling that  $\phi_j^K(\mathbf{x}) \geq 0$  and  $\sum_{j=0}^d \phi_j^K(\mathbf{x}) = 1$ , we have

$$\sum_{k=0}^d v_k^2 \leq \sum_{k=0}^d \left( \sum_{j=0}^d |u_j^K| \phi_j^K(\mathbf{y}_k) \right)^2 \leq \sum_{k=0}^d \left( \sum_{j=0}^d |u_j^K|^2 \phi_j^K(\mathbf{y}_k) \right) \leq (d+1) \sum_{j=0}^d |u_j^K|^2,$$

which gives the right inequality of (38).

To prove the left inequality of (38), from

$$\mathbf{x} = \sum_{j=0}^d \mathbf{x}_j^K \phi_j^K(\mathbf{x}), \quad 1 = \sum_{j=0}^d \phi_j^K(\mathbf{x}),$$

we have

$$\begin{bmatrix} \phi_0^K \\ \vdots \\ \phi_d^K \end{bmatrix} = E^{-1} \begin{bmatrix} \mathbf{x} \\ 1 \end{bmatrix}, \quad E = \begin{bmatrix} \mathbf{x}_0^K & \mathbf{x}_1^K & \cdots & \mathbf{x}_d^K \\ 1 & 1 & \cdots & 1 \end{bmatrix}.$$

It is known that  $\det(E) = d!|K|$ . Using this and letting  $\vec{u}^K = (u_0^K, \dots, u_d^K)^T$ , we have

$$\begin{aligned} \sum_{k=0}^d v_k^2 &= \sum_{k=0}^d (\vec{u}^K)^T E^{-1} \begin{bmatrix} \mathbf{y}_k \\ 1 \end{bmatrix} \begin{bmatrix} \mathbf{y}_k \\ 1 \end{bmatrix}^T E^{-T} \vec{u}^K \\ &= (\vec{u}^K)^T E^{-1} \sum_{k=0}^d \begin{bmatrix} \mathbf{y}_k \\ 1 \end{bmatrix} \begin{bmatrix} \mathbf{y}_k \\ 1 \end{bmatrix}^T E^{-T} \vec{u}^K \\ &= (\vec{u}^K)^T E^{-1} B B^T E^{-T} \vec{u}^K, \end{aligned} \quad (39)$$

where

$$B = \begin{bmatrix} \mathbf{y}_0 & \mathbf{y}_1 & \cdots & \mathbf{y}_d \\ 1 & 1 & \cdots & 1 \end{bmatrix}.$$

The right inequality of (38) implies that

$$\lambda_{\max}(E^{-1} B B^T E^{-T}) \leq (d+1).$$

On the other hand,

$$\det(E^{-1} B B^T E^{-T}) = \frac{\det(B)^2}{\det(E)^2} = \frac{|\tilde{K}|^2}{|K|^2}.$$

Since the determinant of a matrix is equal to the product of its eigenvalues, we get

$$\lambda_{\min}(E^{-1} B B^T E^{-T}) \geq \frac{\det(E^{-1} B B^T E^{-T})}{\lambda_{\max}(E^{-1} B B^T E^{-T})^d} \geq \frac{|\tilde{K}|^2}{(d+1)^d |K|^2}.$$

Combining this with (39) we obtain the left inequality of (38).  $\square$

**Theorem 3.2.** *If we choose*

$$h_{FD} \leq \frac{a_h}{(d+1)\sqrt{d}}, \quad (40)$$

where  $a_h$  is the minimum element height of  $\mathcal{T}_h$ , then the transfer matrix  $I_h^{FD}$  associated with piecewise linear interpolation has the following properties.

(i)

$$\frac{1}{(d+1)\sqrt{d}} \cdot \frac{a_h}{h} \leq D_{h,j} \leq N_{val} N_{FD}^h, \quad \forall j = 1, \dots, N_v \quad (41)$$

and thus,  $D_h$  is invertible. Here,  $h$  is the maximum element diameter of  $\mathcal{T}_h$ .

(ii) The minimum eigenvalue of  $(I_h^{FD})^T I_h^{FD}$  is bounded below by

$$\lambda_{\min}((I_h^{FD})^T I_h^{FD}) \geq C \left(\frac{a_h}{h}\right)^{2d}, \quad (42)$$

where  $C$  is a positive constant.

(iii)  $I_h^{FD}$  has full column rank.

*Proof.* (i) Lemma 3.2 implies that any element  $K$  of  $\mathcal{T}_h$  contains a cube of side length  $\frac{2a_h}{(d+1)\sqrt{d}}$ . Thus, when  $h_{\text{FD}}$  satisfies (40),  $K$  contains at least a cubic cell of  $\mathcal{T}_{\text{FD}}$ . As a consequence, for any vertex (say  $\mathbf{x}_j$ ) of  $K$ , there is a node (say  $\mathbf{x}_k^{\text{FD}}$ ) of  $\mathcal{T}_{\text{FD}}$  that is in  $K$  and its distance to the facet opposing  $\mathbf{x}_j$  is at least  $h_{\text{FD}}$ . From Lemma 3.3, the barycentric coordinate of  $\mathbf{x}_k^{\text{FD}}$  at  $\mathbf{x}_j$  is greater than or equal to  $h_{\text{FD}}/a_j \geq a_h/(h(d+1)\sqrt{d})$ . Then, (41) follows from (36) and  $D_h$  is invertible.

(ii) For any function  $\vec{u} = \{u(\mathbf{x}_j), j = 1, \dots, N_v\}$ ,

$$\vec{u}^T (I_h^{\text{FD}})^T I_h^{\text{FD}} \vec{u} = \sum_{K \in \mathcal{T}_h} \sum_{\mathbf{x}_k^{\text{FD}} \in K} (I_h u|_K(\mathbf{x}_k^{\text{FD}}))^2. \quad (43)$$

As mentioned in the proof of (i), each element of  $\mathcal{T}_h$  contains at least a cubic cell of  $\mathcal{T}_{\text{FD}}$ . We can take  $\tilde{K}$  in Lemma 3.4 as a simplex formed by any  $d+1$  vertices of the cubic cell. Then  $|\tilde{K}|/|K| \geq C(a_h/h)^d$  for some constant  $C$ . From Lemma 3.4, we have

$$\sum_{\mathbf{x}_k^{\text{FD}} \in K} (I_h u|_K(\mathbf{x}_k^{\text{FD}}))^2 \geq C \left(\frac{a_h}{h}\right)^{2d} \sum_{j=0}^d (u_j^K)^2,$$

which yields (42).

