



A fast direct solver for nonlocal operators in wavelet coordinates

Helmut Harbrecht^{a,1}, Michael Multerer^{b,*}

^a *Departement Mathematik und Informatik, Universität Basel, Spiegelgasse 1, 4051 Basel, Switzerland*

^b *Institute of Computational Science, Università della Svizzera italiana, Via Buffi 13, 6900 Lugano, Switzerland*



ARTICLE INFO

Article history:

Received 3 July 2020

Received in revised form 4 November 2020

Accepted 2 December 2020

Available online 16 December 2020

Keywords:

Nonlocal operator

Direct solver

Wavelet matrix compression

Polarizable continuum model

Fractional Laplacian

Gaussian random fields

ABSTRACT

In this article, we consider fast direct solvers for nonlocal operators. The pivotal idea is to combine a wavelet representation of the system matrix, yielding a quasi-sparse matrix, with the nested dissection ordering scheme. The latter drastically reduces the fill-in during the factorization of the system matrix by means of a Cholesky decomposition or an LU decomposition, respectively. This way, we end up with the exact inverse of the compressed system matrix with only a moderate increase of the number of nonzero entries in the matrix.

To illustrate the efficacy of the approach, we conduct numerical experiments for different highly relevant applications of nonlocal operators: We consider (i) the direct solution of boundary integral equations in three spatial dimensions, issuing from the polarizable continuum model, (ii) a parabolic problem for the fractional Laplacian in integral form and (iii) the fast simulation of Gaussian random fields.

© 2020 The Author(s). Published by Elsevier Inc. This is an open access article under the CC BY license (<http://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Various problems in science and engineering lead to nonlocal operators and corresponding operator equations. Examples arise from physical problems like field calculations and Riesz energy problems, from machine learning, and also from stochastic simulations and uncertainty quantification.

Traditional discretizations of nonlocal operators result in densely populated system matrices. This feature renders the computation very costly in both respects, the computation time and computer memory requirements. Therefore, over recent decades, different ideas for the data sparse approximation of nonlocal operators have been developed. Most prominent examples of such methods are the fast multipole method [19], the panel clustering [22], the wavelet matrix compression [1,9], and the hierarchical matrix format [21]. These techniques are able to represent nonlocal operators in linear or almost linear cost with respect to the number of degrees of freedom used for their discretization.

The present article relies on a compression of the system matrix by wavelets. Especially, the matrix representation of the nonlocal operator in wavelet coordinates is quasi-sparse, i.e. most matrix entries are negligible and can be treated as zero without compromising the overall accuracy. Discarding the non-relevant matrix entries is called matrix compression.

* Corresponding author.

E-mail addresses: helmut.harbrecht@unibas.ch (H. Harbrecht), michael.multerer@usi.ch (M. Multerer).

¹ This research was funded in part by the Swiss National Science Foundation through 407540_167186 NFP 75 Big Data.

Roughly speaking, nonlocal operators become local operators in wavelet coordinates. A fully discrete version of the wavelet matrix compression has been developed in [29]. It computes the compressed operator within discretization accuracy with linear cost.

Based on the sparsity pattern of the system matrix, which is solely determined by the order of the underlying operator, we employ a fill-in reducing reordering of the matrix entries by means of nested dissection, see [16,36]. This reordering in turn allows for the rapid inversion of the system matrix by the Cholesky decomposition or more generally by the LU decomposition. In particular, besides the rigorously controllable error for the matrix compression in the wavelet format and the roundoff errors in the computation of the matrix factorization, no additional approximation errors are introduced. This is a major difference to other approaches for the discretization and the arithmetics of nonlocal operators, e.g. by means of hierarchical matrices, \mathcal{H} -matrices for short. As the hierarchical matrix format is not closed under arithmetic operations, a recompression step after each arithmetic (block) operation has to be performed, which results into accumulating and hardly controllable consistency errors for matrix factorizations, see [20,21]. We remark that fill-in reducing strategies in the context of \mathcal{H} -matrices are well established. For example, the combination of \mathcal{H} -matrices and nested dissection is suggested for the \mathcal{H} -LU decomposition of finite element stiffness matrices in [18]. Variants of this approach can be found in [3,4,32,34,44,48]. In order to demonstrate the efficacy of the suggested approach, we consider applications from different fields. Namely, we consider (i) a boundary integral equation arising from the polarizable continuum model in quantum chemistry as a classical example for a nonlocal operator equation, (ii) a parabolic problem for the fractional Laplacian, and finally (iii) the fast numerical simulation of Gaussian random fields as an important example from computational uncertainty quantification.

One of the most widespread methods to include solvent effects in quantum chemistry is by making use of a continuum dielectric: the solvent is represented by a continuum which surrounds the molecule. Solute-solvent interactions are then described through appropriate functions supported on the molecule's surface. For an overview of continuum solvation models, we refer the reader to [46], and in particular for the polarizable continuum model to [6,7,38]. Wavelet matrix compression for the polarizable continuum model has been considered in [2,47]. Especially, in [23], the use of an incomplete Cholesky decomposition based preconditioner has been suggested, which is however inferior to the approach presented here.

The fractional Laplacian is an operator which generalizes the notion of spatial derivatives to fractional orders. It appears in image analysis, kinetic equations, phase transitions and nonlocal heat conduction, just to mention some applications. We refer to the review article [13] and the references therein for further details. In particular, we will focus here on the definition of the fractional Laplacian in its integral form, as it can be found in [12] and also in [13]. To the best of our knowledge, the numerical treatment of the parabolic problem for the fractional Laplacian by means of wavelets has not been addressed in literature yet.

The rapid simulation of (Gaussian) random fields with a prescribed covariance structure is of paramount importance in computational uncertainty quantification. The fast methods, which have been suggested so far are based on the computation of matrix square roots employing low-rank factorizations, block-wise low-rank factorizations, such as obtained by hierarchical matrices, or the discretization of the action of the matrix square root on a given vector by Krylov subspace methods, see [25,14]. Further approaches compute the Karhunen-Loève expansion by circulant embeddings and fast Fourier techniques, see [17,41], or employ the contour integral method, see [31]. Alternatively, samples from the random field can also be obtained by the stochastic partial differential equation approach, see [35]. In contrast to these approaches, we consider here the direct simulation of the random field by the Cholesky decomposition of the covariance matrix, which is very sparse in wavelet coordinates.

This article is organized as follows. Wavelet bases and their properties are specified in Section 2. Section 3 briefly repeats the main features of the fully discrete wavelet matrix compression scheme from [29]. Then, in Section 4, for the sake of completeness, the idea of nested dissection is briefly outlined. Section 5 presents the three different applications considered in this article, while Section 6 is devoted to related numerical experiments. Finally, Section 7 contains some concluding remarks.

In the following, in order to avoid the repeated use of generic but unspecified constants, we write $C \lesssim D$ to indicate that C can be bounded by a multiple of D , independently of parameters which C and D may depend on. Then, $C \gtrsim D$ is defined as $D \lesssim C$, while we write $C \sim D$, iff $C \lesssim D$ and $C \gtrsim D$.

2. Wavelets and multiresolution analysis

Let D denote a domain in \mathbb{R}^n or a manifold in \mathbb{R}^{n+1} . A *multiresolution analysis* consists of a nested family of finite dimensional approximation spaces

$$\{0\} = V_{-1} \subset V_0 \subset V_1 \subset \cdots \subset V_j \subset \cdots \subset L^2(D), \quad (1)$$

such that

$$\overline{\bigcup_{j \geq 0} V_j} = L^2(D) \quad \text{and} \quad \dim V_j \sim 2^{jn}.$$

We will refer to j as the *level* of V_j in the multiresolution analysis. Each space V_j is endowed with a *single-scale basis*

$$\Phi_j = \{\varphi_{j,\mathbf{k}} : \mathbf{k} \in \Delta_j\},$$

i.e. $V_j = \text{span } \Phi_j$, where Δ_j denotes a suitable index set with cardinality $|\Delta_j| \sim 2^{jn}$. For convenience, we shall in the sequel write bases on the form of row vectors, such that, for $\mathbf{v} = [v_{\mathbf{k}}]_{\mathbf{k} \in \Delta_j} \in \ell^2(\Delta_j)$, the corresponding function can simply be written as a dot product according to

$$v_j = \Phi_j \mathbf{v} = \sum_{\mathbf{k} \in \Delta_j} v_{\mathbf{k}} \varphi_{j,\mathbf{k}}.$$

In addition, we shall assume that the single-scale bases Φ_j are *uniformly stable*, i.e.

