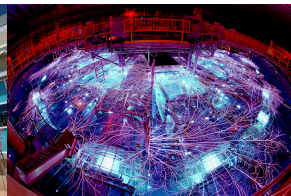


Exceptional service in the national interest



Domain Decomposition Solvers for Nonlocal Equations

Christian Glusa

SIAM PP 22
February 23, 2022



Sandia National Laboratories is a multimission laboratory managed and operated by National Technology and Engineering Solutions of Sandia, LLC, a wholly owned subsidiary of Honeywell International, Inc., for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-NA0003525. SAND NO. XXXXXXXXXX

Sandia National Laboratories is a multimission laboratory managed and operated by National Technology & Engineering Solutions of Sandia, LLC, a wholly owned subsidiary of Honeywell International Inc., for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-NA0003525.

Elliptic nonlocal operators

Let $\delta \in (0, \infty]$ be the *horizon*, $\Omega \subset \mathbb{R}^d$ a bounded open domain, define the *interaction domain*

$$\Omega_I := \{\vec{y} \in \mathbb{R}^d \setminus \Omega : |\vec{x} - \vec{y}| \leq \delta, \text{ for } \vec{x} \in \Omega\}.$$

We want to numerically solve equations involving the nonlocal operator

$$\mathcal{L}u(\vec{x}) = \text{p.v.} \int_{\Omega \cup \Omega_I} (u(\vec{y}) - u(\vec{x})) \gamma(\vec{x}, \vec{y}) d\vec{y}, \quad \vec{x} \in \Omega,$$

with

$$\begin{aligned} \gamma(\vec{x}, \vec{y}) &= \phi(\vec{x}, \vec{y}) |\vec{x} - \vec{y}|^{-\beta(\vec{x}, \vec{y})} \chi_{|\vec{x} - \vec{y}| \leq \delta}, & \vec{x}, \vec{y} \in \Omega \cup \Omega_I, \\ \phi(\vec{x}, \vec{y}) &> 0. \end{aligned}$$

■ Examples:

- Integral fractional Laplacian: $\phi \sim \text{const}$, $\beta = d + 2s$, $s \in (0, 1)$, $\delta = \infty$
- Tempered fractional Laplacian: $\phi(\vec{x}, \vec{y}) \sim \exp(-\lambda|\vec{x} - \vec{y}|)$
- Truncated fractional Laplacian: δ finite
- Variable order fractional Laplacians with varying coefficient: $\beta(\vec{x}, \vec{y}) = d + 2s(\vec{x}, \vec{y})$, $\phi(\vec{x}, \vec{y}) > 0$
- Integrable kernels: constant kernel ($\beta = 0$), “peridynamic” kernel ($\beta = 1$)

■ Assumptions (for now):

- γ is symmetric.
- Interaction domain is defined wrt ℓ_2 -norm.

■ Nonlocal Poisson's equation:

$$\begin{aligned} -\mathcal{L}u &= f && \text{in } \Omega, \\ u &= 0 && \text{in } \Omega_I. \end{aligned}$$

■ Nonlocal heat equation:

$$\begin{aligned} u_t - \mathcal{L}u &= f && \text{in } (0, T) \times \Omega, \\ u &= 0 && \text{in } (0, T) \times \Omega_I, \\ u &= u_0 && \text{on } \{0\} \times \Omega. \end{aligned}$$

■ Source control

■ Parameter learning:

$$\min_{u, s, \delta, \dots} \frac{1}{2} \|u - u_d\|_{L^2}^2 + \mathcal{R}(s, \delta, \dots)$$

subject to nonlocal equation.

■ Remark: Homogeneous Dirichlet “boundary” condition for simplicity.

Goal

Assemble and solve nonlocal equations in similar complexity & memory as their local counterparts, i.e. $\mathcal{O}(n \log n)$.

- We consider

$$a(u, v) = \frac{1}{2} \int_{\Omega} d\vec{x} \int_{\Omega} d\vec{y} [(u(\vec{x}) - u(\vec{y})) (v(\vec{x}) - v(\vec{y}))] \gamma(\vec{x}, \vec{y}) \\ + \int_{\Omega} d\vec{x} \int_{\Omega_l} d\vec{y} u(\vec{x}) v(\vec{x}) \gamma(\vec{x}, \vec{y}).$$

posed on $\tilde{H}^s(\Omega)$ or $L^2(\Omega)$ respectively, where

$$H^s(\Omega) := \left\{ u \in L^2(\Omega) \mid \|u\|_{H^s(\Omega)} < \infty \right\}, \quad \tilde{H}^s(\Omega) := \left\{ u \in H^s(\mathbb{R}^d) \mid u = 0 \text{ in } \Omega^c \right\},$$

and

$$\|u\|_{H^s(\Omega)}^2 = \|u\|_{L^2(\Omega)}^2 + \int_{\Omega} d\vec{x} \int_{\Omega} d\vec{y} \frac{(u(\vec{x}) - u(\vec{y}))^2}{|\vec{x} - \vec{y}|^{d+2s}}, \\ \|u\|_{\tilde{H}^s(\Omega)}^2 = \int_{\mathbb{R}^d} d\vec{x} \int_{\mathbb{R}^d} d\vec{y} \frac{(u(\vec{x}) - u(\vec{y}))^2}{|\vec{x} - \vec{y}|^{d+2s}}.$$

