Technische Universität Graz

Complexity Analysis of a Fast Directional Matrix-Vector Multiplication

G. Of, R. Watschinger

**Berichte aus dem
Institut für Angewandte Mathematik**

Technische Universität Graz

Complexity Analysis of a Fast Directional Matrix-Vector Multiplication

G. Of, R. Watschinger

Berichte aus dem Institut für Angewandte Mathematik

Technische Universität Graz
Institut für Angewandte Mathematik
Steyrergasse 30
A 8010 Graz

WWW: `http://www.applied.math.tugraz.at`

Complexity Analysis of a Fast Directional Matrix-Vector Multiplication

Günther Of[0000−0003−2258−7001] and Raphael Watschinger[0000−0002−7750−8561]

Graz University of Technology, Institute of Applied Mathematics, Steyrergasse 30, 8010 Graz, Austria, `of@tugraz.at`, `watschinger@math.tugraz.at`

Abstract. We consider a fast, data-sparse directional method to realize matrix-vector products related to point evaluations of the Helmholtz kernel. The method is based on a hierarchical partitioning of the point sets and the matrix. The considered directional multi-level approximation of the Helmholtz kernel can be applied even on high-frequency levels efficiently. We provide a detailed analysis of the almost linear asymptotic complexity of the presented method. Our numerical experiments are in good agreement with the provided theory.

Keywords: Helmholtz · Fast multipole method · Hierarchical matrix.

1 Introduction

In this paper we consider an efficient method for the computation of the matrix-vector product for a fully populated matrix $A \in \mathbb{C}^{N_T \times N_S}$ with entries

$$A[j,k] = f(x_j, y_k), \tag{1}$$
$$f(x,y) = \frac{\exp(i\kappa|x-y|)}{4\pi|x-y|}$$

where f is the Helmholtz kernel, $\kappa > 0$ the wave number and $P_T = \{x_j\}_{j=1}^{N_T}$ and $P_S = \{y_k\}_{k=1}^{N_S}$ are two sets of points in $\mathbb{R}^3$. Similar matrices arise in the solution of boundary value problems for the Helmholtz equation by boundary element methods. Using standard matrix-vector multiplication is prohibitive for large N_T and N_S due to the asymptotic runtime and storage complexity $\mathcal{O}(N_T N_S)$.

Due to the oscillating behavior of the Helmholtz kernel, existing standard fast methods for the reduction of the complexity do not perform well for relatively large wave numbers κ. Therefore, a variety of methods have been developed. There are several versions of the fast multipole method (FMM) based on different expansions of the Helmholtz kernel f. A first version suitable for high frequency regimes is given in [16] and an overview of the early developments can be found in [14]. Of further interest are the methods in [8,12], which rely on plane wave expansions, and the wideband method in [7] which switches between different expansions in low and high frequency regimes.

Directional methods allow to overcome the deficiencies of standard schemes in high frequency regimes, too. The basic idea of these methods is that the

Helmholtz kernel f can locally be smoothed by a plane wave. In the context of fast methods this idea was first considered in [6] and later in [9]. In [13] the idea is picked up and combined with an approximation of the kernel via interpolation. [3,5] follow a similar path in the context of $\mathcal{H}^2$-matrices providing a rigorous analysis. A slightly different method is proposed in [1], where the directional smoothing is combined with a nested cross approximation of the kernel.

In this paper we present a directional method in the spirit of [13] based on a uniform clustering of the point sets. We choose this approach due to the applicability of the involved interpolation to other kernels and a smooth transition between low and high frequency regimes in contrast to the wideband FMM in [7]. We give a description of the method in Sect. 2 and an asymptotic complexity analysis in Sect. 3. While [13] provides already a brief analysis we present a detailed one not unlike the one in [3], but focusing on points distributed in 3D volumes instead of points on 2D manifolds and allowing two distinct sets of points. In addition, we exploit the uniformity for a significant storage reduction compared to non-uniform approaches. This reduction and the claimed almost linear asymptotic behavior can be observed in our numerical tests in Sect. 4.

2   Derivation of the Fast Directional Method

In this section we present a method for fast matrix-vector multiplications for the matrix A in (1) based on a hierarchical partitioning of the sets of points into boxes and a directional multi-level approximation of the Helmholtz kernel f on suitable pairs of such boxes.

2.1   Box Cluster Trees

The desired matrix partition can efficiently be constructed from a hierarchical tree clustering of the point sets into axis-parallel boxes. In what follows we define uniform box cluster trees which are constructed by a uniform subdivision of an initial box, see, e.g., [10]. In particular, we construct a uniform box cluster tree $\mathcal{T}_T$ for a given set of points $P_T = \{x_j\}_{j=1}^{N_T}$ in an axis-parallel box $T = (a_1, b_1] \times \ldots \times (a_3, b_3] \subset \mathbb{R}^3$ by Algorithm 1. As additional parameter we have the maximal number of points per leaf $n_{\max}$. We use standard notions of levels and leaves in trees known from graph theory. In addition we define

- the *index set* $\hat{t} := \{j \in \{1, \ldots, N_T\} : x_j \in t\}$ for a box $t \in \mathcal{T}_T$,
- the *level sets of the tree* by $\mathcal{T}_T^{(\ell)} := \{t \in \mathcal{T}_T : \text{level}(t) = \ell\}$,
- the *depth* $p(\mathcal{T}_T) := \max\{\text{level}(t) : t \in \mathcal{T}_T\}$ of the cluster tree $\mathcal{T}_T$,
- the *set $\mathcal{L}_T$ of all leaves* of $\mathcal{T}_T$.

In general, Alg. 1 creates an adaptive, i.e. unbalanced cluster tree depending on the point distribution. Other construction principles for box cluster trees such as bisection [15, Sect. 3.1.1] tailor the tree to the point sets yielding more balanced trees. However, the boxes at a given level ℓ of such a tree can vary strongly in shape, while the ones of a uniform box cluster tree are identical up

---

Algorithm 1 Construction of a uniform box cluster tree $\mathcal{T}_T$

1: **input**: Points $P_T = \{x_j\}_{j=1}^{N_T}$ inside a box $T = (a_1, b_1] \times \ldots \times (a_3, b_3]$, maximal number $n_{\max}$ of points per leaf.
2: Construct an empty tree $\mathcal{T}_T$ and add T as its root.
3: Call REFINECLUSTER(T, $\mathcal{T}_T$)

4: **function** REFINECLUSTER($T = (a_1, b_1] \times \ldots \times (a_3, b_3]$, $\mathcal{T}$)
5:     **if** $\#\{x_j : x_j \in T\} > n_{\max}$ **then**
6:         Compute center $c_1 = (a_1 + b_1)/2$, $c_2 = (a_2 + b_2)/2$, $c_3 = (a_3 + b_3)/2$.
7:         Uniformly subdivide T into 8 boxes $T_1 = (a_1, c_1] \times \ldots \times (a_3, c_3]$, $\ldots$,
8:         $T_8 = (c_1, b_1] \times \ldots \times (c_3, b_3]$.
9:         **for** $k = 1, \ldots, 8$ **do**
10:             **if** $\#\{x_j : x_j \in T_k\} \geq 1$ **then**
11:                 Add T_k to $\mathcal{T}$ as child of T.
12:                 Call REFINECLUSTER(T_k, $\mathcal{T}$).

---

to translation. We will exploit this uniformity to avoid recomputations and to reduce the storage costs of the presented method.

2.2   A Directional Kernel Approximation

In this section we describe a method to approximate the Helmholtz kernel f on a suitable pair of boxes t and s by a separable expansion, which will allow for low rank approximations of suitable subblocks of the matrix A in (1). Due to the oscillatory part $\exp(i\kappa|x-y|)$ of f, standard approaches like tensor interpolation of the kernel are not effective for relatively large κ as pointed out in [1,13]. Therefore, we consider a directional approach which first appeared in [6] and [9] and was later used in [5] and [13] among others. The basic idea is that the oscillatory part $\exp(i\kappa|x-y|)$ of f can be smoothened by a plane wave term $\exp(-i\kappa\langle x-y, c\rangle)$ in a cone around a direction $c \in \mathbb{R}^3$ with $|c| = 1$. We can rewrite the Helmholtz kernel f by expanding the numerator and the denominator by a plane wave term yielding

$$f(x, y) = f_c(x, y) \exp(i\kappa\langle x, c\rangle) \exp(-i\kappa\langle y, c\rangle), \tag{2}$$

$$f_c(x, y) := f(x, y) \exp(-i\kappa\langle x - y, c\rangle) = \frac{\exp(i\kappa(|x - y| - \langle x - y, c\rangle))}{4\pi|x - y|}. \tag{3}$$

The modified kernel function f_c is somewhat smoother than f on suitable boxes t and s. In fact, if two points $x \in t$ and $y \in s$ satisfy $(x - y)/|x - y| \approx c$, then $f_c(x, y) \approx (4\pi|x - y|)^{-1}$, i.e. the oscillations of f are locally damped in f_c. Therefore tensor interpolation can be applied to approximate f_c instead of f on suitable axis-parallel boxes t and s and we get

$$f_c(x, y) \approx \sum_{\nu \in M} \sum_{\mu \in M} f_c(\xi_{t,\nu}, \xi_{s,\mu}) L_{t,\nu}^{(m)}(x) L_{s,\mu}^{(m)}(y), \tag{4}$$

where ν and μ are multi-indices in the set $M = \{1, \ldots, m+1\}^3$, $\xi_{t,\nu}$ are tensor products of 1D Chebyshev nodes of order $m+1$ transformed to the box $t = (a_1, b_1] \times \ldots \times (a_3, b_3]$, i.e. $\xi_{t,\nu} = (\xi_{[a_1,b_1],\nu_1}, \xi_{[a_2,b_2],\nu_2}, \xi_{[a_3,b_3],\nu_3})$ with

$$\xi_{[a_j,b_j],\nu_j} = \frac{a_j + b_j}{2} + \frac{b_j - a_j}{2} \cos\left(\frac{2\nu_j - 1}{2\pi(m+1)}\right), \quad \nu_j \in \{1, \ldots, m+1\},$$

and $L_{t,\nu}^{(m)}$ are the corresponding Lagrange polynomials, which are tensor products of the 1D Lagrange polynomials corresponding to the interpolation nodes $\{\xi_{[a_j,b_j],\nu_j}\}_{\nu_j=1}^{m+1}$.