(iii) is a consequence of (ii).  $\square$

**Remark 3.3.** The choice (40) is needed for the theoretical guarantee of the full column rank of  $I_h^{\text{FD}}$  and the invertibility of  $D_h$ . However, the requirement is only a sufficient condition. Numerical experiment shows that we can use much larger  $h_{\text{FD}}$ , for instance,  $h_{\text{FD}} = a_h$ , which works well for the examples we have tested.  $\square$

## 3.2 Linear systems, stability, and convergence

The GoFD discretization of the homogeneous Dirichlet problem (1) on the unstructured mesh  $\mathcal{T}_h$  is defined as

$$\frac{1}{h_{\text{FD}}^{2s}} D_h^{-1} (I_h^{\text{FD}})^T A_{\text{FD}} I_h^{\text{FD}} \vec{u}_h = D_h^{-1} (I_h^{\text{FD}})^T \vec{f}_{\text{FD}}, \quad (44)$$

where

$$\begin{aligned} \vec{u}_h &= \{u_j \approx u(\mathbf{x}_j), j = 1, \dots, N_v; u_j = 0, \text{ for } \mathbf{x}_j \in \Omega^c\}, \\ \vec{f}_{\text{FD}} &= \{f(\mathbf{x}_k^{\text{FD}}), k = 1, \dots, N_v^{\text{FD}}; f(\mathbf{x}_k^{\text{FD}}) = 0, \text{ for } \mathbf{x}_k^{\text{FD}} \in \Omega^c\}. \end{aligned}$$

Notice that  $u$  is approximated on the vertices of  $\mathcal{T}_h$  while the right-hand side function  $f$  is calculated on the vertices of  $\mathcal{T}_{\text{FD}}$ . We can also use the values of  $f$  at the vertices of  $\mathcal{T}_h$ . In this case, we have

$$\frac{1}{h_{\text{FD}}^{2s}} D_h^{-1} (I_h^{\text{FD}})^T A_{\text{FD}} I_h^{\text{FD}} \vec{u}_h = \vec{f}_h. \quad (45)$$

Numerical experiment shows that (44) and (45) produce comparable results. Since (44) provides some convenience in defining the local truncation error (cf. (48)), we use (44) in this work.

The system (44) can be simplified into

$$(I_h^{\text{FD}})^T A_{\text{FD}} I_h^{\text{FD}} \vec{u}_h = h_{\text{FD}}^{2s} (I_h^{\text{FD}})^T \vec{f}_{\text{FD}}. \quad (46)$$

One may notice that  $D_h^{-1}$  does not appear in the above equation. This is due to the special choice of the right-hand side function of (44). It appears in (45) for a different choice of the right-hand side function. Moreover, since  $I_h^{\text{FD}}$  is sparse and the multiplication of  $A_{\text{FD}}$  with vectors can be carried out efficiently using FFT (cf. Section 2.2), (46) is amenable to iterative solution with Krylov subspace methods. The conjugate gradient method (CG) is used in our computation. Recall that the cost for each iteration of CG is proportional to the cost of computing the matrix-vector multiplication  $(I_h^{\text{FD}})^T A_{\text{FD}} I_h^{\text{FD}} \vec{u}_h$ , which can be estimated as  $\mathcal{O}(N^d \log(N^d)) + \mathcal{O}(N_v)$ , where  $(2N + 1)$  is the number of grid points of  $\mathcal{T}_{\text{FD}}$  in each axial direction and  $N_v$  is the number of vertices of  $\mathcal{T}_h$ . When  $\mathcal{T}_h$  is quasi-uniform, the choice  $h_{\text{FD}} = \mathcal{O}(a_h)$  leads to  $N^d = \mathcal{O}(N_e) = \mathcal{O}(N_v)$ , where  $N_e$  denotes the number of elements of  $\mathcal{T}_h$ . This gives the cost of each CG iteration for solving (46) as  $\mathcal{O}(N_e \log N_e)$ . When  $\mathcal{T}_h$  is not quasi-uniform, it is difficult to estimate the cost since, in this case,  $a_h$  can be very different from  $h$ .

**Remark 3.4.** It is worth noting that only the block of the system (46) corresponding to the interior vertices is solved in the actual computation since the unknown variables on  $\partial\Omega$  are known.  $\square$

**Theorem 3.3** (Stability). *If  $h_{\text{FD}}$  satisfies (40), then the solution of (44) satisfies*

$$\vec{u}_h^T \vec{u}_h \leq C N_{\text{val}} N_{\text{FD}}^h \left( \frac{h}{a_h} \right)^{4d} \vec{f}_{\text{FD}}^T \vec{f}_{\text{FD}}, \quad (47)$$

where  $h$  and  $a_h$  are the maximum diameter and minimum height of elements of  $\mathcal{T}_h$ .

*Proof.* Multiplying  $\vec{u}_h^T$  from left with (46) and using the Cauchy-Schwarz inequality and Lemma 3.1, we get

$$\begin{aligned} \vec{u}_h^T (I_h^{\text{FD}})^T A_{\text{FD}} I_h^{\text{FD}} \vec{u}_h &= h_{\text{FD}}^{2s} (I_h^{\text{FD}} \vec{u}_h)^T \vec{f}_{\text{FD}} \\ &\leq h_{\text{FD}}^{2s} ((I_h^{\text{FD}} \vec{u}_h)^T I_h^{\text{FD}} \vec{u}_h)^{\frac{1}{2}} ((\vec{f}_{\text{FD}})^T \vec{f}_{\text{FD}})^{\frac{1}{2}} \\ &\leq C h_{\text{FD}}^{2s} (N_{\text{val}} N_{\text{FD}}^h)^{\frac{1}{2}} (\vec{u}_h^T \vec{u}_h)^{\frac{1}{2}} ((\vec{f}_{\text{FD}})^T \vec{f}_{\text{FD}})^{\frac{1}{2}}. \end{aligned}$$

Moreover, from Proposition 2.1 and Theorem 3.2 we have

$$\vec{u}_h^T (I_h^{\text{FD}})^T A_{\text{FD}} I_h^{\text{FD}} \vec{u}_h \geq C h_{\text{FD}}^{2s} \vec{u}_h^T (I_h^{\text{FD}})^T I_h^{\text{FD}} \vec{u}_h \geq C h_{\text{FD}}^{2s} \left( \frac{a_h}{h} \right)^{2d} \vec{u}_h^T \vec{u}_h.$$

Combining the above results, we get

$$\vec{u}_h^T \vec{u}_h \leq C (N_{\text{val}} N_{\text{FD}}^h)^{\frac{1}{2}} \left( \frac{h}{a_h} \right)^{2d} (\vec{u}_h^T \vec{u}_h)^{\frac{1}{2}} ((\vec{f}_{\text{FD}})^T \vec{f}_{\text{FD}})^{\frac{1}{2}},$$

which gives rise to (47).  $\square$

Denote the exact solution of (1) by  $u = u^e(\mathbf{x})$ . We define the local truncation error as

$$\vec{\tau}_{\text{FD}} = \vec{f}_{\text{FD}} - \frac{1}{h_{\text{FD}}^{2s}} A_{\text{FD}} I_h^{\text{FD}} \vec{u}_h^e, \quad (48)$$

where

$$\vec{u}_h^e = \{u^e(\mathbf{x}_j), j = 1, \dots, N_v; u^e(\mathbf{x}_j) = 0, \text{ for } \mathbf{x}_j \in \Omega^c\}.$$