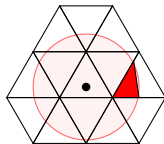
- For $\delta = \infty$, if $\gamma(\vec{x}, \vec{y}) = \nabla_{\vec{y}} \cdot \vec{\Gamma}(\vec{x}, \vec{y})$, can reduce integral from $\Omega \times \Omega^c$ to $\Omega \times \partial\Omega$.
(E.g. $\Gamma(\vec{x}, \vec{y}) \sim \frac{\vec{x} - \vec{y}}{|\vec{x} - \vec{y}|^{d+2s}}$ for the constant-order fractional kernel.)

- Partition domain into shape-regular mesh $\mathcal{P}_h = \{K\}$ with edges e on the boundary $\partial\Omega$.
- Set $V_h \subset \tilde{H}^s(\Omega)$ the space of continuous, piecewise linear functions.

$$a(u, v) = \frac{1}{2} \sum_K \sum_{\tilde{K}} \int_K d\vec{x} \int_{\tilde{K}} d\vec{y} (u(\vec{x}) - u(\vec{y})) (v(\vec{x}) - v(\vec{y})) \gamma(\vec{x}, \vec{y}) \\ + \sum_K \sum_e \int_K d\vec{x} u(\vec{x}) v(\vec{x}) \int_e d\vec{y} \vec{n}_e \cdot \Gamma(\vec{x}, \vec{y}).$$

$\dim V_h =: n$

- Approximate cut elements with simplices, $\mathcal{O}(h_K^2)$ error¹

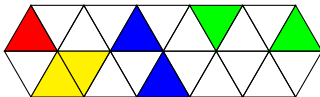


¹Marta D'Elia, Max Gunzburger, and Christian Vollmann. "A cookbook for approximating Euclidean balls and for quadrature rules in finite element methods for nonlocal problems". In: *Mathematical Models and Methods in Applied Sciences* 31.08 (2021), pp. 1505–1567.

- In subassembly procedure, use quadrature to evaluate element pair contributions:

$$a^{K \times \tilde{K}}(\phi_i, \phi_j) = \frac{1}{2} \int_K d\vec{x} \int_{\tilde{K}} d\vec{y} (\phi_i(\vec{x}) - \phi_i(\vec{y})) (\phi_j(\vec{x}) - \phi_j(\vec{y})) \gamma(\vec{x}, \vec{y})$$

- Treatment for element pairs $K \cap \tilde{K} \neq \emptyset$:



- split $K \times \tilde{K}$ into sub-simplices,
- Duffy transform onto a hypercube, with Jacobian canceling the singularity.
- Choose quadrature order so that quadrature error \leq discretization error²:
 - $|\log h_K|$ if the elements coincide (red),
 - $|\log h_K|^2$ if the elements share only an edge (yellow),
 - $|\log h_K|^3$ if the elements share only a vertex (blue),
 - $|\log h_K|^4$ if the elements are "near neighbours" (green), and
 - C if the elements are well separated.

²Mark Ainsworth and Christian Glusa. "Aspects of an adaptive finite element method for the fractional Laplacian: A priori and a posteriori error estimates, efficient implementation and multigrid solver". In: *Computer Methods in Applied Mechanics and Engineering* (2017).

$\mathcal{O}(n \log n)$ approximations to the stiffness matrix

Depending on δ and h :

- Straightforward discretization can lead to a fully dense matrix.
- Assembly and solve would have at least $\mathcal{O}(n^2)$ complexity and memory requirement.

Better approach

Panel clustering / Fast Multipole Method / hierarchical matrix approximation

- Find low-rank representations of off-diagonal matrix blocks:
lots of methods, varying level of intrusiveness.
- Important: we don't want to assemble a dense matrix and then compress it.
- Approximation incurs error. Control it so that it is dominated by discretization error.

\mathcal{H}^2 -matrix approximation^{3,4}

FE assembly and matrix-vector product in $\mathcal{O}(n \log^{2d} n)$ operations.

³Mark Ainsworth and Christian 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*. Springer, 2018, pp. 17–57.

⁴Mark Ainsworth and Christian Glusa. "Aspects of an adaptive finite element method for the fractional Laplacian: A priori and a posteriori error estimates, efficient implementation and multigrid solver". In: *Computer Methods in Applied Mechanics and Engineering* (2017).