Inserting approximation (4) into (2) and grouping the terms depending on x and y, respectively, yields the desired separable approximation

$$f(x,y) \approx \sum_{\nu \in M} \sum_{\mu \in M} f_c(\xi_{t,\nu}, \xi_{s,\mu}) L_{t,c,\nu}^{(m)}(x) \overline{L_{s,c,\mu}^{(m)}(y)}, \tag{5}$$

$$L_{t,c,\nu}^{(m)}(x) := L_{t,\nu}^{(m)}(x) \exp(i\kappa\langle x, c\rangle). \tag{6}$$

The directional approximation (5) of f can be used to approximate the sub-matrix $A|_{\hat{t} \times \hat{s}}$ of the matrix A in (1) restricted to the entries of the index sets $\hat{t}$ and $\hat{s}$ for two suitable axis-parallel boxes t and s, i.e.,

$$A|_{\hat{t} \times \hat{s}}[j, k] = f(x_j, y_k) \approx \sum_{\nu \in M} \sum_{\mu \in M} f_c(\xi_{t,\nu}, \xi_{s,\mu}) L_{t,c,\nu}^{(m)}(x_j) \overline{L_{s,c,\mu}^{(m)}(y_k)}. \tag{7}$$

In matrix notation this reads

$$A|_{\hat{t} \times \hat{s}} \approx L_{t,c} A_{c,t \times s} L_{s,c}^*, \tag{8}$$

where we define the *coupling matrix* $A_{c,t \times s} \in \mathbb{C}^{(m+1)^3 \times (m+1)^3}$ by

$$A_{c,t \times s}[j, k] := f_c(\xi_{t,\alpha_j}, \xi_{s,\beta_k}), \quad j, k \in \{1, \ldots, (m+1)^3\}, \tag{9}$$

for suitably ordered multi-indices $\alpha_j, \beta_k \in M = \{1, \ldots, m+1\}^3$, the *directional interpolation matrix* $L_{t,c} \in \mathbb{C}^{\hat{t} \times (m+1)^3}$ by

$$L_{t,c}[j, k] := L_{t,c,\alpha_k}^{(m)}(x_j), \quad j \in \hat{t}, k \in \{1, \ldots, (m+1)^3\}, \tag{10}$$

and $L_{s,c}$ analogously. In particular, instead of the original $\#\hat{t} \cdot \#\hat{s}$ matrix entries only $(m+1)^3(\#\hat{t} + \#\hat{s} + (m+1)^3)$ entries have to be computed for the approximation in (8), which is significantly less if $(m+1)^3 \ll \#\hat{t}, \#\hat{s}$.

In the following admissibility conditions we will specify for which boxes t and s and which direction c the approximation in (5) is applicable. Similar criteria have been considered in [1,5,13]. In particular, the criteria lead to exponential convergence of the approximation with respect to the interpolation degree [5,17].

Definition 1 (Directional admissibility [5, cf. Sect. 3.3]). *Let $t, s \subset \mathbb{R}^3$ be two axis-parallel boxes and let $c \in \mathbb{R}^3$ be a direction with $|c| = 1$ or $c = 0$. Denote the midpoints of t and s by m_t and m_s, respectively. Let two constants $\eta_1 > 0$ and $\eta_2 > 0$ be chosen suitably. Define the diameter $\operatorname{diam}(t)$ and the distance $\operatorname{dist}(t, s)$ by*

$$\operatorname{diam}(t) := \sup_{x_1, x_2 \in t} |x_1 - x_2|, \quad \operatorname{dist}(t, s) := \inf_{x \in t, y \in s} |x - y|.$$

We say that t and s are directionally admissible with respect to c if the separation criterion

$$\max\{\operatorname{diam}(t), \operatorname{diam}(s)\} \leq \eta_2 \operatorname{dist}(t, s), \tag{A1}$$

and the two cone admissibility criteria

$$\kappa \left| \frac{m_t - m_s}{|m_t - m_s|} - c \right| \leq \frac{\eta_1}{\max\{\operatorname{diam}(t), \operatorname{diam}(s)\}}, \tag{A2}$$

$$\kappa \max\{\operatorname{diam}(t), \operatorname{diam}(s)\}^2 \leq \eta_2 \operatorname{dist}(t, s) \tag{A3}$$

are satisfied.

Criterion (A1) is a standard separation criterion, see, e.g., [10] and [11, Sect. 4.2.3]. It ensures that the boxes t and s are well-separated allowing for an approximation of general non-oscillating kernels.

Criterion (A3) is similar to (A1), since it also controls the distance of two boxes t and s. Note that (A1) follows immediately from (A3) in case that $\kappa \max\{\operatorname{diam}(t), \operatorname{diam}(s)\} > 1$ and vice versa in the opposite case. As stated in [3, Sect. 3], (A3) can also be understood as a bound on the angle between all vectors $x - y$ for $x \in t$ and $y \in s$ that shrinks if κ or $\max\{\operatorname{diam}(t), \operatorname{diam}(s)\}$ increases. Hence, (A3) guarantees that the angle between $x - y$ and a direction c is small if the angle between the difference of the midpoints $m_t - m_s$ and c is already small, which is enforced by (A2).

Indeed, criterion (A2) is used to assign a suitable direction c to two non-overlapping boxes t and s. While the choice $c = (m_t - m_s)/|m_t - m_s|$ would always guarantee (A2), we want to choose c from a small, finite set of directions. This allows to use the same direction c for a fixed box t and several boxes s_j and, therefore, to use the same interpolation matrix $L_{t,c}$ for the approximation of various blocks $A|_{\hat{t} \times \hat{s}_j}$ as in (8). A possible way to construct suitable sets of directions and further details on criterion (A2) are discussed in Sect. 2.4. First, we want to discuss how to use criteria (A1) and (A3) to construct a suitable partition of the matrix A in (1) based on the clustering described in Sect. 2.1.

2.3 Partitioning of the Matrix

In general, the sets of evaluation points P_T and P_S for the matrix A in (1) are contained in overlapping boxes T and S. Therefore, the full matrix A cannot be approximated directly. For this reason, we recursively construct a partition of A by Alg. 2, which we organize in a block tree $\mathcal{T}_{T \times S}$ ([11, Sect. 5.5]).

Algorithm 2 Construction of a block tree $\mathcal{T}_{T \times S}$

1: **input**: Box cluster trees $\mathcal{T}_T$ and $\mathcal{T}_S$, parameter η_2 for the criteria (A1) and (A3).
2: Set $b = (t_1^0, s_1^0)$, i.e. the pair of roots of $\mathcal{T}_T$ and $\mathcal{T}_S$.
3: Construct an empty tree $\mathcal{T}_{T \times S}$ and add b as its root.
4: Call REFINEBLOCK(b, $\mathcal{T}_{T \times S}$).

5: **function** REFINEBLOCK($b = (t, s)$, $\mathcal{T}_{T \times S}$)
6:      **if** $t \in \mathcal{L}_T$ **or** $s \in \mathcal{L}_S$ **then**
7:          **return**
8:      **if** t and s violate (A1) or (A3) **then**
9:          **for** $t' \in \text{child}(t)$ **do**
10:             **for** $s' \in \text{child}(s)$ **do**
11:                 Add $b' = (t', s')$ to $\mathcal{T}_{T \times S}$ as child of b.
12:                 Call REFINEBLOCK(b', $\mathcal{T}_{T \times S}$).

Definition 2. *Let $\mathcal{T}_T$ and $\mathcal{T}_S$ be two uniform box cluster trees and let $\eta_2 > 0$. A block tree $\mathcal{T}_{T \times S}$ is constructed by Alg. 2. The set of all leaves of $\mathcal{T}_{T \times S}$ is denoted by $\mathcal{L}_{T \times S}$ and split into the set of admissible (i.e. approximable) leaves and the set of inadmissible leaves*

$$\mathcal{L}_{T \times S}^+ := \{b = (t, s) \in \mathcal{L}_{T \times S} : t \text{ and } s \text{ satisfy (A1) and (A3)}\},$$
$$\mathcal{L}_{T \times S}^- := \mathcal{L}_{T \times S} \setminus \mathcal{L}_{T \times S}^+.$$

For a given block tree $\mathcal{T}_{T \times S}$ the pairs of indices $\hat{t} \times \hat{s}$ of all leaves $(t, s) \in \mathcal{L}_{T \times S}$ form a partition of the full index set $\{1, \ldots, N_T\} \times \{1, \ldots, N_S\}$, i.e. of the matrix A. The matrix blocks corresponding to admissible blocks $b \in \mathcal{L}_{T \times S}^+$ can be approximated by the directional interpolation (8). Inadmissible blocks related to $b \in \mathcal{L}_{T \times S}^-$ are computed directly.

2.4   Choice of Directions

As we would like to use relatively small numbers of directions c in the directional approximations (8), we consider a fixed set of directions $D^{(\ell)}$ for all blocks (t, s) at a given level ℓ of the block tree. These sets $D^{(\ell)}$ should be constructed in such a way that for all blocks (t, s) at level ℓ in $\mathcal{L}_{T \times S}^+$ there exists a direction $c \in D^{(\ell)}$ such that criterion (A2) holds for some fixed η_1.