$\vec{\tau}_{\text{FD}}$  can be rewritten into

$$\vec{\tau}_{\text{FD}} = \vec{f}_{\text{FD}} - \frac{1}{h_{\text{FD}}^{2s}} A_{\text{FD}} \vec{u}_{\text{FD}}^e + \frac{1}{h_{\text{FD}}^{2s}} A_{\text{FD}} (\vec{u}_{\text{FD}}^e - I_h^{\text{FD}} \vec{u}_h^e). \quad (49)$$

Thus,  $\vec{\tau}_{\text{FD}}$  can be viewed as a combination of the discretization error on the uniform grid  $\mathcal{T}_{\text{FD}}$  and the interpolation error from  $\mathcal{T}_h$  to  $\mathcal{T}_{\text{FD}}$ . From (48), we have

$$(I_h^{\text{FD}})^T A_{\text{FD}} I_h^{\text{FD}} \vec{u}_h^e = h_{\text{FD}}^{2s} (I_h^{\text{FD}})^T \vec{f}_{\text{FD}} - h_{\text{FD}}^{2s} (I_h^{\text{FD}})^T \vec{\tau}_{\text{FD}}.$$

Subtracting (46) from the above equation, we obtain the error equation as

$$(I_h^{\text{FD}})^T A_{\text{FD}} I_h^{\text{FD}} \vec{e}_h = -h_{\text{FD}}^{2s} (I_h^{\text{FD}})^T \vec{\tau}_{\text{FD}}, \quad (50)$$

where the error is defined as  $\vec{e}_h = \vec{u}_h - \vec{u}_h^e$ . From Theorem 3.3, we have the following corollary.

**Corollary 3.1** (Convergence). *If  $h_{\text{FD}}$  satisfies (40), the error for the GoFD scheme (44) is bounded by*

$$\vec{e}_h^T \vec{e}_h \leq C N_{\text{val}} N_{\text{FD}}^h \left( \frac{h}{a_h} \right)^{4d} \vec{\tau}_{\text{FD}}^T \vec{\tau}_{\text{FD}}. \quad (51)$$

**Remark 3.5.** Here we do not attempt to give a rigorous analysis of the local truncation error since it is still challenging to do so for the uniform FD discretization for solutions of optimal regularity (see Remark 2.1). Instead, we provide some intuitions here. From (49) we see that the local truncation error consists of two parts, one from the uniform FD discretization and the other from linear interpolation. It is known [9, Proposition 1.2] that the linear interpolation error in  $L^2$  norm is  $\mathcal{O}(h^{\min(1, s+1/2-\epsilon)})$  for functions in  $H^{s+1/2-\epsilon}(\Omega)$  for any  $\epsilon > 0$ . Moreover, it can be proved that  $h_{\text{FD}}^{-2s} A_{\text{FD}}$  is bounded in  $H^s(\Omega)$ . Thus, we can expect that the local truncation error (and thus the error by Corollary 3.1) for the GoFD scheme (44) is  $\mathcal{O}(h^{\min(1, s+1/2-\epsilon)})$  in  $L^2$  norm if the local truncation error of the uniform FD discretization is in the same order (cf. Remark 2.1).  $\square$

**Remark 3.6.** Interestingly, Borthagaray et al. [9] and Acosta et al. [2] show that the error of the linear finite element approximation of (1) in  $L^2$  norm is  $\mathcal{O}(h^{\min(1, s+1/2-\epsilon)})$  for quasi-uniform meshes and  $\mathcal{O}(\bar{h}^{1+s})$  for graded meshes. Here,  $\bar{h} \equiv N_e^{-\frac{1}{d}}$  is the average element diameter commonly used to measure convergence order in mesh adaptation. The convergence order,  $\mathcal{O}(\bar{h}^{1+s})$ , has also been established by Ainsworth and Glusa [3] for adaptive finite element approximations. Numerical results in Section 5 show that GoFD has similar convergence behavior for quasi-uniform meshes and second-order convergence (in  $L^2$  norm) for adaptive meshes.  $\square$

### 3.3 Preconditioning with sparse matrices

Various types of preconditioners have been developed for  $A_{\text{FD}}$ , including circulant preconditioners [10] and the direct use of the Laplacian [28]. In principle, we can use these preconditioners to replace  $A_{\text{FD}}$  in the stiffness matrix  $(I_h^{\text{FD}})^T A_{\text{FD}} I_h^{\text{FD}}$  and obtain a preconditioner for (46). Here we consider preconditioners based on sparse matrices. Notice that the fractional Laplacian approaches to the Laplacian operator as  $s \rightarrow 1$  and the identity operator as  $s \rightarrow 0$ . Thus, it is reasonable to build an efficient preconditioner based on the Laplacian at least when  $s$  is close to 1. First, we choose a

sparsity pattern based on the FD discretization of the Laplacian. For example, we can take the 5-point pattern (cf. (9)) or the 9-point pattern. Then, we form a sparse matrix using the entries of  $A_{\text{FD}}$  at the positions specified by the pattern. We denote these matrices by  $A_{\text{FD}}^{(5)}$  and  $A_{\text{FD}}^{(9)}$ , respectively. Next, we define

$$A_h^{(5)} = (I_h^{\text{FD}})^T A_{\text{FD}}^{(5)} I_h^{\text{FD}}, \quad A_h^{(9)} = (I_h^{\text{FD}})^T A_{\text{FD}}^{(9)} I_h^{\text{FD}}. \quad (52)$$

Finally, the preconditioners for (46) are obtained using the incomplete Cholesky decomposition of  $A_h^{(5)}$  and  $A_h^{(9)}$  with level-1 fill-ins. Notice that all of  $A_{\text{FD}}^{(5)}$  and  $A_{\text{FD}}^{(9)}$  and therefore,  $A_h^{(5)}$  and  $A_h^{(9)}$  are sparse and they can be computed economically. Effectiveness of these preconditioners will be demonstrated in numerical examples.

## 4 Mesh adaptation

It is known (e.g. see [9, 33]) that the solution of (1) has low regularity especially near the boundary of  $\Omega$ . Thus, it is useful to use mesh adaptation in the numerical solution of (1) to improve accuracy and convergence order. We recall that GoFD described in the previous section uses unstructured meshes for  $\Omega$ , which not only works for arbitrary geometry of  $\Omega$  but also allows easy incorporation with existing mesh adaptation algorithms.

