

**Solution of a Second Order Elliptic
Partial Differential Equation with
Varying Complex Coefficients: An
Application for Computing
Effective Complex Electrical
Properties of Materials represented
by 3D Images**

Johnny Valbuena Soler

A thesis submitted for the degree of
Doctor of Philosophy
The Australian National University

Except where otherwise indicated, this thesis is my own original work.

A handwritten signature in blue ink, appearing to read 'Johnny Valbuena Soler', with a large, sweeping flourish extending to the right.

Johnny Valbuena Soler
October 23, 2017

To the memory of my beloved and adorable mother, Maria.

© Johnny Valbuena Soler 2017

Acknowledgments

First and foremost, I would like to express my deepest and most heartfelt gratitude to Prof Wolfgang Hackbusch for believing in me and offering me his continuous support and dedication. Without his guidance, his expertise and his persistent help, I would still be "lost" in my research. I could not imagine having a better supervisor and advisor than Prof Hackbusch, whose passion for mathematics is utterly contagious. We spent many long hours on the whiteboard and his personal enthusiasm kept me engaged with my research and gave me the necessary encouragement to keep going. I am also very grateful to his beloved wife, Mrs. Ingrid for welcoming me in their home in Kiel, Germany, and preparing some delicious meals for us, while we were discussing numerical applications.

I would also like to thank Prof Tim Senden and specially Prof Adrian Sheppard, the chair of my supervisory panel, for not giving up on me and for their assistance and generous support along my long journey.

A very special thank you goes to Prof. Steffen Börm, Chair of Scientific Computing at the University of Kiel for all his help and unconditional support in making my multiple stays in Kiel comfortable and affordable, but also for always keeping a door open when I needed to unwind and talk about ideas for my future career. I would also like to thank Prof Börm's group: Linda, Nadine, Sven, Jens, and Dirk.

I express my sincere thanks to Dr. Ronald Kriemann for all his support to use and install the \mathcal{H} -Lib^{Pro} library, and the discussion of the convergence results.

My appreciation also extends to Mrs Ritter, the administrative assistant at the University of Kiel Guest House, for her constant help in finding accommodation for me and assisting me in my non-existent knowledge of the German language.

I would not have reached the last stage of the PhD process without the administrative constant support and help from Mrs. Luidmila Mangos (Luda). Many thanks for everything you have done to assist me in submitting my PhD.

I am also forever grateful to my dear friend Tony Karrys, who was always very supportive and a true friend through the many years I stayed in University House. His sense of humour and his immense heart kept me going through the difficult times that one can endure when doing a PhD.

This acknowledgement would not be complete if I did not express my utmost and heartfelt thanks to my very dear friend Amalia. With her constant generous support and her true friendship, she made my road much less rugged. I would also like to thank my friends Babu, Alon, David, Jill, Vivian, Nicolas, Marion and my little friend Nonoshe for their constant support, their humour and their friendship.

Finally, I would like to thank my partner Eleonora, for her incredible patience, her love and her unconditional support in this valuable experience of my life.

Abstract

Materials may be characterised by using their electrical properties which establish how they interact when an electric field is applied at various frequency ranges. This interaction is used to determine properties of materials such as moisture content, bulk density, bio-content, chemical concentration and stress-strain. In the case of the physical characteristics of rocks, the response of the minerals under the influence of an electric field is different at distinct frequencies due to their chemical compounds. It affects the electrical properties.

The computing of complex effective permittivity and complex effective conductivity of materials plays an important role due to its applications in different fields. The response of these properties under the influence of an alternating current field is used to characterise materials. The development of an approach to calculate these properties involves the solution of the second order elliptic partial differential equation as $\nabla \cdot [Q(\omega)\nabla u(x, y, z)] = 0$, where $Q(\omega) \in \mathbb{C}^{3 \times 3}$ represents the physical parameters of the different phases in the material, and $u(x, y, z)$ is the electric potential. The main difficulty in solving this equation comes from the high contrast of the coefficients in the distinct phases of the material.

