Efficient Inversion of the Galerkin Matrix of General Second Order Elliptic Operators with Non-Smooth Coefficients

by

Mario Bebendorf

Preprint no.: 6 2004
Efficient Inversion of the Galerkin Matrix of General Second Order Elliptic Operators with Non-smooth Coefficients

Mario Bebendorf
Fakultät für Mathematik und Informatik
Universität Leipzig
Augustusplatz 10/11
D-04109 Leipzig, Germany
bebendorf@math.uni-leipzig.de
February 26, 2004

Mathematics Subject Classification (2000): 35C20, 65F05, 65F50, 65N30

Abstract

This article deals with the efficient (approximate) inversion of finite element stiffness matrices of general second order elliptic operators with $L^\infty$-coefficients. It will be shown that the inverse stiffness matrix can be approximated by hierarchical matrices ($H$-matrices). Furthermore, numerical results will demonstrate that it is possible to compute an approximate inverse with almost linear complexity.

1 Introduction

We are concerned with the numerical solution of large finite element systems of Dirichlet problems

$$Lu = f \quad \text{in } \Omega$$
$$u = g \quad \text{on } \partial \Omega$$

with second order elliptic operators

$$Lu = -\text{div} [A \nabla u + bu] + c \cdot \nabla u + du$$ (1.1)

on bounded Lipschitz domains $\Omega \subset \mathbb{R}^n$. The Galerkin matrix of such operators is sparse but has a bandwidth of order $N^{1-1/n}$, where $N$ is the number of degrees of freedom. Therefore, direct methods are well suited for small problem sizes, but are not competitive if $N$ is large. In the latter case usually iterative methods are more efficient. On the other hand, if the coefficient matrix is ill-conditioned these methods suffer from slow convergence.

The aim of this article is to show that hierarchical matrices ($H$-matrices) introduced by Hackbusch et al. [15, 16] can fill this gap. As we will see, they provide a means by which an approximation of the inverse stiffness matrix can be generated and handled with logarithmic-linear complexity. Furthermore, no grid hierarchy is required and $H$-matrices are robust in

*This work was supported by the DFG priority program SPP 1146 “Modellierung inkrementeller Umformverfahren”
the sense that their efficiency does not depend on the smoothness and only slightly on the size of the coefficients. The $\mathcal{H}$-approximant might be used directly to solve the finite element system or it might be used as a black-box preconditioner in an iterative scheme. If it is used as a preconditioner there is no need to approximate the inverse with high accuracy, and the complexity can therefore even be reduced. Another application of the inverse are Schur complements, which play a central role for example in domain decomposition methods.

The structure of $\mathcal{H}$-matrices was originally designed to efficiently represent integral operators with asymptotically smooth kernel. For this application the existence of $\mathcal{H}$-matrix approximants is well understood. Even efficient algorithms for the generation of the approximants exist, see [1, 3]. Close to the application of integral operators are inverses of elliptic operators since they have an integral representation with the Green function as kernel function:

$$(L^{-1}\varphi)(x) = \int_{\Omega} G(x,y)\varphi(y) \, dy \quad \text{for all } \varphi \in C^\infty_0(\Omega).$$

(1.2)

If $L$ has smooth coefficients, the Green function is smooth (except for $x = y$) and the mentioned existence theorems apply. However, the algorithms for building the approximants cannot be used, since they are either based on the matrix entries or on the kernel function, neither of which is accessible in general.

In this article the case of $L^\infty$-coefficients is treated. In this case it is not obvious that an $\mathcal{H}$-matrix approximant exists, since according to the De Georgi-Nash theorem (see [10]), $G$ is only locally Hölder continuous. Therefore, proofs cannot rely on the smoothness of the kernel function as they did for integral operators. In [4] we were able to show that the inverse stiffness matrix of the principal parts, i.e., $b = c = 0$ and $d = 0$, can be approximated by $\mathcal{H}$-matrices. In the present article this result will be extended to operators (1.1) with lower order terms without restrictions on their size. Furthermore, we will present numerical results that by an $\mathcal{H}$-matrix inversion based on the Frobenius formulas one is able to (approximately) invert the stiffness matrix with logarithmic-linear complexity.

The structure of the rest of this article is as follows: In Section 2 a brief review of the structure of $\mathcal{H}$-matrices will be given. All necessary results and notations for the theorems of this article from the field of $\mathcal{H}$-matrices will be presented. Section 3 contains the existence theory of degenerate kernel approximants of the Green function, i.e.,

$$G(x,y) \approx \sum_{i=1}^{k} u_i(x) v_i(y), \quad x \in D_1 \text{ and } y \in D_2,$$

on an appropriate pair of domains $(D_1, D_2)$. The usual way to proof existence of a degenerate kernel approximant is to exploit the smoothness of the kernel function. In the case of $L^\infty$-coefficients the Green function of the inverse differential operator is not smooth. Therefore, another technique, which is based on the interior regularity of elliptic problems, has to be used. We will show that as long as $L$ is invertible the Green function can be approximated by a degenerate kernel even in the case of dominating lower-order terms. From the numerical results it will be seen that these terms enter the constants only in a moderate way.

This result is then employed using (1.2) to show that the discrete inverse of $L$ can be approximated by $\mathcal{H}$-matrices, which in turn leads to the existence of $\mathcal{H}$-matrix approximants to the inverse stiffness matrix. In Section 4 it is described in detail how to implement an efficient $\mathcal{H}$-matrix inversion. The presented $\mathcal{H}$-inversion is based on the Frobenius formulas and is used to produce numerical results for operators with non-smooth coefficients. We will see that this algorithm is able to compute an approximate inverse with almost linear complexity.
2 Hierarchical matrices

This section gives a brief overview over the structure of $\mathcal{H}$-matrices originally introduced by Hackbusch et al. [15, 16]. We will describe the two principles on which the efficiency of $\mathcal{H}$-matrices is based. These are the hierarchical partitioning of the matrix into blocks and the blockwise restriction to low-rank matrices. These principles were also used in the mosaic-skeleton method [19].

In contrast to other efficient methods like wavelet techniques [6, 7, 8], fast multipole and panel clustering, see [13], [17] and the references therein, $\mathcal{H}$-matrices concentrate on the matrix-level. They are purely algebraic in the sense that once the $\mathcal{H}$-matrix approximant is built, no further information about the underlying problem is needed.

Let us assume that $M \in \mathbb{R}^{N \times N}$ has indices

$$m_{ij} = a(\varphi_j, \varphi_i),$$

where $\varphi_i$ are basis functions with support $X_i := \text{supp} \varphi_i$, $i \in I := \{1, \ldots, N\}$ and $a$ is a bilinear form. In this section we assume that there is a partition $P$ of the indices $I \times I$ of $M$ such that each block $b = s \times t$, $s, t \subset I$, can be approximated by a matrix of low-rank, i.e.,

$$M_b \approx UV^T, \quad U \in \mathbb{R}^{s \times k}, V \in \mathbb{R}^{t \times k},$$

where $k$ is small compared with $|s|$ and $|t|$. Obviously, by $M_b$ we denote the subblock in the intersection of the rows $s$ and columns $t$ of $M$. From Example 2.4 it will be seen that the stiffness matrix of operators of type (1.1) possesses this property. Section 3 will extend this to the inverse stiffness matrix.

2.1 The cluster tree

In order to exploit the fact that there is a partition such that each block can be approximated by a matrix of low rank, we first have to find it from the set of possible subsets of $I \times I$. This set however is too large to be searched for a partition that will satisfy our needs. Therefore, the set of subsets $b = s \times t$ is restricted to those which consist of index sets $s$ and $t$ stemming from a cluster tree $T_I$. A tree $T_I$ satisfying the following conditions is called a cluster tree:

1. $I$ is the root of $T_I$

2. if $t \in T_I$ is not a leaf, then $t$ has sons $t_1, t_2 \in T_I$, so that $t = t_1 \cup t_2$.

The set of sons of $t$ is denoted by $S(t)$, while $L(T_I)$ stands for the set of leaves of the tree $T_I$. The support of a cluster $t$ is the union of the supports of the basis functions corresponding to the indices in $t$:

$$X_t := \bigcup_{i \in t} X_i.$$

A cluster tree is usually generated by recursive subdivision of $I$ so as to minimize the diameter of each part. For practical purposes the recursion should be stopped if a certain cardinality $n_{\text{min}}$ of the clusters is reached, rather than subdividing the clusters until only one index is left. The depth of $T_I$ will be denoted by $p$. For reasonable cluster trees one would always expect $p = \mathcal{O}(\log N)$. A strategy based on the principle component analysis is used in [2]. The complexity of building the cluster tree in the case of quasi-uniform grids can be estimated as $\mathcal{O}(N \log N)$. 

3
Remark 2.1 Sometimes, the number of sons of a cluster in the previous definition of a cluster tree is not restricted to two. However, this generalization has not proved useful in practice.

