Max-Planck-Institut
für Mathematik
in den Naturwissenschaften
Leipzig

# On $\mathcal{H}^2$-Matrices

by

*Wolfgang Hackbusch, Boris Khoromskij*
*and Stefan A. Sauter*

Preprint no.: 50 1999

# On $\mathcal{H}^2$-Matrices

Wolfgang Hackbusch, Boris Khoromskij
Max-Planck-Institut *Mathematik in den Naturwissenschaften*
Inselstr. 22-26, D-04103 Leipzig, Germany
email: wh@mis.mpg.de

Stefan A. Sauter
Universität Leipzig, Mathematisches Institut,
Augustusplatz 10-11, D-04109 Leipzig, Germany
email: sas@mathematik.uni-leipzig.de

**Abstract**

A class of matrices ($\mathcal{H}$-matrices) has recently been introduced by one of the authors. These matrices have the following properties: (i) They are sparse in the sense that only few data are needed for their representation. (ii) The matrix-vector multiplication is of almost linear complexity. (iii) In general, sums and products of these matrices are no longer in the same set, but their truncations to the $\mathcal{H}$-matrix format are again of almost linear complexity. (iv) The same statement holds for the inverse of an $\mathcal{H}$-matrix.

The term "almost linear complexity" used above means that estimates are given by $O(n \log^\alpha n)$. The logarithmic factor can be avoided by a further improvement, which is described in the present paper. We prove that the storage requirements and the cost of the matrix-vector multiplication is strictly linear in the dimension $n$, while still (full) system matrices of the boundary element method can be approximated up to the discretisation error.

*AMS Subject Classifications:* 65F05, 65F30, 65F50, 65N38, 68P05, 45B05, 35C20
*Key words:* Hierarchical matrices, hierarchical bases, full matrices, fast matrix-vector multiplication, BEM, FEM.

## 1 Introduction

For linear systems with sparse $n \times n$-matrices several optimal iteration methods are known, where optimality is characterised by an estimation of the arithmetic operations by $O(n)$. A different situation is given in the case of full matrices. Then standard techniques require an storage amount of $O(n^2)$ and $O(n^2)$ operations for the matrix-vector multiplication. Other arithmetic operations like matrix-matrix multiplications or the inversion even lead to $O(n^3)$ operations.

Full matrices are directly obtained by the discretisation of integral equations as they are common in the boundary element method (BEM; cf. [2]). Another source of a full matrix is the inverse of a sparse FEM matrix which, e.g., appears in the Schur complement of any saddle point problem (cf. [1, Section 11.7]). In both cases, the matrices are affected with a discretisation error. Therefore, one may replace the full matrix $M$ by a more convenient matrix $M'$, provided that the error $M - M'$ is of the size of the discretisation error.

The hierarchical matrices (abbreviated as $\mathcal{H}$-matrices) define a set of matrices which provides the approximations $M'$ discussed above. As described in detail in [3] and [4], $\mathcal{H}$-matrices have the following properties:

(i) They are data-sparse in the sense that the size data to be stored is almost linear in the dimension $n$.

(ii) The matrix-vector multiplication is of almost linear complexity.

(iii) In general, sums and products of these matrices are no longer in the same set, but their truncations to the $\mathcal{H}$-matrix format are again of almost linear complexity.

(iv) The same statement holds for the inverse of an $\mathcal{H}$-matrix.

The basic (hierarchical) structure of $\mathcal{H}$-matrices is the *cluster tree* which is already introduced in the *panel clustering method* (see [6] and [7] or [2, Section 9.7]).

The term "almost linear complexity" used above means that estimates are given by $O(n \log^\alpha n)$. The logarithmic factor can be avoided by a further improvement which leads to the $\mathcal{H}^2$-matrices (hierarchical bases $\mathcal{H}$-matrices) introduced and analysed in this paper. These matrices are already mentioned in [3] under the name "uniform $\mathcal{H}$-matrices". The essential analysis is given in Section 4. The approximation by a Taylor polynomial of fixed degree is replaced by a variable degree. Although we use lower approximation degrees for most of the matrix-blocks, the overall error estimate does not deteriorate.

# 2 Hierarchical Bases $\mathcal{H}$-Matrices

After presenting the introductory example (§2.1), we define the cluster tree (§2.2), which is the basis of the standard $\mathcal{H}$-matrices (§2.3). Finally, in Subsection 2.4, we introduce the hierarchical bases $\mathcal{H}$-matrices, which we call $\mathcal{H}^2$-matrices.

## 2.1 Introductory Example

The matrices, which we have in mind, may stem from integral or differential equations. In the latter case, it is of interest to represent the inverse matrix as an $\mathcal{H}$-matrix[1]. Here, we consider the example of the integral operator

$$(Ku)(x) = \int_0^1 \log(|x-y|)u(y)dy. \tag{2.1}$$

Its finite element discretisation with piecewise constant basis functions corresponding to the interval partitioning

$$J_i = [(i-1)h, ih], \quad h := 1/n, \quad 1 \le i \le n \tag{2.2}$$

leads to the matrix

$$M = (m_{ij})_{i,j \in I}, \quad m_{ij} := \int_{J_i} \int_{J_j} \log(|x-y|)dxdy, \tag{2.3}$$

where

$$I = \{1, \dots, n\} \tag{2.4}$$

is the underlying *index set*. As further simplification, we assume that $n$ is a power of 2:

$$n = 2^p. \tag{2.5}$$

In boundary element applications (BEM), one has to replace the unit interval by a surface, the equidistant partitioning by a general triangulation and the kernel function $\log(|x-y|)$ by some appropriate singularity function (cf. [2]). However, in order not to distract the attention of the reader from the main ideas, we consider the matrix $M$ from (2.3). The kernel $\log(|x-y|)$ shares typical properties with the kernels arising in general BEM applications. The results of this paper can be extended to general BEM problems as well.

The matrix $M$ from (2.3) is a full matrix, i.e., the usual storage amount is $O(n^2)$ instead of $O(n)$ for standard sparse matrices. Furthermore, a simple matrix-vector multiplication $M * x$ requires $O(n^2)$ operations. The aim of the $\mathcal{H}^2$-matrix concept is to replace $M$ by an approximation $M'$ such that the error $M - M'$ is of the size of the discretisation error (therefore negligible), while *storage(M)* and *cost(M * x)* amounts to $O(n)$ instead of $O(n^2)$.

The discussion of the error $M - M'$ is performed in Section 4. The details about the storage and matrix-vector multiplication cost are given in Section 3.

[1] Formally, the inverse can be considered as a discretisation of an integral operator with the Green function as kernel.

## 2.2 The Cluster Tree

We start with the full index set $I_1^0 := I = \{1, \dots, n\}$ from (2.4) and split it into the parts $I_1^1 := \{1, \dots, \frac{n}{2} = 2^{p-1}\}$, $I_2^1 := \{2^{p-1}+1, \dots, n\}$. Similarly, the new sets are divided so that, in general,

$$I_i^\ell := \{(i-1)2^{p-\ell}+1, \dots, i\, 2^{p-\ell}\} \qquad \text{for } 0 \le \ell \le p,\ 1 \le i \le 2^\ell. \tag{2.6}$$

The superscript $\ell$ indicates the *level*. At level $p$, we reach the one-element sets $I_1^p = \{1\}, \dots, I_n^p = \{n\}$.

Obviously, these sets form a tree $T$ (the so-called *cluster tree*).

**Remark 2.1** *(a) $I$ is the root of $T$. (b) The sets $I_i^\ell$ are the vertices ("clusters") of $T$ at level $\ell$. (c) $T$ is a binary tree: $I_i^\ell$ has two sons $I_{2i-1}^{\ell+1}$ and $I_{2i}^{\ell+1}$ if $\ell < p$. (d) The sets $I_i^p$ are the leaves of $T$. (e) The cardinality of $I_i^\ell$ is $\#I_i^\ell = 2^{p-\ell}$.*

In the following, we use the variables $\tau$ and $\sigma$ for the vertices of the tree $T$ and call $\tau \in T$ a *cluster*. Usually, the sons of $\tau \in T$ are denoted by $\tau', \tau''$.

An isomorphic description occurs when we replace the index $i$ by the interval $J_i$ from (2.2) which is the support of the $i$th basis function. Then a cluster $\tau \in T$ corresponds to the interval

$$J(\tau) := \bigcup \{J_\alpha : \alpha \in \tau\}. \tag{2.7}$$

The partitioning of the set $I$ into $I_1^0, I_1^1$ corresponds to the definition of a block structure of a vector (over the index set $I$). The tree structure of $T$ allows to continue the block decomposition in a hierarchical way. The *hierarchical matrices* based on this tree structure are abbreviated as $\mathcal{H}$*-matrices.*

## 2.3 $\mathcal{H}$-Matrices

### 2.3.1 The Model Partitioning

Since we are dealing with matrices, we have to consider the index set $I \times I$. In the following, we describe a particular partitioning $P_2$ of $I \times I$ such that

$$I \times I = \dot{\bigcup} \{b : b \in P_2\}, \tag{2.8}$$

where each block $b \in P_2$ is of the form $b = I_i^\ell \times I_j^\ell$ for some $0 \le \ell \le p$, $1 \le i, j \le 2^\ell$. The subscript 2 in $P_2$ should indicate that $P_2$ partitions the *two*fold product index set $I \times I$. In the interesting case, the blocks $b \in P_2$ corresponding to the block matrix $M^b := (m_{\alpha\beta})_{(\alpha,\beta)\in b}$ do not all belong to only one level $\ell$. The level number $\ell$ of a block $b$ is written as $level(b)$.

The easiest way to introduce the partitioning $P_2$ is by a recursive description of the matrix block structure. For this purpose we consider four different matrix formats: $\mathcal{R}$-, $\mathcal{N}$-, $\mathcal{N}^*$- and, finally, the $\mathcal{H}$-matrices.

$\mathcal{R}$*-matrices* are matrices of rank $\le k$. The value of $k$ and its possible dependence on $b$ will be discussed later. These $\mathcal{R}$-matrices can be represented in the form

$$\sum\nolimits_{i=1}^{k} [a_i, c_i], \quad \text{where } [a_i, c_i] := a_i * c_i^H \tag{2.9}$$

with column vectors $a_i$ and row vectors $c_i^H$. The set of $\mathcal{R}$-matrices is denoted by $\mathcal{M}_\mathcal{R}$.