Since the bound on the right-hand side of (A2) increases for decreasing diameters of t and s and these diameters are halved for each new level of the uniform box cluster trees, the number of directions in $D^{(\ell)}$ can be reduced with increasing level ℓ. If the maximum of the diameters of two boxes t and s at level $\tilde{\ell}$ is so small that the bound on the right-hand side of (A2) is greater than κ, then (A2) holds for $c = 0$ for all following levels. In this case, a plane wave term is not needed for the approximation of the Helmholtz kernel f, and the approximation (8) coincides with a standard tensor interpolation. We call the other levels

---

Algorithm 3 Construction of directions $D^{(\ell)}$

1: **input**: Largest high frequency level $\ell_{\mathrm{hf}} \geq -1$.
2: **for** $\ell = \ell_{\mathrm{hf}} + 1, \ell_{\mathrm{hf}} + 2, \ldots, \min\{p(\mathcal{T}_T), p(\mathcal{T}_S)\}$ **do**
3:     Set $D^{(\ell)} = \{0\}$.
4: Construct the six faces $\{E_j^{(\ell_{\mathrm{hf}})}\}_{j=1}^6$ of the cube $[-1,1]^3$, i.e.
    $E_1^{(\ell_{\mathrm{hf}})} = \{-1\} \times [-1,1]^2,\ E_2^{(\ell_{\mathrm{hf}})} = \{1\} \times [-1,1]^2,\ \ldots,\ E_6^{(\ell_{\mathrm{hf}})} = [-1,1]^2 \times \{1\}$.
5: Set $D^{(\ell_{\mathrm{hf}})} = \{c_j^{(\ell_{\mathrm{hf}})}\}_{j=1}^6$ where $c_j^{(\ell_{\mathrm{hf}})}$ is the midpoint of $E_j^{(\ell_{\mathrm{hf}})}$, i.e.
    $c_1^{(\ell_{\mathrm{hf}})} = (-1,0,0),\ c_2^{(\ell_{\mathrm{hf}})} = (1,0,0),\ \ldots,\ c_6^{(\ell_{\mathrm{hf}})} = (0,0,1)$.
6: **for** $\ell = \ell_{\mathrm{hf}} - 1, \ldots, 0$ **do**
7:     Set $D^{(\ell)} = \emptyset$.
8:     **for** all faces $E_j^{(\ell+1)}$, $j = 1, \ldots, 6 \cdot 4^{\ell_{\mathrm{hf}}-\ell-1}$ **do**
9:         Uniformly subdivide $E_j^{(\ell+1)}$ into 4 faces $E_{4(j-1)+1}^{(\ell)}, \ldots, E_{4j}^{(\ell)}$.
10:         Construct normalized midpoints $c_{4(j-1)+1}^{(\ell)}, \ldots, c_{4j}^{(\ell)}$ of $E_{4(j-1)+1}^{(\ell)}, \ldots, E_{4j}^{(\ell)}$.
11:         Add directions $c_{4(j-1)+1}^{(\ell)}, \ldots, c_{4j}^{(\ell)}$ to $D^{(\ell)}$.

---

satisfying

$$\frac{\eta_1}{\kappa \max\{\mathrm{diam}\,(t), \mathrm{diam}\,(s)\}} \leq 1, \quad \text{for all } t \in \mathcal{T}_T^\ell, s \in \mathcal{T}_S^\ell, \tag{11}$$

high frequency levels and denote the *largest high frequency level* as ℓ_{hf}, or set $\ell_{\mathrm{hf}} = -1$ in case that all levels $\ell \geq 0$ are low frequency levels, i.e. do not satisfy (11). The value of ℓ_{hf} depends on η_1 and the uniform box cluster trees $\mathcal{T}_T$ and $\mathcal{T}_S$. In practice, we choose a suitable level ℓ_{hf} instead of η_1 and construct the *sets of directions* $D^{(\ell)}$, using more and more directions for levels $\ell < \ell_{\mathrm{hf}}$. Our construction by Alg. 3 combines ideas from [9, Sect. 4.1] and [3, Sect. 3].

Finally, we assign a direction $c \in D^{(\ell)}$ to a pair of boxes t and s which is close to the normalized difference $(m_t - m_s)/|m_t - m_s|$ of the midpoints of t and s and, hence, can be used for the directional approximation (8). For this purpose, we define a mapping $\mathrm{dir}_{(\ell)}$ for each level $\ell \in \mathbb{N}_0$, which maps a vector v in $\mathbb{R}^3 \backslash \{0\}$ to a direction $c_j^{(\ell)}$ such that the intersection point of the ray $\{\lambda v : \lambda > 0\}$ and the surface of the cube $[-1,1]^3$ lies in the face $E_j^{(\ell)}$ (cf. Alg. 3).

Definition 3. *Let $\ell_{\mathrm{hf}} \geq -1$ and let the directions $D^{(\ell)}$ and the faces $\{E_j^{(\ell)}\}$ be constructed by Alg. 3. We define the mapping $\mathrm{dir}_{(\ell)} : \mathbb{R}^3 \to D^{(\ell)} \cup \{0\}$ for each $\ell \in \mathbb{N}_0$ as follows:*

- *If $\ell > \ell_{\mathrm{hf}}$ we set $\mathrm{dir}_{(\ell)}(v) = 0$ for all $v \in \mathbb{R}^3$.*
- *If $\ell \leq \ell_{\mathrm{hf}}$ we set $\mathrm{dir}_{(\ell)}(0) = 0$. For all $v \in \mathbb{R}^3 \backslash \{0\}$ we set $\mathrm{dir}_{(\ell)}(v) = c_{j(v)}^{(\ell)}$ where*

$$j(v) := \min\{j : \psi_Q(v) \in E_j^{(\ell)}\}, \quad \psi_Q(v) := \frac{1}{\max_{j \in \{1,\ldots,3\}} |v_j|}\, v$$

to avoid ambiguity.

For two boxes $t, s \subset \mathbb{R}^3$ and a level $\ell \in \mathbb{N}_0$ we define the direction $c_{(\ell)}(t, s)$ by

$$c_{(\ell)}(t, s) := \mathrm{dir}_{(\ell)} \left(\frac{m_t - m_s}{|m_t - m_s|} \right).$$

In this way (A2) is satisfied for two boxes t, s, and the direction $c_{(\ell)}(t, s)$ for a constant η_1 which depends linearly on the product $\kappa q_{\ell_{\mathrm{hf}}}$ [17, Thm. 2.19]. Here $q_{\ell_{\mathrm{hf}}}$ denotes the maximal diameter of all boxes at level ℓ_{hf} in the trees $\mathcal{T}_T$ and $\mathcal{T}_S$.

2.5   Transfer Operations

The approximation of an admissible subblock $A|_{\hat{t} \times \hat{s}}$ of A in (7) can be further enhanced. If t is a non-leaf box at level ℓ in a box cluster tree $\mathcal{T}_T$ with children $t_1, \ldots, t_k$, the directional interpolation matrix $L_{t,c}$ can be approximated using the matrices $L_{t_j,c_{\ell+1}}$ for a suitable direction $c_{\ell+1}$. We describe this approach following [5, Sect. 2.2.2].

Let us rewrite the generating functions $L_{t,c,\nu}^{(m)}(x)$ of $L_{t,c}$ in (6) by

$$L_{t,c,\nu}^{(m)}(x) = \exp(i\kappa\langle x, c_{\ell+1}\rangle) \left[\exp(i\kappa\langle x, c - c_{\ell+1}\rangle) L_{t,\nu}^{(m)}(x) \right].$$

If $c_{\ell+1}$ is sufficiently close to c, the term in square brackets is smooth and can be interpolated for points x in a child box t_j yielding

$$\exp(i\kappa\langle x, c - c_{\ell+1}\rangle) L_{t,\nu}^{(m)}(x) \approx \sum_{\tilde{\nu} \in M} \exp(i\kappa\langle \xi_{t_j,\tilde{\nu}}, c - c_{\ell+1}\rangle) L_{t,\nu}^{(m)}(\xi_{t_j,\tilde{\nu}}) L_{t_j,\tilde{\nu}}^{(m)}(x).$$

This provides an approximation of the restriction of $L_{t,c,\nu}^{(m)}$ to the child t_j

$$L_{t,c,\nu}^{(m)}\big|_{t_j}(x) \approx \sum_{\tilde{\nu} \in M} \left(\left[\exp(i\kappa\langle \xi_{t_j,\tilde{\nu}}, c - c_{\ell+1}\rangle) L_{t,\nu}^{(m)}(\xi_{t_j,\tilde{\nu}}) \right] L_{t_j,c_{\ell+1},\tilde{\nu}}^{(m)}(x) \right).$$

In matrix notation the related restriction to the index set $\hat{t}_j$ reads as

$$L_{t,c}\big|_{\hat{t}_j \times (m+1)^3} \approx L_{t_j,c_{\ell+1}} E_{t_j,c}, \tag{12}$$

where the entries of the transfer matrix $E_{t_j,c} \in \mathbb{C}^{(m+1)^3 \times (m+1)^3}$ are defined by

$$E_{t_j,c}[k, \ell] := \exp(i\kappa\langle \xi_{t_j,\nu_k}, c - c_{\ell+1}\rangle) L_{t,\nu_\ell}^{(m)}(\xi_{t_j,\nu_k}), \tag{13}$$

for all $k, \ell \in \{1, \ldots, (m+1)^3\}$.

A suitable choice [17, Thm. 2.19] for the direction $c_{\ell+1}$ is given by $\mathrm{dir}_{(\ell+1)}(c)$, with $\mathrm{dir}_{(\ell+1)}$ given in Def. 3. Since this direction depends only on c and the level ℓ of the box t, it is reasonable to omit the dependence of the transfer matrix $E_{t_j,c}$ on $c_{\ell+1}$ in the notation.

2.6 Main Algorithm

In the previous sections, we have described how to partition the matrix (1) and how to approximate suitable subblocks. Here we explain the complete algorithm for a a matrix-vector multiplication $g = Av$.

The idea is to execute the multiplication blockwise according to the partition induced by the leaves $\mathcal{L}_{T \times S}$ of the block tree $\mathcal{T}_{T \times S}$. Inadmissible blocks from $\mathcal{L}_{T \times S}^{-}$ are multiplied directly with the target vector v. For admissible blocks from $\mathcal{L}_{T \times S}^{+}$ we use the decomposition (8) and split the multiplication into three phases. This is similar to the usual three-phase algorithm for $\mathcal{H}^2$-matrices [11, Sect. 8.7] and the FMM [10] with adaptations due to the directional approximation. We describe the scheme first for one block corresponding to an admissible pair of boxes t and s at level ℓ and then give a description of the complete algorithm.