There is an efficient approach that is used to compute the effective electrical conductivity of material under the influence of a static field. The material is represented in a 3D image. The Finite Element method and periodic boundary conditions are used to build an energy function which is minimised using the Conjugate Gradient algorithm. It allows to obtain the electric potentials, and then the computation of the conductivity is carried out. Moreover, this approach can also be used to calculate effective permittivity of material when a static field is applied, just by making a few changes in the approach. However, it is not possible to modify this approach to compute complex effective permittivity and complex effective conductivity because if one wants to obtain the electric potentials, a complex energy function has to be minimised.

This research is focused on developing a numerical scheme that allows to solve the second order elliptic partial differential equation with varying complex coefficients in order to obtain the electrical potentials. In the initial stage, a few tools of functional analysis are used to transfer the strong formulation into the variational or weak formulation in the appropriate functional space. A demonstration is made to prove that the sesquilinear form is bounded and V -elliptic. These conditions are necessary to use the Lax-Milgram theorem which guarantees that there is a solution, and that it is unique. In order to find the best approximation u_h to the solution u of the variational problem, the Galerkin method and the orthogonality condition between u and u_h are used to produce the best u_h in a given approximating subspace

in a finite-dimensional space. The process of construction of the finite-dimensional subspace is carried out using the Finite Element method.

The first stage of the numerical scheme consists in constructing a complex system of linear equations that arises from the second order elliptic partial differential equation. This is carried out by using the physical parameters of the material represented in a 3D image, the frequency where an electric field is applied, the employment of the Finite Element method, and the application of the Dirichlet and the Neumann boundary conditions. The second phase in the numerical scheme focuses on solving the complex system. The solution is computed using the technique of Hierarchical Matrices in combination with a Linear Method and the Generalised Minimal Residual Method algorithm. A C code was written to implement the scheme. The code uses the NetCDF library to read the 3D image and the \mathcal{H} -Lib^{Pro} library to work with the Hierarchical Matrices.

The scheme was evaluated using three artificial materials and three types of rocks with their 3D images, their electrical parameters, and their ranges of frequencies where the electric fields are applied. A complex system of linear equations is generated by each frequency within the range of each sample. In total, there are 199 complex systems of linear equations generated from the six different samples that were used to assess the scheme. The performance of the scheme is measured in terms of the convergence rate and the frequency. The numerical results show that the scheme is a robust tool to solve the second order elliptic partial differential equation to obtain the electric potentials, which are needed to compute the complex effective electrical properties.

Notation

$\ \cdot\ _{Y \leftarrow X}$	norm of a mapping (matrix) from X into Y
$(\cdot, \cdot), (\cdot, \cdot)_{0,\Omega}, (\cdot, \cdot)_{L^2(\Omega)}$	scalar product
$\langle \cdot, \cdot \rangle$	scalar product
$\langle \cdot, \cdot \rangle_{V \times V'}$	duality form
$\ \cdot\ _2$	Euclidean norm
Δ	gradient
\oplus_r	formatted matrix addition with truncation to rank r
\odot	formatted matrix-matrix multiplication
$\bullet _b, \bullet _{\tau \times \sigma}$	restriction of a matrix to a block b or $\tau \times \sigma$
η	factor in admissibility condition
$\#S$	cardinality of a set S
α, β, γ	indices of the index set
ε_{ij}	dielectric constant in the direction $i = j = \{x, y, z\}$
ε_0	dielectric constant of air
$\rho(M)$	spectral radius of a matrix M
τ, σ	symbols representing clusters
ω	frequency
Γ	the boundary of the Ω
Γ_D	Dirichlet boundary condition
Γ_N	Neumann boundary condition
$\Phi(u, b, L)$	function describing an iteration
Ω	an open set in \mathbb{R}^n or a domain
Ω_h	a grid
$a(\cdot, \cdot)$	sesquilinear form
$A_{\alpha\beta}, a_{\alpha\beta}, A_{ij}, a_{ij}$	entries of the matrix A
V_{m+1}^H	Hermitian transport matrix
\mathbb{C}	complex numbers
\mathbb{C}^I	complex space of the vectors corresponding to the index set I
$\mathbb{C}^{I \times I}$	complex space of the matrices corresponding to the index set I
$\mathfrak{D}(\Phi)$	domain of the iteration Φ
e^m	error $x^m - x$ of the m th iterate
\mathbf{f}	vector
f_i	entries of the vector \mathbf{f}
$G(A)$	graph of the matrix A
$H^1(\Omega), H_0^1(\Omega), H_{\Gamma_D}^1(\Omega)$	Sobolev Spaces
h	step size
\mathcal{H}_p	matrix model format