2.2 Admissibility condition

In order to be able to approximate each block $b$ of $M$ by a low-rank matrix, $b$ has to satisfy a certain condition. This so-called admissibility condition will be the criterion for choosing whether $b$ belongs to $P$. In the field of elliptic partial differential equations the following condition on $b=s \times t$ has proved useful:

$$\min\{\text{diam } X_s, \text{diam } X_t\} \leq \eta \text{ dist}(X_s, X_t),$$  \hspace{1cm} (2.4)

where $\eta > 0$ is a given real number. We will see that under quite general assumptions this condition allows to approximate the Green function of $L$ by a degenerate kernel, i.e., there are functions $u_i, v_i, i=1, \ldots, k$, so that

$$G(x, y) \approx \sum_{i=1}^{k} u_i(x)v_i(y) \text{ in } X_s \times X_t,$$  \hspace{1cm} (2.5)

where $k$ depends only logarithmically on $N$. Since by (2.3) the entries of $b$ depend only on the values of $a$ on the domain $X_s \times X_t$, the degenerate approximation of $G$ on $X_s \times X_t$ will finally lead to a low-rank approximation of the block $b$.

Condition (2.4) was also used to prove convergence of the ACA algorithm for the efficient generation of $\mathcal{H}$-matrix approximants in the case of integral equations, cf. [1, 3].

Remark 2.2 In the case of unstructured grids the computation of the distance in (2.4) between two supports $X_s$ and $X_t$ is too costly. Therefore, for practical purposes the supports are enclosing into sets of a simpler structure, e.g. boxes or spheres.

2.3 Block cluster tree

Based on a cluster tree $T_I$ which contains a hierarchy of partitions of $I$, we are able to construct the so-called block cluster tree $T_{I \times I}$ describing a hierarchy of partitions of $I \times I$ by the following rule:

procedure build_block_cluster_tree($s \times t$)
begin
  if $(s, t)$ does not satisfy (2.4) and $s, t \notin \mathcal{L}(T_I)$ then begin
    $S(s \times t) := \{s' \times t' : s' \in S(s), t' \in S(t)\}$
    for $s' \times t' \in S(s \times t)$ do build\_block\_cluster\_tree($s' \times t'$)
  end
  else $S(s \times t) := \emptyset$
end

Applying $\text{build\_block\_cluster\_tree}$ to $I \times I$ we obtain a cluster tree for the index set $I \times I$. The set of leaves $P := \mathcal{L}(T_{I \times I})$ is a partition of $I \times I$ with blocks $b = s \times t \in P$ either satisfying (2.4) or consisting of clusters $t$ and $s$ one of which is a leaf in $T_I$. The complexity of building the block cluster tree in the case of quasi-uniform grids can be estimated as $O(\eta^{-n}N \log N)$, cf. [2].
We are now in a position to define the set of $\mathcal{H}$-matrices for a partition $P$ with blockwise rank $k$

$$\mathcal{H}(P, k) := \{ M \in \mathbb{R}^{I \times I} : \text{rank } M_b \leq k \text{ for all } b \in P \}.$$  

Note that $\mathcal{H}(P, k)$ is not a linear space, since in general the sum of two rank-$k$ matrices exceeds rank $k$.

**Remark 2.3** For a block $B \in \mathbb{R}^{s \times t}$ the low-rank representation $B = UV^T$, $U \in \mathbb{R}^{s \times k}$, $V \in \mathbb{R}^{t \times k}$, is only advantageous compared with the entrywise representation, if $k(|s| + |t|) \leq |s||t|$. For the sake of simplicity in this article we will however assume that each block has the low-rank representation. Employing the entrywise representation for appropriate blocks will accelerate the algorithms.

**Example 2.4** The stiffness matrix $S$ of the differential operator $L$ from (1.1) is in $\mathcal{H}(P, n_{\min})$. If $b \in P$ satisfies (2.4) then the supports of the basis functions are pairwise disjoint. Hence, the matrix entries in this block vanish. In the remaining case $b$ does not satisfy (2.4). Then the size of one of the clusters is less or equal $n_{\min}$. In both cases the rank of $S_b$ does not exceed $n_{\min}$.

### 2.4 Storage and matrix-vector multiplication

The cost of a multiplication of an $\mathcal{H}$-matrix $M \in \mathcal{H}(P, k)$ and its transposed $M^T$ by a vector $x \in \mathbb{R}^N$ is inherited from the blockwise matrix-vector multiplication:

$$Mx = \sum_{s \times t \in P} M_{s \times t} x_t$$

and

$$M^T x = \sum_{s \times t \in P} (M_{s \times t})^T x_s.$$

Since each block $s \times t$ has the representation $M_{s \times t} = UV^T$, $U \in \mathbb{R}^{s \times k}$, $V \in \mathbb{R}^{t \times k}$ (see Remark 2.3), $O(k(|s| + |t|))$ units of memory are needed to store $M_{s \times t}$ and the matrix-vector products

$$M_{s \times t} x_t = UV^T x_t$$

and

$$(M_{s \times t})^T x_s = VU^T x_s$$

can be done in $O(k(|s| + |t|))$ operations. Exploiting the hierarchical structure of $M$ it can therefore be shown that both storing $M$ and multiplying $M$ and $M^T$ by a vector has $O(\eta^{-n}kN \log N)$ complexity. For a rigorous analysis the reader is referred to [2]. Therefore, $\mathcal{H}$-matrices are well suited for iterative schemes such as Krylov subspace methods.

### 3 Approximation of FE inverses

In Example 2.4 it was mentioned that the stiffness matrix of a general elliptic operator with $L^\infty$-coefficients can be represented as an $\mathcal{H}$-matrix. In this section it will be proved that moreover its inverse can be approximated by an $\mathcal{H}$-matrix. For this purpose it will first be shown that the Green function of $L$ and the bounded Lipschitz domain $\Omega \subset \mathbb{R}^n$ can be approximated on a pair $D_1 \times D_2$ of domains satisfying

$$\text{diam } D_2 \leq \eta \text{ dist} (D_1, D_2).$$

Since we consider Dirichlet problems we assume that $L : H_0^1(\Omega) \rightarrow H^{-1}(\Omega)$ is an invertible second order partial differential operator

$$Lu = -\text{div} [A \nabla u + bu] + c \cdot \nabla u + du,$$  

(3.6)
where $A(x) \in \mathbb{R}^{n \times n}$ is symmetric with entries $a_{ij} \in L^\infty(\Omega)$ and

$$0 < \lambda \leq \lambda(x) \leq \Lambda$$

for all eigenvalues $\lambda(x)$ of $A(x)$ and almost all $x \in \Omega$. The bound $\Lambda/\lambda$ on the condition numbers of $A$ will be denoted by $\kappa := \Lambda/\lambda$. Furthermore, let $b(x), c(x) \in \mathbb{R}^n$ and $d(x) \in \mathbb{R}$ with $b_i, c_i, d \in L^\infty(\Omega)$, $i = 1, \ldots, n$.

### 3.1 Degenerate approximation of the Green function

In [4] we have investigated the principal part $L_0$ of such operators, i.e., operators with $b = c = 0$ and $d = 0$. For these operators it is shown in [12] that in the case $n \geq 3$ a Green function $G_0 : \Omega \times \Omega \to \mathbb{R}$ with the following properties exists:

$$(3.8a) \quad G_0(x, \cdot) \in H^1(\Omega \setminus B_r(x)) \cap W_0^{1,1}(\Omega) \quad \text{for all } x \in \Omega \text{ and all } r > 0,$$

$$(3.8b) \quad a(G_0(x, \cdot), \varphi) = \varphi(x) \quad \text{for all } \varphi \in C_0^\infty(\Omega) \text{ and } x \in \Omega,$$

where $B_r(x)$ is the open ball centered at $x$ with radius $r$ and

$$a(u, v) = \int_\Omega \nabla v \cdot A \nabla u \, dx.$$  

Furthermore, for $x, y \in \Omega$ it holds that

$$|G_0(x, y)| \leq \frac{c_n(\kappa)}{\lambda}|x - y|^{2-n}. \quad (3.10)$$

In the case of invertible operators $L$ of type (3.6) a Green function $G := (L_0^{-1}L)^{-1}G_0$ satisfying (3.8), where

$$a(u, v) = \int_\Omega \nabla v \cdot A \nabla u \, dx + \int_\Omega \nabla v \cdot b u \, dx + \int_\Omega c \cdot \nabla v \, dx + \int_\Omega d v \, dx, \quad (3.11)$$

can be defined. Notice that

$$G - G_0 = [(L_0^{-1}L)^{-1} - I]G_0 = -L^{-1}L_1G_0,$$

where $L_1 := L - L_0$ is the lower order part of $L$. Since $L^{-1}L_1$ is an operator of order $-1$, $L^{-1}L_1G_0$ is smoother than $G_0$. Hence, the singularity of $G_0$ at $x = y$ is carried over to $G$ and we may assume that there is a constant $c_n$ such that for $x, y \in \Omega$ it holds that

$$|G(x, y)| \leq \frac{c_n(\kappa, b, c, d)}{\lambda}|x - y|^{2-n}. \quad (3.12)$$

In the case $n = 2$ the existence of a Green function for operators of type (3.6) has been proved more rigorously in [9]. In this case instead of (3.10) for $x, y \in \Omega$ one has the following bound on the Green function

$$|G(x, y)| \leq \frac{c(\kappa, b, c, d)}{\lambda}\log|x - y|. \quad (3.13)$$

We will make use of the following characteristic relation between $L^{-1}$ and $G$, which is equivalent to (3.8b):

$$(L^{-1}\varphi)(x) = \int_\Omega G(x, y)\varphi(y) \, dy \quad \text{for all } \varphi \in C_0^\infty(\Omega). \quad (3.14)$$

In the sequel $D \subset \mathbb{R}^n$ is a domain. The proof of the following basic lemma is mainly based on the Poincaré inequality, cf. [5], and can be found in [4].