---

### Algorithm 1 Adaptive mesh grid-overlay finite difference method

---

- Given an initial mesh  $\mathcal{T}_h^{(0)}$  for  $\Omega$ .
  - For  $\ell = 1, \dots, \ell_{max}$ 
    - Solve (46) on  $\mathcal{T}_h^{(\ell)}$  for  $u_h^{(\ell)}$ .
    - Generate a new mesh  $\mathcal{T}_h^{(\ell+1)}$  using the MMPDE method based on  $u_h^{(\ell)}$  and  $\mathcal{T}_h^{(\ell)}$ .
  - end  $\ell$
- 

We use here the MMPDE moving mesh method for mesh adaptation. The procedure for combining GoFD with the MMPDE method is given in Algorithm 1. We use  $\ell_{max} = 5$  in our computation. Numerical experiment shows that this is sufficient.

The MMPDE method is used to generate the new mesh  $\mathcal{T}_h^{(\ell+1)}$  for  $\Omega$ . The method has been developed (e.g., see [19, 20, 21]) for general purpose of mesh adaptation and movement. It uses the moving mesh PDE (or moving mesh equations in discrete form) to move vertices continuously in time and in an orderly manner in space. A key idea of the MMPDE method is viewing any nonuniform mesh as a uniform one in some Riemannian metric specified by a tensor  $\mathbb{M} = \mathbb{M}(\mathbf{x})$ . The metric tensor provides the information needed to control the size, shape, and orientation of mesh elements throughout the domain. Various metric tensors have been developed in [22]. For the current work, we employ a Hessian-based metric tensor

$$\mathbb{M}_K = \det \left( I + \frac{1}{\alpha_h} |H_K(u_h^{(\ell)})| \right)^{-\frac{1}{d+4}} \left( I + \frac{1}{\alpha_h} |H_K(u_h^{(\ell)})| \right), \quad \forall K \in \mathcal{T}_h^{(\ell)} \quad (53)$$

where  $\det(\cdot)$  denotes the determinant of a matrix,  $H_K(u_h^{(\ell)})$  is a recovered Hessian of  $u_h^{(\ell)}$  on the element  $K$  (through quadratic least squares fitting),  $|H_K(u_h^{(\ell)})| = \sqrt{H_K(u_h^{(\ell)})^2}$ , and  $\alpha_h$  is a regularization parameter defined through the following algebraic equation:

$$\sum_K |K| \det \left( I + \frac{1}{\alpha_h} |H_K(u_h^{(\ell)})| \right)^{\frac{2}{d+4}} = 2|\Omega|.$$

This metric tensor is known to be optimal for the  $L^2$ -norm of linear interpolation error [22].

It is known (e.g., see [19, 21]) that a uniform simplicial mesh  $\mathcal{T}_h$  in metric  $\mathbb{M}$  satisfies the following equidistribution and alignment conditions,

$$\sqrt{\det(\mathbb{M}_K)} |K| = \frac{\sigma_h}{N_e}, \quad \forall K \in \mathcal{T}_h \quad (54)$$

$$\frac{1}{2} \text{trace} \left( (F'_K)^{-1} \mathbb{M}_K^{-1} (F'_K)^{-T} \right) = \det \left( (F'_K)^{-1} \mathbb{M}_K^{-1} (F'_K)^{-T} \right)^{\frac{1}{2}}, \quad \forall K \in \mathcal{T}_h \quad (55)$$

where  $N_e$  denotes the number of elements in  $\mathcal{T}_h$ ,  $F'_K$  is the Jacobian matrix of the affine mapping  $F_K : \hat{K} \rightarrow K$ ,  $\hat{K}$  is the reference element taken as an equilateral simplex with unit volume, and

$$\sigma_h = \sum_K \sqrt{\det(\mathbb{M}_K)} |K|.$$

The condition (54) requires all elements to have the same size while (55) requires every element  $K$  to be similar to  $\hat{K}$ , in metric  $\mathbb{M}_K$ . An energy function associated with these conditions is given by

$$\begin{aligned} I_h = & \frac{1}{3} \sum_K \sqrt{\det(\mathbb{M}_K)} |K| \text{trace} \left( (F'_K)^{-1} \mathbb{M}_K^{-1} (F'_K)^{-T} \right)^{\frac{3d}{4}} \\ & + \frac{d^{\frac{3d}{4}}}{3} \sum_K \sqrt{\det(\mathbb{M}_K)} |K| \left( \sqrt{\det(\mathbb{M}_K)} \det(F'_K) \right)^{-\frac{3d}{4}}. \end{aligned} \quad (56)$$

This function is a Riemann sum of a continuous functional developed based on mesh equidistribution and alignment (e.g., see [21]).

The energy function  $I_h$  is a function of the coordinates of the vertices of  $\mathcal{T}_h$ , i.e.,  $I_h = I_h(\mathbf{x}_1, \dots, \mathbf{x}_{N_v})$ . An approach for minimizing this function is to integrate the gradient system of  $I_h$ . Thus, we define the moving mesh equations as

$$\frac{d\mathbf{x}_i}{dt} = -\frac{\sqrt{\det(\mathbb{M}(\mathbf{x}_i))}}{\tau} \frac{\partial I_h}{\partial \mathbf{x}_i}, \quad i = 1, \dots, N_v \quad (57)$$

where  $\tau > 0$  is a parameter used to adjust the time scale of mesh movement. The analytical expression of the derivative of  $I_h$  with respect to  $\mathbf{x}_i$  can be found using scalar-by-matrix differentiation [19]. Using this expression, we can rewrite (57) as

$$\frac{d\mathbf{x}_i}{dt} = \frac{\sqrt{\det(\mathbb{M}(\mathbf{x}_i))}}{\tau} \sum_{K \in \omega_i} |K| \mathbf{v}_{i_K}^K, \quad i = 1, \dots, N_v \quad (58)$$

where  $\mathbf{v}_{i_K}^K$  is the local mesh velocity contributed by element  $K$  to the vertex  $\mathbf{x}_i$ . The interested reader is referred to [19, Equations (38), (40), and (41)] for the analytical expression of  $\mathbf{v}_{i_K}^K$ .

The nodal velocity needs to be modified at boundary vertices. For fixed boundary vertices,  $\frac{d\mathbf{x}_i}{dt}$  should be set to be zero. If  $\mathbf{x}_i$  is allowed to slide along the boundary, the component of  $\frac{d\mathbf{x}_i}{dt}$  in the normal direction of the boundary should be set to be zero.

In our computation, the Matlab ODE solver *ode15s* (a variable-step, variable-order solver based on the numerical differentiation formulas of orders 1 to 5) is used to integrate (58), with the Jacobian matrix approximated by finite differences, over  $t \in (0, 1]$  with  $\tau = 10^{-2}$  and the initial mesh  $\mathcal{T}_h^{(\ell)}$ . The obtained mesh is  $\mathcal{T}_h^{(\ell+1)}$ . Notice that the mesh connectivity is kept fixed during the time integration. Thus,  $\mathcal{T}_h^{(\ell+1)}$  has the same connectivity as  $\mathcal{T}_h^{(\ell)}$ .