$\mathcal{H}(r, P)$	set of hierarchical matrices
i, j, k	indices of the ordered index set
I	identity matrix
I	index set
I_α	subset of block indices
I, J, K	index sets
$Init(\Phi, L)$	cost for initialising the iteration Φ
$\mathcal{K}_m(X, v)$	Krylov space
$L(X, Y)$	linear space of bounded operators from X to Y
$L^2(\Omega)$	space of square-integrable functions
\mathbf{L}	the stiffness matrix
L_{ij}	entries of the stiffness matrix
L	matrix
\mathcal{L}	set of consistent linear iteration
$\mathcal{L}(T(I))$	set of leaves of the cluster $T(I)$
M	matrix
N	matrix
\mathbb{N}	natural numbers $\{1, 2, 3, \dots\}$
\mathbb{N}_0	$\mathbb{N} \cup \{0\} = \{0, 1, 2, 3, \dots\}$
P	partition of a hierarchical matrix
P^+, P^-	far field, near field
Q	3×3 complex parameter matrix
Q_{min}	bounding box
$\mathcal{R}(r, I), \mathcal{R}_r$	set of rank- r matrices
\mathbb{R}	real numbers
$span\{\dots\}$	linear space spanned by $\{\dots\}$
$supp(\cdot)$	support of a function
$S_T(\tau)$	set of sons of $\tau \in T$
$\mathcal{T}_{r \leftarrow s}^{\mathcal{R}}$	truncation of a rank- s matrix to rank- r
$\mathcal{T}_r^{\mathcal{R}}$	truncation to rank r
$T(I)$	cluster tree belonging to the index set I
$T(I \times I)$	block cluster tree for $I \times I$
u_n	a grid function
V_h	finite element space
$Work(\Phi, L)$	amount of work of the iteration Φ applied to $Lu = b$
x^m	m -th iterate
X_τ	support of the cluster τ
\mathbf{y}	vector
y_i	entries of the vector \mathbf{y}

Contents

Acknowledgments	iv
Abstract	v
Contents	vi
Notation	vii
1 Introduction	1
1.1 Electrical properties of materials	3
1.2 Physical problem	6
1.2.1 Complex Permittivity	6
1.2.2 Complex Conductivity	8
1.2.3 Applications	9
1.2.4 Difficulty in computing complex effective properties	11
1.3 Mathematical problem	12
1.3.1 Description of the equation	12
1.3.2 Complexity of the numerical solution to the equation	13
1.4 Aim of thesis	15
1.5 Overview	17
2 Second Order Elliptic Partial Differential Equation	19
2.1 Introduction	19
2.1.1 Operators and Linear Functionals	20
2.1.2 Hilbert space	21
2.1.3 $L^2(\Omega)$, $H^1(\Omega)$, and $H_0^1(\Omega)$ spaces	23
2.2 Abstract variational problem	25
2.2.1 Variational problem	25
2.2.2 Dirichlet boundary condition	26
2.2.3 Neumann boundary condition	27
2.3 Galerkin method	27
2.4 Finite Element Method	30
2.5 Numerical solution of the partial differential equation	31
3 Construction of the Complex Linear System of Equations	34
3.1 Introduction	34
3.2 Model problem	34
3.3 Application of the Finite Element Method	37