6
**Lemma 3.1** Let $D$ be convex and $X$ a closed subspace of $L^2(D)$. Then for any $k \in \mathbb{N}$ there is a subspace $V_k \subset X$ satisfying $\dim V_k \leq k$ so that

$$\text{dist}_{L^2(D)}(u, V_k) \leq c_A \frac{\text{diam } D}{\sqrt{k}} \|u\|_{L^2(D)}$$

(3.15)

for each $u \in X \cap H^1(D)$, where $c_A$ depends only on the spatial dimension $n$.

The following set will be used in Lemma 3.1 as $X$:

$$X(D) = \{u \in H^1_{loc}(D) : a(u, \varphi) = 0 \ \forall \varphi \in C^\infty_0(D) \text{ and } u|_{D'\Omega} = 0\},$$

(3.16)

where $a(\cdot, \cdot)$ is the bilinear form defined in (3.11). Hence, the set $X(D)$ consists of $L$-harmonic $H^1_{loc}$-functions vanishing outside of $\Omega$. A proof for the fact that $X(D)$ is a closed subspace of $L^2(D)$ can be found in [4, Lemma 2.2]. We remark that the extension of $G(x, \cdot), x \in \Omega$, to $\mathbb{R}^n$ by zero is in $X(D)$ for all $D \subset \mathbb{R}^n$ satisfying $\text{dist}(x, D) > 0$.

By the following Caccioppoli inequality we are able to estimate the gradient on a compact subset by the norm of the function on an enclosing domain. This inequality provides a means to overcome the lack of regularity of $G$.

**Lemma 3.2** Let $K \subset D$. There is $c_R = c_R(\kappa, \lambda, b, c, d)$ such that

$$\|\nabla u\|_{L^2(K)} \leq \frac{c_R}{\text{dist}(K, \partial D)} \|u\|_{L^2(D)}$$

(3.17)

for all $u \in X(D)$.

**Proof.** Let $\eta \in C^1(D)$ satisfy $0 \leq \eta \leq 1$, $\eta = 1$ in $K$, $\eta = 0$ in a neighborhood of $\partial D$ and $|\nabla \eta| \leq 2/\delta$ in $D$, where we set $\delta = \text{dist}(K, \partial D)$. Since $K' := \text{supp } \eta \subset \subset D$, the definition (3.16) of $X(D)$ implies $u \in H^1(K')$. Hence, $\varphi := \eta^2 u \in H^1_0(D)$ may be used as a test function in $a(u, \varphi) = 0$ due to the dense embedding of $C^\infty_0(D)$ in $H^1_0(D)$:

$$- \int_D (\nabla^2 u) \cdot b \ dx = - \int_D \nabla \cdot (\eta^2 u) \cdot \nabla u \ dx - \int_D \eta^2 |u|^2 \ dx = \int_D (\nabla \eta)^2 u \cdot A \nabla u \ dx$$

$$= \int_D \eta \nabla (\nabla \eta) \cdot A \nabla u \ dx + \int_D \eta^2 \nabla^2 u \cdot A \nabla u \ dx.$$ 

Hence,

$$\int_D \eta^2 |A^{1/2} \nabla u|^2 \ dx = -2 \int_D \eta \nabla (\nabla \eta) \cdot A \nabla u \ dx - 2 \int_D \eta (\nabla \eta) \cdot b |u|^2 \ dx$$

$$- \int_D \eta^2 \nabla (\nabla u) \cdot b \ u \ dx - \int_D \eta^2 \nabla u \cdot \nabla (\nabla u) \ dx - \int_D \eta^2 |u|^2 \ dx.$$ 

For the first integral on the right hand side of the last equation we obtain

$$\left| \int_D \eta \nabla (\nabla \eta) \cdot A \nabla u \ dx \right| \leq \int_D |A^{1/2} \nabla \eta| \cdot |\eta A^{1/2} \nabla u| |u| \ dx$$

$$\leq 2 \sqrt{\frac{\lambda}{\delta}} \|\eta A^{1/2} \nabla u\|_{L^2(D)} \|u\|_{L^2(D)}.$$
The third integral can be estimated as follows

\[
\left| \int_D \eta^2 \langle \nabla u \cdot \mathbf{b} \rangle \, dx \right| \leq \int_D \eta |\eta| |\nabla u| |u| \, dx \leq \|b\|_\infty \int_D |\nabla u| |u| \, dx
\]

\[
\leq \|b\|_\infty \left( \int_D \eta^2 |\nabla u|^2 \, dx \right)^{1/2} \|u\|_{L^2(D)}
\]

\[
\leq \frac{\|b\|_\infty}{\sqrt{\lambda}} \|\eta A^{1/2} \nabla u\|_{L^2(D)} \|u\|_{L^2(D)}
\]

and similarly one has for the forth integral

\[
\left| \int_D \eta^2 u \cdot \nabla u \, dx \right| \leq \frac{\|c\|_\infty}{\sqrt{\lambda}} \|\eta A^{1/2} \nabla u\|_{L^2(D)} \|u\|_{L^2(D)}
\]

Since

\[
2 \|\eta A^{1/2} \nabla u\|_{L^2(D)} \|u\|_{L^2(D)} \leq \frac{1}{\epsilon} \|\eta A^{1/2} \nabla u\|_{L^2(D)}^2 + \epsilon \|u\|_{L^2(D)}^2
\]

with \( \epsilon := 4\sqrt{\lambda}/\delta + \lambda^{-1/2} \|b\| + \|c\|_\infty \) we end up with

\[
\|\eta A^{1/2} \nabla u\|_{L^2(D)}^2 \leq 2 \left( \frac{\epsilon^2}{2} + \frac{4}{\delta} \|b\|_\infty + d \|u\|_{L^2(D)} \right) \|u\|_{L^2(D)}^2.
\]

From \( \delta \leq \text{diam } \Omega \) one gets

\[
\|\eta A^{1/2} \nabla u\|_{L^2(D)} \leq \frac{c}{\delta} \|u\|_{L^2(D)},
\]

where \( c^2 = 16\Lambda + \text{diam } D(\lambda^{-1/2} \|b\| + \|c\|_\infty + d \|u\|_{L^\infty}) + 8\text{diam } D \|b\|_{L^\infty} \), and hence

\[
\|\nabla u\|_{L^2(K)} \leq \|\eta \nabla u\|_{L^2(D)} \leq \lambda^{-1/2} \|\eta A^{1/2} \nabla u\|_{L^2(D)} \leq \frac{c}{\lambda^{1/2} \delta} \|u\|_{L^2(D)}.
\]

\[\blacksquare\]

**Remark 3.3** From the previous proof it can be seen that \( d \) does not enter estimate (3.17) if \( d \geq 0 \).

**Lemma 3.4** Assume that \( D_2 \subset D \) is a convex domain such that for some \( \eta > 0 \) it holds that

\[
0 < \text{diam } D_2 \leq \eta \text{dist}(D_2, \partial D).
\]

Then for any \( \epsilon > 0 \) there is a subspace \( W \subset X(D_2) \) so that

\[
\text{dist}_{L^2(D_2)}(u, W) \leq \epsilon \|u\|_{L^2(D)} \quad \text{for all } u \in X(D)
\]

and \( \dim W \leq c^n \| |\log \epsilon|\|^{n+1} + |\log \epsilon| \) where \( c^n = c_A c_R \epsilon(2 + \eta) \).