## 5 Numerical examples

In this section we present numerical results obtained with GoFD described in the previous sections for one 1D, three 2D, and one 3D examples. Three of those examples come from problem (1) with the following setting in different dimensions,

$$\Omega = B(0, 1), \quad f = \frac{2^{2s}\Gamma(1+s+k)\Gamma(\frac{d}{2}+s+k)}{k!\Gamma(\frac{d}{2}+k)} \cdot P_k^{s, \frac{d}{2}-1}(2|\mathbf{x}|^2 - 1), \quad (59)$$

where  $P_k^{s, \frac{d}{2}-1}(\cdot)$  is the Jacobi polynomial of degree  $k$  with parameters  $(s, \frac{d}{2} - 1)$  and  $B(0, 1)$  is a unit ball centered at the origin. Notice that  $f$  is constant for  $k = 0$ . This problem has an analytical exact solution

$$u = (1 - |\mathbf{x}|^2)_+^s P_k^{s, \frac{d}{2}-1}(2|\mathbf{x}|^2 - 1). \quad (60)$$

In this section, the solution error is plotted against  $N$ , the number of elements in  $\mathcal{T}_h$ . The convergence order is measured in terms of  $\bar{h} \equiv N_e^{-1/d}$ , the average element diameter for both fixed and adaptive meshes. For a fixed (and almost uniform) mesh,  $\bar{h}$  is equivalent to  $h$ , the maximum element diameter while for an adaptive mesh,  $\bar{h}$  makes more sense since the elements can have very different diameters. Moreover, we take  $R$  (half of the size of the overlay cube) as 1.1 times of half of the diameter of  $\Omega$ . We have tried 1.0 and 1.2 times and found no significant difference in the computed solution. Furthermore, we take  $h_{\text{FD}} = a_h$ . This is larger than what is given in the condition (40) but works well for all examples we have tested. This relation also implies that  $\mathcal{T}_h$  will become finer when  $\mathcal{T}_h$  is refined. Particularly,  $h_{\text{FD}}$  can become very small for a highly adaptive mesh  $\mathcal{T}_h$  with a small element height  $a_h$ .

**Example 5.1.** The first example is the 1D version of problem (59). For this problem, the FD scheme described in Section 2 can be used for uniform meshes but not for adaptive ones.

We consider the case with  $k = 0$ . The solution error in  $L^\infty$  and  $L^2$  norm is plotted in Fig. 2 for fixed and adaptive meshes. For fixed (uniform) meshes, the error behaves like  $\mathcal{O}(h^s)$  in  $L^\infty$  norm and  $\mathcal{O}(h^{\min(1, 0.5+s)})$  in  $L^2$  norm. This is consistent with the observations made by other researchers; cf. Remark 2.1. The solution error is also shown for adaptive meshes. Mesh adaptation improves accuracy and convergence order significantly. Indeed, the error decreases like  $\mathcal{O}(\bar{h}^{0.5+s})$  in  $L^\infty$  norm and  $\mathcal{O}(\bar{h}^2)$  in  $L^2$  norm for adaptive meshes.

Results for  $k > 0$  show similar behavior. They are not included here to save space.  $\square$

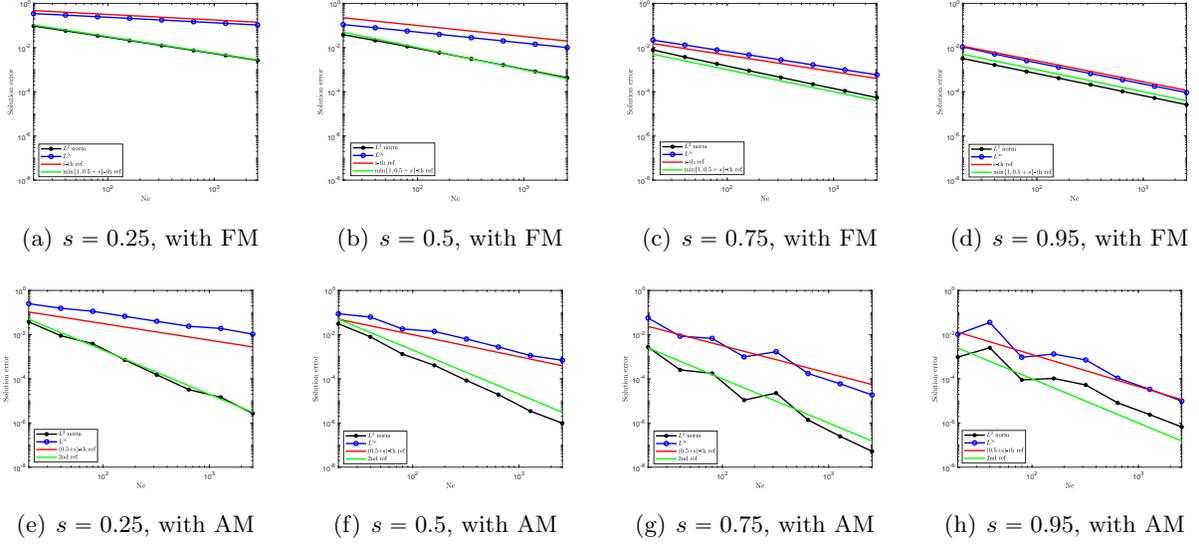


Figure 2: Example 5.1. The solution error is plotted as a function of  $N_e$  (the number of elements of  $\mathcal{T}_h$ ) for  $k = 0$ . FM stands for Fixed Mesh and AM stands for Adaptive Mesh.

**Example 5.2.** The second example is the 2D version of (59). We consider two cases with  $k = 0$  and 5 and  $s = 0.5$ . Fig. 3 shows computed solutions. The convergence histories are shown in Fig. 4. The  $L^2$  norm of the solution error converges like  $\mathcal{O}(h)$  for fixed meshes. This is consistent with finite element approximations (cf. Remark 3.6) since in this case with  $s = 0.5$ ,  $\mathcal{O}(h^{\min(1, 0.5+s)}) = \mathcal{O}(h)$ . On the other hand, the error is second order, i.e.,  $\mathcal{O}(\bar{h}^2)$ , for adaptive meshes. This is higher than the expected rate  $\mathcal{O}(\bar{h}^{1+s}) = \mathcal{O}(\bar{h}^{1.5})$  (cf. Remark 3.6). Higher accuracy with mesh adaptation can also be observed from the computed solutions. For instance, oscillations are visible in Fig. 3(c) but not in Fig. 3(d). Examples of adaptive mesh are shown in Fig. 5.