$$\|\mathbf{v}\|_{\ell^2(\Delta_j)} \sim \|\Phi_j \mathbf{v}\|_{L^2(D)} \quad \text{for all } \mathbf{v} \in \ell^2(\Delta_j)$$

uniformly in j , and that they satisfy the locality condition

$$\text{diam}(\text{supp } \varphi_{j,\mathbf{k}}) \sim 2^{-j}.$$

Additional properties of the spaces V_j are required for using them as trial spaces in a Galerkin scheme. The approximation spaces need to have the *regularity*

$$\gamma := \sup\{s \in \mathbb{R} : V_j \subset H^s(D)\}$$

and the *approximation order* $d \in \mathbb{N}$, that is

$$d = \sup \left\{ s \in \mathbb{R} : \inf_{v_j \in V_j} \|v - v_j\|_{L^2(D)} \lesssim 2^{-js} \|v\|_{H^s(D)} \right\}.$$

Rather than using the multiresolution analysis corresponding to the hierarchy in (1), the pivotal idea of wavelets is to keep track of the increment of information between two consecutive levels $j - 1$ and j . Since we have $V_{j-1} \subset V_j$, we may decompose

$$V_j = V_{j-1} \oplus W_j, \text{ i.e. } V_{j-1} \cup W_j = V_j \text{ and } V_{j-1} \cap W_j = \{0\},$$

with an appropriate *detail space* W_j . Of practical interest is the particular choice of the basis of the detail space W_j in V_j . This basis will be denoted by

$$\Psi_j = \{\psi_{j,\mathbf{k}} : \mathbf{k} \in \nabla_j := \Delta_j \setminus \Delta_{j-1}\}.$$

In particular, we shall assume that the collections $\Phi_{j-1} \cup \Psi_j$ form uniformly stable bases of V_j , as well. If $\Psi = \bigcup_{j \geq 0} \Psi_j$, where $\Psi_0 := \Phi_0$, is even a Riesz-basis of $L_2(D)$, then it is called a *wavelet basis*. We require the functions $\psi_{j,\mathbf{k}}$ to be localized with respect to the corresponding level j , i.e.

$$\text{diam}(\text{supp } \psi_{j,\mathbf{k}}) \sim 2^{-j},$$

and we normalize them such that

$$\|\psi_{j,\mathbf{k}}\|_{L_2(D)} \sim 1.$$

At first glance it would be very convenient to deal with a single orthonormal system of wavelets. However, it has been shown in [9,10,43] that orthogonal wavelets are not optimal for the efficient approximation nonlocal operator equations. For this reason, we rather use *biorthogonal wavelet bases*. In this case, we also have a dual, multiresolution analysis, i.e. dual single-scale bases and wavelets

$$\tilde{\Phi}_j = \{\tilde{\varphi}_{j,\mathbf{k}} : \mathbf{k} \in \Delta_j\}, \quad \tilde{\Psi}_j = \{\tilde{\psi}_{j,\mathbf{k}} : \mathbf{k} \in \nabla_j\},$$

which are coupled to the primal ones by the orthogonality condition

$$(\Phi_j, \tilde{\Phi}_j)_{L^2(D)} = \mathbf{I}, \quad (\Psi_j, \tilde{\Psi}_j)_{L^2(D)} = \mathbf{I}.$$

The corresponding spaces $\tilde{V}_j := \text{span } \tilde{\Phi}_j$ and $\tilde{W}_j := \text{span } \tilde{\Psi}_j$ satisfy

$$V_{j-1} \perp \tilde{W}_j, \quad \tilde{V}_{j-1} \perp W_j. \tag{2}$$

Moreover, the dual spaces are supposed to exhibit some approximation order $\tilde{d} \in \mathbb{N}$ and regularity $\tilde{\gamma} > 0$.

Denoting in complete analogy to the primal basis $\tilde{\Psi} = \bigcup_{j \geq 0} \tilde{\Psi}_j$, where $\tilde{\Psi}_0 := \tilde{\Phi}_0$, every $v \in L^2(D)$ has unique representations

$$v = \tilde{\Psi}(v, \Psi)_{L^2(D)} = \Psi(v, \tilde{\Psi})_{L^2(D)}$$

such that

$$\|v\|_{L^2(D)}^2 \sim \sum_{j \geq 0} \sum_{k \in \nabla_j} \|(v, \tilde{\psi}_{j,k})_{L^2(D)}\|_{\ell^2(\nabla_j)}^2 \sim \sum_{j \geq 0} \sum_{k \in \nabla_j} \|(v, \psi_{j,k})_{L^2(D)}\|_{\ell^2(\nabla_j)}^2.$$

In particular, relation (2) implies that the wavelets exhibit *vanishing moments* of order \tilde{d} , i.e.

$$|(v, \psi_{j,k})_{L^2(D)}| \lesssim 2^{-j(n/2+\tilde{d})} |v|_{W^{\tilde{d},\infty}(\text{supp } \psi_{j,k})}. \tag{3}$$

Herein, the quantity $|v|_{W^{\tilde{d},\infty}(D)} := \sup_{|\alpha|=\tilde{d}} \|\partial^\alpha v\|_{L^\infty(D)}$ is the semi-norm in $W^{\tilde{d},\infty}(D)$. We refer to [8] for further details.

Piecewise constant and bilinear wavelets which provide the above properties have been constructed in [28,30]. In what follows, we will refer to the wavelet basis of V_J by $\Psi_J = \{\psi_\lambda : \lambda \in \nabla_J\}$, where the multi-index $\lambda = (j, \mathbf{k})$ incorporates the scale $j = |\lambda|$ and the spatial location $\mathbf{k} = \mathbf{k}(\lambda)$.

3. Wavelet matrix compression

For a given domain or manifold D and $q \in \mathbb{R}$, let

$$\mathcal{A} : H^q(D) \rightarrow H^{-q}(D)$$

denote a given (continuous and bijective) nonlocal operator of order $2q$. According to the Schwartz kernel theorem, it can be represented in accordance with

$$(\mathcal{A}u)(\mathbf{x}) = \int_D k(\mathbf{x}, \mathbf{y})u(\mathbf{y}) \, d\mathbf{y}, \quad \mathbf{x} \in D, \tag{4}$$

for a suitable kernel function $k : D \times D \rightarrow \mathbb{R}$. The kernel functions under consideration are supposed to be smooth as functions in the variables \mathbf{x} and \mathbf{y} , apart from the diagonal $\{(\mathbf{x}, \mathbf{y}) \in D \times D : \mathbf{x} = \mathbf{y}\}$, and may exhibit a singularity on the diagonal. Such kernel functions arise, for instance, from applying a boundary integral formulation to a second order elliptic problem [42,45]. Typically, they decay like a negative power of the distance of the arguments which depends on the order $2q$ of the operator. More precisely, there holds

$$|\partial_{\mathbf{x}}^\alpha \partial_{\mathbf{y}}^\beta k(\mathbf{x}, \mathbf{y})| \leq c_{\alpha,\beta} \|\mathbf{x} - \mathbf{y}\|^{-n-2q-|\alpha|-|\beta|}. \tag{5}$$

We emphasize that this estimate remains valid for the kernels of arbitrary pseudodifferential operators, see [11] for the details.

Corresponding to the nonlocal operator from (4), we may consider the operator equation

$$\mathcal{A}u = f$$

which gives rise to the Galerkin approach:

find $u_J \in V_J$ such that

$$(\mathcal{A}u_J, v_J)_{L^2(D)} = (f, v_J)_{L^2(D)} \quad \text{for all } v_J \in V_J.$$

Traditionally, this equation is discretized employing the single-scale basis of V_J which results in densely populated system matrices. If $N_J \sim 2^{Jn}$ denotes the number of basis functions in the space V_J , then the system matrix contains $\mathcal{O}(N_J^2)$ nonzero matrix entries. In contrast, by utilizing a wavelet basis in the Galerkin discretization, we end up with a matrix that is quasi-sparse, i.e. it is compressible to $\mathcal{O}(N_J)$ nonzero matrix entries without compromising the overall accuracy. More precisely, by combining (3) and (5), we arrive at the decay estimate

$$(\mathcal{A}\psi_{\lambda'}, \psi_\lambda)_{L^2(D)} \lesssim \frac{2^{-(|\lambda|+|\lambda'|)(\tilde{d}+n/2)}}{\text{dist}(D_\lambda, D_{\lambda'})^{n+2q+2\tilde{d}}}, \tag{6}$$

which is the foundation of the compression estimates in [9]. Herein, $D_\lambda := \text{supp } \psi_\lambda$ and $D_{\lambda'} := \text{supp } \psi_{\lambda'}$ denote the convex hulls of the supports of the wavelets ψ_λ and $\psi_{\lambda'}$.