In the first phase, the forward transformation, the product $\tilde{v}_{s,c} := L_{s,c}^{*} v|_{\hat{s}}$ is computed. If s is a leaf in the cluster tree, this is done directly by (10). This is also known as S2M (source to moment) step in fast multipole methods. If s is not a leaf, approximation (12) with $c_{\ell+1} = \mathrm{dir}_{(\ell+1)}(c)$ is used iteratively to get

$$\tilde{v}_{s,c} := \sum_{s_j \in \mathrm{child}(s)} E_{s_j,c}^{*} \tilde{v}_{s_j, c_{\ell+1}} \approx \sum_{s_j \in \mathrm{child}(s)} E_{s_j,c}^{*} \left[L_{s_j, c_{\ell+1}}^{*} v|_{\hat{s}_j} \right] \approx L_{s,c}^{*} v|_{\hat{s}},$$

by using the products of the children, which is also known as M2M operation (moment to moment). In the second phase, which is called multiplication phase or M2L (moment to local) step, the product

$$\tilde{g}_{t,c} := A_{c,t \times s} \tilde{v}_{s,c}$$

is computed by (9). In the complete algorithm all contributions from various boxes s are added up, i.e.

$$\tilde{g}_{t,c} := \sum_{s:(t,s) \in \mathcal{L}_{T \times S}^{+}} A_{c,t \times s} \tilde{v}_{s,c}.$$

In the third phase, the so-called backward transformation, the product

$$\tilde{g}|_{\hat{t}} := L_{t,c} \tilde{g}_{t,c} \tag{14}$$

is computed. If t is a leaf, this is done directly. This step is known as L2T (local to target) in fast multipole methods. If t is not a leaf, the approximation (12) is used to compute

$$\tilde{g}_{t_j, c_{\ell+1}} = E_{t_j,c}\, \tilde{g}_{t,c}, \tag{15}$$

for all children t_j of t, which is also known as L2L operation (local to local), and the evaluation (14) takes place for descendants which are leaves. In the complete algorithm the local contribution in (15) is added to the existing contribution $\tilde{g}_{t_j, c_{\ell+1}}$ originating from the multiplication phase.

Before we present the complete Alg. 4, we define the sets of active and inherited directions for each box in the cluster trees $\mathcal{T}_T$ and $\mathcal{T}_S$. These are used to keep track of all required directions for boxes t and s in the trees $\mathcal{T}_T$ and $\mathcal{T}_S$. They can be generated during the construction of the block tree $\mathcal{T}_{T \times S}$.

Definition 4. *Let $\mathcal{T}_T$ and $\mathcal{T}_S$ be two uniform box cluster trees, $\mathcal{T}_{T \times S}$ the corresponding block tree and $\ell_{\mathrm{hf}} \geq -1$. Recalling Alg. 3 and Def. 3 we define for all $\ell \geq 0$ and all $t \in \mathcal{T}_T^{(\ell)}$ the set of active directions by*

$$D(t) := \{c \in D^{(\ell)} : \exists\ s \in \mathcal{T}_S^{(\ell)} \ \text{such that} \ (t, s) \in \mathcal{L}_{T \times S}^+ \ \text{and} \ c = c_{(\ell)}(t, s)\}.$$

The set of inherited directions $\hat{D}(t)$ is defined recursively by setting $\hat{D}(t_1^0) = \emptyset$ for the root t_1^0 of $\mathcal{T}_T$, and for all $\ell > 0$ and all $t \in \mathcal{T}_T^{(\ell)}$ by setting

$$\hat{D}(t) := \{\hat{c} \in D^{(\ell)} : \exists\ c \in D(t') \cup \hat{D}(t') \ \text{such that} \ \hat{c} = \mathrm{dir}_{(\ell)}(c),\ t' = \mathrm{parent}(t)\}.$$

Analogously, the sets of active directions $D(s)$ and inherited directions $\hat{D}(s)$ are defined for clusters $s \in \mathcal{T}_S$.

Algorithm 4 Fast directional matrix vector multiplication $g \approx Av$

1: **input**: Box cluster trees $\mathcal{T}_T$ and $\mathcal{T}_S$, block tree $\mathcal{T}_{T \times S}$, interpolation degree m,
         sets of directions $D(t)$, $\hat{D}(t)$, $D(s)$, $\hat{D}(s)$ for all boxes t, s in $\mathcal{T}_T$, $\mathcal{T}_S$.
2: Initialize $g = 0$.
3: ▷ *Forward transformation*
4: **for** all leaves $s \in \mathcal{L}_S$ **do**
5:     **for** all directions $c \in D(s) \cup \hat{D}(s)$ **do**
6:         Compute $\tilde{v}_{s,c} = L_{s,c}^* v|_{\hat{s}}$.
7: **for** all levels $\ell = p(\mathcal{T}_S) - 1,\ \ldots,\ 0$ **do**
8:     **for** all non-leaf boxes $s \in \mathcal{T}_S^{(\ell)} \backslash \mathcal{L}_S$ **do**
9:         **for** all directions $c \in D(s) \cup \hat{D}(s)$ **do**
10:            Set $\tilde{v}_{s,c} = 0$.
11:            **for** all $s' \in \mathrm{child}(s)$ **do**
12:                Update $\tilde{v}_{s,c} \mathrel{+}= E_{s',c}^* \tilde{v}_{s',c'}$, where $c' = \mathrm{dir}_{(\ell+1)}(c)$.
13: ▷ *Multiplication phase*
14: **for** all boxes $t \in \mathcal{T}_T$ **do**
15:     **for** all directions $c \in D(t) \cup \hat{D}(t)$ **do**
16:         Initialize $\tilde{g}_{t,c} = 0$.
17:     **for** all boxes $s \in \mathcal{T}_S$ such that $(t, s) \in \mathcal{L}_{T \times S}^+$ **do**
18:         Update $\tilde{g}_{t,c} \mathrel{+}= A_{c,t \times s} \tilde{v}_{s,c}$, where $c = \mathrm{dir}_{(\ell)}(t, s)$ and $\ell = \mathrm{level}(t)$.
19: ▷ *Backward transformation*
20: **for** all levels $\ell = 0,\ \ldots,\ p(\mathcal{T}_T) - 1$ **do**
21:     **for** all non-leaf boxes $t \in \mathcal{T}_T^{(\ell)} \backslash \mathcal{L}_T$ **do**
22:         **for** all directions $c \in D(t) \cup \hat{D}(t)$ **do**
23:             **for** all $t' \in \mathrm{child}(t)$ **do**
24:                 Update $\tilde{g}_{t',c'} \mathrel{+}= E_{t',c} \tilde{g}_{t,c}$, where $c' = \mathrm{dir}_{(\ell+1)}(c)$.
25: **for** all leaves $t \in \mathcal{L}_T$ **do**
26:     **for** all directions $c \in D(t) \cup \hat{D}(t)$ **do**
27:         Update $g|_{\hat{t}} \mathrel{+}= L_{t,c} \tilde{g}_{t,c}$.
28: ▷ *Nearfield evaluation*
29: **for** all blocks $b = (t, s) \in \mathcal{L}_{T \times S}^-$ **do**
30:     Update $g|_{\hat{t}} \mathrel{+}= A|_{\hat{t} \times \hat{s}} v|_{\hat{s}}$.

2.7   Implementation Details

In this section, we describe how to exploit the uniformity of the box cluster trees to reduce the storage required by the transfer matrices $E_{t',c}$ defined in (13) and the coupling matrices $A_{c,t\times s}$ defined in (9). This is crucial as there is a large number of such matrices involved in the computations in Alg. 4.

For a level $\ell \geq 0$, a box $t \in \mathcal{T}_T^\ell$, a child t' and directions c and $c' = \mathrm{dir}_{(\ell+1)}(c)$ we consider the transfer matrix $E_{t',c}$ which has the entries

$$E_{t',c}[j,k] = \exp(i\kappa\langle\xi_{t',\nu_j}, c - c'\rangle)L_{t,\nu_k}^{(m)}(\xi_{t',\nu_j}), \quad j,k \in \{1,\ldots,(m+1)^3\}.$$

This matrix can be split into a directional and a non-directional part by

$$E_{t',c} = E_{t',c}^{\mathrm{d}}E_{t'},$$

where we define the directional part $E_{t',c}^{\mathrm{d}}$ and the non-directional part $E_{t'}$ by

$$E_{t',c}^{\mathrm{d}} := \mathrm{diag}\left(\{\exp(i\kappa\langle\xi_{t',\tilde{\nu}}, c - c'\rangle)\}_{\tilde{\nu}\in M}\right),$$
$$E_{t'}[j,k] := L_{t,\nu_k}^{(m)}(\xi_{t',\nu_j}), \quad j,k \in \{1,\ldots,(m+1)^3\}.$$

Let us consider the non-directional part $E_{t'}$ first. The value of the Lagrange polynomial $L_{t,\nu}^{(m)}$ depends only on the position of the evaluation point $\xi_{t',\mu}$ relative to the box t. Together with the uniformity of the box cluster tree $\mathcal{T}_T$, this implies that each $E_{t'}$ is identical to one of 8 non-directional transfer matrices in a reference configuration. Only these reference matrices of size $(m+1)^3 \times (m+1)^3$ have to be computed and stored. The directional part $E_{t',c}^{\mathrm{d}}$ changes for varying boxes t, child boxes t' or directions c. Since it is diagonal, however, only $(m+1)^3$ entries instead of $(m+1)^6$ entries need to be computed. Furthermore, for low frequency levels $\ell > \ell_{\mathrm{hf}}$ the directional part $E_{t',c}^{\mathrm{d}}$ becomes the identity and no additional computations are required.

Next we consider the coupling matrices $A_{c,t\times s}$ defined in (9) for admissible blocks (t,s) in a block tree $\mathcal{T}_{T\times S}$. $A_{c,t\times s}$ depends on the difference of the cluster centers only, see (3). Due to the uniformity of the box cluster trees, many of the coupling matrices coincide. In particular, it suffices to compute and store all required coupling matrices for all levels ℓ only once for a reference configuration and assign them to the appropriate blocks $(t,s) \in \mathcal{L}_{T\times S}^+$.