3.4	Description of the 3D image data	38
4	Iterative Methods	43
4.1	Introduction	43
4.2	Iterative Methods	43
4.3	Linear Iterative Methods	44
4.3.1	Storage, computation work and efficacy	47
4.4	Richardson Iteration	48
4.5	Krylov Methods	48
4.5.1	Arnoldi algorithm	50
4.6	Generalised Minimal Residual Method	51
5	\mathcal{H}ierarchical Matrices	53
5.1	Introduction	53
5.2	\mathcal{H} -Matrices Construction	54
5.2.1	Cluster Trees	54
5.2.2	Block Cluster Tree	55
5.2.3	Admissible Blocks	56
5.3	Low-Rank Matrices	57
5.4	\mathcal{H} -Matrices	60
5.4.1	Format and Storage	60
5.4.2	Matrix-Vector Multiplication	62
5.4.3	Matrix-Matrix Multiplication	63
5.5	\mathcal{H} -LU Decomposition	65
5.5.1	Sparse Matrices	68
5.5.2	The \mathcal{H} -LU Decomposition for Sparse Matrices	69
5.5.3	Construction of the Cluster Trees and Admissibility Condition	71
5.5.4	Algebraic LU Decomposition	72
5.6	\mathcal{H} -LU Iteration	72
5.7	\mathcal{H} -Matrices for Solving Complex Linear Systems of Equations Using \mathcal{H} -Lib ^{Pro}	73
6	Numerical Results	76
6.1	Introduction	76
6.2	Artificial Samples	77
6.3	Rock Samples	85
7	Conclusions and Future work	94
7.1	Conclusions	94
7.2	Future Work	95
A	Graph and Matrix Graph	98

B LU Decomposition Procedures	99
B.1 The Forward and Backward substitution procedures	99
B.2 The Forward matrix and Forward transpose Matrix procedures	100
C \mathcal{H}-Lib^{Pro} Code	101
C.1 \mathcal{H} -LU decomposition at one frequency	101
C.2 \mathcal{H} -LU decomposition for a range of frequencies	103

List of Figures

1.1.1	The textural model filled with water and oil (left). The representation of the textural model with random distribution of oblate ellipsoids(right). Image after Abdullah et al. [2007]	3
1.1.2	(a) Real well sorted media. (b) Real poorly sorted media. Image after Boggs [20011].	4
3.2.1	(a) A porous material. (b) Discretisation of the porous material (<i>cont.</i>) .	35
3.2.1	(c). The grid points of the discretisation where the active points are represented by the black circles and the boundary points are illustrated by the red circles.	36
3.4.1	The construction process of the complex linear system for each frequency within a range using a 3D image, the physical properties of the material, Finite Element method and the boundary conditions. The complex linear systems are solved by using \mathcal{H} -Matrices.	42
5.2.1	Supports X_τ and X_σ . Image after Hackbusch [2015].	56
5.4.1	Block partitions. Image after Hackbusch [2015].	60
5.5.1	(a) The graphs G_1 and G_2 and the separator γ . (b) The graphs represented in the graph matrix.	69
5.5.2	The two level recursion: (a) The subgraphs $G_3, G_4, G_5,$ and $G_6,$ and the separators $\gamma, \gamma_1,$ and $\gamma_2.$ (b) The graph matrix of the subgraphs. (c) The corresponding block matrices of the \mathcal{H} -matrix.	70
5.5.3	The factor L where the white colour represents the zero blocks. Image after Hackbusch [2016].	72
6.2.1	The convergence behaviour of the solutions of the complex systems of linear equations generated by the interval of frequencies from the sphere sample is plotted vs the frequency. Only one \mathcal{H} -LU decomposition at the first frequency is computed and it is used to solve all complex systems.	78