Based on (6), we shall neglect all matrix entries for which the distance of the supports between the associated ansatz and test wavelets is larger than a level dependent cut-off parameter $\mathcal{B}_{j,j'}$. An additional compression, reflected by a cut-off parameter $\mathcal{B}_{j,j'}^s$, is achieved by neglecting several of those matrix entries, for which the corresponding trial and test functions have overlapping supports.

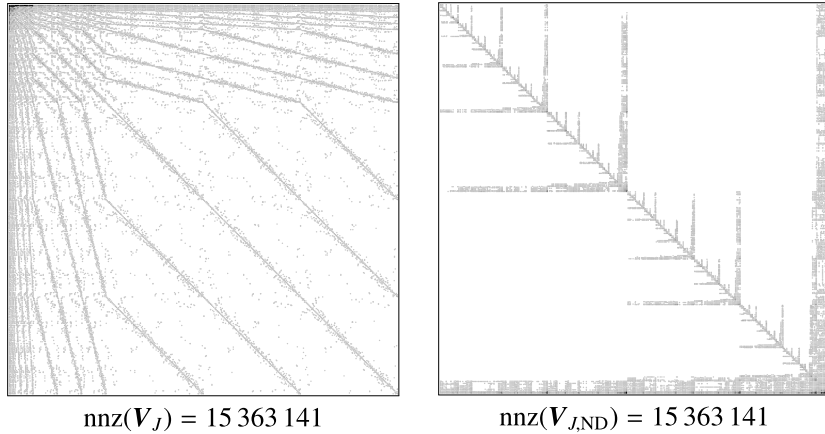


Fig. 1. Sparsity patterns of \mathbf{V} (left) and its nested dissection reordering $\mathbf{V}_{J,ND}$ (right) for the single layer operator on the benzene geometry and $N_J = 93184$. Each dot corresponds to a submatrix of size 64×64 . Lighter blocks have less entries than darker blocks.

To formulate this result, we introduce the abbreviation $D_\lambda^s := \text{sing supp } \psi_\lambda$ which denotes the *singular support* of the wavelet ψ_λ , i.e. that subset of D where the wavelet is non-smooth.

Theorem 1 (A-priori compression [9]). Let D_λ and D_λ^s be given as above and define the compressed system matrix \mathbf{A}_J , corresponding to the boundary integral operator \mathcal{A} , by

$$[\mathbf{A}_J]_{\lambda,\lambda'} := \begin{cases} 0, & \text{dist}(D_\lambda, D_{\lambda'}) > \mathcal{B}_{|\lambda|,|\lambda'|} \text{ and } |\lambda|, |\lambda'| > 0, \\ 0, & \text{dist}(D_\lambda, D_{\lambda'}) \leq 2^{-\min\{|\lambda|,|\lambda'|\}} \text{ and} \\ & \text{dist}(D_\lambda^s, D_{\lambda'}^s) > \mathcal{B}_{|\lambda|,|\lambda'|}^s \text{ if } |\lambda'| > |\lambda| \geq 0, \\ & \text{dist}(D_\lambda, D_{\lambda'}^s) > \mathcal{B}_{|\lambda|,|\lambda'|}^s \text{ if } |\lambda| > |\lambda'| \geq 0, \\ (\mathcal{A}\psi_{\lambda'}, \psi_\lambda)_{L^2(D)}, & \text{otherwise.} \end{cases} \tag{7}$$

Fixing

$$a > 1, \quad d < \delta < \tilde{d} + 2q, \tag{8}$$

the cut-off parameters $\mathcal{B}_{j,j'}$ and $\mathcal{B}_{j,j'}^s$ are set according to

$$\mathcal{B}_{j,j'} := a \max \left\{ 2^{-\min\{j,j'\}}, 2^{\frac{2J(\delta-q)-(j+j')(\delta+\tilde{d})}{2(d+q)}} \right\}, \tag{9}$$

$$\mathcal{B}_{j,j'}^s := a \max \left\{ 2^{-\max\{j,j'\}}, 2^{\frac{2J(\delta-q)-(j+j')\delta-\max\{j,j'\}\tilde{d}}{d+2q}} \right\}.$$

Then, the system matrix \mathbf{A}_J only has $O(N_J)$ nonzero entries. In addition, the error estimate

$$\|u - u_J\|_{H^{2q-d}(D)} \lesssim 2^{-2J(d-q)} \|u\|_{H^d(D)} \tag{10}$$

holds for the solution u_J of the compressed Galerkin system provided that u and D are sufficiently regular.

The compressed system matrix can be assembled with linear cost if the exponentially convergent hp -quadrature method proposed in [29] is employed for the computation of matrix entries. Moreover, for performing faster matrix-vector multiplications, an additional a-posteriori compression might be applied which reduces again the number of nonzero entries by a factor 2–5, see [9]. The pattern of the compressed system matrix shows the typical *finger structure*, see the left hand side of Fig. 1.

4. Nested dissection

The representation of the system matrix corresponding to a nonlocal operator with respect to an appropriate wavelet basis leads to a quasi-sparse matrix, i.e. a matrix with many small entries which can be neglected without compromising accuracy. Performing a thresholding procedure as discussed in the previous section then yields a sparse system matrix whose symmetric sparsity pattern is solely determined by the order of the underlying operator, see the left hand side of Fig. 1.

The factorization of the system matrix represented in the canonical levelwise ordering leads to a massive fill-in. This means that a huge amount of nonzero entries is generated by a Cholesky decomposition or an LU decomposition, typically resulting in dense matrix factors. In order to obtain much sparser factorizations, we employ a nested dissection ordering, cf. [16,36], see the right hand side of Fig. 1.

Nested dissection is a divide and conquer algorithm whose foundation is a graph theoretical observation. To each matrix $\mathbf{A} \in \mathbb{R}^{N \times N}$ with a symmetric sparsity pattern, we may assign an undirected graph $G = (V, E)$ with vertices $V = \{1, 2, \dots, N\}$ and edges $E = \{\{i, j\} : a_{i,j} \neq 0\}$. Then, a symmetric permutation \mathbf{PAP}^T of the rows and columns of \mathbf{A} amounts to a permutation $\pi(V)$ of the nodes in V . In particular, we have the following important result from [40], compare also [39], which we formulate here only for the Cholesky decomposition $\mathbf{PAP}^T = \mathbf{LL}^T$.

Lemma 1 ([40]). *Assuming that no cancellation of nonzero entries in the Cholesky decomposition of \mathbf{PAP}^T takes place, then $\ell_{i,j} \neq 0$ for $i > j$, iff there is a path $i = v_1, v_2, \dots, v_{k+1} = j$ in G such that $\pi(v_t) < \min\{\pi(i), \pi(j)\}$ for $2 \leq t \leq k$.*

The lemma states that the Cholesky decomposition connects all nodes i and j , resulting in a nonzero entry $\ell_{i,j}$, for which there exists a path of nodes that have been eliminated before i and j .

Finding an optimal ordering is a hard problem in general. Therefore, we resort to the following strategy, which is known as nested dissection ordering: We split $V = V_1 \cup V_2 \cup S$ such that $E \cap \{\{v_1, v_2\} : v_1 \in V_1, v_2 \in V_2\} = \emptyset$, i.e. the removal of the vertices of the separator S and its adjacent edges results into two disjoint subgraphs. Hence, employing an ordering which puts first the nodes into V_1 and V_2 and afterwards the nodes in S , leads to a matrix structure of the form

$$\mathbf{PAP}^T = \begin{bmatrix} \mathbf{A}_{V_1, V_1} & & \mathbf{A}_{V_1, S} \\ & \mathbf{A}_{V_2, V_2} & \mathbf{A}_{V_2, S} \\ \mathbf{A}_{S, V_1} & \mathbf{A}_{S, V_2} & \mathbf{A}_{S, S} \end{bmatrix}.$$

Recursively applying this procedure then yields a structure similar to the one on the right hand side of Fig. 1. For obvious reasons, it is desirable to have a minimal separator S , which evenly splits V into two subsets. We refer to [36] and the references therein for a comprehensive discussion of this topic. In order to obtain suitable separators for the computations in this article, we will adopt the strategy from [33], which performs very well in terms of reducing the fill-in.