**Proof.** Let \( \ell := |\log \epsilon| \). We consider a nested sequence of convex domains

\[
K_j = \{ x \in \mathbb{R}^n : \text{dist}(x, D_2) \leq r_j \}
\]

with real numbers \( r_j := (1 - j/\ell)\text{dist}(D_2, \partial D) \), \( j = 0, \ldots, \ell \). Notice that \( D_2 = K_\ell \subset K_{\ell-1} \subset \ldots \subset K_0 \subset D \). Using the definition (3.16) of the space \( X \) we set \( X_j := X(K_j) \).
Applying Lemma 3.1 to $K_j$ with the choice $k := [(c_AC_2(2+\eta)\ell^{-1/\ell})^n]$ we can find a subspace $V_j \subset X_j$ satisfying $\dim V_j \leq k$ and

$$\text{dist}_{L^2(K_j)}(v, V_j) \leq c_A \frac{\text{diam } K_j}{\sqrt{k}} \|\nabla v\|_{L^2(K_j)}$$

(3.19)

for all $v \in X_j \cap H^1(K_j)$. From Lemma 3.2 applied to $(K_j, K_j^{-1})$ instead of $(K, D)$ we obtain

$$\|\nabla v\|_{L^2(K_j^{-1})} \leq \frac{c_R}{\text{dist}(K_j, \partial K_{j^{-1}})} \|v\|_{L^2(K_j^{-1})} = c_R \frac{\ell}{r_0} \|v\|_{L^2(K_j^{-1})}$$

(3.20)

for all $v \in X_{j^{-1}}$. Since any $v \in X_{j^{-1}}$ also belongs to $X_j \cap H^1(K_j)$, the estimates (3.19) and (3.20) together with $\dim K_j \leq (2 + \eta)r_0$ may be combined to

$$\text{dist}_{L^2(K_j)}(v, V_j) \leq \varepsilon^{1/\ell} \|v\|_{L^2(K_j^{-1})} \quad \text{for all } v \in X_{j^{-1}}.$$  

(3.21)

Let $u \in X(D)$ and $v_0 := u|_{K_0} \in X_0$. By the last estimate we have $v_0|_{K_1} = u_1 + v_1$ with $u_1 \in V_1$ and

$$\|v_1\|_{L^2(K_1)} \leq \varepsilon^{1/\ell} \|v_0\|_{L^2(K_0)}.$$  

Consequently, $v_1$ belongs to $X_1$. Similarly, for all $j = 1, \ldots, \ell$ we are able to find an approximant $u_j \in V_j$ so that $v_{j-1}|_{K_j} = u_j + v_j$ and $\|v_j\|_{L^2(K_j)} \leq \varepsilon^{1/\ell} \|v_{j-1}\|_{L^2(K_{j-1})}$. Using the restrictions of $V_j$ to the smallest domain $D_2 = K_\ell$ let

$$W := \text{span}\{V_j|_{D_2}, \ j = 1, \ldots, \ell\}.$$  

Then $W$ is a subspace of $X(D_2)$ and since $v_0|_{D_2} = v_\ell + \sum_{j=1}^{\ell} u_j|_{D_2}$ we are led to

$$\text{dist}_{L^2(D_2)}(v_0, W) \leq \|v_\ell\|_{L^2(D_2)} \leq \left(\varepsilon^{1/\ell}\right)^\ell \|v_0\|_{L^2(K_0)} \leq \varepsilon \|u\|_{L^2(D)},$$

where the last inequality is due to $K_0 \subset D$.

The dimension of $W$ is bounded by $\sum_{j=1}^{\ell} \dim V_j \leq \ell k$. Since $\varepsilon^{-1/\ell} \leq \varepsilon$ we obtain $\dim W \leq (c_A c_R^2(2 + \eta))^{n+1} + \ell$.\hfill \blackqed

The previous lemma will now be applied to the Green functions $G(x, \cdot)$ with $x \in D_1 \subset \Omega$. For this purpose let $g_x$ be the extension of $G(x, \cdot)$ to $\mathbb{R}^n \setminus \overline{D_1}$, i.e.,

$$g_x(y) := \begin{cases} G(x, y), & y \in \Omega \setminus \overline{D_1} \\ 0, & y \in \mathbb{R}^n \setminus \Omega \end{cases}.$$  

(3.22)

Then $g_x$ is in $X(\mathbb{R}^n \setminus \overline{D_1})$. Note that its approximant $G_k(x, \cdot)$ from the following theorem is of the desired form (2.5).

**Theorem 3.5** Let $D_1 \subset \Omega$ and $D_2 \subset \mathbb{R}^n$ convex. Assume that there is $\eta > 0$ such that

$$0 < \text{diam } D_2 \leq \eta \text{dist}(D_1, D_2).$$

Then for any $\varepsilon > 0$ there is a separable approximation

$$G_k(x, y) = \sum_{i=1}^k u_i(x)v_i(y) \quad \text{with } k \leq k_\varepsilon := c_\eta \|\log \varepsilon\|_{n+1} + \|\log \varepsilon\|,$$
we have triangulation. Hence, the step size $h$ satisfies estimate (3.23).

In agreement with the assumptions of Section 2 let $N = \dim V_h$ be the dimension and \{\varphi_i\}_{i \in I}$ a basis of $V_h$, where $I := \{1, \ldots, N\}$ is used as an index set. The notation for the support of the finite element basis function is generalized to subsets $t \subset I$ as follows:

$$X_i := \text{supp} \varphi_i \subset \Omega \quad \text{for } i \in I, \quad X_t := \bigcup_{i \in t} X_i \quad \text{for } t \subset I.$$  \hfill (3.25)

In order to avoid technical complications, we consider a quasi-uniform and shape-regular triangulation. Hence, the step size $h := \max_{i \in I} \text{diam } X_i$ fulfills

$$\text{vol } X_i \geq c_i h^n.$$  \hfill (3.26)

The supports $X_i$ may overlap. In accordance with the standard finite element discretization we require that each triangle belongs to the support of a bounded number of basis functions, i.e., there is a constant $c_M > 0$ so that

$$c_M \text{vol } X_t \geq \sum_{i \in t} \text{vol } X_i.$$  \hfill (3.27)

Proof. Let $D := \{y \in \mathbb{R}^n : 2\eta \text{dist}(y, D_2) < \text{diam } D_2\}$. Note that because of $\text{dist}(D_1, D) > 0$, we have $g_x \in X(D)$ for all $x \in D_1$. Since in addition $\text{diam } D_2 \leq 2\eta \text{dist}(D_2, \partial D)$, Lemma 3.4 can be applied with $\eta$ replaced by $2\eta$. Let \{\varphi_1, \ldots, \varphi_k\} be a basis of the subspace $W \subset X(D_2)$ with $k = \dim W \leq c_{2\eta}^2 \|\log \varepsilon\|_{H^1} + \|\log \varepsilon\|$. By means of (3.18) $g_x$ can be decomposed into $g_x = \hat{g}_x + r_x$ with $\hat{g}_x \in W$ and $\|r_x\|_{L^2(D_2)} \leq \varepsilon \|\hat{g}_x\|_{L^2(D_2)}$. Since $g_x$ and $\hat{g}_x$ vanish outside of $\Omega$, we actually have $\|r_x\|_{L^2(D_2)} \leq \varepsilon \|G(x, \cdot)\|_{L^2(D_2)}$. Expressing $\hat{g}_x$ by means of the basis of $W$, we obtain

$$\hat{g}_x = \sum_{i=1}^k u_i(x) \varphi_i$$

with coefficients $u_i(x)$ depending on the index $x \in D_1$. The function $G_k(x, y) := \sum_{i=1}^k u_i(x) \varphi_i(y)$ satisfies estimate (3.23).

Note that since the constant $c_A$ does not depend on $\Omega$, the geometry enters $c_\eta$ only through the diameter. Hence, the shape of the domain does not influence our approximation result.

The existence of degenerate approximants to the Green function will now be used to prove existence of $H$-matrix approximants to the discrete inverse of $L$ and the inverse stiffness matrix.

### 3.2 $H(P,k)$-Approximation of discrete operators

Using a finite element discretization, $H^1_0(\Omega)$ is approximated by $V_h \subset H^1_0(\Omega)$, i.e., for all $v \in H^1_0(\Omega)$

$$\inf_{v_h \in V_h} \|v - v_h\|_{H^1} \to 0 \quad \text{for } h \to 0.$$  \hfill (3.24)

In agreement with the assumptions of Section 2 let $N = \dim V_h$ be the dimension and \{\varphi_i\}_{i \in I}$ a basis of $V_h$, where $I := \{1, \ldots, N\}$ is used as an index set. The notation for the support of the finite element basis function is generalized to subsets $t \subset I$ as follows:

$$X_i := \text{supp } \varphi_i \subset \Omega \quad \text{for } i \in I, \quad X_t := \bigcup_{i \in t} X_i \quad \text{for } t \subset I.$$  \hfill (3.25)

In order to avoid technical complications, we consider a quasi-uniform and shape-regular triangulation. Hence, the step size $h := \max_{i \in I} \text{diam } X_i$ fulfills

$$\text{vol } X_i \geq c_i h^n.$$  \hfill (3.26)

The supports $X_i$ may overlap. In accordance with the standard finite element discretization we require that each triangle belongs to the support of a bounded number of basis functions, i.e., there is a constant $c_M > 0$ so that

$$c_M \text{vol } X_t \geq \sum_{i \in t} \text{vol } X_i.$$  \hfill (3.27)
We use the notation $J$ for the natural bijection $J : \mathbb{R}^N \to V_h$ defined by $Jx = \sum_{i \in I} x_i \varphi_i$. For quasi-uniform and shape-regular triangulations it is known (see [14, Theorem 8.8.1]) that there are constants $0 < c_{1,1} \leq c_{1,2}$ (independent of $h$ and $N$) such that

$$c_{1,1} \|x\|_h \leq \|Jx\|_{L^2(\Omega)} \leq c_{1,2} \|x\|_h \quad \text{for all } x \in \mathbb{R}^N,$$

where $\| \cdot \|_h$ is the naturally scaled Euclidean norm induced by the scalar product $\langle x, y \rangle_h = h^n \sum_{i \in I} x_i y_i$. Since $J$ is also a function from $\mathbb{R}^N$ to $H^1_0(\Omega)$, the adjoint $J^* \in L(H^{-1}(\Omega), \mathbb{R}^N)$ with respect to $\langle \cdot, \cdot \rangle_h$ is defined. We define the following three $N \times N$ matrices,

$$S = J^* L J, \quad B = J^* L^{-1} J \quad \text{and} \quad M = J^* J.$$

$S$ is the stiffness matrix, $B$ the Galerkin discretization of the inverse of $L$ and $M$ is the mass matrix. The matrices $S$ and $M$ are sparse, while $B$ as well as $S^{-1}$ and $M^{-1}$ are dense.