The dimension of the coupling matrices (9) increases cubically in the interpolation degree m. A compression of these matrices by a low rank approximation

$$A_{c,t\times s} \approx U_{c,t\times s}V_{c,t\times s}^*,$$

with $U_{c,t\times s}$, $V_{c,t\times s} \in \mathbb{C}^{(m+1)^3 \times k}$ for some low rank k, increases the performance of the algorithm (cf. [13]). Such approximations exist because the coupling matrices are generated by smooth functions. For their construction, we apply a partially pivoted ACA [2,15] in our implementation and the examples in Sect. 4, but do not analyze its effect on the complexity in the following section. A more involved compression strategy is described in [4].

3   Complexity Analysis

To analyze the complexity of Alg. 4 for fast directional matrix vector multiplications, we estimate the number of directional interpolation matrices and transfer matrices in Thm. 1, give then an estimate for the number of coupling matrices in Thm. 2 and 3 and finally estimate the number of nearfield matrices in Thm. 4. We start by establishing the general setting.

Throughout this section we fix the wave number $\kappa > 0$ and the sets of points $P_T = \{x_j\}_{j=1}^{N_T}$ and $P_S = \{y_k\}_{k=1}^{N_S}$, which may but do not have to coincide, and set $N = \max\{N_T, N_S\}$. In all considerations $\mathcal{T}_T$ and $\mathcal{T}_S$ denote two uniform box cluster trees as constructed in Alg. 1 for a fixed parameter $n_{\max}$. We set the maximum and the minimum of the depths of the trees $\mathcal{T}_T$ and $\mathcal{T}_S$

$$p_{\max} := \max\{p(\mathcal{T}_T), p(\mathcal{T}_S)\}, \quad p_{\min} := \min\{p(\mathcal{T}_T), p(\mathcal{T}_S)\}.$$

The diameters of all boxes at a fixed level ℓ of $\mathcal{T}_T$ are identical and denoted as $q_\ell(\mathcal{T}_T)$ just like the diameters $q_\ell(\mathcal{T}_S)$ of boxes at level ℓ in $\mathcal{T}_S$. For all levels $\ell \leq p_{\min}$ we define

$$q_\ell := \max\{q_\ell(\mathcal{T}_T), q_\ell(\mathcal{T}_S)\}.$$

The related block tree $\mathcal{T}_{T \times S}$ is constructed by Alg. 2 for a fixed parameter η_2. For the directional approximation we use a small, fixed interpolation degree m and the directions $D^{(\ell)}$, constructed by Alg. 3 for a fixed choice of the largest high frequency level $\ell_{\mathrm{hf}} \geq -1$.

For the complexity analysis we will need a few assumptions which we collect and discuss here. We assume that there exist small constants c_{geo}, $c_{\max}$, c_{ad} and $c_{\mathrm{un}} \in \mathbb{R}_+$ such that the following assumptions hold true:

$$n_{\max} \leq c_{\max}(m+1)^3, \tag{16}$$

$$q_0 \leq c_{\mathrm{un}} \min\{q_0(\mathcal{T}_T), q_0(\mathcal{T}_S)\}, \tag{17}$$

$$p_{\max} \leq \log_8(N) + c_{\mathrm{ad}}, \tag{18}$$

$$\kappa q_0 \leq c_{\mathrm{geo}} \sqrt[3]{N}. \tag{19}$$

In addition, ℓ_{hf} is assumed to be chosen such that

$$\ell_{\mathrm{hf}} + 1 \leq p_{\max} + c_{\mathrm{hf}}, \tag{20}$$

for a small constant $c_{\mathrm{hf}} \in \mathbb{N}_0$. Furthermore, we require that (33) holds, which we introduce and discuss later. Let us shortly discuss above assumptions. By equation (16) we ensure that the maximal number of points in leaf boxes of the cluster trees is reasonably small. Assumption (17) means that the diameters of the root boxes of $\mathcal{T}_T$ and $\mathcal{T}_S$ should be of comparable size. While this is not satisfied in general, one can enforce it by an initial subdivision of the greater box and application of the method to the resulting subboxes. Eqn. (18) is an indirect assumption on the sets of points P_T and P_S, which holds if the points are distributed more or less uniformly in a 3D domain. Also Eqn. (19) is reasonable only if points are distributed rather uniformly in a 3D volume, and guarantees

that the wave length $\lambda = 2\pi/\kappa$ is resolved in that case, which is required in typical physical applications. Finally, Eqn. (20) is a bound on the largest high frequency level ℓ_{hf} and allows to bound the number of directions constructed in Alg. 3. With these assumptions we can start with the complexity analysis, which is based on the following obvious, but important observation.

Remark 1. In Alg. 4 every directional interpolation matrix $L_{t,c}$ and $L_{s,c}$, every transfer matrix $E_{t',c}$ and $E_{s',c}$, every coupling matrix $A_{c,t\times s}$ and every nearfield matrix $A|_{\hat{t}\times\hat{s}}$ is multiplied with a suitable vector exactly once. All entries of these matrices can be computed with $\mathcal{O}(1)$ operations. Since the complexity of the application of a matrix to a vector is proportional to the number of its entries, it suffices to count all these matrices and their respective entries to estimate the storage and runtime complexity of Alg. 4.

Theorem 1. *Let assumption* (20) *hold true. Then there exists a constant c_{LE} depending only on c_{hf} such that the number N_{LE} of applied transfer matrices $E_{t',c}$ and $E_{s',c}$ and directional interpolation matrices $L_{t,c}$ and $L_{s,c}$ in Alg. 4 is bounded by*

$$N_{\mathrm{LE}} \le c_{\mathrm{LE}}\, 8^{p_{\max}}. \tag{21}$$

If (16) *and* (18) *apply in addition, these matrices can be stored and applied with complexity $\mathcal{O}(N)$.*

Proof. We start to estimate the number $N_{\mathrm{LE},T}$ of applied transfer matrices $E_{t',c}$ for L2L operations in lines 20–24 of Alg. 4 and directional interpolation matrices $L_{t,c}$ for L2T operations in lines 25–27. For this purpose we estimate the number of such matrices for each box t in $\mathcal{T}_T$.

Let us first assume, that $t \in \mathcal{T}_T^\ell$ is a non-leaf box at level ℓ. In this case a transfer matrix is applied for each direction $c \in D(t) \cup \hat{D}(t)$ and each box $t' \in \mathrm{child}(t)$, but no directional interpolation matrix. The number of directions in $D(t) \cup \hat{D}(t)$ is bounded by $\#D^{(\ell)}$, which is $6 \cdot 4^{\ell_{\mathrm{hf}}-\ell}$ if $\ell \le \ell_{\mathrm{hf}}$ and 1 else, and $\#\,\mathrm{child}(t) \le 8$ for all t due to the uniformity of the box cluster tree. Therefore, the total number $N_{\mathrm{LE}}(t)$ of transfer and directional interpolation matrices needed for a non-leaf box $t \in \mathcal{T}_T^\ell$ is bounded by

$$B^{(\ell)} = \begin{cases} 48 \cdot 4^{\ell_{\mathrm{hf}}-\ell}, & \text{if } \ell \le \ell_{\mathrm{hf}}, \\ 8, & \text{otherwise.} \end{cases}$$

If $t \in \mathcal{T}_T^\ell$ is a leaf box then we only need a directional interpolation matrix for each direction $c \in D(t) \cup \hat{D}(t)$ but no transfer matrix. Therefore, $N_{\mathrm{LE}}(t)$ is bounded by $6 \cdot 4^{\ell_{\mathrm{hf}}-\ell}$ if $\ell \le \ell_{\mathrm{hf}}$ and by 1 otherwise. Since this bound is less than $B^{(\ell)}$ for all levels ℓ, there holds $N_{\mathrm{LE}}(t) \le B^{(\ell)}$ for all boxes $t \in \mathcal{T}_T^\ell$.

The number $N_{\mathrm{LE},T}$ of all directional interpolation matrices and transfer matrices for boxes $t \in \mathcal{T}_T$ can hence be estimated by

$$N_{\mathrm{LE},T} = \sum_{\ell=0}^{p(\mathcal{T}_T)} \sum_{t \in \mathcal{T}_T^\ell} N_{\mathrm{LE}}(t) \le \sum_{\ell=0}^{p(\mathcal{T}_T)} \#\mathcal{T}_T^\ell \, B^{(\ell)}.$$

Due to the uniformity of the box cluster tree there holds $\#\mathcal{T}_T^\ell \leq 8^\ell$. Let us first assume that all levels in $\mathcal{T}_T$ are high frequency levels, i.e. $p(\mathcal{T}_T) \leq \ell_{\mathrm{hf}}$. Then we can further estimate

$$N_{\mathrm{LE},T} \leq \sum_{\ell=0}^{p(\mathcal{T}_T)} 48 \cdot 4^{\ell_{\mathrm{hf}}-\ell} 8^\ell < 48 \cdot 4^{\ell_{\mathrm{hf}}} 2^{p(\mathcal{T}_T)+1} \leq 24 \cdot 4^{c_{\mathrm{hf}}} 8^{p_{\max}}, \qquad (22)$$

where we used assumption (20) in the last step. If instead $p(\mathcal{T}_T) > \ell_{\mathrm{hf}}$, we get

$$N_{\mathrm{LE},T} \leq \sum_{\ell=0}^{\ell_{\mathrm{hf}}} 48 \cdot 4^{\ell_{\mathrm{hf}}-\ell} 8^\ell + \sum_{\ell=\ell_{\mathrm{hf}}+1}^{p(\mathcal{T}_T)} 8^{\ell+1}$$
$$\leq 12 \cdot 8^{\ell_{\mathrm{hf}}+1} + 8\left(8^{p(\mathcal{T}_T)+1} - 8^{\ell_{\mathrm{hf}}+1}\right) \leq 68 \cdot 8^{p_{\max}}. \qquad (23)$$

Analogously, we can estimate the number $N_{\mathrm{LE},S}$ of transfer matrices and directional interpolation matrices needed for the S2M and M2M operations in lines 4–12 of Alg. 4. Therefore, the estimate on the number N_{LE} of all transfer and directional interpolation matrices in (21) holds with $c_{\mathrm{LE}} = 2 \cdot \max(68, 24 \cdot 4^{c_{\mathrm{hf}}})$.