5. Applications

5.1. Polarizable continuum model

Continuum solvation models are widely used to model quantum effects of molecules in liquid solutions, compare [46] for an overview. In the *polarizable continuum model* (PCM), introduced in [38], the molecule under study (the solute) is located inside a cavity D , surrounded by a homogeneous dielectric (the solvent) with dielectric constant $\epsilon \geq 1$. The solute-solvent interactions between the charge distributions which compose the solute and the dielectric are reduced to those of electrostatic origin.

For a given charge $\rho \in H^{-1}(D)$, located inside the cavity, the solute-solvent interaction is expressed by the *apparent surface charge* $\sigma \in H^{-1/2}(\partial D)$. It is given by the integral equation

$$\mathcal{V}\sigma = \left(\frac{1+\epsilon}{2} + (1-\epsilon)\mathcal{K} \right)^{-1} \mathcal{N}_\rho - \mathcal{N}_\rho \quad \text{on } \partial D, \tag{11}$$

where \mathcal{V} is the *single layer operator*

$$(\mathcal{V}u)(\mathbf{x}) = \int_{\partial D} \frac{u(\mathbf{y})}{4\pi \|\mathbf{x} - \mathbf{y}\|^3} d\sigma_{\mathbf{y}},$$

\mathcal{K} is the *double layer operator*

$$(\mathcal{K}u)(\mathbf{x}) = \int_{\partial D} u(\mathbf{y}) \frac{\langle \mathbf{n}(\mathbf{y}), \mathbf{x} - \mathbf{y} \rangle}{4\pi \|\mathbf{x} - \mathbf{y}\|^3} d\sigma_{\mathbf{y}},$$

and \mathcal{N}_ρ denotes the *Newton potential* of the given charge

$$\mathcal{N}_\rho(\mathbf{x}) := \int_{\partial D} \frac{\rho(\mathbf{y})}{4\pi \|\mathbf{x} - \mathbf{y}\|} d\mathbf{y}.$$

The discretization of the boundary integral equation (11) by means of a Galerkin scheme is performed as follows, compare [26,27]: We make the ansatz

$$\sigma_J = \sum_{\lambda} \sigma_{\lambda} \psi_{\lambda}$$

and introduce the mass matrix

$$\mathbf{G}_J = [(\psi_{\lambda'}, \psi_{\lambda})_{L^2(D)}]_{\lambda, \lambda'}$$

and the system matrices

$$\mathbf{V}_J = [(\mathcal{V}\psi_{\lambda'}, \psi_{\lambda})_{L^2(D)}]_{\lambda, \lambda'}, \quad \mathbf{K}_J = [(\mathcal{K}\psi_{\lambda'}, \psi_{\lambda})_{L^2(D)}]_{\lambda, \lambda'}$$

Then, for a given data vector $\mathbf{f}_J = [(\mathcal{N}_{\rho}, \psi_{\lambda})_{L^2(\partial D)}]_{\lambda}$, we need to solve the linear system of equations

$$\mathbf{V}_J \boldsymbol{\sigma}_J = \mathbf{G}_J \left(\frac{1+\epsilon}{2} \mathbf{G}_J + (1-\epsilon) \mathbf{K}_J \right)^{-1} \mathbf{f}_J - \mathbf{f}_J \tag{12}$$

in order to determine the sought apparent surface charge.

In quantum chemical simulations, for example when solving the Hartree-Fock equations in a self consistent field approximation, one has to compute the interaction energies between the different particles. This amounts to the determination of different apparent surface charges. Therefore, the fast solution of (12) for multiple right hand sides is indispensable for fast simulations in quantum chemistry.

5.2. Parabolic diffusion problem for the fractional Laplacian

For a given domain $D \subset \mathbb{R}^n$ and $0 < s < 1/2$, the fractional Laplacian $\mathcal{L}_s : H^s(D)/\mathbb{R} \rightarrow H^{-s}(D)/\mathbb{R}$ is given by

$$(\mathcal{L}_s u)(\mathbf{x}) := 2 \int_D \frac{u(\mathbf{y}) - u(\mathbf{x})}{\|\mathbf{x} - \mathbf{y}\|^{n+2s}} d\mathbf{y}, \quad \mathbf{x} \in D,$$

compare [12,13]. We intent to solve the following parabolic diffusion problem

$$\partial_t u - \mathcal{L}_s u = f \quad \text{in } D$$

for the fractional Laplacian. To this end, we employ the θ -scheme in time and a Galerkin discretization of the problem in space. This leads to the linear system of equations

$$\mathbf{G}_J \frac{\mathbf{u}_J(t_{i+1}) - \mathbf{u}_J(t_i)}{t_{i+1} - t_i} - \mathbf{L}_J \{ (1-\theta) \mathbf{u}_J(t_i) + \theta \mathbf{u}_J(t_{i+1}) \} = (1-\theta) \mathbf{f}_J(t_i) + \theta \mathbf{f}_J(t_{i+1}). \tag{13}$$

Here,

$$\mathbf{L}_J = [(\mathcal{L}_s \psi_{\lambda'}, \psi_{\lambda})_{L^2(D)}]_{\lambda, \lambda'}, \quad \mathbf{G}_J = [(\psi_{\lambda'}, \psi_{\lambda})_{L^2(D)}]_{\lambda, \lambda'}$$

are the system matrix of the fractional Laplacian and the mass matrix, respectively. In each time step, we have hence to invert the matrix $\mathbf{A}_J = \mathbf{G}_J - \theta(t_{i+1} - t_i) \mathbf{L}_J$ for computing the new solution $\mathbf{u}_J(t_{i+1})$ from the solution $\mathbf{u}_J(t_i)$ of the previous time step t_i . A factorization of the system matrix \mathbf{A}_J is favourable in this situation since the system matrix does not change with respect to time.

5.3. Gaussian random fields

Let $(\Omega, \mathcal{F}, \mathbb{P})$ denote a complete and separable probability space. We consider a Gaussian random field

$$a : D \times \Omega \rightarrow \mathbb{R}$$

with expectation

$$\mathbb{E}[a](\mathbf{x}) := \int_{\Omega} a(\mathbf{x}, \omega) d\mathbb{P}(\omega)$$

and covariance

$$\text{Cov}[a](\mathbf{x}, \mathbf{x}') := \int_{\Omega} (a(\mathbf{x}, \omega) - \mathbb{E}[a](\mathbf{x})) (a(\mathbf{x}', \omega) - \mathbb{E}[a](\mathbf{x}')) d\mathbb{P}(\omega).$$

If the expectation and the covariance are known, we may represent the field a by its *Karhunen-Loève expansion*

$$a(\mathbf{x}, \omega) = \mathbb{E}[a](\mathbf{x}) + \sum_{k=1}^{\infty} \sqrt{\mu_k} a_k(\mathbf{x}) Y_k(\omega).$$

Herein, (μ_k, a_k) , $k = 1, 2, \dots$, denote the eigen pairs of the Hilbert-Schmidt operator

$$(Cv)(\mathbf{x}) := \int_D \text{Cov}[a](\mathbf{x}, \mathbf{x}') v(\mathbf{x}') d\mathbf{x}',$$

while Y_1, Y_2, \dots are independent and standard normally distributed random variables.

In order to discretize the Karhunen-Loève expansion, we proceed in complete analogy to [25] and compute the orthogonal projection of C onto V_J . Let $\mathbf{C}_J \in \mathbb{R}^{N_J \times N_J}$ denote the corresponding coefficient matrix. Obviously, a suitable basis for the orthogonal projection is given by the orthogonalized wavelet basis $\Psi_J \mathbf{G}_J^{-1/2}$, where

$$\mathbf{G}_J = [(\psi_{\lambda'}, \psi_{\lambda})_{L^2(D)}]_{\lambda, \lambda'}$$

denotes the mass matrix and $\mathbf{G}_J^{-1/2}$ is an inverse matrix root. Thus, in accordance with [25], we arrive at the discretized random field

$$a_J(\mathbf{x}, \omega) = \Psi_J(\mathbf{x}) \mathbf{G}_J^{-1/2} (\bar{\mathbf{a}}_J + \mathbf{V} \Sigma \mathbf{Y}(\omega)), \tag{14}$$

where $\mathbf{Y} := [Y_1, Y_2, \dots, Y_{N_J}]^T$ is a standard normally distributed Gaussian vector, $\bar{\mathbf{a}}_J$ is the orthogonal projection of $\mathbb{E}[a]$ onto V_J and $\mathbf{V} \Sigma \mathbf{V}^T = \mathbf{C}_J$ is the spectral decomposition of \mathbf{C}_J .