**Remark 3.6** $M$ is positive definite and $S$ is invertible for sufficiently small $h$. Since the principal part of $L$ is coercive and the lower order terms constitute a compact operator due to the compact embedding of $L^2$ in $H^{-1}$, $L$ satisfies Garding’s inequality

$$(Lu, u)_{L^2(\Omega)} \geq \gamma \|u\|_{H^1(\Omega)}^2 - c \|u\|_{L^2(\Omega)}^2 \quad \text{for all } u \in H^1(\Omega).$$

From the Céa-Polški Lemma, cf. [18], $S$ is invertible, if $h$ is sufficiently small.

We need the following lemma [11] by which the spectral norm of an $\mathcal{H}$-matrix can be estimated by its blockwise norms. $P$ is again assumed to be generated as in Section 2.3.

**Lemma 3.7** There is a constant $c_{sp}$ such that for any matrix $M \in \mathcal{H}(P, k)$ the following inequality holds:

$$\|M\|_2 \leq c_{sp} p \max_{b \in P} \|M_b\|_2.$$

**Theorem 3.8** Let $X_t$ be convex for all $t \in T_1$. For any $\varepsilon > 0$ let $k_\varepsilon \in \mathbb{N}$ be chosen as in Theorem 3.5. Then for $k \geq \max\{k_\varepsilon, n_{\min}\}$ there is $B_\mathcal{H} \in \mathcal{H}(P, k)$ such that

$$\|B - B_\mathcal{H}\|_2 \leq c_{sp} \varepsilon,$$

(3.29)

where $c_{sp} = c_{sp}(k, b, c, d, \eta, \Omega)$ depends on $\eta$ from (2.4) and $\text{diam } \Omega$. $p$ is the depth of the cluster tree $T_1$ defined in Section 2.1.

**Proof.** Let $b = s \times t \in P$ with $\min\{\#s, \#t\} \leq n_{\min}$. In this case we simply set

$$(B_\mathcal{H})_b := B_b = (J^* L^{-1} J)_b.$$

Since the block $(B_\mathcal{H})_b$ has at most $n_{\min}$ columns or rows, rank $(B_\mathcal{H})_b \leq k$ holds.

If $b = s \times t \in P$ with $\min\{\#s, \#t\} > n_{\min}$, then $b$ satisfies (2.4). Applying Theorem 3.5 with $D_1 = X_s$, $D_2 = X_t$ there is $G_b(x, y) = \sum_{i=1}^k u_i^t(x) v_i^s(y)$ such that

$$\|G - G_b\|_{L^2(X_s \times X_t)} \leq \varepsilon \|G\|_{L^2(X_s \times X_t)},$$

where $X_t := \{ x \in \Omega : 2\eta \text{dist}(x, X_t) \leq \text{diam } X_t \}$. Let the functions $u_i^t$ and $v_i^s$ be extended to $\Omega$ by zero. We define the integral operator

$$K_b \varphi = \int_{\Omega} \hat{G}_b(\cdot, y) \varphi(y) \, dy \quad \text{for supp } \varphi \subset \overline{\Omega}$$

11
and set \((B_{\mathcal{H}})_{b} = (J^{*}K_{b}J)_{b}\). The rank of \((B_{\mathcal{H}})_{b}\) is bounded by \(k_{c}\) since each term \(u_{i}^{b}(x)v_{j}^{b}(y)\) in \(\tilde{G}_{b}\) produces one rank-1 matrix in \((J^{*}K_{b}J)_{b}\).

Let \(x \in \mathbb{R}^{d}\) and \(y \in \mathbb{R}^{e}\). To see that \((B_{\mathcal{H}})_{b}\) approximates the block \(B_{b}\), remember the representation \((3.14)\) of \(L^{-1}\) and use \((3.28)\). The estimate

\[
\langle (B - B_{\mathcal{H}})_{b}x, y \rangle_{h} = \langle (J^{*}(L^{-1} - K_{b})J)_{b}x, y \rangle_{h} = \langle ((L^{-1} - K_{b})_{b}x, Jy) \rangle_{L^{2}}
\]

\[
\leq \|G_{b} - \tilde{G}_{b}\|_{L^{2}(X_{s} \times X_{s})} \|Jx\|_{L^{2}(X_{s})} \|Jy\|_{L^{2}(X_{s})}
\]

\[
\leq \varepsilon \|G\|_{L^{2}(X_{s} \times X_{s})} \|Jx\|_{L^{2}(\Omega)} \|Jy\|_{L^{2}(\Omega)}
\]

\[
\leq \varepsilon^2 \|G\|_{L^{2}(X_{s} \times X_{s})} \|x\|_{h} \|y\|_{h}
\]

proves \(\|B - B_{\mathcal{H}}\|_{2} \leq \varepsilon \|G\|_{L^{2}(X_{s} \times X_{s})}\).

Although \(G(x, \cdot) \in W^{1,1}(\Omega)\) for all \(x \in \Omega\), \(G(\cdot, \cdot)\) does not belong to \(L^{2}(\Omega \times \Omega)\) as soon as \(n \geq 4\). From \((3.12)\) it can be seen that \(\|G\|_{L^{2}(X_{s} \times X_{t})}\) may increase when the sets \(X_{s}, X_{t}\) are approaching each other. The construction of \(\hat{X}_{t}\) however ensures

\[
\delta := \text{dist}(X_{s}, \hat{X}_{t}) \geq \frac{1}{2} \text{dist}(X_{s}, X_{t}) \geq \frac{1}{2\eta} \text{diam } X_{s}
\]
as well as \(2\eta \delta \geq \text{diam } X_{t}\) due to \((2.4)\). Hence \((3.12)\) implies for the case \(n \geq 3\)

\[
\|G\|_{L^{2}(X_{s} \times \hat{X}_{t})} \leq \frac{c_{n}(\kappa, b, c, d)}{\lambda} \delta^{2-n} \sqrt{\text{vol } X_{s}/(\text{vol } \hat{X}_{t})}.
\]

Using \(\text{vol } \hat{X}_{t} \leq \omega_{n}(\frac{1}{2} \text{diam } \hat{X}_{t})^{n} \leq \omega_{n}(\eta + 1/2)^{n} \delta^{n}\) and \(\text{vol } X_{s} \leq \omega_{n}(\eta \delta)^{n}\), where \(\omega_{n}\) is volume of the unit ball in \(\mathbb{R}^{n}\), we see that

\[
\|G\|_{L^{2}(X_{s} \times \hat{X}_{t})} \leq \tilde{c}_{\eta} \frac{c_{n}(\kappa, b, c, d)}{\lambda} \delta^{2} \text{ with } \tilde{c}_{\eta} := \omega_{n}(\eta + 1/2)^{n/2}.
\]

The rough estimate \(\delta \leq \text{diam } \Omega\) together with Lemma 3.7 yields \((3.29)\). Using \((3.13)\) the case \(d = 2\) can be treated in a similar way.

\[\blacksquare\]

**Remark 3.9** Assume that each (possibly non-convex) set \(X_{t}\) has a convex superset \(Y_{t}\) satisfying the admissibility condition \((2.4)\). Then Theorem 3.8 remains valid for \(X_{s} \times X_{t}\). Therefore, according to Remark 2.2 the assumption on the convexity of \(X_{t}\) in Theorem 3.8 is reasonable even for practical purposes.

The previous theorem shows that we are able to approximate the discrete inverse of \(L\) by \(\mathcal{H}\)-matrices. Our aim however is to prove that the inverse of the stiffness matrix \(S\) possesses this property. For this purpose we use the fact that \(S^{-1}\) can be approximated by \(M^{-1}BM^{-1}\). The last product, in turn, can be approximated by an \(\mathcal{H}\)-matrix. In [4] we have already presented the details of the above arguments. Since they are quite technical we just give the main results without proofs.