To prove the complexity statement we observe that every transfer matrix (13) has $(m+1)^6$ entries and every directional interpolation matrix (10) has at most $n_{\max}(m+1)^3 \leq c_{\max}(m+1)^6$ entries by assumption (16). Therefore, the linear complexity is a direct consequence of (21), if in addition (18) holds. □

Theorem 2. *Let assumption (17) hold true. Then there exists a constant c_{C} depending only on c_{un} and η_2, such that the number N_{C} of all coupling matrices $A_{c,t\times s}$ in Alg. 4 is bounded by*

$$N_{\mathrm{C}} \leq c_{\mathrm{C}}\left(p_{\min}(q_0\kappa)^3 + 8^{p_{\min}}\right). \qquad (24)$$

If in addition (18) and (19) hold true, these matrices can be stored and applied with complexity $\mathcal{O}(N\log(N))$. If (19) is replaced by the stronger assumption

$$\kappa q_0 \leq c\sqrt[3]{N/\log(N)}, \qquad (25)$$

then the complexity is reduced to $\mathcal{O}(N)$.

Proof. In this proof we pursue similar ideas as in [3, cf. proof of Lem. 8]. We assume that the depth of $\mathcal{T}_{T\times S}$ is not zero, because otherwise $N_{\mathrm{C}} \leq 1$ and the assertion is trivial. Our strategy is to estimate the numbers $N_{\mathrm{C}}^{(\ell)}$ of coupling matrices at all relevant levels $\ell = 1, \ldots, p_{\min}$.

In line 17 of Alg. 4 we see that the number of coupling matrices needed for a box $t \in \mathcal{T}_T^{(\ell)}$ is given by $\#\{s : (t,s) \in \mathcal{L}_{T\times S}^+\}$. For such blocks $(t,s) \in \mathcal{L}_{T\times S}^+$ the parent(s) is in the nearfield $\mathcal{N}(t_{\mathrm{p}})$ of $t_{\mathrm{p}} := \mathrm{parent}(t)$ by construction of the block tree in Alg. 2, where

$$\mathcal{N}(t_{\mathrm{p}}) := \{s_{\mathrm{p}} \in \mathcal{T}_S^{(\ell-1)} : s_{\mathrm{p}} \text{ and } t_{\mathrm{p}} \text{ violate (A1) or (A3)}\}.$$

Using this property and the uniformity of the box cluster trees we can estimate

$$N_{\mathrm{C}}^{(\ell)} = \sum_{t\in\mathcal{T}_T^{(\ell)}} \sum_{(t,s)\in\mathcal{L}_{S\times T}^+} 1 \le \sum_{t\in\mathcal{T}_T^{(\ell)}} 8 \cdot \#\mathcal{N}(\text{parent}(t)) \le 8^{\ell+1} N_{\mathcal{N},T}^{(\ell-1)}, \qquad (26)$$

where $N_{\mathcal{N},T}^{(\ell-1)}$ is an upper bound for the number of boxes in the nearfield of a box at level $\ell-1$ in $\mathcal{T}_T$ which we estimate in the following.

We cover the nearfield $\mathcal{N}(t)$ of a fixed box $t \in \mathcal{T}_T^{(\ell)}$ by a ball $B_{r_\ell}(m_t)$ with radius r_ℓ and center m_t and take the ratio of the volume of the ball and the one of a box to estimate $N_{\mathcal{N},T}^{(\ell)}$ for $\ell \ge 1$. We have to distinguish the cases of the two admissibility criteria (A1) and (A3). For this purpose, let $\tilde{\ell}$ be such that $\kappa q_j > 1$, if and only if $j \le \tilde{\ell}$. Such an $\tilde{\ell}$ exists since q_j decreases monotonically for increasing level j. In particular, we set $\tilde{\ell} = -1$, if $\kappa q_j \le 1$ for all $j \ge 0$. If $j \le \tilde{\ell}$ criterion (A3) implies (A1) as mentioned in Sect. 2.2. Vice versa, (A1) implies (A3) if $j > \tilde{\ell}$.

Let us first assume that $\ell \le \tilde{\ell}$ and consider an arbitrary box $s \in \mathcal{N}(t)$. Then t and s violate (A3), which means that $\eta_2 \operatorname{dist}(t,s) < \kappa q_\ell^2$, i.e. there exist $x \in \bar{t}$ and $y \in \bar{s}$ such that $|x-y| < \kappa q_\ell^2/\eta_2$. Hence, we can estimate

$$\begin{aligned} \max_{z\in\bar{s}}|z-m_t| &\le \max_{z\in\bar{s}}\left(|z-y|+|x-y|+|x-m_t|\right) \\ &\le q_\ell + \frac{\kappa q_\ell^2}{\eta_2} + \frac{q_\ell}{2} \le \left(\frac{3}{2}+\frac{1}{\eta_2}\right)\kappa q_\ell^2 =: r_\ell, \end{aligned} \qquad (27)$$

where we used $\kappa q_\ell > 1$ for the last estimate. Therefore, every box $s \in \mathcal{N}(t)$ is contained in the ball $B_{r_\ell}(m_t)$ with r_ℓ from (27). If instead $\ell > \tilde{\ell}$ we analogously show

$$\mathcal{N}(t) \subset B_{r_\ell}(m_t), \quad r_\ell = \left(\frac{3}{2}+\frac{1}{\eta_2}\right) q_\ell. \qquad (28)$$

With the ball $B_{r_\ell}(m_t)$ covering $\mathcal{N}(t)$ we can estimate

$$\#\mathcal{N}(t) \le \frac{|B_{r_\ell}(m_t)|}{v_\ell(\mathcal{T}_S)} = \frac{(4\pi/3)r_\ell^3}{3^{-3/2}q_\ell(\mathcal{T}_S)^3} = 4\pi\sqrt{3}\left(\frac{r_\ell}{q_\ell(\mathcal{T}_S)}\right)^3, \qquad (29)$$

where $v_\ell(\mathcal{T}_S) = 3^{-3/2}q_\ell(\mathcal{T}_S)^3$ denotes the volume of boxes $s \in \mathcal{T}_S^{(\ell)}$. Since $t \in \mathcal{T}_T^{(\ell)}$ was arbitrary, the bound in (29) holds also for $N_{\mathcal{N},T}^{(\ell)}$ instead of $\mathcal{N}(t)$.

Summarizing (26) and above findings, we get for the number N_{C} of all coupling matrices the estimate

$$\begin{aligned} N_{\mathrm{C}} &= \sum_{\ell=1}^{p_{\min}} N_{\mathrm{C}}^{(\ell)} \le \sum_{\ell=1}^{p_{\min}} 8^{\ell+1} N_{\mathcal{N},T}^{(\ell-1)} \le \sum_{\ell=1}^{p_{\min}} 8^{\ell+1} 4\pi\sqrt{3}\left(\frac{r_{\ell-1}}{q_{\ell-1}(\mathcal{T}_S)}\right)^3 \\ &\le 4\pi\sqrt{3}\left(\frac{3}{2}+\frac{1}{\eta_2}\right)^3 \left(\sum_{\ell=1}^{\tilde{\ell}+1} 8^{\ell+1}(\kappa c_{\mathrm{un}}q_\ell)^3 + \sum_{\ell=\tilde{\ell}+2}^{p_{\min}} 8^{\ell+1}c_{\mathrm{un}}^3\right) \\ &\le c_{\mathrm{C}}\left(\sum_{\ell=1}^{\tilde{\ell}+1}(\kappa q_0)^3 + 8^{p_{\min}}\right) \le c_{\mathrm{C}}(p_{\min}(\kappa q_0)^3 + 8^{p_{\min}}), \end{aligned} \qquad (30)$$

where we assumed that $1 \leq \tilde{\ell} + 1 < p_{\min}$ and used assumption (17) and the relation $q_0 = 2^{\ell} q_{\ell}$. If either $\tilde{\ell} + 1 \geq p_{\min}$ or $\tilde{\ell} = -1$, one can repeat the estimates in (30) and ends up with a similar result where one can cancel $8^{p_{\min}}$ in the first case and $p_{\min}(\kappa q_0)^3$ in the second case. The assertions about the complexity follow directly from (30) with assumptions (18) and (19) or (25), respectively, since every coupling matrix (9) has $(m+1)^6 = \mathcal{O}(1)$ entries.   □

In Thm. 2 we have estimated the number N_{C} of all coupling matrices, which corresponds to the number of admissible blocks $\mathcal{L}^+_{T \times S}$. As explained in Sect. 2.7, we store reoccuring matrices only once to reduce the related storage costs drastically as we will see in the next theorem and in Sect. 4. Since one needs to know all blocks in $\mathcal{L}^+_{T \times S}$ in Alg. 4 and storing them has complexity $\mathcal{O}(N_{\mathrm{C}})$, storing each matrix only once does not reduce the overall storage complexity of the method asymptotically.

Theorem 3. *Let the root boxes T and S of $\mathcal{T}_T$ and $\mathcal{T}_S$ be identical up to translation. Then the number N_{SC} of coupling matrices $A_{c,t \times s}$ which have to be stored can be estimated by*

$$N_{\mathrm{SC}} \leq p_{\min} \max\{c_{\mathrm{C}}, c_{\mathrm{C}}^{2/3}(\kappa q_0)^2\}. \tag{31}$$

If (18) and (19) hold, the corresponding storage complexity is $\mathcal{O}(N^{2/3} \log(N))$.