The $\mathcal{N}$*-matrices* correspond to off-diagonal blocks $b = I_i^\ell \times I_{i+1}^\ell$ ($\mathcal{N}$ abbreviates "neighbourhood"). For $\ell = p$, $\mathcal{N}$-matrices are simple $1 \times 1$-matrices. For $\ell = p-1, \dots, 1$, the following recursive definition holds: Abbreviate $m = 2^{p-\ell}$. An $m \times m$ matrix $M$ has the $\mathcal{N}$-format if

$$M = \begin{bmatrix} M_{11} & M_{12} \\ M_{21} & M_{22} \end{bmatrix} \text{ with } \tfrac{m}{2} \times \tfrac{m}{2}\ \mathcal{R}\text{-matrices } M_{11}, M_{12}, M_{22} \text{ and } \mathcal{N}\text{-matrix } M_{21}. \tag{2.10}$$

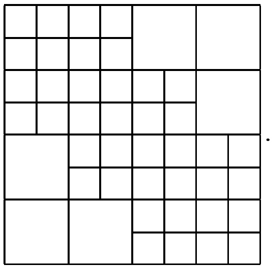
Similarly, we define the transposed type: $M$ is an $\mathcal{N}^*$*-matrix* if $M^T$ is of $\mathcal{N}$-type, i.e., in (2.10) $M_{11}, M_{21}, M_{22}$ are $\mathcal{R}$-matrices and $M_{12}$ is an $\mathcal{N}^*$-matrix. The sets of $\mathcal{N}$- and $\mathcal{N}^*$-matrices are denoted by $\mathcal{M}_\mathcal{N}$ and $\mathcal{M}_{\mathcal{N}^*}$.

Finally, the $\mathcal{H}$*-matrices* ("hierarchical matrices") are defined in

**Definition 2.2** *Let $M$ be an $n \times n$-matrix with $n = 2^p$. Then $M$ is an $\mathcal{H}$-matrix (notation: $M \in \mathcal{M}_\mathcal{H}$) if either $n = 1$ ($p = 0$) or if the partitioning into $2 \times 2$ blocks of size $\frac{n}{2} \times \frac{n}{2}$ leads to*

$$M = \begin{bmatrix} M_{11} & M_{12} \\ M_{21} & M_{22} \end{bmatrix} \textit{ with } M_{11}, M_{22} \in \mathcal{M}_\mathcal{H},\ M_{12} \in \mathcal{M}_\mathcal{N},\ M_{21} \in \mathcal{M}_{\mathcal{N}^*}. \tag{2.11}$$

In the case of $p = 3$, the resulting block structure of an $8 \times 8$-matrix is



. (2.12)

Let $P_2 \subset \mathbb{P}(I \times I)$ be the set of the finally resulting blocks in (2.11). In the case of (2.12), $P_2$ consists of 40 $1 \times 1$-blocks and 6 $2 \times 2$-blocks.

Here, we remark that we need a partitioning with two properties: On the one hand side, the partitioning should contain as few blocks as possible to reduce the costs for storage and operations, while on the other hand the blocks should be small enough so that the resulting matrix is a sufficiently good approximation of the true matrix. We shall see that $P_2$ from (2.11) is a good compromise.

The rank $k$ involved in $\mathcal{M}_{\mathcal{R}}$ is not necessarily a constant. In the following, $k : P_2 \to \mathbb{N}$ is a function of the block $b \in P_2$. Then, a submatrix $M^b$ over the index block $b \in P_2$ belongs to $\mathcal{M}_{\mathcal{R}}$ if the block $M^b$ satisfies $\operatorname{rank}(M^b) \leq k(b)$. The following definition is equivalent to Definition 2.2, if we choose the partitioning $P_2$ from above.

**Definition 2.3** *Let $P_2$ be a block partitioning of $I \times I$ and $k : P_2 \to \mathbb{N}$. The underlying field of the matrices is $\mathbb{K}$. The set of $\mathcal{H}$-matrices induced by $P_2$ and $k$ is*

$$\mathcal{M}_{\mathcal{H},k}(I \times I, P_2) := \{M \in \mathbb{K}^{I \times I} : \text{ each block } M^b,\ b \in P_2,\ \text{satisfies } \operatorname{rank}(M^b) \leq k(b)\}. \tag{2.13}$$

## 2.4 Definition of $\mathcal{H}^2$-Matrices

Up to now, we made use of the cluster tree $T$, which yields a hierarchy among the clusters and leads to the optimal partitioning $P_2$. Next, we introduce another hierarchical structure connected with the vectors $a_i, c_i$ from (2.9). This second hierarchy gives rise to the exponent 2 in the name $\mathcal{H}^2$-matrices (or[2] *hierarchical basis $\mathcal{H}$-matrices*).

### 2.4.1 Hierarchical Bases for Row and Column Vectors of $\mathcal{H}^2$-Matrices

So far, an $\mathcal{R}$-matrix $\sum_{i=1}^{k(b)}[a_i, c_i]$ from (2.9) could be formed with arbitrary vectors $a_i, c_i$. Another situation occurs if we fix two bases $\{a_i\}, \{c_i\}$ depending on the block $b \in P_2$. Any block $b$ has the form $b = \tau \times \sigma$ with clusters $\tau, \sigma \in T$. We require that $\{a_i\}$ depends only on the row-index cluster $\tau$, while $\{c_i\}$ depends only on the column-index cluster $\sigma$ :

$$\mathcal{V}_a(\tau) = \operatorname{span}\{a_i^\tau : 1 \leq i \leq k(\tau)\} \subseteq \mathbb{K}^\tau, \quad \mathcal{V}_c(\sigma) = \operatorname{span}\{c_j^\sigma : 1 \leq j \leq k(\sigma)\} \subseteq \mathbb{K}^\sigma. \tag{2.14}$$

The notation $a_i^\tau \in \mathbb{K}^\tau$ means that the vector $a_i^\tau$ has components $a_{i,\nu}^\tau$ only for $\nu \in \tau$, while $c_j^\sigma \in \mathbb{K}^\sigma$ has coefficients $c_{j,\nu}^\sigma$ only for $\nu \in \sigma$.

The corresponding $\mathcal{R}$-matrices are elements of the tensor vector space

$$\mathcal{V}(b) = \operatorname{span}\{[a_i^\tau, c_j^\sigma] : 1 \leq i \leq k(\tau), 1 \leq j \leq k(\sigma)\} = \mathcal{V}_a(\tau) \times \mathcal{V}_c(\sigma) \quad \text{for } b = \tau \times \sigma. \tag{2.15}$$

In our model case, the clusters $\tau, \sigma$ of $b = \tau \times \sigma$ belong to the same level. If we make the natural assumption that the rank $k$ is a function $k_\ell$ of the level $\ell$ only, $k(\tau) = k(\sigma) = k_{level(b)} =: k(b)$ follows.

The storage requirements are less than for $\mathcal{H}$-matrices. Since the vectors $a_i^\tau, c_j^\sigma$ are pre-defined, only the coefficients with respect to the basis $\{[a_i^\tau, c_j^\sigma]\}$ for $\mathcal{V}(b)$ from (2.15) are to be stored.

**Remark 2.4** *Let (2.14) be given. It needs $k(\tau) * k(\sigma)$ coefficients $\zeta_{ij}$ to code an $\mathcal{R}$-matrix $\sum_{i,j} \zeta_{ij}[a_i^\tau, c_j^\sigma]$.*

[2] The "hierarchical bases" which appear in our context, have another hierarchical structure than the hierarchical bases known from the finite element method. Also the hierarchical structure of wavelet bases is different.

### 2.4.2 Restrictions

Consider a cluster $\tau \in T$ being not a leaf. Its sons are denoted by $\tau', \tau''$ (the tree of the model problem is binary). The decomposition $\tau = \tau' \cup \tau''$ describes a block partitioning of the vector $a_i^\tau$ into the block-vectors

$$(a^\tau_{i,\nu})_{\nu\in\tau'} = R_a^{\tau',\tau}\, a_i^\tau \quad \text{and} \quad (a^\tau_{i,\nu})_{\nu\in\tau''} = R_a^{\tau'',\tau}\, a_i^\tau. \tag{2.16}$$

The *restriction operator* $R_a^{\tau',\tau}$ denotes the mapping from the full vector into a block-vector. Conversely, we can represent the vector $a_i^\tau$ as the composition

$$a_i^\tau = \begin{pmatrix} R_a^{\tau',\tau}\, a_i^\tau \\ R_a^{\tau'',\tau}\, a_i^\tau \end{pmatrix}, \tag{2.17}$$

if we first enumerate the indices of $\tau'$ and then those of $\tau''$.

