

Max-Planck-Institut
für Mathematik
in den Naturwissenschaften
Leipzig

Domain decomposition based \mathcal{H} -matrix
preconditioners for the skin problem

(revised version: August 2008)

by

Boris N. Khoromskij, and Alexander Litvinenko

Preprint no.: 95

2006



Domain decomposition based \mathcal{H} -matrix preconditioners for the skin problem

B.N. Khoromskij and A. Litvinenko

Max Planck Institute for Mathematics in the Sciences, Inselstr. 22, Leipzig, 04103, Germany, bokh@mis.mpg.de, litvinen@tu-bs.de

Abstract

In this paper we propose and analyse a new hierarchical Cholesky (\mathcal{H} -Cholesky) factorization based preconditioner for iterative solving the elliptic equations with highly jumping coefficients arising in the so-called skin-modelling problem in 3D [8]. First, we construct the block-diagonal approximation to the FE stiffness matrix, which is well suited to the “perforated” structure of the coefficients. We apply the \mathcal{H} -Cholesky factorization of this block-diagonal matrix as a preconditioner in the pcg iteration. It is shown that the new preconditioner is robust with respect to jumps in the coefficients and it requires less storage and computing time than the standard \mathcal{H} -Cholesky factorization.

Key words: skin problem, \mathcal{H} -matrix approximation, hierarchical Cholesky, jumping coefficients, domain decomposition.

1 Introduction

In papers [1, 9, 11] the authors successfully applied the pcg, gmres, bicgstab iterations with \mathcal{H} -matrix based preconditioners to different types of second order elliptic differential equations. In some cases \mathcal{H} -matrix inverse can be used even as a direct solver [6, 4]. In this paper we consider the elliptic equation with highly jumping coefficients,

$$\begin{aligned} -\operatorname{div}(\alpha(x)\nabla u) &= f(x) & x \in \Omega \subset \mathbb{R}^d, d = 2, 3, \\ u &= 0 & x \in \gamma, \\ \frac{\partial u}{\partial n} &= g & x \in \Gamma \setminus \gamma, \end{aligned} \tag{1}$$

where $\Gamma = \partial\Omega$, $\gamma \subset \partial\Omega$ corresponds to a piece of the boundary with the Dirichlet boundary condition. This equation was used for the numerical modeling of the so-called skin problem that describes penetration of drugs through the skin (cf. [8]). To simplify the model we choose Ω as a fragment with 8 cells Ω_c and the lipid layer Ω_l (see Fig. 2), where $\Omega = \Omega_c \cup \bar{\Omega}_l$, $\Omega_c = \cup_{i=1}^8 \Omega_{c,i}$,

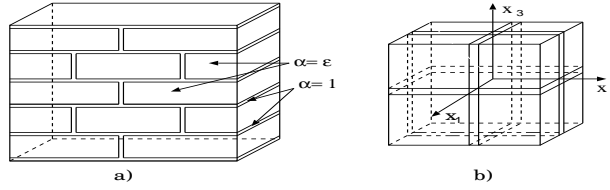


Fig. 1. (left) A skin fragment consists of the lipid layer and disjoint cells. **(right)** The simplified model of a skin fragment contains 8 cells with the lipid layer in between. $\Omega = [-1, 1]^3$, $\alpha(x) = \varepsilon$ inside the cells and $\alpha(x) = 1$ in the lipid layer.

Ω_l is a closed set. Fig. 1 (right) shows cells and the lipid layer in between. Typical feature for the skin problem is the highly jumping coefficients: the penetration coefficient inside the cells is very small, $\alpha(x) = \varepsilon \in [10^{-5} - 10^{-3}]$, but it is relatively large in the lipid layer ($\alpha(x) = 1$). In this problem the Dirichlet boundary condition describes the presence of drugs on the boundary γ of the skin fragment. The nonzero Neumann condition on $\Gamma \setminus \gamma$ specifies the penetration through the surface $\Gamma \setminus \gamma$. The right-hand side in (1) presents external forces.

It is known that for problems with jumping coefficients (see (1)) the condition number $\text{cond}(A)$ of the FE stiffness matrix A is proportional to $h^{-2} \sup_{x,y \in \Omega} \alpha(x)/\alpha(y)$, where $\alpha(x)$ denotes the jumping coefficient and h is the step size of a finite element scheme. In the case of a large condition number one requires the efficient preconditioner W , so that $\text{cond}(W^{-1}A) \simeq 1$.