Hence, for the Galerkin projection, we find the identities

$$\begin{aligned} \bar{\mathbf{a}}_J^G &:= [(\mathbb{E}[a], \psi_{\lambda})_{L^2(D)}]_{\lambda} = \mathbf{G}_J^{1/2} \bar{\mathbf{a}}_J, \\ \mathbf{C}_J^G &:= [(C\psi_{\lambda'}, \psi_{\lambda})_{L^2(D)}]_{\lambda, \lambda'} = \mathbf{G}_J^{1/2} \mathbf{C}_J \mathbf{G}_J^{1/2}. \end{aligned}$$

Thus, we obtain for the expectation

$$\bar{\mathbf{a}}_J = \mathbf{G}_J^{-1/2} \bar{\mathbf{a}}_J^G,$$

while the covariance satisfies

$$\mathbf{C}_J = (\mathbf{V} \Sigma)(\mathbf{V} \Sigma)^T = \mathbf{G}_J^{-1/2} \mathbf{C}_J^G \mathbf{G}_J^{-1/2} = (\mathbf{G}_J^{-1/2} \mathbf{L})(\mathbf{G}_J^{-1/2} \mathbf{L})^T,$$

where $\mathbf{L} \mathbf{L}^T = \mathbf{C}_J^G$ is the Cholesky decomposition of \mathbf{C}_J^G .

It is well known that any two matrix square roots $\mathbf{R} \mathbf{R}^T = \widehat{\mathbf{R}} \widehat{\mathbf{R}}^T = \mathbf{C}_J$ only differ by an isometry. Consequently, there exists an orthogonal matrix \mathbf{Q} such that $\mathbf{V} \Sigma = \mathbf{G}_J^{-1/2} \mathbf{L} \mathbf{Q}$. Introducing the transformed random vector $\mathbf{X}(\omega) := \mathbf{Q} \mathbf{Y}(\omega)$, we infer

$$\int_{\Omega} \mathbf{X}(\omega) \mathbf{X}^T(\omega) d\mathbb{P}(\omega) = \mathbf{Q} \int_{\Omega} \mathbf{X}(\omega) \mathbf{X}^T(\omega) d\mathbb{P}(\omega) \mathbf{Q}^T = \mathbf{Q} \mathbf{I} \mathbf{Q}^T = \mathbf{I}.$$

Therefore, $\mathbf{X}(\omega)$ is a standard normally distributed Gaussian vector as well and we end up with the representation

$$a_J(\mathbf{x}, \omega) = \Psi_J(\mathbf{x}) \mathbf{G}_J^{-1} (\bar{\mathbf{a}}_J^G + \mathbf{L} \mathbf{X}(\omega)). \tag{15}$$

If the singular values in the Karhunen-Loève expansion (14) decay only slowly, as it is typically the case for non-smooth covariance functions, the numerical solution of the associated eigenvalue problem becomes prohibitive. In such cases, the direct simulation of the random field by means of a sparse Cholesky decomposition in (15) is computationally superior. Therefore, the proposed method is in particular useful for rough random fields, which issue from non-smooth covariance functions.

We remark that in case of a smooth covariance function, the matrix \mathbf{C}_J^G becomes numerically positive semidefinite. In such cases, the pivoted Cholesky decomposition, see [24], immediately yields a low-rank decomposition of \mathbf{C}_J^G and is computationally superior to other data sparse representations, we refer to [25] for a comprehensive discussion of this complementary approach.

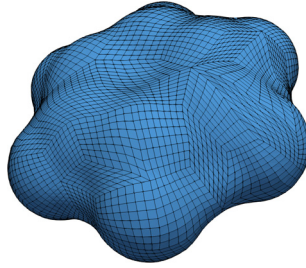


Fig. 2. Solvent excluding surface of benzene.

Table 1
Computation times and numbers of nonzero entries in case of PCM.

N_J	t_{WEM}	t_{ND}	$t_{\text{Chol}}(\mathbf{V}_J)$	$t_{\text{LU}}(\mathbf{A}_J)$	$\text{anz}(\mathbf{V}_J)$	$\text{anz}(\mathbf{L}_J)$
1456	0.96	0.032	0.19	0.22	134	209
5824	7.10	0.15	0.19	1.59	163	328
23296	52.32	0.74	1.05	14.03	159	419
93184	309.35	3.91	7.66	189.70	165	639
372736	2265.91	19.38	58.28	2211.57	173	890

6. Numerical results

In order to obtain consistent computation times, all computations reported in this section have been carried out on a single core of a compute server with Intel Xeon E5-2650 v3 @2.30 GHz CPUs and 512 GB DDR4 @2133 MHz main memory. The implementation of the wavelet matrix compression has been performed in ANSI C, while nested dissection orderings, Cholesky decompositions and LU decompositions have been computed using MATLAB 2020a, compare [37]. In particular, for the nested dissection ordering, we employ the function `dissect`.

6.1. Polarizable continuum model

For PCM, we consider the assembly and the factorization of the matrices

$$\mathbf{V}_J \quad \text{and} \quad \mathbf{A}_J := \frac{1+\epsilon}{2}\mathbf{G}_J + (1-\epsilon)\mathbf{K}_J,$$

which are required to set up the linear system of equations (12). The dielectric constant is chosen as $\epsilon = 78.39$, corresponding to the solvent water. The molecule under study is benzene, whose solvent excluding surface is depicted in Fig. 2. For the wavelet matrix compression, we use piecewise constant wavelets with three vanishing moments, as developed in [28].

Table 1 shows the numerical results. The first column, labelled by N_J , corresponds to the number of surface elements. The second column, labelled by t_{WEM} , contains the combined computation times in seconds for the assembly of \mathbf{V}_J and \mathbf{A}_J by the wavelet matrix compression. The third column, labelled by t_{ND} , provides the computation times in seconds for the nested dissection ordering, as \mathbf{V}_J and \mathbf{A}_J have identical sparsity patterns, the same reordering can be applied to both of them. The fourth column, labelled by $t_{\text{Chol}}(\mathbf{V}_J)$, denotes the computation times in seconds for the Cholesky factorization of \mathbf{V}_J , while the fifth column shows the computation times in seconds of the LU decomposition of \mathbf{A}_J . For the purpose of measuring the fill-in relative to the matrix size, we introduce for a sparse matrix $\mathbf{A} \in \mathbb{R}^{N \times N}$ the *average number of nonzeros per row*

$$\text{anz}(\mathbf{A}) := \frac{\text{nnz}(\mathbf{A})}{N}.$$

The values for $\text{anz}(\mathbf{V}_J)$ and $\text{anz}(\mathbf{A}_J)$ are identical and can be found in the sixth row of Table 1, while the values $\text{anz}(\mathbf{L}_J)$ for the Cholesky factor of $\text{anz}(\mathbf{V}_J)$ are given in the last column of the table. The sparsity pattern of the L-factor of the LU decomposition of \mathbf{A}_J coincides with that of \mathbf{L}_J , while the U-factor has somewhat less coefficients compared to the L-factor. Hence, the average number of nonzeros per row for the LU factorization of \mathbf{A}_J matches that of $\text{anz}(\mathbf{L}_J)$.

The sparsity patterns for \mathbf{V}_J , \mathbf{L}_J and for the U-factor are shown in Fig. 3 for $N_J = 93184$. In order to obtain a neat representation in this figure, we have merged matrix blocks of size 64×64 . Darker blocks have a higher density of entries, while lower blocks have a lower density of entries.

As can clearly be inferred from Table 1, the times for the computing the nested dissection ordering and the subsequent Cholesky decomposition are negligible with respect to the wavelet matrix compression. The average number of nonzero entries per row stays rather low and only increases for increasing numbers of unknowns up to a factor of approximately 5 for $N_J = 372736$.