Similarly, the restrictions $R_c^{\sigma',\sigma}$ are defined for the row vectors $c_j^\sigma$ and yield

$$c_j^\sigma = \begin{pmatrix} R_c^{\sigma',\sigma}\, c_j^\sigma \\ R_c^{\sigma'',\sigma}\, c_j^\sigma \end{pmatrix}. \tag{2.18}$$

### 2.4.3 Consistency Conditions

Let $\tau, \tau', \tau'' \in T$ be as before. The *consistency relation* between the spaces $\mathcal{V}_a(\tau)$ and $\mathcal{V}_a(\tau'), \mathcal{V}_a(\tau'')$ is

$$\mathcal{V}_a(\tau') = R_a^{\tau',\tau}\, \mathcal{V}_a(\tau), \qquad \mathcal{V}_a(\tau'') = R_a^{\tau'',\tau}\, \mathcal{V}_a(\tau). \tag{2.19}$$

Similarly, we require the analogous relations for the spaces $\mathcal{V}_c(\sigma)$, $\mathcal{V}_c(\sigma')$, $\mathcal{V}_c(\sigma'')$ :

$$\mathcal{V}_c(\sigma') = R_c^{\sigma',\sigma}\, \mathcal{V}_c(\sigma), \qquad \mathcal{V}_c(\sigma'') = R_c^{\sigma'',\sigma}\, \mathcal{V}_c(\sigma). \tag{2.20}$$

One important conclusion from $R_a^{\tau',\tau}\, \mathcal{V}_a(\tau) \subseteq \mathcal{V}_a(\tau')$ is

**Remark 2.5** *It is not necessary to store the vectors $a_i^\tau$ explicitly. Instead, one can store the coefficients $\alpha_{ij}^{\tau,\tau'}$ of the representation*

$$R_a^{\tau',\tau} a_i^\tau = \sum_{j=1}^{k(\tau')} \alpha_{ij}^{\tau,\tau'} a_j^{\tau'} \qquad \textit{for } 1 \le i \le k(\tau) \tag{2.21}$$

*and the analogously defined coefficients $\alpha_{ij}^{\tau,\tau''}$.*

The other direction $\mathcal{V}_a(\tau') \subseteq R_a^{\tau',\tau}\, \mathcal{V}_a(\tau)$ implies

**Remark 2.6** *The dimension $k$ must be a monotone function of the vertices, i.e., $k(\tau') \le k(\tau)$ if $\tau'$ is a son of $\tau$. If $k = k_\ell$ depends on the level $\ell$ as discussed above, $k_{\ell+1} \le k_\ell$ holds.*

### 2.4.4 Normal Form

Among all vectors $\{a_i^\tau : 1 \le i \le k(\tau), \tau \in T\}$ satisfying (2.19) and (2.21), we can choose a basis in such a way that the sum in (2.21) runs over $1 \le j \le \min(i, k(\tau'))$, i.e.,

$$R_a^{\tau',\tau} a_i^\tau = \sum\nolimits_{j=1}^{\min(i,k(\tau'))} \alpha_{ij}^{\tau,\tau'} a_j^{\tau'} \qquad \text{for } 1 \le i \le k(\tau). \tag{2.22}$$

Similarly,

$$R_c^{\sigma',\sigma} c_j^\sigma = \sum\nolimits_{i=1}^{\min(j,k(\sigma'))} \gamma_{ji}^{\sigma,\sigma'} c_i^{\sigma'} \qquad \text{for } 1 \le j \le k(\sigma). \tag{2.23}$$

Furthermore, the vectors could be chosen to be orthonormal, i.e., $\langle a_i^\tau, a_j^\tau \rangle = \delta_{ij}$ for $1 \le i, j \le k(\tau)$ with Kronecker's symbol $\delta_{ij}$. However, it is even more convenient if the respective bases $\{a_i^\tau\}$ and $\{c_j^\tau\}$ of $\mathcal{V}_a(\tau)$ and $\mathcal{V}_c(\tau)$ (which may be different!) are *bi-orthonormal*, i.e.,

$$\langle a_i^\tau, c_j^\tau \rangle = \delta_{ij} \qquad \text{for } 1 \le i, j \le k(\tau). \tag{2.24}$$

Finally, we remark that for level $p$, where all clusters $\tau$ contain only one index, $k(\tau) = 1$ holds and all basis vectors are the unit vector $a_1^\tau = c_1^\tau = (1)$.

### 2.4.5 Case of Constant $k(b)$

The simplest case is a constant rank $k(\tau) = k_{const}$. Since we called $\{a_i^\tau : 1 \le i \le k(\tau)\}$ a basis, $\dim \mathcal{V}_a = k(\tau)$ holds and thus $\#\tau \ge k(\tau)$ is required. Therefore $k(\tau) = k_{const}$ cannot hold for small blocks of level $\ell$, where $\#\tau = 2^{p-\ell} < k_{const}$. Hence, the exact requirement is

$$k(\tau) = \min\{k_{const}, 2^{p-level(\tau)}\} \qquad \text{for all } \tau \in T. \tag{2.25}$$

If the $\mathcal{H}$-matrix $M'$ has to approximate an BEM matrix $M$ up to the error $O(h^\gamma)$ with $\gamma$ being the consistency order, the choice of $k_{const}$ should be of the order $k_{const} = \log n = p = p - level(I)$.

### 2.4.6 Case of Variable $k(b)$

Formula (2.25) gives a first advise to choose a smaller rank $k(\tau)$ for small blocks. As we shall see later, it is reasonable to choose $k(\tau)$ due to a rule like $k(\tau) := p - level(\tau) + 1$ or, more general, $k\,(\tau) := \alpha\,(p - level\,(\tau)) + \beta$ for some $\alpha, \beta \ge 1$ (see (4.8) below).

It is a result of the approximation considerations in Chapter 4 that this choice does not deteriorate the approximation quality. On the other hand, it is obvious that for the larger number of smaller blocks we have to deal with less coefficients $\zeta_{ij}$ (see Remark 2.4) and $\alpha_{ij}^{\tau,\tau'}$, $\gamma_{ji}^{\sigma,\sigma'}$ (see (2.22), (2.23)). Therefore, a smaller rank yields lower costs for the storage and for the various arithmetic operations. In this context, the key inequality is (2.26) expressing the fact that the sum over all vertices weighted by $k(\tau)^\gamma$ for any (fixed) $\gamma$ remains bounded linearly in $n$ :

$$\sum_{\tau \in T} k(\tau)^\gamma = \sum_{\ell=0}^{p} 2^\ell (\alpha\,(p-\ell) + \beta)^\gamma \;\sim\; n \qquad \text{for all } \gamma \in \mathbb{N}. \tag{2.26}$$

In the variable case, conditions (2.19) and (2.20) are nontrivial, since the restriction of $k(\tau)$ must lead to a vector space $\mathcal{V}_a(\tau')$ of a *lower* dimension $k(\tau')$. Here, it is interesting to consider equations (2.17) and (2.21) as the fundamental construction of the basis vectors of $\mathcal{V}_a(\tau)$.

# 3 Storage and Complexity Bounds

Next we prove that the storage size is $O(n)$ without any logarithmic factor (§3.1). Then we describe the matrix-vector multiplication algorithm in §3.2 and show its $O(n)$ complexity.

## 3.1 Storage Requirements

According to Remark 2.5, we have to store the matrices

$$\begin{aligned} A^{\tau,\tau'} &:= (\alpha_{ij}^{\tau,\tau'})_{1 \le i \le k(\tau), 1 \le j \le k(\tau')} && \text{for } \tau, \tau' \in T,\ \tau' \text{ son of } \tau, \\ C^{\sigma,\sigma'} &:= (\gamma_{ji}^{\sigma,\sigma'})_{1 \le j \le k(\sigma), 1 \le i \le k(\sigma')} && \text{for } \sigma, \sigma' \in T,\ \sigma' \text{ son of } \sigma, \end{aligned} \tag{3.1}$$

of the size $k_\ell \times k_{\ell+1}$, where $\ell = level(\tau) = level(\sigma) \in \{0, 1, \ldots, p-1\}$ (cf. (2.22), (2.23)). There are $2^{\ell+1}$ different pairs $\tau, \tau'$ with $\ell = level(\tau)$ and $\tau'$ son of $\tau$. Assuming $k_\ell \le \alpha\,(p-\ell) + \beta$ as proposed in §2.4.6 and used in Definition 4.7 below, the required storage amounts to

$$\sum\nolimits_{\ell=0}^{p-1} 2^{\ell+1} k_\ell k_{\ell+1} \le \sum\nolimits_{\ell=0}^{p-1} 2^{\ell+1} (\alpha\,(p-\ell) + \beta)(\alpha\,(p-\ell-1) + \beta).$$

Thanks to (2.26), we obtain

**Remark 3.1** *The storage needed for all transfer matrices $A^{\tau,\tau'}$, $C^{\sigma,\sigma'}$ is proportional to $n$.*

Next we consider the storage of the coefficients

$$Z^b = (\zeta_{ij}^b)_{1 \le i,j \le k(b)} \qquad \text{for } b = \tau \times \sigma \in P_2$$

of the block-matrix $M^b = \sum_{i,j} \zeta_{ij}^b [a_i^\tau, c_j^\sigma]$ (cf. Remark 2.4). The total storage is $\sum_{b \in P_2} k(b)^2$. Let $P_2\,(\ell) := \{b \in P_2 : level\,(b) = \ell\}$. The recursions discussed in [3] show $\#P_2\,(\ell) \sim 2^\ell$. Hence, $\sum_{b \in P_2} k(b)^2 \le C \sum_{\ell=0}^{p} (p-\ell)^2 2^\ell \sim n$ proves

**Remark 3.2** *The storage needed for all block coefficients matrices $Z^b$, $b \in P_2$, is bounded by $O(n)$.*

## 3.2 Description of the Fast Matrix-Vector Multiplication Algorithm

The fast matrix-vector multiplication algorithm is performed in three steps: (i) forward transformation (see §3.2.2), (ii) block-multiplication phase (see §3.2.3), and (iii) backward transformation (see §3.2.4). All steps are shown to require only $O(n)$ operations, hence the matrix-vector multiplication algorithm has linear complexity.

### 3.2.1 Block-Matrix times Vector

First we consider the multiplication of a block $M^b$, $b = \tau \times \sigma \in P_2$, with a vector $\hat{x}_\sigma \in \mathcal{V}_a(\sigma)$. We denote the coefficient vector with respect to the basis $\{a_i^\sigma : 1 \le i \le k(\sigma)\}$ by $\hat{\mathbf{x}}_\sigma$, i.e.,

$$\hat{x}_\sigma = \sum\nolimits_{i=1}^{k(\sigma)} \hat{\mathbf{x}}_{\sigma,i}\, a_i^\sigma . \tag{3.2}$$

**Remark 3.3** *(a) Let $b = \tau \times \sigma \in P_2$, $M^b = \sum_{i,j} \zeta_{ij}^b [a_i^\tau, c_j^\sigma]$ with $Z^b = (\zeta_{ij}^b)_{1 \le i,j \le k(b)}$ and $\hat{x}_\sigma \in \mathcal{V}_a(\sigma)$. Then $y_\tau = M^b \hat{x}_\sigma$ has the coefficient vector $\mathbf{y}_\tau = Z^b \hat{\mathbf{x}}_\sigma$ with respect to the basis $\{a_i^\tau : 1 \le i \le k(\tau)\}$.*

*(b) Let $x_\sigma \in \mathbb{K}^\sigma$ have the decomposition $x_\sigma = \hat{x}_\sigma + x_\sigma^\perp$ with $\hat{x}_\sigma \in \mathcal{V}_a(\sigma)$ and $x_\sigma^\perp \perp \mathcal{V}_c(\sigma)$. Then $M^b x_\sigma = M^b \hat{x}_\sigma$ holds and part (a) applies to $M^b \hat{x}_\sigma$.*

*Proof.* Part (b) is trivial. For (a) note that $M_\sigma^b \hat{x}_\sigma = \left(\sum_{i,j} \zeta_{ij}^b [a_i^\tau, c_j^\sigma]\right) \left(\sum_h \hat{\mathbf{x}}_{\sigma,h}\, a_h^\sigma\right) = \sum_{i,j} \zeta_{ij}^b \hat{\mathbf{x}}_{\sigma,j} a_i^\tau$ because of $\left\langle c_j^\sigma, \sum_h \hat{\mathbf{x}}_{\sigma,h}\, a_h^\sigma \right\rangle = \hat{\mathbf{x}}_{\sigma,j}$ (cf. (2.24)). ■

### 3.2.2 Forward Transformations

Let a vector $x \in \mathbb{K}^I$ be given. $Mx$ is to be computed, where $M$ is an $\mathcal{H}^2$-matrix. Due to Remark 3.3, we have to represent the block-vector $x_\sigma := (x_i)_{i \in \sigma}$ as the sum $x_\sigma = \hat{x}_\sigma + x_\sigma^\perp$, where the coefficient vector $\hat{\mathbf{x}}_\sigma$ of $\hat{x}_\sigma$ must be available. Since $M$ contains blocks of all levels, we need the coefficient vectors $\hat{\mathbf{x}}_\sigma$ for all $\sigma \in T$.