The rest of this paper is structured as follows. In Section 2 we describe the FEM discretisation of (1). We recall the main idea of the \mathcal{H} -matrix techniques in Section 3. Section 4 describes the new preconditioner and presents the condition number estimates. Numerics for the 3D model problem is discussed in Section 5.

2 Discretisation by FEM

We choose a rectangular quasi-uniform triangulation τ_h which is compatible with the lipid layer, i.e., $\tau_h := \tau_h^l \cup \tau_h^c$, where τ_h^l is a triangulation of the lipid layer and τ_h^c is a triangulation of cells (see Fig. 2). In the presented example Ω_l contains only two grid layers.

Let $V_h := \text{span}\{b_1, \dots, b_n\}$ be the set of piecewise linear functions with respect to τ_h such that

$$V_h \subset H_{0,\gamma}^1(\Omega) := \{u \in H^1(\Omega) : u|_\gamma = 0\}, \quad (2)$$

where b_j , $j \in I_\Omega := \{1, \dots, n\}$, is the set of corresponding hat-functions. The related Galerkin discretisation of the problem (1) reads as:

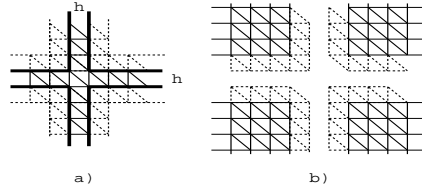


Fig. 2. a) A 2D grid of the lipid layer of width h (bounded by bold lines). b) Fragments of four cells ($\alpha = \varepsilon$). The finite elements restricted by dotted lines in (a) and (b) are needed for constructing the stiffness matrices A_{11} and A_{22} , respectively.

$$\text{find } u_h \in V_h, \text{ so that } a(u_h, v) = c(v) \text{ for all } v \in V_h \quad (3)$$

with respective bilinear form a and linear functional c given by

$$a(u, v) = \int_{\Omega} \alpha(x) (\nabla u, \nabla v) dx, \quad c(v) := \int_{\Omega} f v dx + \int_{\Gamma \setminus \gamma} g v d\Gamma. \quad (4)$$

The system of linear algebraic equations for the coefficients vector \mathbf{u} reads as

$$A_{\varepsilon} \mathbf{u} = \mathbf{c}, \text{ where } A_{\varepsilon} = \{a(b_j, b_i)\}_{i,j \in I_{\Omega}} \in \mathbb{R}^{n \times n}, \mathbf{c} = \{c(b_i)\}_{i \in I_{\Omega}} \in \mathbb{R}^n. \quad (5)$$

The lipid layer Ω_l between the cells specifies the natural decomposition of Ω . The thickness of this layer is proportional to the step size h . Note that with the proper ordering of the index set I_{Ω} , we can represent the global stiffness matrix in the following block form

$$A_{\varepsilon} = \begin{bmatrix} A_{11} & \varepsilon A_{12} \\ \varepsilon A_{21} & \varepsilon A_{22} \end{bmatrix}, \quad A_{11} = A_0 + \varepsilon B_{11}. \quad (6)$$

Here A_{11} , εA_{22} are the stiffness matrices which correspond to the lipid layer and to the rest of the domain, respectively, εA_{12} and εA_{21} are coupling matrices. In turn, A_0 discretises the Neumann problem in Ω_l with homogeneous Neumann data on the inner boundaries $\partial\Omega_c$. In the case $\varepsilon = 1$, the matrix A_{ε} corresponds to the discrete Laplace operator.

In the following we focus on a construction of the efficient preconditioner to the matrix A_{ε} , while the analysis of the discretisation accuracy remains beyond the scope of the present paper. However, the error analysis can be based on the standard FEM theory under certain regularity assumptions.