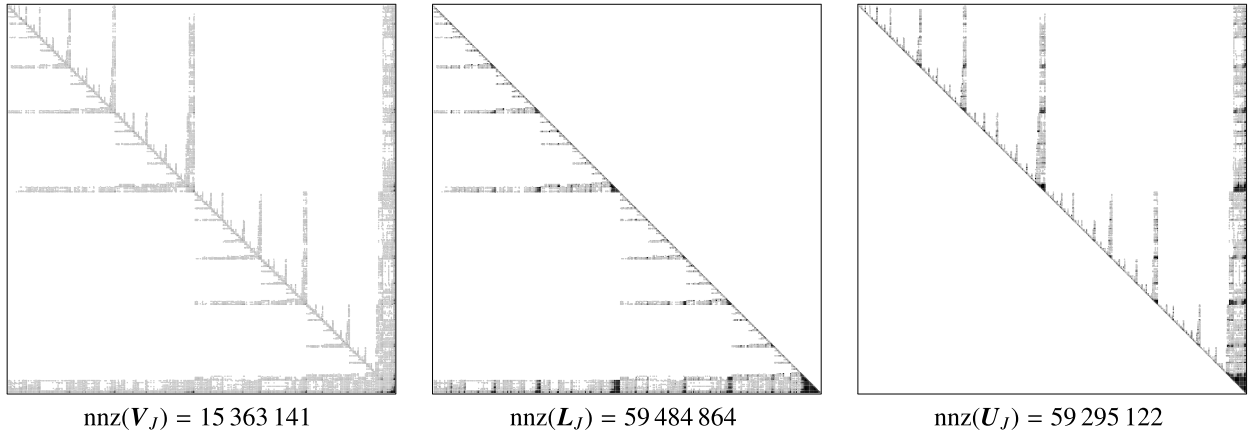


Fig. 3. Sparsity patterns of V_J as well as A_J (left), the Cholesky factor L_J (middle), and the upper triangular matrix U_J (right) for PCM and $N_J = 93\,184$. Each dot corresponds to a submatrix of size 64×64 . Lighter blocks have less entries than darker blocks.

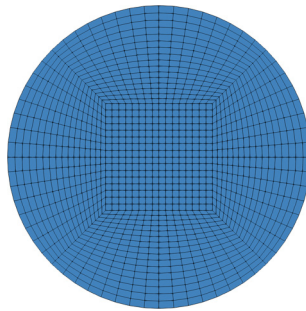


Fig. 4. Computational geometry for the fractional Laplacian.

It can be seen from the fifth column of Table 1 that the LU decomposition is significantly slower than the Cholesky decomposition. This issues from the fact that all matrices are stored in a sparse column major format, resulting in a large overhead for the access of matrix rows. We remark that, in principle, the LU decomposition could be accelerated by an appropriate data structure, which enables direct access to the rows of the matrix as well.

6.2. Parabolic diffusion problem for the fractional Laplacian

We consider the parabolic problem for the fractional Laplacian with $s = 3/8$. The right hand side is a Gaussian heat spot moving on a circular trajectory, given by

$$f(\mathbf{x}, t) = 100 \exp\left(-40(x_1 - \cos(2\pi t))^2 - 40(x_2 - \sin(2\pi t))^2\right),$$

while the initial condition is set to 0. For the solution of the ordinary differential equation in time, we employ the θ -scheme with $\theta = 1/2$, which yields the Crank-Nicolson method [5]. The time interval is given by $[0, T] = [0, 3]$ and we discretize this time interval by 150 equidistant time steps.

In the θ -scheme (13), we need to invert the matrix

$$A_J = G_J + \theta(t_{i+1} - t_i)L_J,$$

which is assembled with the help of wavelet matrix compression using Haar wavelets. The computational geometry is the unit disc depicted in Fig. 4. The solution on the space-time cylinder is visualized for $N_J = 20\,480$ in Fig. 5.

We have tabulated the similar characteristics from the previous example for the matrix A_J in Table 2, while the sparsity patterns of A_J and L_J are shown in Fig. 6 for $N_J = 81\,920$ and after reordering. We observe a similar behaviour as for PCM, the computation times for nested dissection and the Cholesky decomposition are negligible compared to the wavelet matrix compression. In view of the fill-in, there is an increase of approximately a factor of 5 for the average number of nonzero entries per row for $N_J = 1\,310\,720$. The computation times for the θ -scheme in seconds for the 150 time steps are shown in the column labelled by t_θ . As can be seen, we obtain a solution time of roughly 11 minutes for $N_J = 1\,310\,720$ unknowns in the spatial discretization.

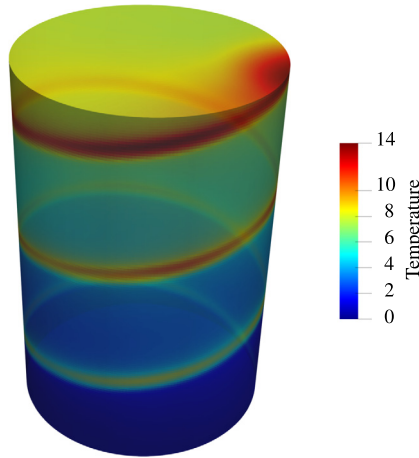


Fig. 5. Solution of the fractional Laplacian on the space-time cylinder. (For interpretation of the colours in the figure(s), the reader is referred to the web version of this article.)

Table 2
Computation times and numbers of nonzero entries in case of the fractional Laplacian.

N_J	t_{WEM}	t_{ND}	$t_{Chol}(A_J)$	t_θ	$anz(A_J)$	$anz(L_J)$
1280	0.63	0.02	0.01	0.19	88	140
5120	3.35	0.11	0.10	0.96	106	222
20480	22.20	0.53	0.59	5.34	117	315
81920	161.83	2.71	3.89	27.94	129	474
327680	724.60	12.87	23.49	131.30	135	576
1310720	3875.30	62.62	161.65	633.50	142	743

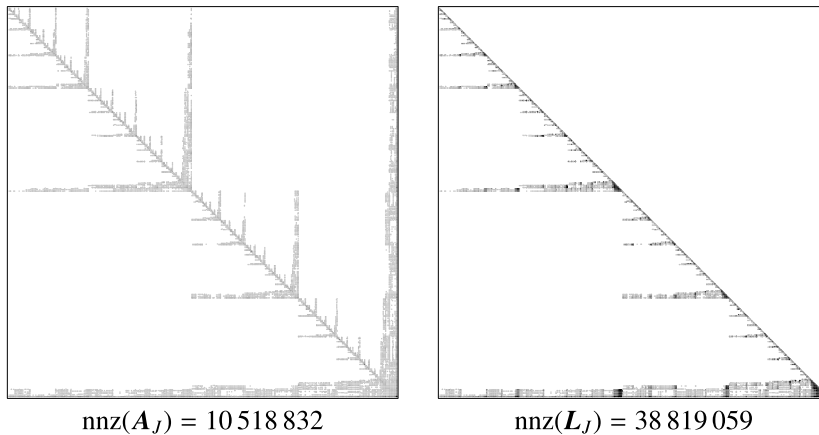


Fig. 6. Sparsity patterns of A_J (left) and the Cholesky factor L_J (right) for the fractional Laplacian and $N_J = 81920$. Each dot corresponds to a submatrix of size 64×64 . Lighter blocks have less entries than darker blocks.

6.3. Gaussian random field

For the simulation of a Gaussian random field, we consider an L-shaped domain with three holes, compare Fig. 7. The domain has a side length of 4, while the holes have a diameter of 0.8. For the wavelet matrix compression, we use piecewise bilinear biorthogonal wavelets with four vanishing moments, see [28]. The expectation is set to $\mathbb{E}[a](\mathbf{x}) \equiv 0$, while the covariance is given by the circular kernel

$$\text{Cov}[a](r_\theta) = \begin{cases} \frac{2}{\pi} \left(\arccos(r_\theta) - r_\theta \sqrt{1 - r_\theta^2} \right), & \text{if } r_\theta < 1, \\ 0, & \text{else,} \end{cases} \quad \text{where } r_\theta := \frac{\|\mathbf{x} - \mathbf{x}'\|_2}{\theta}$$

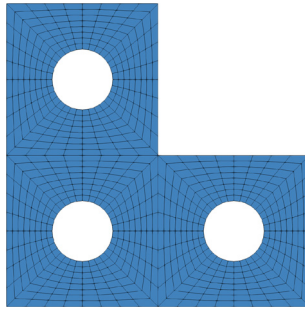


Fig. 7. Computational geometry for the Gaussian random field.