We introduce the notation $T(\ell) := \{\tau \in T : \mathit{level}\,(\tau) = \ell\}$ for all $0 \le \ell \le p$.

The following computations start at level $p$ and proceed to level $0$ :

- *Start at level $\ell = p$.* Let $\sigma = \{s\} \in T(p)$. The one-dimensional block-vector $x_\sigma = (x_i)_{i \in \sigma} = (x_s)$ is identical to the coefficient vector $\mathbf{x}_\sigma$, since the basis is the unit vector $a_1^\sigma = (1)$. Hence, $\hat{\mathbf{x}}_\sigma$ is known without any computation and $x_\sigma^\perp = 0$ holds.

- *Recursion $\ell + 1 \to \ell$ for $p > \ell \ge 0$.* Assume that the coefficient vectors $\hat{\mathbf{x}}_\tau$ of the first summand in $x_\tau = \hat{x}_\tau + x_\tau^\perp$ are already computed for all $\tau \in T(\ell+1)$. For all $\sigma \in T(\ell)$ the new coefficient vectors $\hat{\mathbf{x}}_\sigma$ are constructed as follows. Let $\sigma', \sigma'' \in T(\ell+1)$ be the sons of $\sigma$. The decomposition from level $\ell+1$ yields
$$x_\sigma = x_\sigma^I + x_\sigma^{II} \quad \text{with } x_\sigma^I := \begin{bmatrix} \hat{x}_{\sigma'} \\ \hat{x}_{\sigma''} \end{bmatrix} \text{ and } x_\sigma^{II} := \begin{bmatrix} x_{\sigma'}^\perp \\ x_{\sigma''}^\perp \end{bmatrix}.$$

  The latter term $x_\sigma^{II}$ is orthogonal to $\mathcal{V}_c(\sigma)$, since $\langle c_j^\sigma, x_\sigma^{II} \rangle = \left\langle R_c^{\sigma',\sigma} c_j^\sigma, x_{\sigma'}^\perp \right\rangle + \left\langle R_c^{\sigma'',\sigma} c_j^\sigma, x_{\sigma''}^\perp \right\rangle = 0 + 0 = 0$. The first term is to be split into $x_\sigma^I = \hat{x}_\sigma + x_\sigma^{III}$ determined by $\hat{x}_\sigma \in \mathcal{V}_a(\sigma)$, $x_\sigma^{III} \perp \mathcal{V}_c(\sigma)$; then $x_\sigma = \hat{x}_\sigma + x_\sigma^\perp$ with $x_\sigma^\perp := x_\sigma^{II} + x_\sigma^{III}$ is the desired decomposition. The entries of the coefficient vectors $\hat{\mathbf{x}}_\sigma$ are determined by $\hat{\mathbf{x}}_{\sigma,j} = \langle c_j^\sigma, \hat{x}_\sigma \rangle = \langle c_j^\sigma, x_\sigma^I \rangle$. Using the construction (2.18) of $c_j^\sigma$ (cf. (2.24)), we obtain
$$\hat{\mathbf{x}}_{\sigma,j} = \langle c_j^\sigma, x_\sigma^I \rangle = \left\langle \begin{bmatrix} R_c^{\sigma',\sigma}\, c_j^\sigma \\ R_c^{\sigma'',\sigma}\, c_j^\sigma \end{bmatrix}, \begin{bmatrix} \hat{x}_{\sigma'} \\ \hat{x}_{\sigma''} \end{bmatrix} \right\rangle = \left\langle R_c^{\sigma',\sigma}\, c_j^\sigma, \hat{x}_{\sigma'} \right\rangle + \left\langle R_c^{\sigma'',\sigma}\, c_j^\sigma, \hat{x}_{\sigma''} \right\rangle .$$

  Inserting the representation (2.23) of $R_c^{\sigma',\sigma}\, c_j^\sigma$ and $\hat{x}_{\sigma'} = \sum_h \hat{\mathbf{x}}_{\sigma',h}\, a_h^{\sigma'}$, we result in[3]
$$\left\langle R_c^{\sigma',\sigma}\, c_j^\sigma, \hat{x}_{\sigma'} \right\rangle = \left\langle \sum\nolimits_i \gamma_{ji}^{\sigma,\sigma'} c_i^{\sigma'}, \sum\nolimits_h \hat{\mathbf{x}}_{\sigma',h}\, a_h^{\sigma'} \right\rangle = \sum\nolimits_i \gamma_{ji}^{\sigma,\sigma'} \hat{\mathbf{x}}_{\sigma',i} = (C^{\sigma,\sigma'} \hat{\mathbf{x}}_{\sigma'})_j$$

  with $C^{\sigma,\sigma'}$ defined in (3.1). Since the second term is similar, the final representation is
$$\hat{\mathbf{x}}_\sigma = C^{\sigma,\sigma'} \hat{\mathbf{x}}_{\sigma'} + C^{\sigma,\sigma''} \hat{\mathbf{x}}_{\sigma''}. \tag{3.3}$$

[3] Without the biorthogonality (2.24), equation (3.3) is obtained with another matrix $\hat{C}^{\sigma,\sigma'}$.

By assumption, the coefficient vectors $\hat{\mathbf{x}}_{\sigma'}, \hat{\mathbf{x}}_{\sigma''}$ are known. Therefore, only two matrix-vector multiplications by the $k_\ell \times k_{\ell+1}$-matrices $C^{\sigma,\sigma'}$,$C^{\sigma,\sigma''}$ are needed to compute the desired coefficient vectors $\hat{\mathbf{x}}_\sigma$ for $\sigma \in T(\ell)$.

Since the number of operations needed in (3.3) is proportional to the number of entries in $C^{\sigma,\sigma'}$, $C^{\sigma,\sigma''}$, Remark 3.1 implies

**Remark 3.4** *The performance of (3.3) for all $\sigma \in T(\ell)$, $\ell = p-1, \dots, 0$, requires $O(n)$ operations and yields the coefficient vectors $\hat{\mathbf{x}}_\sigma$ for all $\sigma \in T$.*

### 3.2.3 Multiplication Phase

For all blocks $M^b$, $b = \tau \times \sigma \in P_2$, the intermediate products $y_\tau^b := M^b x_\sigma$ are to be computed, i.e., according to Remark 3.3 the coefficient vectors $\mathbf{y}_\tau^b = Z^b \hat{\mathbf{x}}_\sigma$ of $y_\tau^b$ are to be computed. The upper index $b$ is used in $\mathbf{y}_\tau^b$, since for the same $\tau \in T$ several $\mathbf{y}_\tau^b$ for different $b$ may occur (namely $b = \tau \times \sigma$ and $b^* = \tau \times \sigma^*$ with $\sigma \neq \sigma^*$).

The number of operations for all products $Z^b \hat{\mathbf{x}}_\sigma$, $b = \tau \times \sigma \in P_2$, is again proportional to the entries in all matrices $Z^b$. Therefore, Remark 3.2 implies

**Remark 3.5** *The matrix-vector multiplications $\mathbf{y}_\tau^b := Z^b \hat{\mathbf{x}}_\sigma$ for all $b = \tau \times \sigma \in P_2$ requires $O(n)$ operations.*

### 3.2.4 Backward Transformations

In the final step we have to gather all partial results $\mathbf{y}_\tau^b$ obtained in the previous phase. Here we use a backward transformation starting at level $\ell = 0$ and proceeding to $\ell = p$. On each level $\ell$, we compute $\mathbf{y}_\tau$ for all $\tau \in T(\ell)$, where $\mathbf{y}_\tau$ is the coefficient vector for the sum $y_\tau$ defined by

$$y_{\tau,i} := \sum_{b'=\tau'\times\sigma'\in P_2 \text{ with } \tau'\supseteq\tau} (y_{\tau'}^{b'})_i \qquad \text{for } i \in \tau.$$

Note that all $\tau' \supseteq \tau$ belong to some $T(\ell')$ with $\ell' \leq \ell$. As before, we set $P_2(\ell) := \{b \in P_2 \text{ with } level(b) = \ell\}$.

- *Start at level $\ell = 0$.* Since the partitioning $P_2$ contains no block of level 0 (the only level-0-block is $I \times I$ and not admissible), the start is given by

$$\mathbf{y}_I := 0,$$

where $I \in T(0)$ is the only cluster of level 0.