The finite element approximation is connected with the Ritz projection \(P_{h} = J A^{-1} J^{*}L : H_{0}^{1}(\Omega) \rightarrow V_{h}\). If \(u \in H_{0}^{1}(\Omega)\) is the solution of the variational problem \(a(u, v) = f(v), u_{h} = P_{h}u\) is its finite element solution. The FE error is then given by

\[
e_{h}(u) := \|u - P_{h}u\|_{L^{2}(\Omega)},
\]
and the weakest form of the finite element convergence is described by
\[ e_h(u) \leq \varepsilon_h \| f \|_{L^2(\Omega)} \quad \text{for all } u = L^{-1} f, \; f \in L^2(\Omega), \quad (3.30) \]
where \( \varepsilon_h \rightarrow 0 \) as \( h \rightarrow 0 \).

**Remark 3.10** Due to our quite weak assumptions on the smoothness of the coefficients in (3.6), one cannot specify the behavior of \( \varepsilon_h \) for \( h \rightarrow 0 \).

**Lemma 3.11** It holds that \( \| S^{-1} - M^{-1} BM^{-1} \|_2 \leq 2 c_j^2 J \| \varepsilon_h \| \).

Since the product of two \( \mathcal{H} \)-matrices is an \( \mathcal{H} \)-matrix with augmented rank, cf. [11], it remains to show that \( M^{-1} \) can be approximated by an \( \mathcal{H} \)-matrix \( N_H \). Then, \( C_H := N_H B_H N_H \) approximates \( S^{-1} \).

**Lemma 3.12** For any \( \varepsilon > 0 \), there is \( N_H \in \mathcal{H}(P,k) \) satisfying
\[ \| M^{-1} - N_H \|_2 \leq \varepsilon \| M^{-1} \|_2 \]
with \( k = O(|\log \varepsilon|^n) \).

Gathering all previous results we obtain the existence of \( \mathcal{H} \)-matrix approximants to the inverse stiffness matrix.

**Theorem 3.13** Let \( \varepsilon_h > 0 \) be the finite element error from (3.30) and \( p \) the depth of the cluster tree \( T_I \) defined in Section 2.1. Then there is a constant \( \tilde{c} > 0 \) defining \( k := \tilde{c} p^2 \log n \frac{1}{\varepsilon_h} \) and there is \( C_H \in \mathcal{H}(P,k) \) such that
\[ \| S^{-1} - C_H \|_2 \leq c_n \varepsilon_h, \quad (3.31) \]
where \( c_n = c_n(\kappa, \lambda, \| L^{-1} \|_{H^1 \rightarrow H^{-1}}, b, c, d, \eta, \text{diam } \Omega) \). If \( \varepsilon_h = O(h^\beta) \) for some \( \beta > 0 \), \( k = O(\log^{n+3} N) \) holds.

Theorem 3.13 states that \( S^{-1} \) can be approximated to an accuracy determined by the FE error, which is sufficient since the accuracy of the solution cannot be improved by a better approximation of \( S^{-1} \). In the following section, however, an inversion algorithm is devised, which, as we will see in the numerical results, can reach any prescribed accuracy.

### 4 Algorithms

Since \( \mathcal{H}(P,k) \) is not a linear space, we have to replace the usual matrix operations by truncated ones. Starting from the \( \mathcal{H} \)-matrix addition, we define an \( \mathcal{H} \)-matrix multiplication. Using these modified operations it is possible to define an \( \mathcal{H} \)-matrix inversion based on the Frobenius formulas. These ideas already appeared in the early papers on \( \mathcal{H} \)-matrices, cf. [15, 16].

#### 4.1 Truncated addition

In order to make the sum of two \( \mathcal{H}(P,k) \)-matrices be in \( \mathcal{H}(P,k) \), we have to add them blockwise and truncate each sum \( UV^T \), \( U = (U_1, U_2) \in \mathbb{R}^{s \times 2k}, \; V = (V_1, V_2) \in \mathbb{R}^{t \times 2k} \), of two rank-\( k \) blocks \( U_1 V_1^T \) and \( U_2 V_2^T \) to a matrix of rank at most \( k \). For this purpose we have to assume that for a given precision \( \varepsilon > 0 \) a matrix \( R \) of rank \( \ell \leq k \) exists such that \( \| UV^T - R \|_2 < \varepsilon \). The matrix \( R \) can be found by the following algorithm, which was also used in [2] for finding the approximant of lowest rank in an \( \varepsilon \)-neighborhood of a low-rank matrix.
The previous truncation algorithm needs \( O(k^2(|s| + |t|)) \) operations if \( b = s \times t \). Hence, exploiting the block-hierarchy the complexity for the \( H \)-matrix addition of two matrices from \( H(P, k) \) can be shown to be of order \( \eta^{-n} k^2 N \log N \).

### 4.2 Truncated matrix-matrix multiplication

Since the partition \( P \) consists of the leaves of the block cluster tree \( T_{I \times I} \), we are able to recursively define a modified matrix-matrix multiplication \( C \overset{\oplus}{=} A \odot B, A \in H(P, k), B \in H(P, k) \), making use of the partitioned matrix-matrix multiplication. Let \( r \times s, s \times t, r \times t \in T_{I \times I} \) be block clusters. In order to define what is meant with \( C_{r \times t} \overset{\oplus}{=} A_{r \times s} \odot B_{s \times t} \) we have to distinguish three cases:

1. All three blocks \( r \times s, s \times t \) and \( r \times t \) have sons in the tree \( T_{I \times I} \).

   \[
   \begin{bmatrix}
   C_{r_1 \times t_1} & C_{r_1 \times t_2} \\
   C_{r_2 \times t_1} & C_{r_2 \times t_2}
   \end{bmatrix}
   \overset{\oplus}{=}
   \begin{bmatrix}
   A_{r_1 \times s_1} & A_{r_1 \times s_2} \\
   A_{r_2 \times s_1} & A_{r_2 \times s_2}
   \end{bmatrix}
   \odot
   \begin{bmatrix}
   B_{s_1 \times t_1} & B_{s_1 \times t_2} \\
   B_{s_2 \times t_1} & B_{s_2 \times t_2}
   \end{bmatrix}
   \]

   is recursively defined by

   \[
   \begin{align*}
   C_{r_1 \times t_1} & \overset{\oplus}{=} A_{r_1 \times s_1} \odot B_{s_1 \times t_1}, \\
   C_{r_1 \times t_2} & \overset{\oplus}{=} A_{r_1 \times s_1} \odot B_{s_1 \times t_2}, \\
   C_{r_2 \times t_1} & \overset{\oplus}{=} A_{r_2 \times s_1} \odot B_{s_1 \times t_1}, \\
   C_{r_2 \times t_2} & \overset{\oplus}{=} A_{r_2 \times s_1} \odot B_{s_1 \times t_2}.
   \end{align*}
   \]

2. One of the blocks \( r \times s \) and \( s \times t \) is a leaf in \( T_{I \times I} \).

   Assume that \( s \times t \) is a leaf, then \( B_{s \times t} \) has a representation \( B_{s \times t} = U_B V_B^T, U_B \in \mathbb{R}^{s \times k}, V_B \in \mathbb{R}^{t \times k} \).

   \[
   C_{r \times t} \overset{\oplus}{=} A_{r \times s} \odot B_{s \times t}
   \]

   where
in this case is defined as
\[ C_{r \times t} := C_{r \times t} \oplus A_{r \times s} U_B V_B^T, \]
where \( A_{r \times s} U_B \) are \( \mathcal{H} \)-matrix-vector products.

3. \( r \times t \) has no sons in \( T_{r \times t} \), and \( r \times s, s \times t \) have sons in the tree \( T_{r \times t} \).

For the definition of
\[ C_{r \times t} \oplus \left[ \begin{array}{cc} A_{r_1 \times s_1} & A_{r_1 \times s_2} \\ A_{r_2 \times s_1} & A_{r_2 \times s_2} \end{array} \right] \oplus \left[ \begin{array}{cc} B_{s_1 \times t_1} & B_{s_1 \times t_2} \\ B_{s_2 \times t_1} & B_{s_2 \times t_2} \end{array} \right] \]
we introduce matrices \( R_1, R_2, R_3 \) and \( R_4 \) by
\[ R_1 = R_2 = R_3 = R_4 = 0 \]
and
\[ R_1 \oplus A_{r_1 \times s_1} \oplus B_{s_1 \times t_1}, \quad R_1 \oplus A_{r_1 \times s_2} \oplus B_{s_2 \times t_1}, \]
\[ R_2 \oplus A_{r_1 \times s_1} \oplus B_{s_1 \times t_2}, \quad R_2 \oplus A_{r_1 \times s_2} \oplus B_{s_2 \times t_2}, \]
\[ R_3 \oplus A_{r_2 \times s_1} \oplus B_{s_1 \times t_1}, \quad R_3 \oplus A_{r_2 \times s_2} \oplus B_{s_1 \times t_1}, \]
\[ R_4 \oplus A_{r_2 \times s_1} \oplus B_{s_1 \times t_2}, \quad R_4 \oplus A_{r_2 \times s_2} \oplus B_{s_2 \times t_2}, \]
and set
\[ C_{r \times t} := C_{r \times t} \oplus_k \left[ \left( \begin{array}{cc} R_1 & 0 \\ 0 & R_2 \end{array} \right) \oplus_k \left( \begin{array}{cc} 0 & 0 \\ R_3 & 0 \end{array} \right) \oplus_k \left( \begin{array}{cc} 0 & 0 \\ 0 & R_4 \end{array} \right) \right]. \]

If the truncation accuracy \( \varepsilon \) was chosen to be the machine precision then \( C \oplus A \circ B \) would coincide with \( C := C + AB \). The complexity of the truncated matrix-matrix multiplication can be estimated as \( O(\eta^{-2} k^2 N \log^2 N) \), cf. [11].