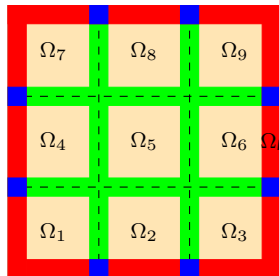
- $\mathcal{O}(n \log n)$ matrix-vector product in all cases \rightarrow can explore iterative solvers
- Steady-state:
 - Fractional kernel, $\delta = \infty$ ⁵: $\kappa(\mathbf{A}) \sim h^{-2s} \sim n^{2s/d}$
 - Fractional kernel, $\delta \leq \delta_0$ ⁶: $\kappa(\mathbf{A}) \sim \delta^{2s-2} h^{-2s} \sim \delta^{2s-2} n^{2s/d}$
 - Constant kernel, δ finite⁴: $\kappa(\mathbf{A}) \sim \delta^{-2}$
- Time-dependent:
 - $\kappa(\mathbf{M} + \Delta t \mathbf{A}) \sim 1 + \Delta t \kappa(\mathbf{A})$
 - Depending on time-stepper and CFL condition, this is well-conditioned for small s , large δ .
- Scalable solver options:
 - Multigrid
 - Geometric (GMG)
 - Algebraic (AMG)
 - Domain decomposition
 - Substructuring
 - Schwarz methods
 - Krylov methods

The matrix is well-conditioned in the certain parameter regimes, e.g.

 - constant kernel, δ large, or
 - fractional kernel, s small, δ large.

⁵Mark Ainsworth, William McLean, and Thanh Tran. "The conditioning of boundary element equations on locally refined meshes and preconditioning by diagonal scaling". In: *SIAM Journal on Numerical Analysis* 36.6 (1999), pp. 1901–1932.

⁶Burak Aksoylu and Zuhair Unlu. "Conditioning analysis of nonlocal integral operators in fractional Sobolev spaces". In: *SIAM Journal on Numerical Analysis* 52.2 (2014), pp. 653–677.



- Assume $\delta = \mathcal{O}(h)$.
- Cover with overlapping subdomains
 $\Omega \cup \Omega_l = \bigcup \Omega_i$, $\text{diam}(\Omega_i \cap \Omega_j) \sim \delta$ for adjacent subdomains.
- Duplicate unknowns in overlaps:

$$\mathbf{A}\vec{u} = \vec{f} \Leftrightarrow \begin{pmatrix} \mathbf{A}_{\epsilon\epsilon} & \mathbf{M}^T \\ \mathbf{M} & 0 \end{pmatrix} \begin{pmatrix} \vec{u}_\epsilon \\ \vec{\lambda} \end{pmatrix} = \begin{pmatrix} \vec{f}_\epsilon \\ 0 \end{pmatrix}$$

- $\mathbf{A}_{\epsilon\epsilon}$ is block diagonal by subdomain, partition-of-unity type scaling included.
- For floating subdomains, local matrix \mathbf{A}_p is singular.
- \mathbf{M} has entries $\{\pm 1, 0\}$, encodes the identity constraints on the overlaps (non-redundant).

⁷Giacomo Capodaglio, Marta D'Elia, Pavel Bochev, and Max Gunzburger. "An energy-based coupling approach to nonlocal interface problems". In: *Computers & Fluids* 207 (2020), p. 104593.

⁸Xiao Xu, Christian Glusa, Marta D'Elia, and John T. Foster. "A FETI approach to domain decomposition for meshfree discretizations of nonlocal problems". In: *Computer Methods in Applied Mechanics and Engineering* 387 (2021), p. 114148.

⁹WIP with Bochev, Capodaglio, D'Elia, Gunzburger, Klar, Vollmann

Reduced system and Dirichlet preconditioner

- Let nullspace of $\mathbf{A}_{\epsilon\epsilon}$ be given by \mathbf{Z} .
- Eliminate primal variables from

$$\begin{pmatrix} \mathbf{A}_{\epsilon\epsilon} & \mathbf{M}^T \\ \mathbf{M} & 0 \end{pmatrix} \begin{pmatrix} \vec{u}_\epsilon \\ \vec{\lambda} \end{pmatrix} = \begin{pmatrix} \vec{f}_\epsilon \\ 0 \end{pmatrix}$$

and obtain

$$\begin{aligned} \mathbf{P}_0 \mathbf{K} \vec{\lambda} &= \mathbf{P}_0 (\mathbf{M} \mathbf{A}_{\epsilon\epsilon}^\dagger \vec{f}_\epsilon) \\ \mathbf{G}^T \vec{\lambda} &= \mathbf{Z}^T \vec{f}_\epsilon, \end{aligned}$$

where $\mathbf{K} = \mathbf{M} \mathbf{A}_{\epsilon\epsilon}^\dagger \mathbf{M}^T$, $\mathbf{G} = \mathbf{M} \mathbf{Z}$, $\mathbf{P}_0 = \mathbf{I} - \mathbf{G}(\mathbf{G}^T \mathbf{G})^\dagger \mathbf{G}^T$.