- *Recursion $\ell \to \ell + 1$ for $p > \ell \geq 0$.* Assume that the coefficient vectors $\mathbf{y}_\tau$ for all $\tau \in T(\ell)$ are already computed. Let $\tau', \tau'' \in T(\ell+1)$ be the sons of some $\tau \in T(\ell)$. The vector $y_\tau = \sum_i \mathbf{y}_{\tau,i} a_i^\tau$ corresponding to $\mathbf{y}_\tau$ equals $\begin{bmatrix} R_a^{\tau',\tau}\, y_\tau \\ R_a^{\tau'',\tau}\, y_\tau \end{bmatrix}$ by the definition of $R_a^{\tau',\tau}$ and $R_a^{\tau'',\tau}$. The coefficient vectors $\hat{\mathbf{y}}_{\tau'}$ and $\hat{\mathbf{y}}_{\tau''}$ of $R_a^{\tau',\tau}\, y_\tau$ and $R_a^{\tau'',\tau}\, y_\tau$ are given by

$$\hat{\mathbf{y}}_{\tau'} = (A^{\tau,\tau'})^T \mathbf{y}_\tau, \qquad \hat{\mathbf{y}}_{\tau''} = (A^{\tau,\tau''})^T \mathbf{y}_\tau, \tag{3.4}$$

as one concludes from $R_a^{\tau',\tau}\, y_\tau = R_a^{\tau',\tau} \sum_i \mathbf{y}_{\tau,i} a_i^\tau = \sum_i \mathbf{y}_{\tau,i} R_a^{\tau',\tau} a_i^\tau = \sum_{i,j} \mathbf{y}_{\tau,i} \alpha_{ij}^{\tau,\tau'} a_j^{\tau'}$.

Next, we have to add all contributions from blocks of level $\ell + 1$ :

$$\mathbf{y}_{\tau'} = \hat{\mathbf{y}}_{\tau'} + \sum_{\sigma' \text{ with } b'=\tau'\times\sigma'\in P_2(\ell+1)} \mathbf{y}_{\tau'}^{b'}. \tag{3.5}$$

**Remark 3.6** *The number of operations involved in the backward transformations (3.4) and (3.5) is $2\sum_{\ell=0}^{p-1} k_\ell k_{\ell+1} \# T(\ell+1) + \sum_{\ell=0}^{p-1} \# P_2(\ell+1) = 2\sum_{\ell=0}^{p-1} k_\ell k_{\ell+1} 2^{\ell+1} + \# P_2 \sim n$.*

- *Result at level $\ell = p$.* The resulting coefficient vectors $\mathbf{y}_\tau$ for all one-element clusters $\tau = \{i\} \in T(p)$ coincide with the component $y_i$ of $y = Mx$. Therefore, the matrix-vector multiplication is completed.

### 3.3 Other Matrix Operations

Different from general $\mathcal{H}$-matrices, the sum of two $\mathcal{H}^2$-matrices (with the same partitioning and the same hierarchical bases) can be performed *exactly*. Since only the matrices $Z^b$, $b \in P_2$, are to be added, the cost is clearly $O(n)$.

We do not discuss the matrix-matrix multiplications in detail, but it may be mentioned that the product of two blocks is rather cheap since we have to perform scalar products of the form $\left\langle c_j^\sigma, a_i^\sigma \right\rangle$, which are trivial because of (2.24).

### 3.4 Constant $k(\tau)$

The proof of the following statement is left to the reader.

**Proposition 3.7** *Let $k_{const} \in \{1, \dots, n\}$. Choose the rank $k(\tau)$ according to (2.25). Then the storage size of $A^{\tau,\tau'}$, $C^{\sigma,\sigma'}$, $Z^b$ (see Remarks 3.1-2) as well as the matrix-vector multiplication cost amounts to $O(n \cdot k_{const})$.*

## 4 Approximation by Variable Order

In this section, we will explain how the approximation of the integral operator (2.1) via $\mathcal{H}^2$-matrices with variable order $k$ can be realised.

### 4.1 Galerkin Matrix

Let $b = \tau \times \sigma \in P_2$ and $(i,j) \in b$. The matrix element $m_{ij}$ is defined in (2.3) by $\int_{J_i}\int_{J_j} s(x,y)dxdy$, where $s(x,y) = \log(|x-y|)$. If we find an expansion

$$s(x,y) \approx \tilde{s}(x,y) := \sum_{\alpha,\beta=1}^{k(b)} \gamma_{\alpha,\beta}\varphi_\alpha(x)\psi_\beta(y)$$

which is sufficiently accurate on the rectangle $J(\tau) \times J(\sigma)$ (cf. (2.7)), the Galerkin matrix based on $\tilde{s}$ instead of $s$ has the entries

$$\tilde{m}_{ij} := \sum_{\alpha,\beta=1}^{k(b)} \gamma_{\alpha,\beta} a_{\alpha,i} c_{\beta,j} \quad \text{for } (i,j) \in b \text{ with}$$

$$a_{\alpha,i} = \int_{J_i} \varphi_\alpha(x)dx \text{ and } c_{\beta,j} = \int_{J_j} \psi_\beta(y)dy.$$

Obviously, the block matrix $(\tilde{m}_{ij})_{(i,j)\in b} = \sum_{\alpha,\beta=1}^{k(b)} \gamma_{\alpha,\beta} a_\alpha c_\beta^T$ is of the desired form, if the spaces spanned by $a_\alpha$ or $c_\beta$ satisfy the respective consistency conditions (2.19) or (2.20).

Concerning the consistency conditions, we state the following criterion.

**Remark 4.1** *For $b = \tau \times \sigma$, let $\varphi_\alpha^\tau(x)$ and $\psi_\beta^\sigma(y)$, $1 \le \alpha, \beta \le k(b)$, be the functions involved in the approximation of $s(x,y)$ by*

$$\tilde{s}(x,y) := \sum_{\alpha,\beta=1}^{k(b)} \gamma_{\alpha,\beta}\varphi_\alpha^\tau(x)\psi_\beta^\sigma(y) \quad \textit{for } (x,y) \in J(\tau) \times J(\sigma).$$

*Set $\mathcal{W}_a(\tau) := \operatorname{span}\{\varphi_\alpha^\tau : 1 \le \alpha \le k(\tau)\}$ and $\mathcal{W}_c(\sigma) := \operatorname{span}\{\psi_\beta^\sigma : 1 \le \alpha \le k(\sigma)\}$. Then the condition*

$$\mathcal{W}_a(\tau') = \{\varphi|_{J(\tau')} : \varphi \in \mathcal{W}_a(\tau)\} \qquad \textit{for all sons } \tau' \textit{ of } \tau \tag{4.1}$$

*implies the consistency condition (2.19). Similar for (2.20).*

Note that $\dim \mathcal{V}_a(\tau) \le \dim \mathcal{W}_a(\tau)$ holds, where the strict inequality may occur. Since nothing is to be discussed about the $1 \times 1$-blocks $b$ in $P_2$ (belonging to level $p$), we restrict our considerations to the subset of the "far field" blocks:

$$P_2^{far} := \{b \in P_2 : \ level(b) < p\}. \tag{4.2}$$

In the following, the approximation $\tilde{s}(x,y)$ is based on Taylor expansions[4] of the kernel functions. However,

[4] Other expansions based on projections or interpolations are possible.

in order to satisfy (4.1), the arising polynomials of degree $k(\tau)-1$ must be replaced by very particular functions.

## 4.2 Taylor Expansions

First, we have to introduce some notations. $J := [0,1]$ denotes the integration domain.

In order to simplify the notation, we replace the rank $k(\tau)$ by $k(\tau)+1$, since then $k(\tau)$ also coincides with the polynomial degree. In other words, the summation $\sum_{i=1}^{k(\tau)}$ is replaced by $\sum_{i=0}^{k(\tau)}$.

**Definition 4.2** *Let $\omega \subseteq J$ and $\check{c}_\omega$ be the smallest interval containing $\omega$. The Čebyšev centre $z_\omega$ of $\omega$ is the midpoint of $\check{c}_\omega$ and the Čebyšev radius $r_\omega$ equals the halved interval length of $\check{c}_\omega$.*

**Definition 4.3** *For $\omega, w \subseteq J$, the difference domain $d_{\omega,w}$ is given by*

$$d_{\omega,w} := \omega - w := \{z \in \mathbb{R} : \exists\, (x,y) \in \omega \times w \text{ with } z = x - y\}.$$

**Definition 4.4** *Let $\omega \subseteq J$ and $f : \omega \to \mathbb{R}$ be sufficiently smooth. The Taylor operator $T_\omega^{(m)}$ of order $m \in \mathbb{N}$ is given by*

$$T_\omega^{(m)}[f](x) = \sum_{\nu=0}^{m-1} \frac{1}{\nu!} f^{(\nu)}(z_\omega)(x - z_\omega)^\nu .$$

**Lemma 4.5** *Let $\omega, w \subseteq J$ satisfy $\eta \operatorname{dist}(\omega, w) \geq r_\omega + r_w$ with some $\eta \in (0,1]$. On the difference domain $d = d_{\omega,w}$, the Taylor expansion of the function $s : d \to \mathbb{R}$,*

$$s(z) = \log|z| \tag{4.3}$$

*about $z_d$ of order $m$ satisfies*

$$\left| s(z) - T_d^{(m)}[s](z) \right| \leq \begin{cases} |\log \operatorname{dist}(\omega, w)| & m = 0 \\ \eta^m / m & m > 0 \end{cases} \qquad \text{for all } z \in d. \tag{4.4}$$

*Proof.* The remainder $R_d^{(m)}(z)$ of the Taylor expansion can be estimated (by using $r_d = r_\omega + r_w$)

$$\begin{aligned} \left| R_d^{(m)}(z) \right| &\leq r_d^m \sup_{\xi \in d} \frac{\left| s^{(m)}(\xi) \right|}{m!} = r_d^m \sup_{\xi \in d} \begin{cases} |\log|\xi|| & \text{if } m = 0 \\ \frac{1}{m}|\xi|^{-m} & \text{if } m > 0 \end{cases} \\ &\leq (r_\omega + r_w)^m \begin{cases} |\log \operatorname{dist}(\omega, w)| & (m = 0) \\ \frac{1}{m} \operatorname{dist}^{-m}(\omega, w) & (m > 0) \end{cases} \leq \begin{cases} |\log \operatorname{dist}(\omega, w)| & (m = 0) \\ \eta^m / m & (m > 0) \end{cases} . \end{aligned}$$

■

**Corollary 4.6** *Let $\omega, w$ be as in Lemma 4.5. The Taylor approximation of the kernel function* $\log(|x-y|)$ *is denoted by $T_d^{(m)}[s](x-y)$. The explicit representation*

$$T_d^{(m)}[s](x-y) = \sum_{\nu+\mu \leq m-1} \kappa_{\omega,w}^{(\nu,\mu)} \Phi_\omega^{(\nu)}(x) \Phi_w^{(\mu)}(y) \tag{4.5}$$