We now examine the effectiveness of the preconditioner described in Section 3.3. The convergence history for the conjugate gradient method (CG) with/without preconditioning is shown in Fig. 6. We can see that the preconditioner reduces the number of iterations significantly. Moreover, the preconditioner is more effective when  $s$  is closer to 1. Meanwhile, a smaller number of iterations is required to reach the same accuracy for  $s = 0.5$  than  $s = 0.9$ . These observations are consistent with the fact that the fractional Laplacian approaches to the Laplacian as  $s \rightarrow 1$  and the identity operator as  $s \rightarrow 0$ . As a result, the stiffness matrix of the FD approximation has a smaller condition number and the corresponding linear system is easier to solve for smaller  $s$ . Moreover, the preconditioner, whose pattern is based on that of the FD discretion of the Laplacian, can be expected to be more effective when the fractional Laplacian is closer to the Laplacian.

The CG convergence history is also plotted in Fig. 7 for the preconditioner based on the 5-point pattern. This preconditioner is slightly less effective than that based on the 9-point pattern.  $\square$

**Example 5.3.** Next we consider the 3D version of (59) for  $k = 0$  and  $s = 0.5$ . Fig. 8(a) shows the convergence history in  $L^2$  norm. The solution error converges slightly lower than  $\mathcal{O}(h)$  for fixed meshes and  $\mathcal{O}(\bar{h}^{1.5})$  for adaptive meshes. For this example, we take the 27-point pattern to build the

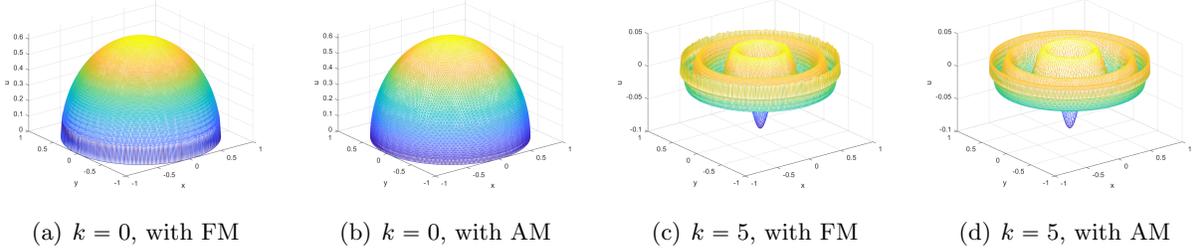


Figure 3: Example 5.2. Computed solutions obtained with meshes of  $N_e = 27130$  for  $s = 0.5$ . FM: fixed mesh and AM: adaptive mesh.

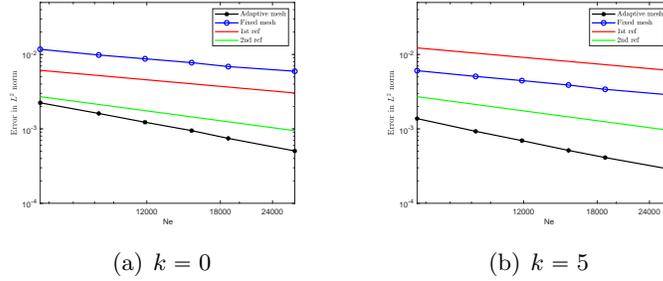


Figure 4: Example 5.2. The  $L^2$  norm of the solution error is plotted as a function of  $N_e$  for  $k = 0$  and  $5$  and  $s = 0.5$  with and without mesh adaptation.

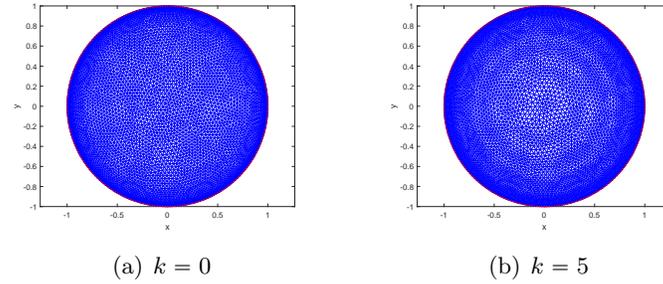


Figure 5: Example 5.2. Adaptive meshes of  $N_e = 11886$  for  $s = 0.5$ .

preconditioner. The CG convergence histories shown in Fig. 8(b) and (c) demonstrate the effectiveness of the preconditioner.  $\square$

**Example 5.4.** This example is (1) with  $f = 1$  and  $\Omega$  as shown in Fig. 1 with  $s = 0.75$ . The geometry of  $\Omega$  is complex, with the wavering outside boundary and two holes inside. An analytical exact solution is not available for this example. A computed solution with an adaptive mesh of  $N_e = 250948$  is used as the reference solution. Numerical results are shown in Fig. 9. The solution error in  $L^2$  norm is about  $\mathcal{O}(h)$  for fixed meshes and  $\mathcal{O}(\bar{h}^2)$  for adaptive meshes. This example demonstrates that GoFD works well with complex geometries.  $\square$

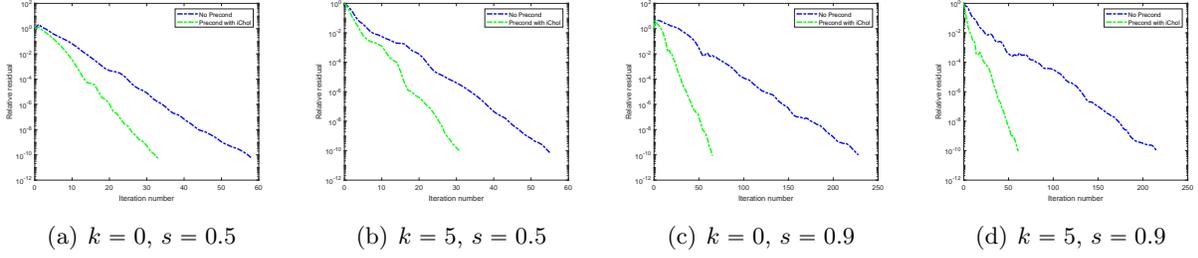


Figure 6: Example 5.2. The CG convergence history is plotted for a fixed mesh of  $N_e = 27130$  and with/without preconditioning (9-point pattern).

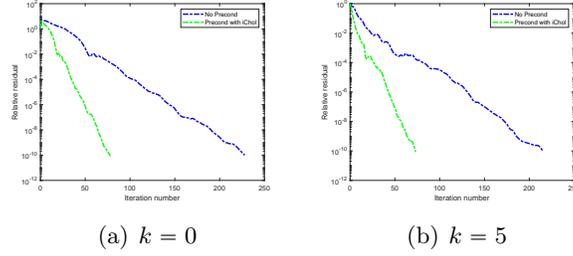


Figure 7: Example 5.2. The CG convergence history is plotted for  $k = 0$  and  $5$  and  $s = 0.9$  with a non-adaptive mesh of  $N_e = 27130$  and with/without preconditioning (5-point pattern).