3 Hierarchical Matrices

The hierarchical matrices (\mathcal{H} -matrices) (cf. [3]) provide the efficient data-sparse representation of fully-populated matrices arising in a wide range of FEM/BEM applications. The main idea of \mathcal{H} -matrices is to approximate certain subblocks $R \in \mathbb{R}^{n \times m}$ of a given matrix by the rank- k matrices, i. e.,

$R \cong AB^T$, with $A \in \mathbb{R}^{n \times k}$ and $B \in \mathbb{R}^{m \times k}$, $k \ll \min(n, m)$. The storage requirement for both matrices A and B is $k(n + m)$ instead of $n \cdot m$ for the matrix R . The advantage of the \mathcal{H} -matrix technique is that the complexity of the \mathcal{H} -matrix addition, multiplication and inversion is $\mathcal{O}(kn \log^q n)$, $q = 1, 2$ (see [3, 5]). Let I be an index set. To build an \mathcal{H} -matrix $M \in \mathbb{R}^{I \times I}$ one needs an admissible block partitioning (see Fig. 3) built on a block cluster tree $T_{I \times I}$ by means of an admissibility condition (see [3, 2]). The admissible block partitioning indicates which blocks can be approximated by low-rank matrices.

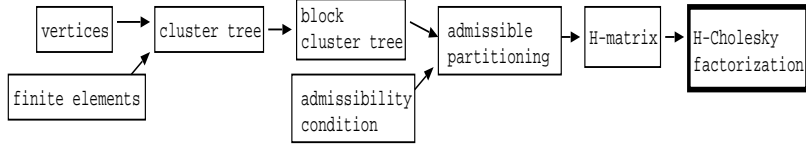


Fig. 3. The scheme of building an \mathcal{H} -matrix and its \mathcal{H} -Cholesky factorization.

Definition 1. We define the set of \mathcal{H} -matrices with the maximal rank k as

$$\mathcal{H}(T_{I \times I}, k) := \{M \in \mathbb{R}^{I \times I} \mid \text{rank}(M|_b) \leq k \text{ for all admissible leaves } b \text{ of } T_{I \times I}\}.$$

Suppose that $A = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} = \begin{bmatrix} L_{11} & 0 \\ L_{21} & L_{22} \end{bmatrix} \begin{bmatrix} U_{11} & U_{12} \\ 0 & U_{22} \end{bmatrix}$. The algorithm that computes the \mathcal{H} -LU factorization (cf. [10, 1]) is the following:

1. compute L_{11} and U_{11} as \mathcal{H} -LU factorization of A_{11} ;
2. compute U_{12} from $L_{11}U_{12} = A_{12}$ (recursive block forward substitution);
3. compute L_{21} from $L_{21}U_{11} = A_{21}$ (recursive block backward substitution);
4. compute L_{22} and U_{22} as \mathcal{H} -LU factorization of $L_{22}U_{22} = A_{22} - L_{21}U_{12}$.

Note that all the steps are executed approximately via truncation to the class of \mathcal{H} -matrices.

4 Block \mathcal{H} -LU Preconditioner \widetilde{W}_2

Let us introduce the \mathcal{H} -Cholesky factorisation of the following symmetric matrices

$$A_\varepsilon = \begin{bmatrix} A_{11} & \varepsilon A_{12} \\ \varepsilon A_{21} & \varepsilon A_{22} \end{bmatrix} \cong L_1 L_1^T =: \widetilde{W}_1, \quad W_2 := \begin{bmatrix} A_{11} & 0 \\ 0 & \varepsilon A_{22} \end{bmatrix} \cong L_2 L_2^T =: \widetilde{W}_2. \quad (7)$$

\mathcal{H} -Cholesky factorisation $L_1 L_1^T$ was successfully applied in [1, 9]. As a new preconditioner we use the \mathcal{H} -Cholesky factorisation of W_2 , which we denote by \widetilde{W}_2 . Examples of the \mathcal{H} -Cholesky factors L_1 and L_2 are shown in Fig. 4.

Remark 1. Note that $\widetilde{W}_2^{-1/2} A_\varepsilon \widetilde{W}_2^{-1/2} = L_2^{-T} A_\varepsilon L_2^{-1}$, i.e., $\widetilde{W}_2^{-1/2} A_\varepsilon \widetilde{W}_2^{-1/2}$ is positive definite and symmetric (the same holds for \widetilde{W}_1). Thus, for solving the initial problem (5) one may apply the pcg method with preconditioner \widetilde{W}_2 .