*holds with $\Phi_\omega^{(\nu)}(x) = (x - z_\omega)^\nu$ and $\kappa_{\omega,w}^{(\nu,\mu)} = (-1)^{\nu+\mu} s^{(\nu+\mu)}(z_d) / (\mu!\nu!)$.*

*Proof.* Reorganising sums and products results in

$$\begin{aligned} T_d^{(m)}[s](x-y) &= \sum_{\nu=0}^{m-1} \frac{1}{\nu!} s^{(\nu)}(z_d)(z - z_d)^\nu = \sum_{\nu=0}^{m-1} \sum_{\mu=0}^{\nu} \frac{(-1)^{\nu-\mu}}{\nu!} s^{(\nu)}(z_d) \binom{\nu}{\mu} (x - z_\omega)^\mu (y - z_w)^{\nu-\mu} \\ &= \sum_{\mu=0}^{m-1} \sum_{\nu=0}^{m-1-\mu} \frac{(-1)^{\nu+\mu} s^{(\nu+\mu)}(z_d)}{\mu!\,\nu!} (x - z_\omega)^\mu (y - z_w)^\nu . \end{aligned}$$

■

## 4.3 The Hierarchical Bases Construction

The essential step in the definition of the variable order approximation method is the definition of the restriction operators. In order to illustrate the underlying idea, we consider a cluster $\tau$ of the cluster tree with sons $\tau', \tau''$. Assume that the spaces $\mathcal{W}_a(\tau')$ and $\mathcal{W}_a(\tau'')$ with bases $\left\{\tilde{\Phi}_{\tau'}^{(i)}\right\}_{i=0}^{k(\tau')}$ and $\left\{\tilde{\Phi}_{\tau''}^{(i)}\right\}_{i=0}^{k(\tau'')}$ are already defined. Then, every function in $\mathcal{W}_a(\tau)$ has the representation

$$u(x) = \begin{cases} \sum_{\mu=0}^{k(\tau')} a_{\tau'}^{(\mu)}\, \tilde{\Phi}_{\tau'}^{(\mu)}(x) & \text{for } x \in J(\tau'), \\ \sum_{\mu=0}^{k(\tau'')} a_{\tau''}^{(\mu)}\, \tilde{\Phi}_{\tau''}^{(\mu)}(x) & \text{for } x \in J(\tau''). \end{cases} \tag{4.6}$$

Then Remark 4.1 guarantees the consistency condition (2.19). Because of $\mathcal{W}_c(\tau) = \mathcal{W}_a(\tau)$, also (2.20) holds.

The difficulty is that (4.6) is not able to represent monomials of degree $k(\tau)$ if $k(\tau) > k(\tau') = k(\tau'')$. Instead, we are looking for approximations of $\Phi_\tau^{(\nu)} := (x - z_\tau)^\nu$ (see Corollary 4.6) by functions $\tilde{\Phi}_\tau^{(\nu)}$ belonging to $\mathcal{W}_a(\tau)$, i.e., having a representation (4.6).

The basis functions $\tilde{\Phi}_\tau^{(\nu)}$ will be chosen as the composition of the *local* Taylor polynomials of degree $k(\tau)$ in $J(\tau')$ and $J(\tau'')$ obtained by expansions of the true monomial $\Phi_\tau^{(\nu)}$ around the respective Čebyšev centres $z_{\tau'}$ and $z_{\tau''}$. The coefficients $a_t^{(\nu,\mu)} := a_t^{(\mu)}[\Phi_\tau^{(\nu)}]$, $t \in \{\tau', \tau''\}$, in (4.6) are

$$a_t^{(\nu,\mu)} := a_t^{(\mu)}[\Phi_\tau^{(\nu)}] = \frac{1}{\mu!}\partial_x^{(\mu)}\left[(x - z_\tau)^\nu\right]|_{x=z_t}. \tag{4.7}$$

We repeat that this construction yields the true monomial $\tilde{\Phi}_\tau^{(\nu)} = \Phi_\tau^{(\nu)}$ if the functions $\tilde{\Phi}_t^{(\mu)}$, $t \in \{\tau', \tau''\}$, in the right-hand side of (4.6) are the true monomials $\Phi_t^{(\mu)}$ and if $\nu \leq k(t)$, $t \in \{\tau', \tau''\}$. However, our variable order assumption leads to the case $\nu = k(\tau) > k(t)$.

**Definition 4.7** *The polynomial degree distribution depends on constants $\alpha, \beta \geq 0$ and is given by*

$$k(\tau) := \alpha\,(p - level\,(\tau)) + \beta. \tag{4.8}$$

*For the leaves $\tau \in T$ (i.e., $level\,(\tau) = p$), we put*

$$\tilde{\Phi}_\tau^{(\nu)}(x) = (x - z_\tau)^\nu \qquad \textit{for all } \nu \in \{0, 1, \ldots, \beta\}.$$

*Assume that the basis functions $\tilde{\Phi}_t^{(\nu)}$ are defined on all clusters $t \in T$ with $level\,(t) \geq \ell$. Then, for $\tau \in T$ with $level\,(\tau) = \ell - 1$, the basis functions $\tilde{\Phi}_\tau^{(\nu)}$ are given by*

$$\tilde{\Phi}_\tau^{(\nu)}(x) := \begin{cases} \sum_{\mu=0}^{k(\tau')} a_{\tau'}^{(\nu,\mu)} \tilde{\Phi}_{\tau'}^{(\mu)}(x) & \textit{for } x \in J(\tau'), \\ \sum_{\mu=0}^{k(\tau'')} a_{\tau''}^{(\nu,\mu)} \tilde{\Phi}_{\tau''}^{(\mu)}(x) & \textit{for } x \in J(\tau''), \end{cases} \tag{4.9}$$

*with $a_t^{(\nu,\mu)}$ from (4.7), where $\tau', \tau''$ are the sons of $\tau$.*

The approximation of the kernel function is given by replacing the Taylor polynomials $\Phi_\tau^{(\nu)}$ in (4.5) by the functions $\tilde{\Phi}_\tau^{(\nu)}$.

**Definition 4.8** *Let $\eta \in (0, 1]$. A block $b = \tau \times \sigma \in P_2$ is $\eta$-admissible if the following condition holds:*

$$\eta \operatorname{dist}(\tau, \sigma) \geq \max\{\operatorname{diam}\tau, \operatorname{diam}\sigma\}. \tag{4.10}$$

We remark that for the partitioning in (2.11) and (2.12) all $b \in P_2^{far}$ (cf. (4.2)) satisfy (4.10) with $\eta = 1$.

**Definition 4.9** *Let $b = \tau \times \sigma$ denote an $\eta$-admissible block. The approximation to the function $s$ as in (4.3) is given by*

$$\tilde{s}_b(x, y) := \sum_{\nu+\mu \leq k(b)} \kappa_{\tau,\sigma}^{(\nu,\mu)} \tilde{\Phi}_\tau^{(\nu)}(x)\, \tilde{\Phi}_\sigma^{(\mu)}(y) \tag{4.11}$$

*where $\kappa_{\tau,\sigma}^{(\nu,\mu)}$ are the Taylor coefficients from (4.5).*

We remark that the approximation of $s$ by $\tilde{s}_b$ in $J(\tau) \times J(\sigma)$ is not the optimal one. First, we could find better coefficients $a_t^{(\nu,\mu)}$ in (4.9) when we look for the best expansion with respect to the basis $\tilde{\Phi}_t^{(\mu)}$ (instead of $\Phi_t^{(\mu)}$). Second, in (4.11) we could allow all indices $0 \leq \nu, \mu \leq k(b)$ instead of $\nu + \mu \leq k(b)$.

## 4.4 Error Analysis

The error analysis of the variable order approximation algorithm consists of a local estimate of the error $s - \tilde{s}_b$ on admissible blocks and a global estimate of the consistency error. We begin with the local estimates. On an $\eta$-admissible block $b = \tau \times \sigma \in P_2$, the local approximation error is defined by

$$e_b(x,y) = s(x,y) - \tilde{s}_b(x,y) \qquad \text{for } (x,y) \in J(\tau) \times J(\sigma) \tag{4.12}$$

and its maximum norm by

$$\varepsilon_b := \|e_b\|_{\infty,b} = \sup_{(x,y)\in J(\tau)\times J(\sigma)} |e_b(x,y)|\,. \tag{4.13}$$

The expansion with the true Taylor polynomials defines the function

$$s_b(x,y) = \sum_{\nu+\mu\le k(b)} \kappa_{\tau,\sigma}^{(\nu,\mu)}\, \Phi_\tau^{(\nu)}(x)\, \Phi_\sigma^{(\mu)}(y)\,.$$

The error is split into

$$e_b = (s - s_b) + (s_b - \tilde{s}_b) =: e_b^I + e_b^{II}, \tag{4.14}$$

where $e_b^I$ is already estimated by (4.4). The error $e_b^{II}$ has the representation:

$$\begin{aligned} e_b^{II}(x,y) = & \sum_{\nu+\mu\le k(b)} \kappa_{\tau,\sigma}^{(\nu,\mu)} \left(\Phi_\tau^{(\nu)}(x) - \tilde{\Phi}_\tau^{(\nu)}(x)\right) \Phi_\sigma^{(\mu)}(y) \\ + & \sum_{\nu+\mu\le k(b)} \kappa_{\tau,\sigma}^{(\nu,\mu)} \tilde{\Phi}_\tau^{(\nu)}(x) \left(\Phi_\sigma^{(\mu)}(y) - \tilde{\Phi}_\sigma^{(\mu)}(y)\right) =: e_b^{III}(x,y) + e_b^{IV}(x,y)\,. \end{aligned} \tag{4.15}$$

The estimate of the difference $\Phi_\tau^{(\nu)} - \tilde{\Phi}_\tau^{(\nu)}$ plays the key role in the following error estimation.