### 4.3 Inversion

We assume that each block \( A_{s \times s}, s \in T_r \), of \( A \in \mathcal{H}(P, k) \) is invertible. This is for example the case if \( A \) is positive definite. The matrix block \( A_{s \times s} \) corresponding to \( s \in T_r \setminus \mathcal{L}(T_r) \) is subdivided into the sons of \( s \times s \):
\[ A_{s \times s} = \left[ \begin{array}{cc} A_{s_1 \times s_1} & A_{s_1 \times s_2} \\ A_{s_2 \times s_1} & A_{s_2 \times s_2} \end{array} \right]. \]

According to the Frobenius formulas for the inverse of \( A \) it holds that:
\[ A_{s \times s}^{-1} = \left[ \begin{array}{cc} A_{s_1 \times s_1}^{-1} + A_{s_1 \times s_2}^{-1} A_{s_1 \times s_2} S^{-1} A_{s_2 \times s_1} A_{s_2 \times s_2}^{-1} A_{s_1 \times s_1}^{-1} & -A_{s_1 \times s_1}^{-1} A_{s_1 \times s_2}^{-1} S^{-1} \\ -S^{-1} A_{s_2 \times s_1} A_{s_1 \times s_1}^{-1} & A_{s_1 \times s_1}^{-1} \end{array} \right], \]
where \( S \) is the Schur complement \( S = A_{s_2 \times s_2} - A_{s_2 \times s_1} A_{s_1 \times s_1}^{-1} A_{s_1 \times s_2} \). The \( \mathcal{H} \)-matrix inverse \( C_{s \times s} \) of \( A_{s \times s} \) is defined by replacing the matrix-matrix multiplication and the addition by the \( \mathcal{H} \)-versions. We need a temporary matrix \( T \in \mathcal{H}(P, k) \), which together with \( C \) is initialized to zero.
procedure invertH(s, A, var C)
begin
if $s \in L(T_I)$ then $C_{s \times s} := A_{s \times s}^{-1}$ is the usual inverse.
else begin
invertH($s_1$, A, C).
$T_{s_1 \times s_2} \oplus C_{s_1 \times s_1} \circ A_{s_1 \times s_2}$.
$T_{s_2 \times s_1} \ominus C_{s_2 \times s_2} \circ C_{s_1 \times s_1}$.
invertH($s_2$, A, C).
end
end

The matrix $A$ is destroyed during the previous algorithm and $C \in H(P, k)$ contains an approximant of $A^{-1}$. The cost for the computation of the $H$-inverse is mainly determined by the cost for the $H$-multiplication. Therefore, an approximation to the inverse of $A$ can be obtained with complexity $O(\eta^{-n}k^2N \log^2 N)$.

5  Numerical experiments

In this section the practical influence of the various terms of the differential operator (3.6) on the efficiency and accuracy of the $H$-inverse is investigated. For simplicity all tests are performed on a uniform triangulation of the unit square $\Omega := (0, 1)^2$ in $\mathbb{R}^2$. In each case the stiffness matrix $S$ is built in the $H$-matrix format, see Example 2.4. Then the inversion algorithm from Section 4.3 is applied to it with a relative truncation accuracy $\varepsilon$. Hence, the rank $k$ is adaptively chosen and is therefore expected to vary among the blocks. All test were carried out on a single processor of a SunFire 6800 – 900MHz.

Let $u_h \in \mathbb{R}^N$ be the finite element solution, i.e., the solution of $Su_h = b$, where $b$ is the vector with the components

$$b_i = \int_{\Omega} f \varphi_i \, dx, \quad i = 1, \ldots, N,$$

and $\tilde{u}_h = Cb$, where $C$ is the computed $H$-matrix approximant of $S^{-1}$. Since

$$\|u_h - \tilde{u}_h\|_2 \leq \|I_N - CS\|_2 \|u_h\|_2$$

(5.32)

the expression $\|I_N - CS\|_2$ is an upper bound on the relative accuracy of $\tilde{u}_h$ compared with the finite element solution $u_h$. Note that $\tilde{u}_h$ cannot be a better approximation of $u$ than $u_h$ is, since the proposed method is built on top of the finite element method. Hence, in the following computations we will rely on the expression $\|I_N - CS\|_2$ as a measure for the accuracy.

5.1  Principal parts

In the first example we consider operators $L = \text{div } A(x) \nabla$. The coefficients $A$ of $L$ are chosen to be of the following form

$$A(x) = \begin{bmatrix} 1 & 0 \\ 0 & \alpha(x) \end{bmatrix}, \quad x \in \Omega,$$
where $\alpha(x) = 1$ in the lower region of Figure 1 and a random number from the interval $[0, a]$ in the remaining part of the unit square. In order to avoid averaging effects the coefficient $\alpha$ possesses a two-level random structure: the randomly chosen coefficient on each triangle is multiplied by a coefficient chosen randomly on a scale of length $\sqrt{h}$, where the grid size $h$ is defined through $h(\sqrt{N/2} + 1) = 1$.

In the following table the accuracy $\|I_N - CS\|_2$ of the $H$-matrix $C$ and the CPU time consumption are compared for different $a$ and different problem sizes $N$. The truncation accuracy $\varepsilon$ is chosen such that $\|I_N - CS\|_2$ is of order $h$.

<table>
<thead>
<tr>
<th>$N$</th>
<th>$h$</th>
<th>$\varepsilon$</th>
<th>$a = 1.0$</th>
<th>$a = 10.0$</th>
<th>$a = 100.0$</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>time [s]</td>
<td>accuracy</td>
<td>time [s]</td>
</tr>
<tr>
<td>14400</td>
<td>8.3e-3</td>
<td>1e-5</td>
<td>29.4</td>
<td>8.5e-3</td>
<td>28.4</td>
</tr>
<tr>
<td>38025</td>
<td>5.1e-3</td>
<td>2e-6</td>
<td>146.7</td>
<td>6.2e-3</td>
<td>138.6</td>
</tr>
<tr>
<td>65025</td>
<td>3.9e-3</td>
<td>5e-7</td>
<td>341.3</td>
<td>3.4e-3</td>
<td>327.6</td>
</tr>
<tr>
<td>129600</td>
<td>2.8e-3</td>
<td>2e-7</td>
<td>1074.9</td>
<td>2.6e-3</td>
<td>1008.0</td>
</tr>
<tr>
<td>278784</td>
<td>1.9e-3</td>
<td>5e-8</td>
<td>3452.1</td>
<td>1.6e-3</td>
<td>3201.7</td>
</tr>
<tr>
<td>529984</td>
<td>1.3e-3</td>
<td>2e-8</td>
<td>9210.0</td>
<td>1.3e-3</td>
<td>8529.0</td>
</tr>
</tbody>
</table>

In the next set of tests the same quantities for a smooth but oscillating coefficient in the principal part are computed, i.e., $\alpha$ is chosen to be the following function

$$\alpha(x) = a(1 + \cos(b2\pi x) \sin(b2\pi y)).$$

By changing the coefficient $a$ we are able to prescribe the amplitude of the oscillation and by $b$ the number of oscillations in $x$- and $y$-direction of $\Omega = (0, 1)^2$. The following table contains the results for $N = 65025$ and $\varepsilon = 5e - 7$. 

Figure 1: The coefficient $\alpha(x)$
The experiments show that in the absence of lower-order terms neither the accuracy nor the CPU times needed to compute the approximant do depend much on the coefficients.

5.2 Convection-diffusion

In this section operators of the type

\[ L = -\Delta + c \cdot \nabla \]

will be considered. In the first example the convection coefficient \( c \) is randomly chosen, i.e., \( c(x) \in [-a, a]^2 \) for \( x \in \Omega \). The following table shows \( \| I_N - CS \|_2 \) for different parameters \( a \).