Lemma 1. For a symmetric and positive definite matrix $A = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix}$ and any vectors u_1 and u_2 of the respective size it holds that

$$|(A_{12}u_2, u_1)| \leq (A_{11}u_1, u_1)^{1/2} \cdot (A_{22}u_2, u_2)^{1/2}.$$

Lemma 2. For any $u \in \mathbb{R}^n$ we have $(A_\varepsilon u, u) \leq 2(W_2 u, u)$ with W_2 defined by (7).

Proposition 1. Let the component u_2 be discrete harmonic extension of u_1 into $\Omega_c \subset \Omega$. Then $\exists c_1 > 0 : c_1(A_{11}u_1, u_1) \geq (A_{22}u_2, u_2)$ with c_1 independent of h .

Proof: This estimate corresponds to the case of the Laplace operator in Ω and can be verified by using standard properties of the harmonic elliptic extension operator and trace estimates. Applying Theorem 4.1.3 in [12], where we set $\Omega_1 = \Omega_l$ and $\Omega_2 = \Omega_c$, leads to the desired bound with constant c_1 depending only on the geometry.

Lemma 3. Assume that Proposition 1 holds. Then $\exists \varepsilon_0 \in (0, 1)$ such that $\forall \varepsilon \in (0, \varepsilon_0]$ we have $((A_\varepsilon - cW_2)u, u) \geq 0$ with $c = \frac{1-\varepsilon}{1+c_1\varepsilon}$, where c_1 is defined in Proposition 1.

Proof: We choose $c = \frac{1-\varepsilon}{1+c_1\varepsilon}$ and then apply Lemma 1 to obtain

$$\begin{aligned} ((A_\varepsilon - cW_2)u, u) &= (1-c)(A_{11}u_1, u_1) + \varepsilon(1-c)(A_{22}u_2, u_2) + 2\varepsilon(A_{12}u_2, u_1) \\ &\geq (1-c)(A_{11}u_1, u_1) + \varepsilon(1-c)(A_{22}u_2, u_2) - 2\varepsilon|(A_{12}u_2, u_1)| \\ &\geq (1-c)((A_{11}u_1, u_1) + \varepsilon(A_{22}u_2, u_2)) - \varepsilon((A_{11}u_1, u_1) + (A_{22}u_2, u_2)) \\ &\geq (1-c-\varepsilon)(A_{11}u_1, u_1) - c\varepsilon(A_{22}u_2, u_2). \end{aligned}$$

Now Proposition 1 ensures

$$((A_\varepsilon - cW_2)u, u) \geq (A_{11}u_1, u_1)(1-c-\varepsilon-c_1c\varepsilon) = 0.$$

Thus, $((A_\varepsilon - cW_2)u, u) \geq 0$ holds with $c = \frac{1-\varepsilon}{1+c_1\varepsilon}$.

Notice that the constant $c = \frac{1-\varepsilon}{1+c_1\varepsilon}$ depends on the geometry of Ω .

In Fig. 4(right) one can see two blocks on the first level of the \mathcal{H} -Cholesky factorisation of W_2 . The first block corresponds to the lipid layer Ω_l , the second one (with 8 subblocks) corresponds to 8 cells. The problems inside the cells can be treated in parallel.

Remark 2. The set of all nodal points in the lipid layer (Fig. 1 (right)) can be decomposed into 12 parts $\overline{\Omega}_l = \cup_{i=1}^{12} \Omega_{l,i}$, which will lead to further simplifications.