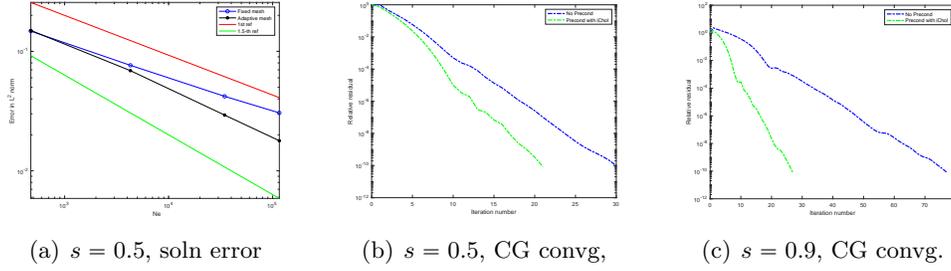


Figure 8: Example 5.3. (a): The solution error as function of  $N$  and (b) and (c): CG convergence histories for a fixed mesh of  $N_e = 116054$  with and without preconditioning.

**Example 5.5.** This example is (1) with  $f = 1$  and  $\Omega$  being  $L$ -shaped with  $s = 0.5$ . An analytical exact solution is not available for this example. A computed solution obtained with an adaptive mesh of  $N_e = 417508$  is used as the reference solution. Numerical results are shown in Figs. 10. The solution error in  $L^2$  norm is about  $\mathcal{O}(h)$  for fixed meshes and  $\mathcal{O}(\bar{h}^2)$  for adaptive meshes.  $\square$

## 6 Conclusions and further comments

In the previous sections we have studied a grid-overlay finite difference method (GoFD) for the numerical approximation of the fractional Laplacian on arbitrary bounded domains. The method uses an

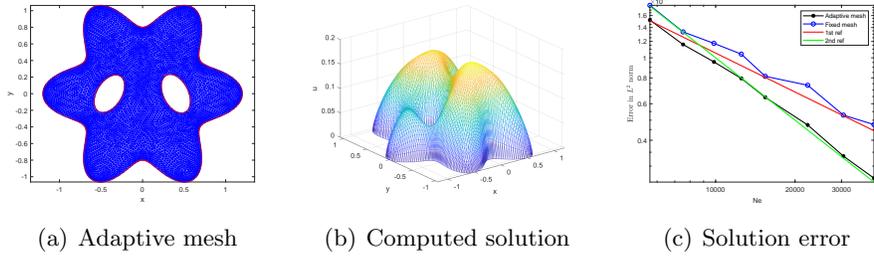


Figure 9: Example 5.4. (a): An adaptive mesh of  $N_e = 9850$ , (b) the corresponding computed solution, and (c) the solution error as function of  $N_e$  for  $s = 0.75$ .

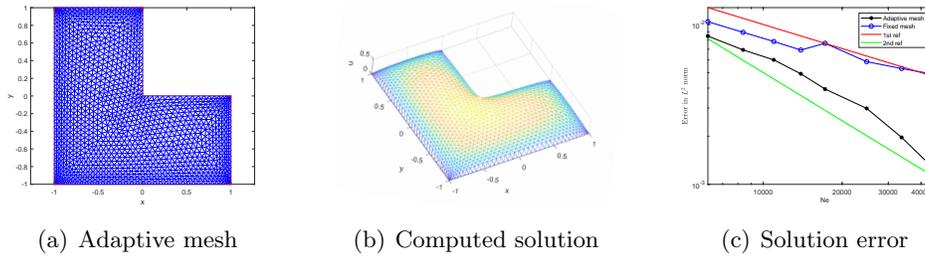


Figure 10: Example 5.5. (a): An adaptive mesh of  $N_e = 2728$ , (b) the corresponding computed solution, and (c) the solution error as function of  $N_e$  for  $s = 0.5$ .

unstructured mesh and an overlaying uniform grid and constructs the approximation matrix  $A_h$  (cf. (2)) based on the uniform-grid FD approximation  $A_{\text{FD}}$  (cf. (13) and (14)) and the transfer matrix  $I_h^{\text{FD}}$  from the unstructured mesh to the uniform grid. The multiplication of  $A_h$  with vectors can be carried out efficiently using FFT and sparse-matrix-vector multiplication. A main result is Theorem 3.1 stating that  $A_h$  is similar to a symmetric and positive definite matrix (and thus invertible) if  $I_h^{\text{FD}}$  has full column rank and positive column sums. A special choice of  $I_h^{\text{FD}}$  is piecewise linear interpolation. Theorem 3.2 states that the full column rank and positive column sums are guaranteed for this special choice if the spacing of the uniform grid satisfies (40). Stability and preconditioning for the resulting linear system have been discussed.

GoFD retains the efficient matrix-vector multiplication advantage of uniform-grid FD methods for the fraction Laplacian while being able to work for domains with complex geometries. Meanwhile, the method can readily be combined with existing adaptive mesh strategies due to its use of unstructured meshes. We have discussed in Section 4 how to combine GoFD with the MMPDE moving mesh method.

Numerical results have been presented for a selection of 1D, 2D and 3D examples. They have demonstrated that GoFD is feasible and convergent and has a convergence order of  $\mathcal{O}(h^{\min(1, 0.5+s)})$  in  $L^2$  norm for fixed meshes. This is consistent with observations known for existing uniform-grid FD and finite element methods. With adaptive meshes, the method shows second-order convergence in 1D and 2D and close to  $\mathcal{O}(\bar{h}^{1+s})$  in 3D. The numerical results have also demonstrated that the preconditioners based on the sparsity pattern of the Laplacian (cf. Section 3.3) are effective in terms

of reducing the number of iterations required to reach the commensurate accuracy.

Finally we comment that we have used unstructured simplicial meshes for  $\Omega$  in this work. The use of simplicial meshes makes it relatively simpler to prove the full column rank of  $I_h^{\text{FD}}$  and implement the transfer. However, it is not necessary to use simplicial meshes. We can use any other boundary fitted meshes or even meshless points. Particularly, we can take  $\mathcal{T}_h$  as a graded mesh. Moreover, we can use data transfer schemes other than linear interpolation that has been considered in this work. These are interesting topics worth future investigations.

## Acknowledgments

The first and second authors were supported in part by the University of Kansas General Research Fund FY23 and the National Natural Science Foundation of China through grant [12101509], respectively.