**Lemma 4.10** *Choose $\alpha \ge 2$ and $\beta \ge \max\left\{\frac{6}{5}\alpha + 1, 2\alpha - 1\right\}$ in (4.8) and put*

$$\omega \ge \max\{2.5, \alpha - 1\}\,. \tag{4.16}$$

*Then, for $\tau \in T$ with $\ell = level(\tau)$, the difference $\Phi_\tau^{(\nu)} - \tilde{\Phi}_\tau^{(\nu)}$ can be estimated by*

$$\|\Phi_\tau^{(\nu)} - \tilde{\Phi}_\tau^{(\nu)}\|_{\infty,\tau} \le \lambda^{-k_\ell} \left(\omega 2^{-\ell}\right)^\nu, \qquad \text{where } \lambda = 3/2.$$

*Proof.* We define an intermediate approximation $\check{\Phi}_\tau^{(\nu)}$ to $\Phi_\tau^{(\nu)}$ by using the true Taylor polynomials in the right-hand side of

$$\check{\Phi}_\tau^{(\nu)}(x) := \begin{cases} \sum_{\mu=0}^{k(\tau')} a_{\tau'}^{(\nu,\mu)} \Phi_{\tau'}^{(\mu)}(x) & \text{for } x \in J(\tau'), \\ \sum_{\mu=0}^{k(\tau'')} a_{\tau''}^{(\nu,\mu)} \Phi_{\tau''}^{(\mu)}(x) & \text{for } x \in J(\tau''). \end{cases} \tag{4.17}$$

The estimate of $\Phi_\tau^{(\nu)} - \tilde{\Phi}_\tau^{(\nu)}$ will be performed on all leaves contained in $\tau$ separately. Let $t$ be a leaf satisfying $t \subseteq \tau$ and define the sequence $(\tau_\ell, \tau_{\ell+1}, \ldots, \tau_p)$ by $t = \tau_p \subset \tau_{p-1} \subset \ldots \subset \tau_{\ell+1} \subset \tau_\ell = \tau$ and $level(\tau_i) = i$. For $x \in J(t)$, the representation

$$\Phi_{\tau_i}^{(\nu)} - \tilde{\Phi}_{\tau_i}^{(\nu)} = \Phi_{\tau_i}^{(\nu)} - \check{\Phi}_{\tau_i}^{(\nu)} + \sum_{\mu=0}^{k_{i+1}} a_{\tau_{i+1}}^{(\nu,\mu)} \left(\Phi_{\tau_{i+1}}^{(\mu)} - \tilde{\Phi}_{\tau_{i+1}}^{(\mu)}\right)$$

holds, where $k_{i+1} = k(\tau_{i+1})$. Furthermore, we replace, for ease of notation, indices $\tau_i$ by $i$, e.g., $z_i$ instead of $z_{\tau_i}$. Define the error quantities

$$\delta_i^{(\nu)} := \|\Phi_i^{(\nu)} - \tilde{\Phi}_i^{(\nu)}\|_{\infty,t}, \qquad \varepsilon_i^{(\nu)} := \|\Phi_i^{(\nu)} - \check{\Phi}_i^{(\nu)}\|_{\infty,t}.$$

Then,

$$\delta_i^{(\nu)} \le \varepsilon_i^{(\nu)} + \sum_{\mu=0}^{k_{i+1}} \left|a_{i+1}^{(\nu,\mu)}\right| \delta_{i+1}^{(\mu)}. \tag{4.18}$$

$\check{\Phi}_i^{(\nu)}$ is the Taylor expansion of $\Phi_i^{(\nu)}$ on $\tau_{i+1}$ of order $k_{i+1}$. Estimating the remainder of the Taylor expansion results in $\varepsilon_i^{(\nu)} = 0$ for all $\nu \leq k_{i+1}$ and, for $\nu > k_{i+1}$,

$$\begin{aligned}\varepsilon_i^{(\nu)} &\leq \frac{r_{i+1}^{k_{i+1}+1}}{(k_{i+1}+1)!} \frac{\nu!}{(\nu - k_{i+1} - 1)!} \left\| (\cdot - z_i)^{\nu - k_{i+1} - 1} \right\|_{\infty, \tau_{i+1}} = \binom{\nu}{k_{i+1}+1} r_i^{\nu - k_{i+1} - 1} r_{i+1}^{k_{i+1}+1} \\ &= \binom{\nu}{k_{i+1}+1} 2^{-i\nu - k_{i+1} - 1}.\end{aligned}$$

The coefficients $a_{i+1}^{(\nu,\mu)}$ in (4.17) vanish for $\mu > \nu$ and, for $\mu \leq \nu$, we have

$$\left| a_{i+1}^{(\nu,\mu)} \right| = \binom{\nu}{\mu} \left| z_{i+1} - z_i \right|^{\nu - \mu} = \binom{\nu}{\mu} r_{i+1}^{\nu-\mu} = \binom{\nu}{\mu} 2^{-(i+1)(\nu-\mu)}.$$

The coefficients $\delta_i^{(\nu)}$ in (4.18) are bounded from above by $\tilde{\delta}_i^{(\nu)}$ defined by

$$\tilde{\delta}_p^{(\nu)} = 0 \qquad \text{for all } \nu \leq k_p \tag{4.19}$$

and, for $i = p-1, p-2, \ldots, \ell$, by

$$\begin{aligned}\tilde{\delta}_i^{(\nu)} &= \sum_{\mu=0}^{\nu} \binom{\nu}{\mu} 2^{-(i+1)(\nu-\mu)} \, \tilde{\delta}_{i+1}^{(\mu)} & \text{for } \nu \leq k_{i+1}, \\ \tilde{\delta}_i^{(\nu)} &= \binom{\nu}{k_{i+1}+1} 2^{-(i+1)\nu - k_{i+1} - 1} + \sum_{\mu=0}^{k_{i+1}} \binom{\nu}{\mu} 2^{-(i+1)(\nu-\mu)} \, \tilde{\delta}_{i+1}^{(\mu)} & \text{for } k_{i+1} < \nu \leq k_i.\end{aligned} \tag{4.20}$$

The estimate $\tilde{\delta}_i^{(\nu)} \leq \lambda^{i-p} \left( \omega 2^{-i} \right)^{\nu}$ stated in Lemma 4.11 yields the proof. ■

**Lemma 4.11** *Let $\alpha, \beta$, and $\omega$ be as in Lemma 4.10 and set $\lambda = 3/2$. Then, the coefficients $\tilde{\delta}_i^{(\nu)}$ can be estimated by $\tilde{\delta}_i^{(\nu)} \leq \lambda^{-k_i} \left( \omega 2^{-i} \right)^{\nu}$.*

The rather technical proof of this lemma can be found in the Appendix.

We have all ingredients to estimate the approximation error on a local block.

**Theorem 4.12** *Let $\alpha, \beta$, and $\omega$ be as in Lemma 4.10. Let $\eta \leq (8\omega)^{-1}$. Then, there exists a constant $C < \infty$ so that, for every $\eta$-admissible block $b = \tau \times \sigma$ with level $(\tau) =$ level $(\sigma) = \ell$, the error $\varepsilon_b$ as in (4.13) can be estimated by*

$$\varepsilon_b \leq C \left( \tfrac{2}{3} \right)^{k(b)}.$$

*Proof.* In view of the splittings (4.14) and (4.15) it remains to estimate $e_b^{III,IV}$. The coefficients $\kappa_{\tau,\sigma}^{(\nu,\mu)}$ from (4.15) can be estimated by

$$\left| \kappa_{\tau,\sigma}^{(\nu,\mu)} \right| \leq \binom{\nu+\mu}{\nu} \begin{cases} \left| \log \operatorname{dist}(\tau,\sigma) \right| & \text{if } \nu + \mu = 0 \\ \operatorname{dist}^{-\nu-\mu}(\tau,\sigma) \,/\, (\nu+\mu) & \text{if } \nu + \mu > 0 \end{cases}.$$

By using $\left| \Phi_\sigma^{(\mu)}(y) \right| = \left| y - z_\sigma \right|^{\mu} \leq r_\sigma^{\mu} = 2^{-(\ell+1)\mu}$ for $y \in J(\sigma)$, we obtain (with $\lambda = 3/2$)

$$\begin{aligned}\left| e_b^{III}(x,y) \right| &\leq \sum_{\nu+\mu \leq k(b)} \left| \kappa_{\tau,\sigma}^{(\nu,\mu)} \right| \left| \Phi_\tau^{(\nu)}(x) - \tilde{\Phi}_\tau^{(\nu)}(x) \right| \left| \Phi_\sigma^{(\mu)}(y) \right| \\ &\leq \sum_{1 \leq \nu+\mu \leq k(b)} \frac{1}{\nu+\mu} \binom{\nu+\mu}{\nu} \frac{\lambda^{-k(b)} \left( \omega 2^{-\ell} \right)^{\nu}}{\operatorname{dist}^{\nu+\mu}(\tau,\sigma)} 2^{-(\ell+1)\mu} \\ &\leq \lambda^{-k_\ell} \sum_{1 \leq \nu+\mu \leq k(b)} \frac{1}{\nu+\mu} \binom{\nu+\mu}{\nu} (2\omega)^{\nu} \eta^{\nu+\mu} 2^{-\nu-\mu}.\end{aligned}$$

Choosing $\eta \leq (4\omega)^{-1}$, we obtain

$$\left|e_b^{III}(x,y)\right| \leq \lambda^{-k(b)} \sum_{\mu=0}^{k} \sum_{\nu=0}^{\mu} \frac{\mu!}{\nu!\,(\mu-\nu)!}\, 2^{-\nu-\mu} = \lambda^{-k(b)} \left(4 - 3\left(\frac{3}{4}\right)^{k(b)}\right) \leq 4\,\lambda^{-k(b)}.$$

The estimate of $e_b^{IV}$ can be obtained in the same fashion by using

$$\left|\tilde{\Phi}_\tau^{(\nu)}(x)\right| \leq \left|\Phi_\tau^{(\nu)}(x)\right| + \left|\tilde{\Phi}_\tau^{(\nu)}(x) - \Phi_\tau^{(\nu)}(x)\right| \leq 2^{-(\ell+1)\nu} + \lambda^{-k(b)} \left(\omega 2^{-\ell}\right)^\nu \leq \left(\tilde{\omega} 2^{-\ell-1}\right)^\nu$$

with $\tilde{\omega} = 2\omega$ and proceeding in an analogous way. ■

We come now to the global estimate of error caused by replacing the kernel function by the variable order approximation. Let $\mathcal{S}$ denote the space of piecewise constant functions on the mesh $\{J_i\}_{i=1}^n$. The $L^2$-norm on an interval $t = (a,b)$ is denoted by $\|\cdot\|_{0,t}$ while we skip the index $t$ for $t = (0,1)$. The global error of the approximation is given by

$$E(u,v) := \sum_{b \in P_2^{far}} \int_b u(x)\, e_b(x,y)\, v(y)\, dydx \qquad \text{for } u, v \in \mathcal{S}.$$