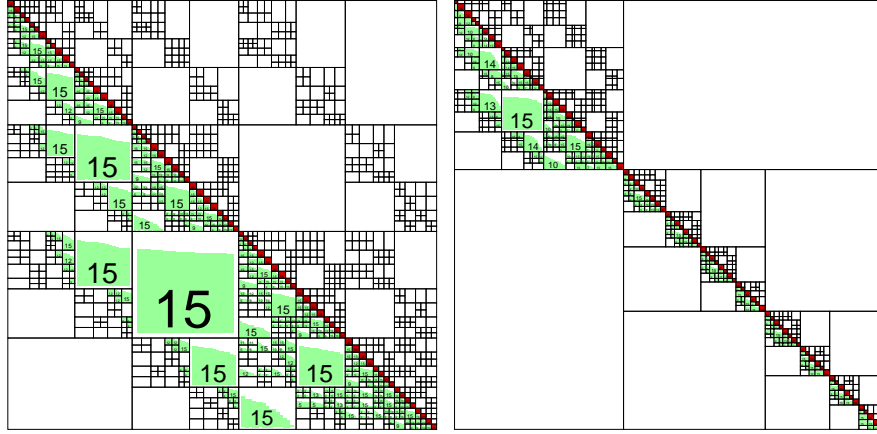


Fig. 4. \mathcal{H} -Cholesky factors of the global stiffness matrix A_ε (**left**) and of the block stiffness matrix W_2 (**right**). The dark blocks $\in \mathbb{R}^{36 \times 36}$ are dense matrices, the gray blocks are low-rank matrices and the white blocks are zero ones. The steps in the grey blocks show the decay of the singular values in a logarithmic scale. The numbers inside the subblocks indicate the local ranks.

5 Numerical tests

In this Section we present numerical results for the 3D Dirichlet problem (see [7] for the 2D case). Figure 2 explains the discretisation of the lipid layer. For simplicity the width of the lipid layer is chosen as h , but it can be a multiple of h .

Table 1 gives the theoretical estimates on the sequential and parallel complexities of \widetilde{W}_1 and \widetilde{W}_2 .

Table 1. Computational complexities of the preconditioners \widetilde{W}_1 and \widetilde{W}_2 . The number of processors is p . The number of degrees of freedom in the lipid layer is n_I (handled by one processor) and the number of dofs on each processor is $n_0 := \frac{n - n_I}{p - 1}$.

	Sequential Complexity	Parallel Complexity
\widetilde{W}_1	$\mathcal{O}(n \log^2 n)$	$\mathcal{O}(n \log^2 n)$
\widetilde{W}_2	$\mathcal{O}(n_I \log^2 n_I) + \mathcal{O}((n - n_I) \log^2 (n - n_I))$	$\max\{\mathcal{O}(n_I \log^2 n_I) + \mathcal{O}(n_0 \log^2 n_0)\}$

Remark 3. The sparsity constant C_{sp} is an important \mathcal{H} -matrix parameter that effects all \mathcal{H} -matrix complexity estimates (see [3, 5]). The smaller C_{sp} the better complexity bound is. For instance, for the problem with 45^3 dofs $C_{sp}(A_\varepsilon) = 108$, while $C_{sp}(W_2) = 30$. For the model geometry with a larger number of cells the difference between the sparsity constants will be even more significant.

Table 2. Comparison of \widetilde{W}_1 and \widetilde{W}_2 in 3D, 40^3 dofs, $\|A_\varepsilon \mathbf{u} - \mathbf{c}\| = 10^{-8}$, $\varepsilon = 10^{-5}$.

rank k	time $t^{(1)}, t^{(2)}$	storage $S^{(1)}, S^{(2)}$	#iter $^{(1)}, ^{(2)}$
1	34.6, 18.7	2e+2, 1e+2	69, 99
2	81.3, 35	3.8e+2, 1.8e+2	46, 91
4	220.5, 81.5	7.5e+2, 3.5e+2	17, 60
6	565.7, 149	1.1e+3, 5.1e+2	11, 74

Table 2 illustrates storage requirements (denoted by $S^{(1)}, S^{(2)}$ and measured in MB) and computing times (denoted by $t^{(1)}, t^{(2)}$ and measured in sec.) for the preconditioners \widetilde{W}_1 and \widetilde{W}_2 , respectively, depending on the \mathcal{H} -matrix rank k . The columns $t^{(1)}, t^{(2)}$ contain the total computing times for setting up the preconditioners \widetilde{W}_1 and \widetilde{W}_2 and for performing the pcg iterations. The columns iter $^{(1)}$ and iter $^{(2)}$ show the number of pcg iterations in both cases (see [7] for more details). One can see that the preconditioner \widetilde{W}_2 requires less storage ($S^{(1)} > S^{(2)}$) and less computing time ($t^{(1)} > t^{(2)}$). Notice that the computation with a smaller rank k in the \mathcal{H} -matrix arithmetic (but with a larger number of iterations) leads to a better performance than in the case with a larger k (but with a smaller number of iterations). Table 3 illustrates linear-logarithmic scaling of the computational time and storage in the problem size n (with fixed maximal rank $k = 5$). Choosing the smaller maximal rank k leads to almost linear complexities. Table 4 shows the number of iterations and the

Table 3. Dependence of the computing time and storage requirements on the problem size, $\|A_\varepsilon \mathbf{u} - \mathbf{c}\| = 10^{-8}$, max. rank= 5 (see Def. 1).