- Use projected CG to solve system.
- \mathbf{P}_0 acts as a “coarse grid”.
- Preconditioner for \mathbf{K} :
 - Let \mathbf{A}_p , \mathbf{M}_p be local parts of $\mathbf{A}_{\epsilon\epsilon}$ and \mathbf{M} .
 - Write $\mathbf{K} = \sum_{p=1}^P \mathbf{M}_p \mathbf{A}_p^\dagger \mathbf{M}_p^T = \sum_{p=1}^P \tilde{\mathbf{M}}_p \mathbf{S}_p^\dagger \tilde{\mathbf{M}}_p^T$.
 - Dirichlet preconditioner: $\mathbf{Q} = \sum_{p=1}^P \tilde{\mathbf{M}}_p \mathbf{S}_p \tilde{\mathbf{M}}_p^T$.
- Results shown use Manuel Klar’s (U of Trier) assembly code
<https://gitlab.uni-trier.de/klar/nonlocal-assembly>

Weak scaling – 2D, constant kernel

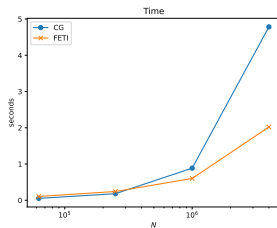
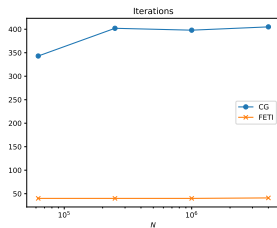


Figure: $\delta = 8e - 3 \rightarrow \kappa \sim \text{const}$

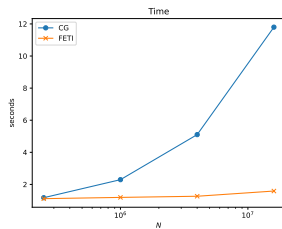
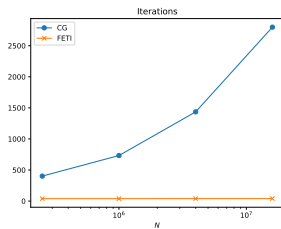


Figure: $\delta = 4h \rightarrow \kappa \sim N$

Weak scaling - 2D, fractional kernel, $s = 0.4$

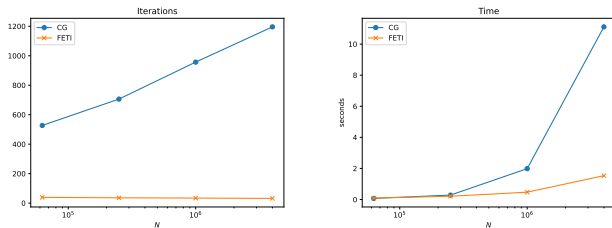


Figure: $\delta = 8e-3 \rightarrow \kappa \sim N^5$

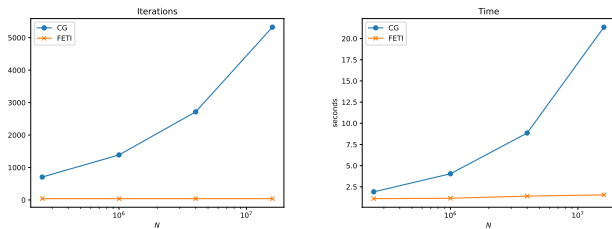


Figure: $\delta = 4h \rightarrow \kappa \sim N$

Strong scaling, 2D

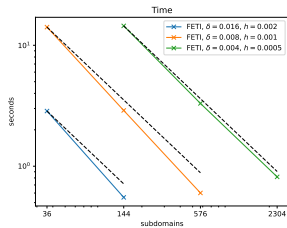


Figure: constant kernel, $\delta = 8h$.

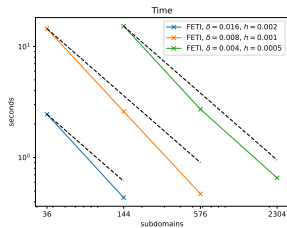


Figure: fractional kernel, $s = 0.4$, $\delta = 8h$.

- Drawback of substructuring: cannot handle $\delta \gg h$.
- Schwarz method
 - overlapping subdomain restrictions $\{\mathbf{R}_p\}$, local matrices $\mathbf{A}_p = \mathbf{R}_p \mathbf{A} \mathbf{R}_p^T$
 - partition of unity $\sum_{p=1}^P \mathbf{R}_p^T \mathbf{D}_p \mathbf{R}_p = \mathbf{I}$, with $\{\mathbf{D}_p\}$ diagonal
 - additive Schwarz preconditioner: $\mathbf{Q}_1 := \sum_{p=1}^P \mathbf{R}_p^T \mathbf{A}_p^{-1} \mathbf{R}_p$, or restricted additive Schwarz
- No global information exchange \rightarrow need a coarse grid
- GenEO approach:

Span coarse space using solutions of subdomain eigenvalue problems

$\mathbf{D}_p \mathbf{A}_p \mathbf{D}_p \vec{v}_{p,k} = \lambda_{p,k} \mathbf{B}_p \vec{v}_{p,k}$, where \mathbf{B}_p is similar to \mathbf{A}_p , but assembled over a modified local mesh.
- Distributed \mathcal{H} -matrix is built using Pierre Marchand's Htool library
<https://github.com/htool-ddm/htool>
- Pierre Jolivet's HPDDM library for DD and GenEO <https://github.com/hpddm/hpddm>
- 2D fractional Poisson problem, $s = 0.75$, $\delta = \infty$

#DoFs	# MPI ranks	dense	\mathcal{H}	GMRES+DD #its (solve time)
65,025	72	31.5 GB	5.4 GB	21 (1.34s)
261,121	288	508 GB	12.6 GB	23 (0.96s)
1,046,529	1152	8160 GB	86 GB	24 (2.4s)
- Caveats:
 - solver setup needs improvement, working on alternative low-rank approximations
 - direct solves and eigenvalue problems (for now) in dense format

¹⁰with Pierre Marchand (INRIA)

- Domain decomposition solvers (and multigrid) are optimal for nonlocal problems.
- Discretized fractional equations are dense, but not structurally dense.
→ approximation of off-diagonal matrix blocks
- Resulting approaches have essentially the same complexity as PDE case.