Proof. From the proof of Thm. 2, in particular (27), (28) and (29), it follows that the number of admissible blocks $(t, s) \in \mathcal{L}^+_{T \times S}$ for a fixed box $t \in \mathcal{T}^{\ell}_T$ can be estimated by

$$\begin{aligned} 8 \cdot \#\mathcal{N}(\mathrm{parent}(t)) &\leq 32\pi\sqrt{3}c_{\mathrm{un}}^3 \left(\frac{3}{2} + \frac{1}{\eta_2}\right)^3 \max\{1, (\kappa q_0)^3 8^{1-\ell}\} \\ &\leq c_{\mathrm{C}} \max\{1, (\kappa q_0)^3 8^{-\ell}\}, \end{aligned} \tag{32}$$

where we used $q_{\ell} = 2^{-\ell} q_0$, and c_{C} is the same constant as in (24). For a different box $t' \in \mathcal{T}^{\ell}_T$ the boxes s' such that $(t', s') \in \mathcal{L}^+_{T \times S}$ are identical to blocks (t, s) up to translation, which follows from the assumption on the root boxes T and S and the uniformity of the trees $\mathcal{T}_T$ and $\mathcal{T}_S$. Hence, the coupling matrices coincide and (32) is a bound for the number $N_{\mathrm{SC}}^{(\ell)}$ of stored coupling matrices at level ℓ. On the other hand, there are at most $8^{2\ell}$ blocks at level ℓ of $\mathcal{T}^+_{T \times S}$, which gives

$$N_{\mathrm{SC}}^{(\ell)} \leq \min\{8^{2\ell}, c_{\mathrm{C}} \max\{1, (\kappa q_0)^3 8^{-\ell}\}\}.$$

The maximum over all ℓ of the expression on the right-hand side is bounded by $8^{2\ell^*}$, where ℓ^* is the intersection point of $8^{2\ell}$ and $c_{\mathrm{C}} \max\{1, (\kappa q_0)^3 8^{-\ell}\}$. By computing this maximum we end up with the general bound

$$N_{\mathrm{SC}}^{(\ell)} \leq \max\left\{c_{\mathrm{C}}, c_{\mathrm{C}}^{2/3}(\kappa q_0)^2\right\} \quad \text{for all } \ell \geq 0.$$

Summation over all levels $\ell = 1, \ldots, p_{\min}$ yields (31). Since every coupling matrix has $(m+1)^6$ entries, it follows that all distinct coupling matrices can be stored with $\mathcal{O}(N^{2/3} \log(N))$ memory units, if assumptions (18) and (19) hold.   □

In Thm. 4, we will perform the complexity analysis of the nearfield evaluation, i.e. lines 29 and 30 of Alg. 4. In unbalanced trees there can be leaf clusters at coarse levels with large nearfields. If there were many of these, the complexity would not be linear. To exclude exceptional settings we make the additional assumption that the number of such leaf clusters is bounded, i.e., there exists a constant $c_{\mathrm{in}} \in \mathbb{N}$ such that

$$\#\mathcal{L}_T^- \leq c_{\mathrm{in}}, \quad \#\mathcal{L}_S^- \leq c_{\mathrm{in}}, \tag{33}$$

where $\mathcal{L}_S^- := \mathcal{L}_S \setminus \mathcal{L}_S^+$ and

$$\begin{aligned}
\mathcal{L}_S^+ := \{s \in \mathcal{L}_S \colon \#\{t : (t,s) \in \mathcal{L}_{T \times S}^-\} \leq c_{\mathrm{nf}} \text{ and} \\
\#\hat{t} \leq c_{\mathrm{m}} n_{\max} \text{ for all } (t,s) \in \mathcal{L}_{T \times S}^-\},
\end{aligned} \tag{34}$$

for some fixed parameters c_{m} and c_{nf}. It follows from (32) that for a leaf box $s \in \mathcal{T}_S^{(\ell)}$ the assumption $\#\{t : (t,s) \in \mathcal{L}_{T \times S}^-\} \leq c_{\mathrm{nf}}$ holds true for sufficiently large constant c_{nf} if $\ell = \mathrm{level}(t)$ is large enough.

Theorem 4. *Let assumptions* (17) *and* (33) *hold true. Then there exists a constant c_{D} depending only on c_{un}, c_{in}, c_{m}, c_{nf}, $n_{\max}$ and η_2 such that the number M_{D} of entries of all nearfield blocks $A|_{\hat{t} \times \hat{s}}$ in Alg. 4 is bounded by*

$$M_{\mathrm{D}} \leq c_{\mathrm{D}}(N_T + N_S + 8^{p_{\min}}). \tag{35}$$

If (18) *holds, the corresponding storage complexity is $\mathcal{O}(N)$.*

Proof. Each nearfield matrix block corresponds to an inadmissible block $(t,s) \in \mathcal{L}_{T \times S}^-$. For such a block there holds $t \in \mathcal{L}_T$ or $s \in \mathcal{L}_S$ by construction. We start counting entries of blocks corresponding to leaves in $\mathcal{L}_S$ by considering the sets $\mathcal{L}_S^+$ and $\mathcal{L}_S^-$.

For the number $M_{\mathrm{D},S}^-$ of nearfield matrix entries corresponding to blocks (t,s) with outlying leaves $s \in \mathcal{L}_S^-$ there holds

$$M_{\mathrm{D},S}^- = \sum_{s \in \mathcal{L}_S^-} \#\hat{s} \sum_{\{t : (t,s) \in \mathcal{L}_{T \times S}^-\}} \#\hat{t} \leq c_{\mathrm{in}} n_{\max} N_T. \tag{36}$$

Here we used that $\#\hat{s} \leq n_{\max}$ holds for all leaf boxes, and that the nearfield $\mathcal{N}(s) = \{t : (t,s) \in \mathcal{L}_{T \times S}^-\}$ of s can contain at most all N_T points in P_T.

Next we estimate the number $M_{\mathrm{D},S}^+$ of nearfield matrix entries corresponding to blocks (t,s) with $s \in \mathcal{L}_S^+$. For fixed $s \in \mathcal{L}_S^+$ there exist at most c_{nf} such blocks (t,s) and the corresponding boxes t contain maximally $c_{\mathrm{m}} n_{\max}$ points by definition of $\mathcal{L}_S^+$ in (34). Furthermore, the level of a box s in an inadmissible block (t,s) can be at most $p_{\min}$ and $\mathcal{T}_S$ can have at most $8^{p_{\min}}$ leaves at levels $\ell \leq p_{\min}$. Hence, we get

$$M_{\mathrm{D},S}^+ = \sum_{s \in \mathcal{L}_S^+} \#\hat{s} \sum_{t \in \mathcal{N}(s)} \#\hat{t} \leq 8^{p_{\min}} c_{\mathrm{nf}} c_{\mathrm{m}} n_{\max}^2. \tag{37}$$

Analogous estimates as (36) and (37) hold true for nearfield matrices corresponding to leaves in $\mathcal{L}_T$. Adding up all these estimates leads to the bound in (35), with the constant $c_\mathrm{D} = 2n_\mathrm{max} \max\{c_\mathrm{in}, c_\mathrm{nf}c_\mathrm{m}n_\mathrm{max}\}$. If (18) holds, the storage complexity $\mathcal{O}(N)$ is an immediate consequence of (35).   □

The following theorem summarizes the results of this section.

Theorem 5. *Let assumptions* (16)–(20) *and* (33) *hold true. Then the complexity of Alg. 4 is* $\mathcal{O}(N \log(N))$. *If* (19) *is replaced by* (25) *the complexity is reduced to* $\mathcal{O}(N)$.

4 Numerical Examples

In this section we want to test the method presented in Sect. 2 and to validate the theoretical results from Sect. 3. For this purpose we use a single core implementation of Alg. 4 in C++ on a computer with 384 GiB RAM and 2 Intel Xeon Gold 5218 CPUs. To reduce the required memory we store only the non-directional parts $E_{t'}$ of the transfer matrices and each coupling matrix once, as described in Sect. 2.7. However, if the matrix is applied several times it can be beneficial to store also the directional interpolation matrices $L_{t,c}$ and nearfield matrix blocks.

For the tests we consider points distributed uniformly inside the cube $[-1, 1]^3$. For various values $k \geq 3$ we choose $\tilde{x}_n = (2n - 1)2^{-k} - 1$ in $[-1, 1]$ for all $n \in \{1, ..., 2^k\}$ and construct the set of points $P_T(k) = \{x_j\}_{j=1}^{N(k)}$ with $N(k) = 8^k$ as tensor products of these one-dimensional points. We choose $P_S(k) = P_T(k)$ and consider the matrix A as in (1) with the wave number $\kappa = 0.1 \cdot 2^k$ and the diagonal set to zero to eliminate the singularities. The approximation derived in Sect. 2 is applicable despite the change of the diagonal because it effects only parts of the matrix which are evaluated directly.

We construct a uniform box cluster tree $\mathcal{T}_T$ for the set P_T using Alg. 1 with the initial box $T = [-1, 1]^3$ and the parameter $n_\mathrm{max} = 512$. With this choice of parameters and points, $\mathcal{T}_T$ is a uniform octree with depth $p(\mathcal{T}_T) = k - 3$, where every leaf contains exactly 512 points. We construct the sets of directions $D^{(\ell)}$ with Alg. 3 and the largest high frequency level $\ell_\mathrm{hf} = k - 4$ and finally we use Alg. 2 to construct the block cluster tree $\mathcal{T}_{T \times T}$ with the parameter $\eta_2 = 5$. The parameters ℓ_hf and η_2 were chosen according to the parameter choice rule in [17, Sect. 3.1.4]. In particular, the choice $\eta_2 = 5$ minimizes the number of inadmissible blocks $b \in \mathcal{L}_{T \times T}^-$ at levels $\ell > \ell_\mathrm{hf}$. Note that due to the uniformity of the tree $\mathcal{T}_T$ and the choice $P_S(k) = P_T(k)$ the block tree $\mathcal{T}_{T \times T}$ has depth $p(\mathcal{T}_T) = \ell_\mathrm{hf} + 1$ and all inadmissible blocks are at level $\ell_\mathrm{hf} + 1$.

The assumptions (16)–(20) are all satisfied for the considered examples for suitable constants c_max, c_un, c_ad, c_geo, and c_hf independent of the sets $P_T(k)$. Assumption (33) holds for $c_\mathrm{in} = 0$, because all leafs in $\mathcal{L}_T$ are at level $k - 3$ and by the choice of η_2 there holds $\#\{s : (t, s) \in \mathcal{L}_{T \times T}^-\} \leq 27$ for all leaves $t \in \mathcal{L}_T$.

In the described setting we apply Alg. 4 for the fast multiplication of the matrix A with a randomly constructed vector v. The interpolation degree $m = 4$ is

Table 1. Computation times and storage requirements for matrix-vector multiplications using Alg. 4 for the matrix A corresponding to sets of points $P_T(k)$ for various values of k. Parameters: $\ell_{\mathrm{hf}} = k - 4$, $\eta_2 = 5$, interpolation degree $m = 4$.