<table>
<thead>
<tr>
<th>( N )</th>
<th>( h )</th>
<th>( \epsilon )</th>
<th>( a = 1.0 )</th>
<th>( a = 10.0 )</th>
<th>( a = 100.0 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>time [s]</td>
<td>accuracy</td>
<td>time [s]</td>
<td>accuracy</td>
<td>time [s]</td>
<td>accuracy</td>
</tr>
<tr>
<td>14641</td>
<td>8.2e-3</td>
<td>3.1</td>
<td>3.1</td>
<td>3.1</td>
<td>3.1</td>
</tr>
<tr>
<td>38416</td>
<td>5.1e-3</td>
<td>4.2</td>
<td>4.2</td>
<td>4.2</td>
<td>4.2</td>
</tr>
<tr>
<td>65025</td>
<td>3.9e-3</td>
<td>2.9</td>
<td>2.9</td>
<td>2.9</td>
<td>2.9</td>
</tr>
<tr>
<td>129600</td>
<td>2.8e-3</td>
<td>2.5</td>
<td>2.5</td>
<td>2.5</td>
<td>2.5</td>
</tr>
<tr>
<td>278784</td>
<td>1.9e-3</td>
<td>1.6</td>
<td>1.6</td>
<td>1.6</td>
<td>1.6</td>
</tr>
<tr>
<td>597529</td>
<td>1.3e-3</td>
<td>1.4</td>
<td>1.4</td>
<td>1.4</td>
<td>1.4</td>
</tr>
</tbody>
</table>

In the next example we investigate operators

\[ Lu = -\epsilon \Delta u + u_x + u_y \]

for different parameters \( \epsilon > 0 \). We are particularly interested in a convection dominated setting.

Since the above tables show that it is possible to find an \( H \)-matrix \( C \) that approximates \( S^{-1} \), from (5.32) it is obvious that \( \tilde{u}_h \) approximates the finite element solution \( u_h \). This is especially true in the presence of boundary layers as is illustrated in the following example. The solution of

\[ -\epsilon \Delta u + u_x + u_y = f, \]

where

\[ f(x, y) = (x + y)(1 - e^{(x-1)/\epsilon} e^{(y-1)/\epsilon}) + (x - y)(e^{(y-1)/\epsilon} - e^{(x-1)/\epsilon}) \]

18
with zero boundary conditions is known to be
\[
u(x, y) = xy(1 - e^{(x-1)/\epsilon})(1 - e^{(y-1)/\epsilon}).
\]

Figure 2 compares the restrictions of \(u\) and \(\tilde{u}_h\) to the set \(\{(x, x), x \in (0, 1)\}\) for \(\epsilon = 0.01\) and \(N = 14641\). Obviously, the proposed inversion procedure is able to handle boundary layers as long as the underlying finite element method is stable.

In the next example we consider convection in a direction that is aligned with the grid, i.e.,
\[
Lu = -\epsilon \Delta u + u_x
\]
for different parameters \(\epsilon > 0\) are investigated.

<table>
<thead>
<tr>
<th>(N)</th>
<th>(\epsilon = 0.1) time [s] accuracy</th>
<th>(\epsilon = 0.01) time [s] accuracy</th>
<th>(\epsilon = 0.001) time [s] accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>14641</td>
<td>30.3  6.3e−3</td>
<td>31.4  6.6e−4</td>
<td>47.4  1.3e−4</td>
</tr>
<tr>
<td>38416</td>
<td>145.7 4.9e−3</td>
<td>161.4 5.7e−4</td>
<td>226.1 6.3e−5</td>
</tr>
<tr>
<td>65025</td>
<td>339.0 2.2e−3</td>
<td>365.7 1.8e−4</td>
<td>463.9 1.8e−5</td>
</tr>
<tr>
<td>129600</td>
<td>1053.9 1.9e−3</td>
<td>1152.1 1.8e−4</td>
<td>1235.8 1.4e−5</td>
</tr>
<tr>
<td>278784</td>
<td>3323.3 1.1e−3</td>
<td>3657.8 1.5e−4</td>
<td>4135.5 6.3e−6</td>
</tr>
<tr>
<td>597529</td>
<td>10457.6 1.0e−3</td>
<td>11646.8 1.3e−4</td>
<td>12208.7 7.9e−6</td>
</tr>
</tbody>
</table>

It is well known that if the ratio \(\epsilon/h\) gets small the finite element discretization suffers from the loss of stability. As a consequence the stiffness matrix becomes ill-conditioned. It may even happen that \(S\) is not invertible. This behavior is observable for small \(N\) if \(\epsilon\) tends to zero: when changing \(\epsilon = 0.01\) to \(\epsilon = 0.001\) the CPU time for \(N = 14641\) increases by a factor of 1.5, while it is almost not influenced in the case \(N = 597529\), where the discretization is stable.

The proposed inversion procedure can also be applied to Shishkin meshes, which should be used for a better convergence of the finite element method.

5.3 Diffusion-reaction

As a third kind of example we consider the operator \(Lu = -\Delta u + du\) with a randomly chosen reaction term \(d(x) \in [0, a]\) for \(x \in \Omega\).
Close to resonance the stiffness matrix is ill-conditioned. The functions depend on the wave number. Therefore, the method can still be applied, but it is only efficient if the wave number is not too large. The same effect can be observed here.

Each column of the following table shows the approximation results for the respective $d$ in the case $N = 129600$. In order to be able to guarantee $\|I_N - CS\|_2 \sim h$ we have to choose a higher truncation accuracy $\varepsilon$ if the modulus of $d$ is increased. Note that $d$ is now a constant.

<table>
<thead>
<tr>
<th>$d$</th>
<th>$a = 10.0$</th>
<th>$a = 100.0$</th>
<th>$a = 1000.0$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$N$</td>
<td>time [s]</td>
<td>accuracy</td>
<td>time [s]</td>
</tr>
<tr>
<td>14641</td>
<td>31.2</td>
<td>5.5e-3</td>
<td>32.3</td>
</tr>
<tr>
<td>38416</td>
<td>154.4</td>
<td>4.9e-3</td>
<td>161.6</td>
</tr>
<tr>
<td>65025</td>
<td>357.1</td>
<td>2.1e-3</td>
<td>369.1</td>
</tr>
<tr>
<td>129600</td>
<td>1106.5</td>
<td>1.9e-3</td>
<td>1133.9</td>
</tr>
<tr>
<td>278784</td>
<td>3502.8</td>
<td>1.2e-3</td>
<td>3610.1</td>
</tr>
<tr>
<td>597529</td>
<td>11100.7</td>
<td>1.1e-3</td>
<td>11500.8</td>
</tr>
</tbody>
</table>

Hence, for large wave numbers the inversion procedure can be applied, but gets less efficient.

As a last example we consider values $d$ that are close to the eigenvalues of the operator $\Delta$. In the case of resonance, i.e., $-d$ is an eigenvalue of $S$, the stiffness matrix $S$ is not invertible. Close to resonance the stiffness matrix is ill-conditioned. The functions

$$u_k(x, y) := \sin(2\pi k x) \sin(2\pi k y), \quad k = 0, 1, \ldots,$$

solve the eigenproblem

$$-\Delta u = \lambda u \quad \text{in } \Omega = (0, 1)^2$$

$$u = 0 \quad \text{on } \partial \Omega$$

with corresponding eigenvalues $\lambda_k := (2\pi k)^2$. In the following table $N = 129600$ and $-d$ is chosen in a neighborhood of the second eigenvalue $\lambda_1 = 4\pi^2$.

<table>
<thead>
<tr>
<th>$d$</th>
<th>$-39.0$</th>
<th>$-39.5$</th>
<th>$-40.0$</th>
<th>$-40.15$</th>
<th>$-40.2$</th>
<th>$-40.25$</th>
<th>$-41.0$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\varepsilon$</td>
<td>2.0e-8</td>
<td>2.0e-8</td>
<td>1.0e-8</td>
<td>2.0e-9</td>
<td>5.0e-10</td>
<td>2.0e-9</td>
<td>2.0e-8</td>
</tr>
<tr>
<td>accuracy</td>
<td>1.9e-3</td>
<td>1.7e-3</td>
<td>2.0e-3</td>
<td>1.6e-3</td>
<td>2.3e-3</td>
<td>1.3e-3</td>
<td>1.5e-3</td>
</tr>
<tr>
<td>time [s]</td>
<td>1109.4</td>
<td>1114.6</td>
<td>1168.2</td>
<td>1299.4</td>
<td>1414.0</td>
<td>1292.6</td>
<td>1109.1</td>
</tr>
</tbody>
</table>

We conclude from the above numerical experiments that the proposed method is robust with respect to non-smooth and anisotropic coefficients. Even convection-diffusion problems with dominating convection can be solved efficiently without special adaption of the algorithm to this class of problems. Hence, the proposed inversion procedure can be applied whenever a
stable discretization of the operator $L$ is available. Helmholtz’ equation can be treated but the algorithms are only efficient for relatively small wave numbers.

**Acknowledgment.** The author wishes to thank the referees for helpful suggestions.

**References**