Thanks for listening!



Funding:

The MATNIP LDRD project (PI: Marta D'Elia) develops for the first time a rigorous nonlocal interface theory based on physical principles that is consistent with the classical theory of partial differential equations when the nonlocality vanishes and is mathematically well-posed. This will improve the predictive capability of nonlocal models and increase their usability at Sandia and, more in general, in the computational-science and engineering community. Furthermore, this theory will provide the groundwork for the development of nonlocal solvers, reducing the burden of prohibitively expensive computations.

References I

- [1] Mark Ainsworth and Christian Glusa. "Aspects of an adaptive finite element method for the fractional Laplacian: A priori and a posteriori error estimates, efficient implementation and multigrid solver". In: *Computer Methods in Applied Mechanics and Engineering* (2017).
- [2] Mark Ainsworth and Christian 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*. Springer, 2018.
- [3] Mark Ainsworth, William McLean, and Thanh Tran. "The conditioning of boundary element equations on locally refined meshes and preconditioning by diagonal scaling". In: *SIAM Journal on Numerical Analysis* 36.6 (1999).
- [4] Burak Aksoylu and Zuhail Unlu. "Conditioning analysis of nonlocal integral operators in fractional Sobolev spaces". In: *SIAM Journal on Numerical Analysis* 52.2 (2014).
- [5] Olena Burkovska, Christian Glusa, and Marta D'Elia. "An optimization-based approach to parameter learning for fractional type nonlocal models". In: *Computers & Mathematics with Applications* (2021).
- [6] Olena Burkovska and Max Gunzburger. "Affine approximation of parametrized kernels and model order reduction for nonlocal and fractional Laplace models". In: *SIAM Journal on Numerical Analysis* 58.3 (2020).
- [7] Giacomo Capodaglio et al. "An energy-based coupling approach to nonlocal interface problems". In: *Computers & Fluids* 207 (2020).

- [8] Marta D'Elia and Christian A. Glusa. *A fractional model for anomalous diffusion with increased variability. Analysis, algorithms and applications to interface problems*. 2021.
- [9] Marta D'Elia, Max Gunzburger, and Christian Vollmann. "A cookbook for approximating Euclidean balls and for quadrature rules in finite element methods for nonlocal problems". In: *Mathematical Models and Methods in Applied Sciences* 31.08 (2021).
- [10] Xiao Xu et al. "A FETI approach to domain decomposition for meshfree discretizations of nonlocal problems". In: *Computer Methods in Applied Mechanics and Engineering* 387 (2021).

$\mathcal{O}(n \log n)$ approximations to the stiffness matrix

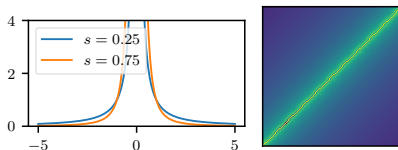


Figure: Left: Fractional kernels in $d = 1$ dimensions. Right: Magnitude of matrix entries.

Depending on δ and h :

- Straightforward discretization can lead to a fully dense matrix.
- Assembly and solve would have at least $\mathcal{O}(n^2)$ complexity and memory requirement.

Better approach

Panel clustering / Fast Multipole Method / hierarchical matrix approximation

- Find low-rank representations of off-diagonal matrix blocks.
- Lots of methods for computing a structurally sparse approximation, varying level of intrusiveness. I will show what I use: *panel clustering*.
- Important: we don't want to assemble a dense matrix and then compress it.
- Approximation incurs error. The game is to control it so that it is dominated by discretization error.

First question: Which sub-blocks of the matrix do we want to compress?

Build tree of clusters of DoFs.

- root contains all unknowns
- subdivision based on coordinates
- distributed computations: first level given by partition of unknowns



Figure: A cluster tree in $d = 1$ dimensions.

- Find cluster pairs (P, Q) that are *admissible* for approximation: sufficient separation compared to sizes.
- Matrix entries that are not part of any admissible cluster pair are assembled directly into the sparse near-field matrix A_{near} .

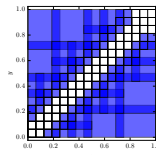


Figure: Elements of admissible cluster pairs in blue. Overlaps in dark blue.

Cluster method – \mathcal{H} -matrices

Let $P, Q \subset \Omega$, P and Q admissible.