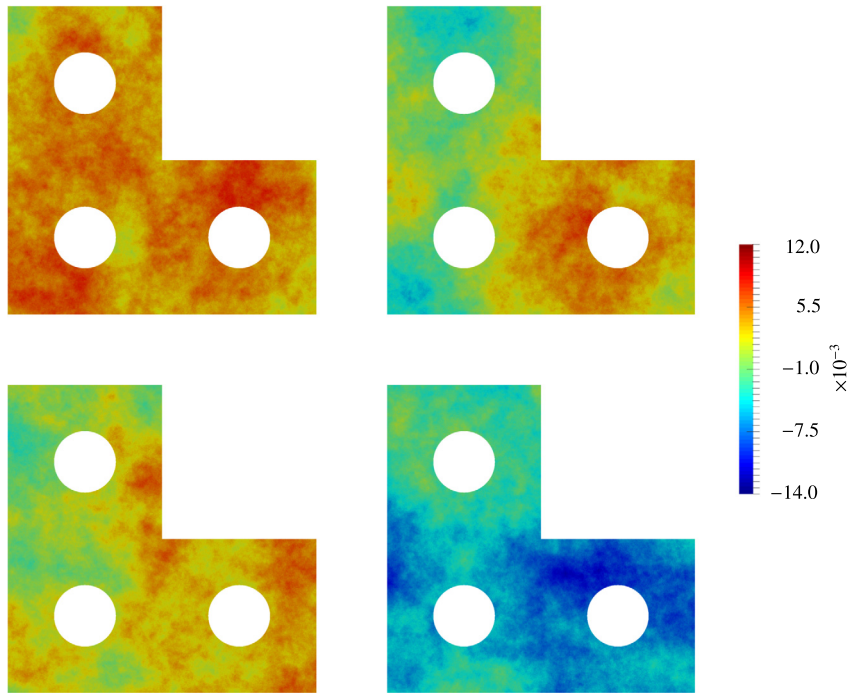


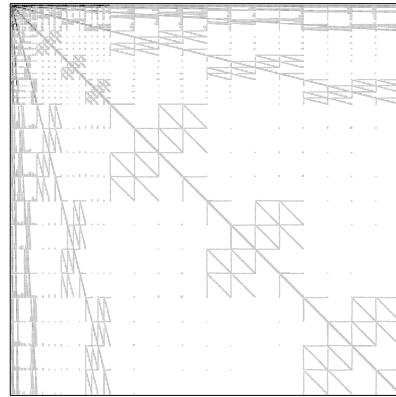
Fig. 8. Four different realizations of the Gaussian random field.

for some $\theta > 0$, see e.g. [15]. In this example, we set $\theta = 6$. Four different realizations of the corresponding field in case of $N_J = 792\,688$ nodes are depicted in Fig. 8.

The pattern of the covariance matrix C_J^G is shown in Fig. 9, while the patterns of the reordered matrix and its Cholesky factor are provided in Fig. 10.

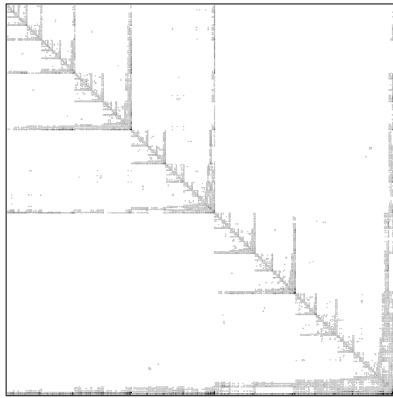
Table 3 shows the computation times and the average numbers of nonzero entries as in the previous examples. In addition, we have the column t_{Sample} , which contains the times for computing a single realization of the random field in seconds, computed by (15). These times have been computed by averaging the computation times over 1000 samples. In order to compute the inverse of the mass matrix, we could in principle reuse the nested dissection ordering which has been computed for the system matrix, as the pattern of the mass matrix is a subset of the pattern of the system matrix. However, this will result in a fill-in similar to the system matrix. Hence, it is favourable to use a different ordering for the mass matrix, resulting in much less fill-in.

As can be seen from Table 3, the times for sampling the random field only increase moderately. We remark that, due to the larger supports of the bilinear wavelets and the higher precision of the discretization, the system matrix contains more entries per row on average. This also leads to higher computation times for the matrix assembly. However, as before the increase of nonzero entries in the Cholesky factor remains very moderate. In addition, we have provided the average number of nonzeros per row for the mass matrix in the column labelled by $\text{anz}(G_J)$. The number of nonzeros for the corresponding Cholesky factor of the mass matrix is given in the column labelled by $\text{anz}(\hat{L}_J)$.

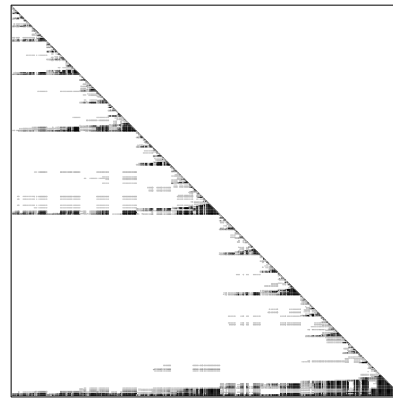


$$\text{nnz}(\mathbf{C}_J^G) = 123\,434\,984$$

Fig. 9. Sparsity patterns of \mathbf{C}_J^G of the random field for $N_J = 199692$. Each dot corresponds to a submatrix of size 108×108 . Lighter blocks have less entries than darker blocks.



$$\text{nnz}(\mathbf{C}_J^G) = 123\,434\,984$$



$$\text{nnz}(\mathbf{L}_J) = 738\,644\,869$$

Fig. 10. Sparsity patterns of \mathbf{C}_J^G (left) and the Cholesky factor \mathbf{L}_J (right) for the random field and $N_J = 199692$. Each dot corresponds to a submatrix of size 108×108 . Lighter blocks have less entries than darker blocks.

Table 3
Computation times and numbers of nonzero entries in case of the Gaussian random field.

N_J	t_{WEM}	t_{ND}	$t_{\text{Chol}}(\mathbf{C}_J^G)$	t_{Sample}	$\text{anz}(\mathbf{C}_J^G)$	$\text{anz}(\mathbf{L}_J)$	$\text{anz}(\mathbf{G}_J)$	$\text{anz}(\hat{\mathbf{L}}_J)$
972	1.27	0.04	0.24	$6.19 \cdot 10^{-4}$	247	247	53	31
3468	12.16	0.16	0.28	$3.88 \cdot 10^{-3}$	356	495	91	72
13068	100.72	0.85	2.29	$3.09 \cdot 10^{-2}$	444	1060	126	124
50700	662.80	4.43	21.11	$2.13 \cdot 10^{-1}$	523	2009	156	196
199692	3717.51	23.72	233.62	$1.46 \cdot 10^0$	619	3699	184	283
792588	25075.57	127.23	4334.80	$1.21 \cdot 10^1$	716	8496	211	390

7. Conclusion

In this article, we have proposed a very efficient direct solver for nonlocal operators. The pivotal idea is to combine the wavelet matrix compression with the nested dissection ordering. Thereby, the fill-in resulting from a Cholesky decomposition or an LU decomposition is drastically reduced. This has been numerically investigated into depth for three relevant applications, namely the polarizable continuum model, a parabolic problem for the fractional Laplacian in integral form, and the fast simulation of Gaussian random fields. In all three cases, wavelet matrix compression yields sparse system matrices, while the fill-in in the matrix factorization remains very low thanks to the nested dissection ordering. This behaviour has been observed for more than 10^6 unknowns in the discretization.

A formidable application of the presented approach is the fast simulation of rough (Gaussian) random fields. In such cases, the numerical solution of the eigenvalue problem for the covariance is computationally prohibitive and, hence, the computation of a Karhunen-Loève expansion is not feasible. In turn, the use of a wavelet basis yields a sparse representation

of the covariance operator and a matrix root is rapidly computable by employing the nested dissection ordering and the Cholesky decomposition.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgement

This research has been funded in parts by the Swiss National Science Foundation (SNF) under project number 407540_167186 NFP 75 Big Data.