- 6.2.2 (a) The convergence of the solutions of the complex systems of equations from the sphere sample is plotted vs the frequency. The different subintervals are represented by distinct colours. The lowest peaks of convergence rate in each subinterval correspond to the frequency where the \mathcal{H} -LU decomposition is computed. For the rest of the frequencies in each subinterval, its corresponding \mathcal{H} -LU decomposition is used to solve the complex systems of equations. (b) The convergence rate vs frequency are plotted for three subintervals of frequency. 80
- 6.2.3 The convergence behaviour of the solutions of the complex systems of linear equations generated by the interval of frequencies from the random voxel sample is plotted vs the frequency. Only one \mathcal{H} -LU decomposition at the first frequency is computed and it is used to solve all complex systems. 81
- 6.2.4 (a) The convergence of the solutions of the complex systems of equations from the random voxel sample is plotted vs the frequency. The different subintervals are represented by distinct colours. The lowest peaks of convergence rate in each subinterval correspond to the frequency where the \mathcal{H} -LU decomposition is computed. For the rest of the frequencies in each subinterval, its corresponding \mathcal{H} -LU decomposition is used to solve the complex systems of equations. (b) The convergence rate vs the frequency are plotted for three subintervals of frequency. 83
- 6.2.5 (a) The convergence iteration of the solutions of the complex systems of equations from the sphere crystal sample is plotted vs the frequency. The different colours represent distinct subintervals of frequency. The lowest peaks of convergence rate in each subinterval are associated to the frequency where the \mathcal{H} -LU decomposition is calculated. For the rest of frequencies in each subinterval, its corresponding \mathcal{H} -LU decomposition is used to solve the complex systems of equations. (b) The different convergence rates are computed in the first subinterval using three accuracy values. 86
- 6.2.6 The convergence speed vs the frequency are plotted for three subintervals of frequency. 87
- 6.3.1 (a) The convergence of the solutions of the complex systems of equations from the heterogeneous rock sample is plotted vs the frequency. The different subintervals are represented by distinct colours. The lowest peaks of convergence rate in each subinterval correspond to the frequency where the \mathcal{H} -LU decomposition is computed. For the rest of the frequencies in each subinterval, its corresponding \mathcal{H} -LU decomposition is used to solve the complex systems of equations. (b) The convergence rate vs the frequency are plotted for three subintervals of frequency. 89

- 6.3.2 (a) The convergence of the solutions of the complex systems of equations from the Bentheimer sample is plotted vs the frequency. The different subintervals are represented by distinct colours. The lowest peaks of convergence rate in each subinterval correspond to the frequency where the \mathcal{H} -LU decomposition is computed. For the rest of the frequencies in each subinterval, its corresponding \mathcal{H} -LU decomposition is used to solve the complex systems of equations. (b) The convergence rate vs the frequency are plotted for three subintervals of frequency. 91
- 6.3.3 (a) The convergence of the solutions of the complex systems of equations from the Berea sample is plotted vs the frequency. The different subintervals are represented by distinct colours. The lowest peaks of convergence rate in each subinterval correspond to the frequency where the \mathcal{H} -LU decomposition is computed. For the rest of the frequencies in each subinterval, its corresponding \mathcal{H} -LU decomposition is used to solve the complex systems of equations. (b) The convergence rate vs the frequency are plotted for three subintervals of frequency. . . 92

Introduction

The development of materials has an incalculable impact on our daily life. This is important for different industries such as aerospace, automotive, biological, chemical, electronic, energy, metals, and telecommunications. Each major technology is underlain by understanding the behaviour and properties of materials. When materials are exposed to external stimuli, they generate some type of response. The properties of materials can be measured in terms of the kind of response and its magnitude under the influence of a specific stimuli.

The response of a material under the application of a magnetic field shows the magnetic properties. When light or electromagnetic radiation is used as a stimulus, the optical properties are represented by an index of refraction and reflectivity. The capacity and thermal conductivity are properties of solids that describe their thermal behaviour. The electrical properties such as electrical conductivity and dielectric constant can be measured when an electric field is used as a stimulus. The elastic modulus, strength and toughness are mechanical properties which are related to the deformation of materials when load or force is applied to them.