Let ϕ, ψ be FE basis functions with $\text{supp } \phi \subset P$, $\text{supp } \psi \subset Q$.

$$a(\phi, \psi) = - \int_{\Omega} \int_{\Omega} \gamma(\vec{x}, \vec{y}) \phi(\vec{x}) \psi(\vec{y}) \, d\vec{x} \, d\vec{y}.$$

Let $\bar{\xi}_{\alpha}^P$ be Chebyshev nodes in P and L_{α}^P the associated Lagrange polynomials. Then

$$\gamma(\vec{x}, \vec{y}) \approx \sum_{\alpha, \beta=1}^{m^d} \gamma(\bar{\xi}_{\alpha}^P, \bar{\xi}_{\beta}^Q) L_{\alpha}^P(\vec{x}) L_{\beta}^Q(\vec{y}), \quad \vec{x} \in P, \vec{y} \in Q.$$

and

$$a(\phi, \psi) \approx - \sum_{\alpha, \beta=1}^{m^d} \gamma(\bar{\xi}_{\alpha}^P, \bar{\xi}_{\beta}^Q) \int_P \phi(\vec{x}) L_{\alpha}^P(\vec{x}) \, d\vec{x} \int_Q \psi(\vec{y}) L_{\beta}^Q(\vec{y}) \, d\vec{y}.$$

- Decouples ϕ and ψ , “sparsifies” off-diagonal matrix blocks.
- Replaces subblock of $a(\cdot, \cdot)$ with a low rank approximation $U_P \Sigma_{(P,Q)} U_Q^T$ with tall and skinny U_P, U_Q .
- If we stop now, we have constructed a so-called \mathcal{H} -matrix approximation:

$$A \approx A_{\text{near}} + A_{\text{far}} = A_{\text{near}} + \sum_{(P,Q) \text{ admissible}} U_P \Sigma_{(P,Q)} U_Q^T.$$

Cluster method – \mathcal{H}^2 -matrices

For \vec{x} in a sub-cluster P of Q , i.e. $P \subset Q$,

$$L_{\alpha}^Q(\vec{x}) = \sum_{\beta=1}^{m^d} L_{\alpha}^Q(\xi_{\beta}^P) L_{\beta}^P(\vec{x}).$$

Need to compute

- Far-field coefficients $\int_P \phi(\vec{x}) L_{\alpha}^P(\vec{x}) d\vec{x}$ only for leaves of the cluster tree,
- shift coefficients $L_{\alpha}^Q(\xi_{\beta}^P)$,
- kernel approximations $\gamma(\xi_{\alpha}^P, \xi_{\beta}^Q)$,
- near-field entries.

\mathcal{H}^2 -matrix approximation¹¹¹²

FE assembly and matrix-vector product in $\mathcal{O}(n \log^{2d} n)$ operations.

- Finite δ : need to be able to form clusters that fit within the horizon.
- Less intrusive but more costly way of computing far-field interactions via entry sampling:
Adaptive Cross Approximation (ACA)

¹¹Mark Ainsworth and Christian 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*. Springer, 2018, pp. 17–57.

¹²Mark Ainsworth and Christian Glusa. "Aspects of an adaptive finite element method for the fractional Laplacian: A priori and a posteriori error estimates, efficient implementation and multigrid solver". In: *Computer Methods in Applied Mechanics and Engineering* (2017).

Parameter learning problem requires operators for different values of s and δ .

- Piecewise Chebyshev interpolation in s :

Lemma

Let $s \in [s_{\min}, s_{\max}] \subset (0, 1)$, $\delta \in (0, \infty)$, and let $\eta > 0$. Assume that $u \in H_{\Omega}^{s+1/2-}(\mathbb{R}^n)$, $v \in H_{\Omega}^s(\mathbb{R}^n)$. There exists a partition of $[s_{\min}, s_{\max}]$ into sub-intervals S_k with interpolation orders $M_k \sim |\log \eta|$.

$$|a(u(s), v; s, \delta) - \tilde{a}(u(s), v; s, \delta)| \leq \eta \|u(s)\|_{H_{\Omega}^{\tilde{s}_2(s)}(\mathbb{R}^n)} \|v\|_{H_{\Omega}^s(\mathbb{R}^n)},$$

and the total number of interpolation nodes satisfies

$$\sum_{k=1}^K (M_k + 1) \leq C |\log \eta|,$$

and the constant C depends on δ and s_{\max} .

- Combined with hierarchical matrix approach: $\mathcal{O}(n \log^{2d+1} n)$ complexity & memory.
- Also allows to evaluate derivatives wrt s .
- Assembly for different values of δ is achieved by splitting the kernel into infinite horizon, singular part, and δ -dependent regular part.

¹³Olena Burkovska and Max Gunzburger. "Affine approximation of parametrized kernels and model order reduction for nonlocal and fractional Laplace models". In: *SIAM Journal on Numerical Analysis* 58.3 (2020), pp. 1469–1494.

¹⁴Olena Burkovska, Christian Glusa, and Marta D'Elia. "An optimization-based approach to parameter learning for fractional type nonlocal models". In: *Computers & Mathematics with Applications* (2021).