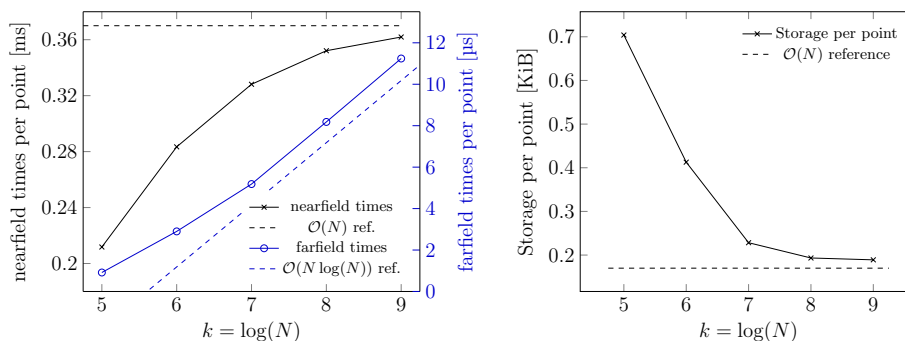| k	N	κ	t_{tot}	t_{s}	t_{nf}	t_{ff}	nf [%]	N_{SC}	N_{C}	[GiB]
5	32768	3.2	7.31	0.34	6.94	0.03	24.41	316	3096	0.02
6	262144	6.4	76.25	1.20	74.29	0.76	4.06	1522	166320	0.10
7	2097152	12.8	702.72	3.71	688.14	10.87	0.58	4554	2640960	0.46
8	16777216	25.6	6060.16	15.24	5907.66	137.26	0.077	9824	33103296	3.09
9	134217728	51.2	50204.00	118.89	48576.20	1508.91	0.010	32036	344979432	24.2

Fig. 1. Plots related to the computations of Table 1. Left image: Nearfield and farfield computation times per point with linear and quasi-linear reference curves in the different scales. Right image: Required storage per point with linear reference curve.

chosen, since it is reasonably high to yield a good approximation quality (e.g. relative error $2 \cdot 10^{-4}$ for $k = 6$) while it is low enough to make the approximations of all admissible blocks efficient.

The results of the computations for various sets of points $P_T(k)$ are given in Table 1 and Fig. 1. The total computational times t_{tot} are split into setup times t_{s}, times t_{nf} of the nearfield part, and computational times t_{ff} of the farfield part. In addition, the percentage of matrix entries in inadmissible blocks (nf), the numbers N_{SC} and N_{C} of stored and applied coupling matrices and the storage requirements ([GiB]) are given. A direct computation for $k = 7$ takes more than 32 hours. Thus the directional approximation is about 160 times faster. For larger examples the difference would be even more pronounced due to the quadratic complexity of the direct computation.

In Fig. 1, we plot computational times and memory consumption per point. As expected from our theoretical results of Sect. 3, we observe linear and almost linear behavior, respectively, for the nearfield and the farfield part of the computations, see the left plot in Fig. 1. As usual there is some preasymptotic behavior in such plots. The right plot in Fig. 1 shows the linear behavior of the memory requirements. Note that we store coupling matrices and transfer matrices only. In particular, we mention the low number N_{SC} of stored coupling matrices compared to the total number N_{C} of coupling matrices in Table 1.

References

1. Bebendorf, M., Kuske, C., Venn, R.: Wideband nested cross approximation for Helmholtz problems. Numer. Math. **130**(1), 1–34 (2015). https://doi.org/10.1007/s00211-014-0656-7
2. Bebendorf, M., Rjasanow, S.: Adaptive low-rank approximation of collocation matrices. Computing **70**(1), 1–24 (2003). https://doi.org/10.1007/s00607-002-1469-6
3. Börm, S.: Directional $\mathcal{H}^2$-matrix compression for high-frequency problems. Numer. Linear. Algebra. Appl. **24**(6), e2112 (2017). https://doi.org/10.1002/nla.2112
4. Börm, S., Börst, C.: Hybrid matrix compression for high-frequency problems (2018), preprint, arXiv:1809.04384
5. Börm, S., Melenk, J.M.: Approximation of the high-frequency Helmholtz kernel by nested directional interpolation: error analysis. Numer. Math. **137**(1), 1–34 (2017). https://doi.org/10.1007/s00211-017-0873-y
6. Brandt, A.: Multilevel computations of integral transforms and particle interactions with oscillatory kernels. Comput. Phys. Commun. **65**(1), 24 – 38 (1991). https://doi.org/10.1016/0010-4655(91)90151-A
7. Cheng, H., Crutchfield, W.Y., Gimbutas, Z., Greengard, L.F., Ethridge, J.F., Huang, J., Rokhlin, V., Yarvin, N., Zhao, J.: A wideband fast multipole method for the Helmholtz equation in three dimensions. J. Comput. Phys. **216**, 300–325 (2006). https://doi.org/10.1016/j.jcp.2005.12.001
8. Darve, E., Havé, P.: Efficient fast multipole method for low-frequency scattering. J. Comput. Phys. **197**(1), 341–363 (2004). https://doi.org/10.1016/j.jcp.2003.12.002
9. Engquist, B., Ying, L.: Fast directional multilevel algorithms for oscillatory kernels. SIAM J. Sci. Comput **29**(4), 1710–1737 (2007). https://doi.org/10.1137/07068583X
10. Greengard, L., Rokhlin, V.: A fast algorithm for particle simulations. J. Comput. Phys. **73**(2), 325–348 (1987). https://doi.org/10.1016/0021-9991(87)90140-9
11. Hackbusch, W.: Hierarchical matrices: algorithms and analysis, SSCM, vol. 49. Springer, Heidelberg (2015). https://doi.org/10.1007/978-3-662-47324-5
12. Hu, B., Chew, W.C.: Fast inhomogeneous plane wave algorithm for scattering from objects above the multilayered medium. IEEE Trans. Geosci. Remote Sens. **39**(5), 1028–1038 (2001). https://doi.org/10.1109/36.921421
13. Messner, M., Schanz, M., Darve, E.: Fast directional multilevel summation for oscillatory kernels based on Chebyshev interpolation. J. Comput. Phys. **231**(4), 1175–1196 (2012). https://doi.org/10.1016/j.jcp.2011.09.027
14. Nishimura, N.: Fast multipole accelerated boundary integral equation methods. Appl. Mech. Rev. **55**(4), 299–324 (2002). https://doi.org/10.1115/1.1482087
15. Rjasanow, S., Steinbach, O.: The Fast Solution of Boundary Integral Equations. Springer-Verlag, Berlin, Heidelberg (2007). https://doi.org/10.1007/0-387-34042-4
16. Rokhlin, V.: Diagonal forms of translation operators for Helmholtz equation in three dimensions. Appl. Comput. Harmon. A. **1**, 82–93 (1993). https://doi.org/10.1006/acha.1993.1006
17. Watschinger, R.: A directional approximation of the Helmholtz kernel and its application to fast matrix-vector multiplications. Master's thesis, Graz University of Technology, Insitute of Applied Mathematics (2019), `https://permalink.obvsg.at/AC15364438`

Erschienene Preprints ab Nummer 2018/3

2018/3   S. Dohr, M. Merta, G. Of, O. Steinbach, J. Zapletal: A parallel solver for a preconditioned space-time boundary element method for the heat equation

2018/4   S. Amstutz, P. Gangl: Toplogical derivative for nonlinear magnetostatic problem

2018/5   O. Steinbach, M. Zank: A Stabilized Space–Time Finite Element Method for the Wave Equation

2018/6   O. Steinbach, H. Yang: A Space-Time Finite Element Method for the Linear Bidomain Equations

2018/7   O. Steinbach, M. Zank: Coercive space-time finite element methods for initial boundary value problems

2018/8   S. Dohr, K. Niino, O. Steinbach: Space-time boundary element methods for the heat equation

2018/8   O. Steinbach, H. Yang: Space-time finite element methods for parabolic evolution equations: Discretization, a posteriori error estimation, adaptivity and solution

2019/1   O. Steinbach (eds.): 15th Austrian Numerical Analysis Day, Book of Abstracts

2019/2   P. Gangl, K. Sturm: A simplified derivation technique of topological derivatives for quasi-linear transmission problems

2019/3   M. Merkel, P. Gangl, S. Schöps: Shape Optimization of Rotating Electric Machines using Isogeometric Analysis and Harmonic Stator-Rotor Coupling

2019/4   P. Gangl and K. Sturm: Asymptotic analysis and topological derivative for 3D quasi-linear magnetostatics

2019/5   M. Holzmann and G. Unger: Boundary integral formulations of eigenvalue problems for elliptic differential operators with singular interactions and their numerical approximation by boundary element methods

2019/6   M. Neumüller, O. Steinbach: Regularization error estimates for distributed control problems in energy spaces

2019/7   U. Langer, M. Schanz, O. Steinbach, W. L. Wendland (eds.): 17th Workshop on Fast Boundary Element Methods in Industrial Applications , Book of Abstracts

2019/8   O. Steinbach, M. Zank: A note on the efficient evaluation of a modified Hilbert transformation

2019/9   P. Gangl: A multi-material topology optimization algorithm based on the topological derivative

2020/1   D. Pacheco, T. Müller, O. Steinbach, G. Brenn: A mixed finite element formulation for generalised Newtonian fluid flows with appropriate natural outflow boundary conditions

2020/2   U. Langer, O. Steinbach, F. Tröltzsch, H. Yang: Unstructured space-time finite element methods for optimal sparse control of parabolic equations

2020/3   D.R.Q. Pacheco, R. Schussnig, O. Steinbach, T.-P. Fries: A fully consistent equal-order finite element method for incompressible flow problems

2020/4   U. Langer, O. Steinbach, F. Tröltzsch, H. Yang: Unstructured space-time finite element methods for optimal control of parabolic equations

2020/5   P. Gangl, K. Sturm, M. Neunteufel, J. Schöberl: Fully and Semi-Automated Shape Differentiation in NGSolve

2020/6   U. Langer, O. Steinbach, F. Tröltzsch, H. Yang: Space-time finite element discretization of parabolic optimal control problems with energy regularization