The analysis of 3D digital images of materials has become the most popular tool to generate information about their structures and properties. A virtual material laboratory to study real complex material was developed by a group of researchers in the Applied Mathematics Department at the Australian National University. Even though their focus is on oil and gas applications, the scheme developed by them is also used to characterise materials in general. Basically, the scheme has four steps: data exploration is the first step and it consists in identifying the configuration of the phases or components in the material using computer visualisation. The second step is the data segmentation which classifies the information in each voxel of the image according to an assumption and a single grayscale image. The representation of the phases in the voxels is very important to quantify the physical properties from the image. The morphological and geometrical analysis is the third step; it produces information about how the different phases are connected and how is their geometrical structures are. The last step is the numerical analysis, that comes as a results of applying the first three steps; the phases, pore, and grains in the materials are represented in different portions in the image. Then, the physical properties are computed assigning the constituent material properties, and solving the adequate equation under suitable boundary conditions [Sakellariou et al., 2007].

Computational rock physics is an approach defined by a group of researchers in the School of Earth Science, Energy, and Environmental Sciences at Stanford University. The approach is based on imaging rock to simulate the physical process at the pore space level in order to calculate physical properties. This consists of three basic steps: a micro-CT-scan machine and X-ray are used to generate the 3D image of a small rock sample. The 3D image is constructed tomographically. A nano-CT scan can be used as well. The second step is the image process and segmentation; during this process diverse artifacts may appear in the raw image and they are eliminated. A gray scale with a small integer number is used to differentiate between the pore space, the mineral matrix and the fluids. The simulation of physical property is the last step, where the segmented image is used to simulate physical processes [Dvorkin et al., 2011].

The computing of material properties using 3D digital images and discrete computational methods have become a very powerful tool. One of the most important references is the work developed by E.J. Garboczi from the National Institute of Standards and Technology. His approach is based on using 2D and 3D images, Finite Difference Method, and Finite Element Method to compute effective linear elasticity, effective thermal conductivity, and effective electrical conductivity in the presence of a static electric field [Garboczi et al., 1999, Garboczi, 1998a]. These properties can be calculated by several sequential Fortran codes written by him [Garboczi, 1998b]. The parallel versions of the codes are reported in [Bohn and Garboczi, 2003]. All the codes have free access. Moreover, Y. Keehm in the group of Computational Rock Physics at Stanford University wrote a PhD thesis about transport properties in porous media, basically following the approach of his group to simulate electrical conductivity in 3D images of Fontainebleau sandstone using the Diffpack software library for the Finite Element Method [Keehm, 2003]. A different numerical technique Finite Volume Method is used by P. Øren and S. Bakke to calculate formation factor as a function of electrical conductivity at zero frequency using a 3D microstructure of sandstone [Øren and Bakke, 2002]. This numerical method is also used by Wei to calculate thermal property of cellular concrete using 3D X-ray Computerised Tomography Images [Wei et al., 2014].

Garboczi's approach is the most common methodology used to simulate elastic and electrical properties of porous media represented in a 3D image [Sain, 2010, Andrä et al., 2013, Dvorkin et al., 2011]. The same scheme has also been used to compute the static electrical conductivity [Richa, 2010, Sun et al., 2014, Arns et al., 2001, Zhan et al., 2009] and the static linear elasticity of porous materials [Arns et al., 2002, Makarynskaa et al., 2008, Madadi et al., 2009]. Furthermore, there are companies such as Lithicom [Ringstad et al., 2013], and INGRAIN [Dvorkin, 2009, INGRAIN, 2009] which employ the same approach. The scheme will be described below.

This thesis is focused on the development of a numerical scheme to solve a second order elliptic partial differential equation in order to compute effective complex electric properties of materials. The scheme is based on a 3D image, on Finite Element methods, and on the Dirichlet and Neumann boundary conditions. The materials are represented in the 3D image. The sparse linear system of equations generated by the

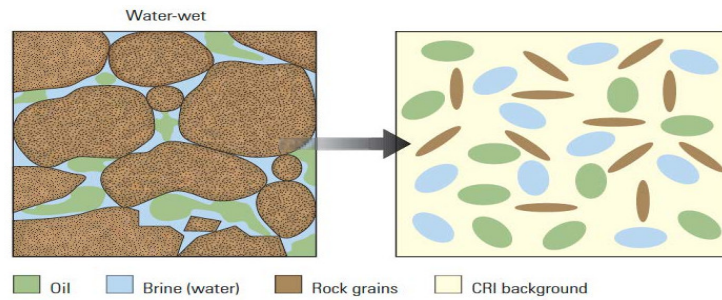


Figure 1.1.1: The textural model filled with water and oil (left). The representation of the textural model with random distribution of oblate ellipsoids(right). Image after Abdullah et al. [2007]

discretisation of the partial differential equation is solved using different numerical techniques. The emphasis of this study is on the numerical aspects for the solution of the equation.