- Hierarchy of meshes from uniform or adaptive refinement
- Restriction / prolongation given by nesting of FE spaces
- Assembly into hierarchical matrix format on every level
- Smoothers:
 - Jacobi,
 - Chebyshev,
 - Gauss-Seidel when CSR matrix format is used.
- Coarse solve: convert to dense or CSR matrix

Numerical Examples in 2D – Timings

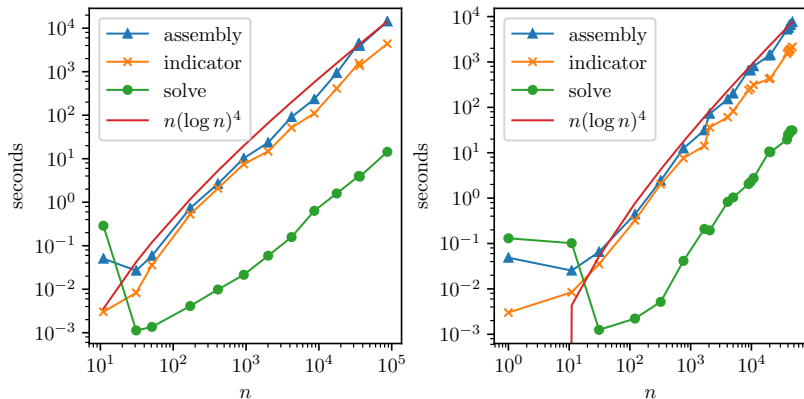
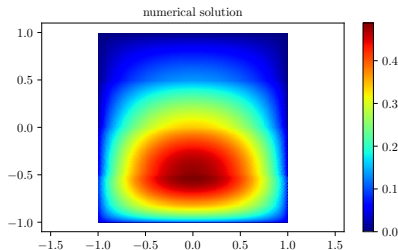


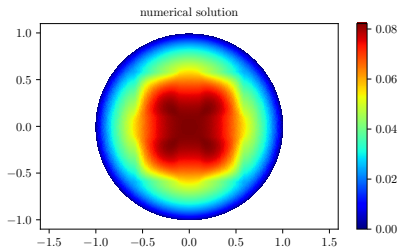
Figure: Timings for assembly of the stiffness matrix for fractional kernels, solution of linear system using GMG and computation of the error indicators for the two-dimensional problem. $s = 0.25$ on the left, $s = 0.75$ on the right.



$$f \equiv 1, \delta = 0.5$$

$$s(\vec{x}, \vec{y}) = \frac{1}{2}(\sigma(\vec{x}_1) + \sigma(\vec{y}_1))$$

$$\sigma(z) = \begin{cases} 1/5 & \text{if } z < -1/2, \\ 2/5 & \text{if } -1/2 \leq z < 0, \\ 3/5 & \text{if } 0 \leq z < 1/2, \\ 4/5 & \text{if } 1/2 \leq z. \end{cases}$$



$$f \equiv 1, \delta = \infty$$

$$s(\vec{x}, \vec{y}) = \begin{cases} 0.25 & \text{if } \vec{x}, \vec{y} \in \text{islands}, \\ 0.75 & \text{if } \vec{x}, \vec{y} \notin \text{islands}, \\ 0.75 & \text{else.} \end{cases}$$

¹⁵Marta D'Elia and Christian A. Glusa. A fractional model for anomalous diffusion with increased variability. Analysis, algorithms and applications to interface problems. (Accepted in Numerical Methods for Partial Differential Equations). 2021.

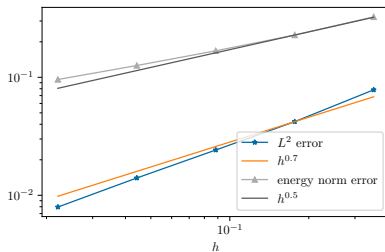
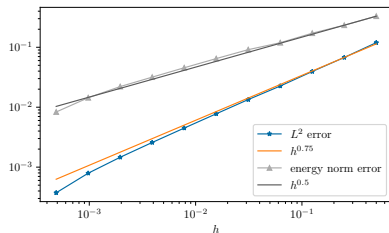


Figure: Convergence in L^2 and energy norm for a 1D example (*left*) and a 2D example with four material layers (*right*).

Rate of convergence, fractional kernels

	$\ e\ $	$\ e\ _{L^2}$
constant kernels (literature)	$h^{1/2-\varepsilon}$	$h^{\min\{1, 1/2+s\}-\varepsilon}$
variable kernels (observed)	$h^{1/2-\varepsilon}$	$h^{\min\{1, 1/2+\underline{s}\}-\varepsilon}$

$$\underline{s} = \min(\vec{x}, \vec{y})$$

\Rightarrow Possibly straightforward extension of regularity theory?