**Theorem 4.13** *Let $\alpha, \beta, \omega$, and $\eta$ be as in Theorem 4.12. There exists a constant $C$ so that, for all $u, v \in S$,*

$$E(u,v) \leq Ch\, \|u\|_0\, \|v\|_0\,.$$

*Proof.* Let $P_2(\ell) := \{b \in P_2 : \mathit{level}\,(b) = \ell\}$ and $T(\ell) := \{\tau \in T : \mathit{level}\,(\tau) = \ell\}$ for all $0 \leq \ell \leq p-1$. For $b = \tau \times \sigma \in P_2(\ell)$, we have $|\tau| = |\sigma| = 2^{-\ell}$. Introduce

$$N_1 = \sup_{0 \leq \ell < p}\ \sup_{\tau \in T(\ell)} \sum_{\sigma \text{ with } \tau\times\sigma \in P_2(\ell)} 1, \qquad N_2 = \sup_{0 \leq \ell < p}\ \sup_{\sigma \in T(\ell)} \sum_{\tau \text{ with } \tau\times\sigma \in P_2(\ell)} 1.$$

Using Theorem 4.12, we obtain with $\lambda = 3/2$

$$\begin{aligned}
E &= \sum_{\ell=0}^{p-1} \sum_{b \in P_2(\ell)} \int_b u(x)\, e_b(x,y)\, v(y)\, dydx \\
&\leq \sum_{\ell=0}^{p-1} \sum_{b=\tau\times\sigma \in P_2(\ell)} C\lambda^{-k_\ell} \sqrt{|\tau|\,|\sigma|}\, \|u\|_{0,\tau}\, \|v\|_{0,\sigma} \\
&\leq Ch \sum_{\ell=0}^{p-1} \lambda^{-k_\ell} 2^{p-\ell} \sqrt{\sum_{b=\tau\times\sigma \in P_2(\ell)} \|u\|_{0,\tau}^2} \sqrt{\sum_{b=\tau\times\sigma \in P_2(\ell)} \|v\|_{0,\sigma}^2} \\
&\leq Ch \sum_{\ell=0}^{p-1} \lambda^{-k_\ell} 2^{p-\ell} \sqrt{\sum_{\tau \in T(\ell)} \|u\|_{0,\tau}^2 \sum_{\sigma:\tau\times\sigma \in P_2(\ell)} 1} \sqrt{\sum_{\sigma \in T(\ell)} \|v\|_{0,\sigma}^2 \sum_{\tau:\tau\times\sigma \in P_2(\ell)} 1} \\
&= Ch\, \|u\|_0\, \|v\|_0 \sqrt{N_1 N_2} \sum_{\ell=0}^{p} \lambda^{-k_\ell} 2^{p-\ell}.
\end{aligned}$$

For $\alpha \geq 2$, we have $\lambda^{-k_\ell} 2^{p-\ell} \leq \left(\frac{9}{8}\right)^{\ell-p}$and $\sum_{\ell=0}^{p} \lambda^{-k_\ell} 2^{p-\ell} \leq \sum_{\ell=0}^{p} \left(\frac{9}{8}\right)^{\ell-p} \leq 9$. One can see from the construction in Definition 2.2 that the numbers $N_1$, $N_2$ can be estimated from above by a constant as well. ■

# References

[1] W. Hackbusch: *Iterative Solution of Large Sparse Systems.* Springer Verlag, New York, 1994.

[2] W. Hackbusch: *Integral Equations. Theory and Numerical Treatment.* ISNM 128. Birkhäuser, Basel, 1995.

[3] W. Hackbusch: *A sparse matrix arithmetic based on $\mathcal{H}$-matrices. Part I: introduction to $\mathcal{H}$-matrices.* Computing **62** (1999).89-108 (http://www.numerik.uni-kiel.de/reports/98-28.html)

[4] W. Hackbusch and B. N. Khoromskij: *A sparse $\mathcal{H}$-matrix arithmetic. Part II: application to multi-dimensional problems.* Preprint Nr 22/1999, Max-Planck-Institut für Mathematik, Leipzig, 1999 (http://www.mis.mpg.de/cgi-bin/preprints.pl)

[5] W. Hackbusch, B. N. Khoromskij, and S. A. Sauter: *On $\mathcal{H}^2$-matrices.* In: Lectures on Applied Mathematics (Bungartz, Hoppe, Zenger, eds.), Springer Verlag, Heidelberg, 1999

[6] W. Hackbusch and Z. P. Nowak: *On the fast matrix multiplication in the boundary element method by panel clustering.* Numer. Math. **54** (1989) 463–491.

[7] S. A. Sauter: *Über die effiziente Verwendung des Galerkin-Verfahrens zur Lösung Fredholmscher Integralgleichungen.* Dissertation. Universität Kiel. 1992.

# 5 Appendix (Proof of Lemma 4.11)

We recall $k_i = \alpha\,(p-i)+\beta$ with $\alpha, \beta \geq 1$. Hence, $\tilde{\delta}_i^{(\nu)} = 0$ for all $i$ and $\nu \leq \beta$ and we assume, for the following, that $\nu \geq \beta + 1$ holds. The proof is given by an induction with respect to $i = p, p-1, \ldots, \ell$.

- $i = p$. Definition (4.19) implies $\tilde{\delta}_p^{(\nu)} = 0$ for all $\nu \leq k_p$.
- Assumption: Assertion holds up to an index $i+1$:

$$\tilde{\delta}_j^{(\nu)} \leq \lambda^{-k_j}\left(\omega 2^{-j}\right)^{\nu} \qquad \text{for all } \nu \leq k_j,\ j \in \{i+1, i+2, \ldots, p\}. \tag{5.1}$$

- $i+1 \to i$:

Plugging (5.1) into the sum in (4.20) results in

$$\begin{aligned}\sum_{\mu=0}^{\nu}\binom{\nu}{\mu}2^{-(i+1)(\nu-\mu)}\tilde{\delta}_{i+1}^{(\mu)} &\leq \sum_{\mu=0}^{\nu}\binom{\nu}{\mu}2^{-(i+1)(\nu-\mu)}\lambda^{-k_{i+1}}\left(\omega 2^{-i-1}\right)^{\mu} \\ &= 2^{-(i+1)\nu}\lambda^{-k_{i+1}}\left(1+\omega\right)^{\nu} = \lambda^{-k_i}\left(\omega 2^{-i}\right)^{\nu}\psi_\nu^I\end{aligned}$$

with $\psi_\nu^I = \lambda^\alpha\left(\frac{1+\omega}{2\omega}\right)^\nu$. Choosing $\omega \geq 2.5$ yields:

$$\lambda^\alpha\left(\frac{1+\omega}{2\omega}\right)^\nu \leq \lambda^\alpha\left(\frac{7}{10}\right)^\nu \leq \lambda^\alpha\left(\frac{7}{10}\right)^{\beta+1}. \tag{5.2}$$

Simple analysis shows that, for

$$\beta \geq \frac{6}{5}\alpha + 1,$$

the right-hand side in (5.2) is bounded from above by $1/2$, i.e., $\psi_\nu^I \leq 1$ and the assertion is proved for $\nu \leq k_{i+1}$.

For $\nu > k_{i+1}$, we estimate the sum as above, while the first term in (4.20) is estimated by

$$\binom{\nu}{k_{i+1}+1}2^{-(i+1)\nu-k_{i+1}-1} = \lambda^{-k_i}\left(\omega 2^{-i}\right)^\nu \psi_\nu^{II}$$

with

$$\begin{aligned}\psi_\nu^{II} &= (2\omega)^{-\nu}\lambda^{k_i}2^{-k_{i+1}-1}\binom{\nu}{k_{i+1}+1} \leq (2\omega)^{-\nu}\,2^{\alpha-1}\left(\frac{\lambda}{2}\right)^{k_i}\binom{k_i}{k_{i+1}+1} \\ &= (2\omega)^{-\nu}\,2^{\alpha-1}\left(\frac{\lambda}{2}\right)^{k_i}\frac{\left(\alpha\,(p-i)+\beta\right)\left(\alpha\,(p-i)+\beta-1\right)\cdots\left(\alpha\,(p-i)+\beta-\alpha+2\right)}{(\alpha-1)!} \\ &\leq (2\omega)^{-\nu}\,2^{\alpha-1}\left(\frac{\lambda}{2}\right)^{k_i}\frac{k_i^{\alpha-1}}{(\alpha-1)!} = (2\omega)^{-\nu}\left(\frac{3}{4}\right)^{k_i}\frac{(2k_i)^{\alpha-1}}{(\alpha-1)!} \leq C_\alpha\,(2\omega)^{-\nu}\end{aligned}$$

with

$$C_\alpha = \left(\frac{3}{4}\right)^x \frac{(2x)^{\alpha-1}}{(\alpha-1)!} \qquad \text{and } x = \frac{\alpha-1}{\log 4/3}.$$

Simple analysis yields:

$$C_\alpha \leq 4\,(\alpha-1)^{\alpha-1}\,.$$

The sum $\psi_\nu^I + \psi_\nu^{II}$ can be estimated by

$$\psi_\nu^I + \psi_\nu^{II} \leq 4\,(\alpha-1)^{\alpha-1}\,(2\omega)^{-\nu} + \lambda^\alpha \left(\frac{1+\omega}{2\omega}\right)^\nu$$

Choosing

$$\alpha \geq 1.5, \qquad \beta \geq \max\left\{\frac{6}{5}\alpha+1, 2\alpha-1\right\}, \qquad \omega \geq \max\left\{2.5, \alpha-1\right\}$$

yields that, for all $\nu \geq \beta+1$, the sum $\psi_\nu^I + \psi_\nu^{II}$ is bounded from above by 1.