References

- [1] G. Beylkin, R. Coifman, V. Rokhlin, The fast wavelet transform and numerical algorithms, *Commun. Pure Appl. Math.* 44 (1991) 141–183.
- [2] M. Bugeanu, R. Di Remigio, K. Mozgawa, S. Reine, H. Harbrecht, L. Frediani, Wavelet formulation of the polarizable continuum model. II. Use of piecewise bilinear boundary elements, *Phys. Chem. Chem. Phys.* 17 (2015) 31566–31581.
- [3] L. Cambier, C. Chen, E.G. Boman, S. Rajamanickam, R.S. Tuminaro, E. Darve, An algebraic sparsified nested dissection algorithm using low-rank approximations, *SIAM J. Matrix Anal. Appl.* 41 (2) (2020) 715–746.
- [4] C. Chen, L. Cambier, E.G. Boman, S. Rajamanickam, R.S. Tuminaro, E. Darve, A robust hierarchical solver for ill-conditioned systems with applications to ice sheet modeling, *J. Comput. Phys.* 396 (2020) 819–836.
- [5] J. Crank, P. Nicolson, A practical method for numerical evaluation of solutions of partial differential equations of the heat conduction type, *Proc. Camb. Philol. Soc.* 43 (1) (1947) 50–67.
- [6] E. Cancés, B. Mennucci, New applications of integral equations methods for solvation continuum models: ionic solutions and liquid crystals, *J. Math. Chem.* 23 (1998) 309–326.
- [7] E. Cancés, B. Mennucci, J. Tomasi, A new integral equation formalism for the polarizable continuum model: theoretical background and applications to isotropic and anisotropic dielectrics, *J. Chem. Phys.* 107 (1997) 3032–3041.
- [8] W. Dahmen, Wavelet and multiscale methods for operator equations, *Acta Numer.* 6 (1997) 55–228.
- [9] W. Dahmen, H. Harbrecht, R. Schneider, Compression techniques for boundary integral equations. Optimal complexity estimates, *SIAM J. Numer. Anal.* 43 (2006) 2251–2271.
- [10] W. Dahmen, S. Prößdorf, R. Schneider, Multiscale methods for pseudo-differential equations on smooth closed manifolds, in: C.K. Chui, L. Montefusco, L. Puccio (Eds.), *Proceedings of the International Conference on Wavelets: Theory, Algorithms, and Applications*, 1994, pp. 385–424.
- [11] J. Dölz, H. Harbrecht, C. Schwab, Covariance regularity and \mathcal{H} -matrix approximation for rough random fields, *Numer. Math.* 135 (4) (2017) 1045–1071.
- [12] M. D’Elia, M. Gunzburger, The fractional Laplacian operator on bounded domains as a special case of the nonlocal diffusion operator, *Comput. Math. Appl.* 66 (7) (2013) 1245–1260.
- [13] Q. Du, M. Gunzburger, R.B. Lehoucq, K. Zhou, Analysis and approximation of nonlocal diffusion problems with volume constraints, *SIAM Rev.* 54 (4) (2012) 667–696.
- [14] M. Feischl, F.Y. Kuo, I.H. Sloan, Fast random field generation with \mathcal{H} -matrices, *Numer. Math.* 140 (3) (2018) 639–676.
- [15] M.G. Genton, Classes of kernels for machine learning: a statistics perspective, *J. Mach. Learn. Res.* 2 (2001) 299–312.
- [16] A. George, Nested dissection of a regular finite element mesh, *SIAM J. Numer. Anal.* 10 (2) (1973) 345–363.
- [17] I.G. Graham, F.Y. Kuo, D. Nuyens, R. Scheichl, I.H. Sloan, Analysis of circulant embedding methods for sampling stationary random fields, *SIAM J. Numer. Anal.* 56 (3) (2018) 1871–1895.
- [18] L. Grasedyck, R. Kriemann, S. Le Borne, Domain decomposition based \mathcal{H} -LU preconditioning, *Numer. Math.* 112 (2009) 565–600.
- [19] L. Greengard, V. Rokhlin, A fast algorithm for particle simulation, *J. Comput. Phys.* 73 (1987) 325–348.
- [20] W. Hackbusch, A sparse matrix arithmetic based on \mathcal{H} -matrices. Part I: Introduction to \mathcal{H} -matrices, *Computing* 64 (1999) 89–108.
- [21] W. Hackbusch, B.N. Khoromskij, A sparse \mathcal{H} -matrix arithmetic: general complexity estimates, *J. Comput. Appl. Math.* 125 (1–2) (2000) 479–501.
- [22] W. Hackbusch, Z.P. Nowak, On the fast matrix multiplication in the boundary element method by panel clustering, *Numer. Math.* 54 (1989) 463–491.
- [23] H. Harbrecht, Preconditioning of wavelet BEM by the incomplete Cholesky factorization, *Comput. Vis. Sci.* 15 (6) (2012) 319–329.
- [24] H. Harbrecht, M. Peters, R. Schneider, On the low-rank approximation by the pivoted Cholesky decomposition, *Appl. Numer. Math.* 62 (4) (2012) 428–440.
- [25] H. Harbrecht, M. Peters, M. Siebenmorgen, Efficient approximation of random fields for numerical applications, *Numer. Linear Algebra Appl.* 22 (4) (2015) 596–617.
- [26] H. Harbrecht, M. Randrianarivony, Wavelet BEM on molecular surfaces. Parametrization and implementation, *Computing* 86 (1) (2009) 1–22.
- [27] H. Harbrecht, M. Randrianarivony, Wavelet BEM on molecular surfaces. Solvent excluded surfaces, *Computing* 92 (4) (2011) 335–364.
- [28] H. Harbrecht, R. Schneider, Biorthogonal wavelet bases for the boundary element method, *Math. Nachr.* 269–270 (2004) 167–188.
- [29] H. Harbrecht, R. Schneider, Wavelet Galerkin schemes for boundary integral equations. Implementation and quadrature, *SIAM J. Sci. Comput.* 27 (4) (2006) 1347–1370.
- [30] H. Harbrecht, R. Stevenson, Wavelets with patchwise cancellation properties, *Math. Comput.* 75 (256) (2006) 1871–1889.
- [31] L. Herrmann, K. Kirchner, C. Schwab, Multilevel approximation of Gaussian random fields: fast simulation, *Math. Models Methods Appl. Sci.* 30 (1) (2020) 181–223.
- [32] K.L. Ho, L. Ying, Hierarchical interpolative factorization for elliptic operators: differential equations, *Commun. Pure Appl. Math.* 69 (2016) 1415–1451.
- [33] G. Karypis, V. Kumar, A fast and high quality multilevel scheme for partitioning irregular graphs, *SIAM J. Sci. Comput.* 20 (1) (1998) 359–392.
- [34] L. Lin, C. Yang, J.C. Meza, J. Lu, L. Ying, W. E. Selln—an algorithm for selected inversion of a sparse symmetric matrix, *ACM Trans. Math. Softw.* 37 (4) (2011) 1–19.
- [35] F. Lindgren, H. Rue, J. Lindström, An explicit link between Gaussian fields and Gaussian Markov random fields: the stochastic partial differential equation approach, *J. R. Stat. Soc. B* 73 (4) (2011) 423–498.
- [36] R.J. Lipton, D.J. Rose, R.E. Tarjan, Generalized nested dissection, *SIAM J. Numer. Anal.* 16 (2) (1979) 346–358.
- [37] Matlab, Version 9.8.0.1396136 (R2020a), The MathWorks Inc., Natick, Massachusetts, 2020.
- [38] S. Miertuš, E. Scrocco, J. Tomasi, Electrostatic interaction of a solute with a continuum. A direct utilization of ab initio molecular potentials for the provision of solvent effects, *Chem. Phys.* 55 (1981) 117–129.

- [39] S. Parter, The use of linear graphs in Gaussian elimination, *SIAM Rev.* 3 (2) (1961) 119–130.
- [40] D.J. Rose, R.E. Tarjan, G.S. Lueker, Algorithmic aspects of vertex eliminations on graphs, *SIAM J. Comput.* 5 (1976) 266–283.
- [41] H. Rue, L. Held, *Gaussian Markov Random Fields. Theory and Applications*, Monographs on Statistics and Applied Probability, vol. 104, Chapman & Hall / CRC Press, London, 2005.
- [42] S. Sauter, C. Schwab, *Boundary Element Methods*, Springer, Berlin-Heidelberg, 2010.
- [43] R. Schneider, *Multiskalen- und Wavelet-Matrixkompression: Analysisbasierte Methoden zur Lösung großer vollbesetzter Gleichungssysteme*, Teubner, Stuttgart, 1998.
- [44] P.G. Schmitz, L. Ying, A fast nested dissection solver for Cartesian 3D elliptic problems using hierarchical matrices, *J. Comput. Phys.* 258 (2014) 227–245.
- [45] O. Steinbach, *Numerical Approximation Methods for Elliptic Boundary Value Problems. Finite and Boundary Elements*, Springer, New York, 2008.
- [46] J. Tomasi, B. Mennucci, R. Cammi, Quantum mechanical continuum solvation models, *Chem. Rev.* 105 (2005) 2999–3094.
- [47] V. Weijo, M. Randrianarivony, H. Harbrecht, L. Frediani, Wavelet formulation of the polarizable continuum model, *J. Comput. Chem.* 31 (7) (2010) 1469–1477.
- [48] Z. Zhao, J. Meza, L.-W. Wang, A divide-and-conquer linear scaling three-dimensional fragment method for large scale electronic structure calculations, *J. Phys. Condens. Matter* 20 (2008) 294203.