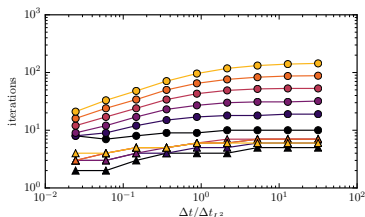
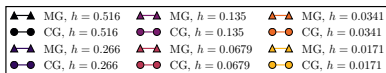
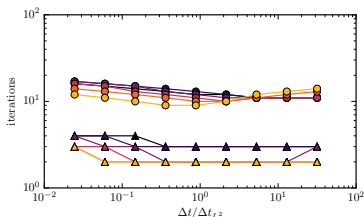
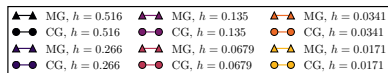


Figure: Fractional kernel. Number of iterations for CG and GMG depending on Δt for $s = 0.25$ (left) and $s = 0.75$ (right). Δt_{L^2} is the time-step that balances discretisation errors in time and space with respect to the L^2 -norm.

Conjugate gradient is a competitive solver when the fractional order s is small and the time step Δt is not too large.

Algebraic multigrid (WIP)

Motivation:

- Adaptively refined / graded meshes can make geometric multigrid painful.
- Use of established algebraic multigrid framework: Trilinos/MueLu
 - Lots of features (more smoothers, coarse solvers, multigrid cycles, etc)
 - Able to handle coefficient and mesh variations
 - Runs on lots of different computing architectures (CPU, threads, GPUs, etc)

Approach:

- Algebraic multigrid constructs coarse problems using sparsity patterns and matrix entries
→ Cannot directly use matrix \mathbf{A} when $\delta \gg h$ and hierarchical matrix format is used.
- Construct hierarchy for an auxiliary operator:
 - PDE operators, e.g. $(\nabla u, \nabla v)$,
 - (distance) Graph Laplacian wrt mesh,
 - near field part of hierarchical matrix, potentially with some filtering.
- Triple matrix products $\mathbf{A}_c = \mathbf{R}\mathbf{A}\mathbf{P}$ where \mathbf{R} and \mathbf{P} are sparse and \mathbf{A} an \mathcal{H} - or \mathcal{H}^2 -matrix
- Recompression of coarse matrix \mathbf{A}_c

#DoFs	# MPI ranks	dense	\mathcal{H}^2	CG #its	CG+AMG #its
11,193	4	0.93 GB	0.18 GB	114	13
45,169	16	15.2 GB	0.89 GB	200	15
181,473	64	245 GB	5.1 GB	350	20

Table: 2d fractional Poisson problem, $s = 0.75$, $\delta = \infty$, unsmoothed prolongators

Advertisement: PyNucleus, a FEM code for nonlocal problems

- Written in Python, lots of optimized kernels compiled to C via Cython.
- Compatible with NumPy/SciPy
- Simplicial meshes in 1D, 2D, (3D); uniform refinement with boundary snapping options
- Mesh (re)partitioning using (PAR)METIS
- Finite Element discretizations: discontinuous P_0 , continuous P_1, P_2, P_3
- Assembly of local differential operators
- Lots of solvers (direct, Krylov, simple preconditioners),
and in particular geometric multigrid
WIP: AMG (Trilinos/MueLu), DD (Htool&HPDDM)
- MPI distributed computations via mpi4py
- Assembly of the nonlocal operators in weak form:

$$a(u, v) = \frac{1}{2} \int_{(\Omega \cup \Omega_I)^2} (u(\vec{x}) - u(\vec{y}))(v(\vec{x}) - v(\vec{y})) \gamma(\vec{x}, \vec{y}) d\vec{y} d\vec{x}$$

into

- CSR sparse matrix ($\delta \sim h$),
- dense matrix ($\delta \gg h$),
- \mathcal{H}^2 hierarchical matrix ($\delta \gg h$; only tested for fractional kernels)
- For fractional kernels: quadrature orders are tuned for optimal convergence.
- Code: <https://github.com/sandialabs/PyNucleus>
- Documentation and examples: <https://sandialabs.github.io/PyNucleus>

Code example

```

1  from PyNucleus import (kernelFactory, nonlocalMeshFactory, dofmapFactory,
2                          functionFactory, HOMOGENEOUS_DIRICHLET, solverFactory)
3
4  # Infinite horizon fractional kernel
5  kernel = kernelFactory('fractional', dim=2, s=0.75, horizon=inf)
6
7  # Mesh for unit disc, no interaction domain for homogeneous Dirichlet
8  mesh, _ = nonlocalMeshFactory('disc', kernel=kernel,
9                                boundaryCondition=HOMOGENEOUS_DIRICHLET,
10                               hTarget=0.15)
11
12  dm = dofmapFactory('P1', mesh) # P1 finite elements
13  f = functionFactory('constant', 1.) # constant forcing
14  b = dm.assembleRHS(f) #  $\int_{\Omega} f \phi_i$ 
15  A = dm.assembleNonlocal(kernel, matrixFormat='h2') #  $a(\phi_i, \phi_j)$ , hierarchical
16  u = dm.zeros() # solution vector
17
18  # solve with diagonally preconditioned CG
19  solver = solverFactory('cg-jacobi', A=A, setup=True)
20  solver(b, u)
21  u.plot()

```

- The documentation contains two examples of how to setup and solve local and nonlocal problems with a lot more explanations.
- The repository contains several drivers that demonstrate some of the code capabilities.