#dofs	time $t^{(2)}$ (sec.)	memory $S^{(2)}$ (Mb)	#iter $^{(2)}$
2200	0.27	5.4	33
15600	7.9	90	59
91100	119.4	1007	98

Table 4. The number of iterations and the computing times depending on the coefficient α for 40^3 dofs, $\|A_\varepsilon \mathbf{u} - \mathbf{c}\| = 10^{-8}$, $k = 1$.

ε	1	10^{-1}	10^{-2}	10^{-4}	10^{-6}	10^{-8}
#iter $^{(1)}, ^{(2)}$	86, 89	77, 100	79, 113	79, 113	82, 116	85, 120
$t^{(1)}, t^{(2)}$ sec.	70, 33	67, 35	63, 37	65, 37	67, 37	67, 38

computing times depending on the coefficient α . The number of iterations is relatively large since we use the low-rank \mathcal{H} -matrix approximation with $k = 1$. The computing time in the case of \widetilde{W}_2 is in a factor two smaller than in the case of \widetilde{W}_1 . This factor is getting larger for problems with increasing number of cells.

We conclude that the preconditioner \widetilde{W}_2 is efficient and robust with respect to the small coefficient $\alpha = \varepsilon$ characterising the skin problem (see Tables 1, 4). It requires less storage and computing time than \widetilde{W}_1 (see Tables 2, 3). The preconditioner \widetilde{W}_2 becomes more efficient for problems with increasing number of cells. The possible disadvantage of W_2 could be a relatively large number of pcg iterations, but it is compensated by their low computational cost (see Table 2).

Acknowledgement: The authors wish to thank Prof. Dr. Hackbusch for valuable comments as well as Dr. Börm and Dr. Grasedyck for the assistance in the usage of HLIB (see [2]).

References

- [1] M. Bebendorf. Hierarchical LU decomposition-based preconditioners for BEM. *Computing*, 74(3):225–247, 2005.
- [2] L. Grasedyck and S. Börm. \mathcal{H} -matrix library: www.hlib.org.
- [3] W. Hackbusch. A sparse matrix arithmetic based on \mathcal{H} -matrices. I. Introduction to \mathcal{H} -matrices. *Computing*, 62(2):89–108, 1999.
- [4] W. Hackbusch. Direct domain decomposition using the hierarchical matrix technique. In *Domain decomposition methods in science and engineering*, pages 39–50 (electronic). Natl. Auton. Univ. Mex., México, 2003.
- [5] W. Hackbusch and B. N. Khoromskij. A sparse \mathcal{H} -matrix arithmetic. II. Application to multi-dimensional problems. *Computing*, 64(1):21–47, 2000.
- [6] W. Hackbusch, B.N. Khoromskij, and R. Kriemann. Direct Schur complement method by domain decomposition based on \mathcal{H} -matrix approximation. *Comput. Vis. Sci.*, 8(3-4):179–188, 2005.
- [7] B.N. Khoromskij and A. Litvinenko. Domain decomposition based \mathcal{H} -matrix preconditioner for the 2D and 3D skin problem. *Max-Planck Institute for Math. in the Sciences, Preprint Leipzig*, (95), 2006.
- [8] B.N. Khoromskij and G. Wittum. *Numerical solution of elliptic differential equations by reduction to the interface*, volume 36 of *Lecture Notes in Computational Science and Engineering*. Springer-Verlag, Berlin, 2004.
- [9] S. Le Borne and L. Grasedyck. \mathcal{H} -matrix preconditioners in convection-dominated problems. *SIAM J. Matrix Anal. Appl.*, 27(4):1172–1183, 2006.
- [10] M. Lintner. The eigenvalue problem for the 2D Laplacian in \mathcal{H} -matrix arithmetic and application to the heat and wave equation. *Computing*, 72(3-4):293–323, 2004.
- [11] A. Litvinenko. Application of hierarchical matrices for solving multiscale problems. *PhD Dissertation, Leipzig University*, 2006.
- [12] A. Quarteroni and A. Valli. *Domain Decomposition Methods for Partial Differential Equations*. Oxford University Press, Oxford, 1999.