## 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] M. Ainsworth and C. Glusa, *Aspects of an adaptive finite element method for the fractional Laplacian: a priori and a posteriori error estimates, efficient implementation and multigrid solver*, *Comput. Methods Appl. Mech. Engrg.*, 327 (2017), 4–35.
- [4] M. Ainsworth and C. Glusa, *Towards an efficient finite element method for the integral fractional Laplacian on polygonal domains*, in *Contemporary computational mathematics – A celebration of the 80th birthday of Ian Sloan*. Vol. 1, 2, Springer, Cham, 2018, 17–57.
- [5] H. Antil, T. Brown, R. Khatri, A. Onwunta, D. Verma, and M. Warma, *Chapter 3 - Optimal control, numerics, and applications of fractional PDEs*, *Handbook of Numerical Analysis*, 23 (2022), 87–114.
- [6] H. Antil, P. Dondl, and L. Striet, *Approximation of integral fractional Laplacian and fractional PDEs via sinc-basis*, *SIAM J. Sci. Comput.*, 43 (2021), A2897–A2922.
- [7] H. Bahouri, J.-Y. Chemin, and R. Danchin, *Fourier analysis and nonlinear partial differential equations*, vol. 343 of *Grundlehren der mathematischen Wissenschaften [Fundamental Principles of Mathematical Sciences]*, Springer, Heidelberg, 2011.
- [8] A. Bonito, W. Lei, and J. E. Pasciak, *Numerical approximation of the integral fractional Laplacian*, *Numer. Math.*, 142 (2019), 235–278.
- [9] J. P. Borthagaray, L. M. Del Pezzo, and S. Martínez, *Finite element approximation for the fractional eigenvalue problem*, *J. Sci. Comput.*, 77 (2018), 308–329.

- [10] R. H. Chan and M. K. Ng, *Conjugate gradient methods for Toeplitz systems*, SIAM Rev., 38 (1996), 427–482.
- [11] N. Du, H.-W. Sun, and H. Wang, *A preconditioned fast finite difference scheme for space-fractional diffusion equations in convex domains*, Comput. Appl. Math., 38 (2019), Paper No. 14.
- [12] Q. Du, L. Ju, and J. Lu, *A discontinuous Galerkin method for one-dimensional time-dependent nonlocal diffusion problems*, Math. Comp., 88 (2019), 123–147.
- [13] Q. Du, L. Ju, J. Lu, and X. Tian, *A discontinuous Galerkin method with penalty for one-dimensional nonlocal diffusion problems*, Commun. Appl. Math. Comput., 2 (2020), 31–55.
- [14] 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.
- [15] M. Faustmann, M. Karkulik, and J. M. Melenk, *Local convergence of the FEM for the integral fractional Laplacian*, SIAM J. Numer. Anal., 60 (2022), 1055–1082.
- [16] L. N. G. Filon, *On a quadrature formula for trigonometric integrals*, Model. Anal. Inf. Sist., 49 (1928), 38–47.
- [17] Z. Hao, Z. Zhang, and R. Du, *Fractional centered difference scheme for high-dimensional integral fractional Laplacian*, J. Comput. Phys., 424 (2021), Paper No. 109851.
- [18] R. A. Horn and C. A. Johnson, *Matrix Analysis*, Cambridge University Press, Cambridge, London, 1985.
- [19] W. Huang and L. Kamenski, *A geometric discretization and a simple implementation for variational mesh generation and adaptation*, J. Comput. Phys., 301 (2015), 322–337.
- [20] W. Huang, Y. Ren, and R. D. Russell, *Moving mesh partial differential equations (MMPDEs) based upon the equidistribution principle*, SIAM J. Numer. Anal., 31 (1994), 709–730.
- [21] W. Huang and R. D. Russell, *Adaptive Moving Mesh Methods*, Springer, New York, 2011. Applied Mathematical Sciences Series, Vol. 174.
- [22] W. Huang and W. Sun, *Variational mesh adaptation II: error estimates and monitor functions*, J. Comput. Phys., 184 (2003), 619–648.
- [23] Y. Huang and A. Oberman, *Numerical methods for the fractional Laplacian: a finite difference-quadrature approach*, SIAM J. Numer. Anal., 52 (2014), 3056–3084.
- [24] Y. Huang and A. Oberman, *Finite difference methods for fractional laplacians*, arXiv preprint arXiv:1611.00164, (2016).
- [25] M. Ilic, F. Liu, I. Turner, and V. Anh, *Numerical approximation of a fractional-in-space diffusion equation. I*, Fract. Calc. Appl. Anal., 8 (2005), 323–341.

- [26] H. Li, R. Liu, and L.-L. Wang, *Efficient hermite spectral-Galerkin methods for nonlocal diffusion equations in unbounded domains*, Numer. Math. Theory Methods Appl., 15 (2022), 1009–1040.
- [27] A. Lischke, G. Pang, M. Gulian, and et al., *What is the fractional Laplacian? A comparative review with new results*, J. Comput. Phys., 404 (2020), 109009.
- [28] V. Minden and L. Ying, *A simple solver for the fractional laplacian in multiple dimensions*, SIAM J. Sci. Comput., 42 (2020), A878–A900.
- [29] M. V. Nevskii, *On some problems for a simplex and a ball in  $\mathbb{R}^n$* , Model. Anal. Inf. Sist., 25 (2018), 680–691.
- [30] M. D. Ortigueira, *Riesz potential operators and inverses via fractional centred derivatives*, Int. J. Math. Math. Sci., (2006), Art. ID 48391.
- [31] M. D. Ortigueira, *Fractional central differences and derivatives*, J. Vib. Control, 14 (2008), 1255–1266.
- [32] H.-K. Pang and H.-W. Sun, *Multigrid method for fractional diffusion equations*, J. Comput. Phys., 231 (2012), 693–703.
- [33] X. Ros-Oton and J. Serra, *The Dirichlet problem for the fractional Laplacian: regularity up to the boundary*, J. Math. Pures Appl., 101 (2014), 275–302.
- [34] N. Saito and G. Zhou, *Analysis of the fictitious domain method with an  $L^2$ -penalty for elliptic problems*, Numer. Funct. Anal. Optim., 36 (2015), 501–527.
- [35] F. Song, C. Xu, and G. E. Karniadakis, *Computing fractional Laplacians on complex-geometry domains: algorithms and simulations*, SIAM J. Sci. Comput., 39 (2017), A1320–A1344.
- [36] J. Sun, D. Nie, and W. Deng, *Algorithm implementation and numerical analysis for the two-dimensional tempered fractional Laplacian*, BIT, 61 (2021), 1421–1452.
- [37] X. Tian and Q. Du, *Analysis and comparison of different approximations to nonlocal diffusion and linear peridynamic equations*, SIAM J. Numer. Anal., 51 (2013), 3458–3482.
- [38] H. Wang and T. S. Basu, *A fast finite difference method for two-dimensional space-fractional diffusion equations*, SIAM J. Sci. Comput., 34 (2012), A2444–A2458.
- [39] Q. Yang, I. Turner, F. Liu, and M. Ilić, *Novel numerical methods for solving the time-space fractional diffusion equation in two dimensions*, SIAM J. Sci. Comput., 33 (2011), 1159–1180.