1.1 Electrical properties of materials

The measurement and calculations of electrical properties of materials are important for their use in different fields such as food science, medicine, biology, agriculture, and chemistry. In particular, the electrical conductivity and permittivity are computed using analytical equations and numerical techniques. The macroscopic properties of inhomogeneous material can be described by analytical or theoretical modelling making use of the Effective Medium Theory, such as conductivity, dielectric permittivity or elastic modulus. Based on the relative fraction of the components in the medium and the physical properties of each fraction, the Effective Medium Theory generates models to approximate the effective properties of the whole material. Some of the models are Bruggnan, Maxwell-Garnett, and Clausius-Mossoti [Choy, 1999].

For example, rocks are commonly inhomogeneous materials due to a mixture of minerals, voids, and cracks. Berryman using the Effective Medium Theory and the Theory of Mixtures describes how to compute effective conductivity, dielectric permittivity, thermal conductivity, and elasticity in this sort of media. The components of rocks, which are the inclusions, are represented by spheres, ellipsoids, needles, and discs. The inclusions are immersed in a host medium that for example corresponds to a fluid [Berryman, 1995].

The Mixing Laws is another analytic approach to calculate effective conductivity and effective permittivity. This is also based on the Effective Medium Theory and the average of polarisation processes that occur in the material [Sihvola, 2008]. For example, the most simple model for a dielectric mixture is given by isotropic dielectric spheres or ellipsoids, or a combination of them which are embedded as an inclusion or guest in an isotropic dielectric environment or host. The macroscopic permit-

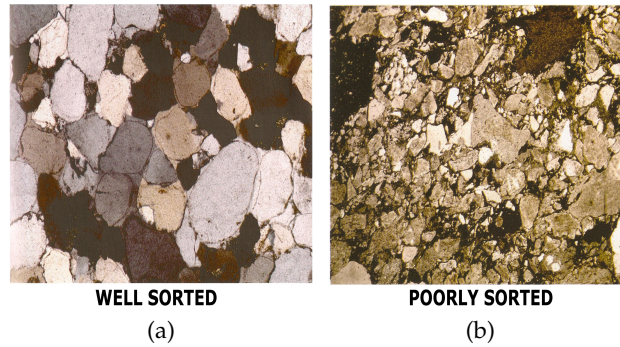


Figure 1.1.2: (a) Real well sorted media. (b) Real poorly sorted media. Image after Boggs [20011].

tivity of the mixture can be calculated if the permittivities of the components are known. There are different formulas to compute the effective permittivity. They are the Maxwell-Garnett formula for spheres and for ellipsoids, homogenisation formulas, coherent potential formulas, power-Law models, and differential mixing models. Several of these models have been used by Seleznev to compute effective conductivity and the effective permittivity of carbonate rocks in different conditions. The experimental and analytical results show that some models matched much better than others. The results are well described in [Seleznev and Boyd, 2004, Seleznev et al., 2006].

The Mixing Laws work effectively for well sorted media, for example the rock in Figure 1.1.2a. Unfortunately, there are a lot of mediums that are poorly sorted (Figure 1.1.2b), hence these formulas are not applicable. This is the main reason to look for alternative forms to compute the electrical properties. The 3D image of materials and numerical techniques can be used to develop a scheme in order to calculate the properties.

The computation of the effective electrical conductivity of material represented in 3D image and under the influence of a static electric field can be carried out using Garboczi's approach [Garboczi, 1998b]. In his scheme, the finite element is represented by a voxel or cube with 8 nodes, and an electrical potential is applied to each node. The approximation of the potential (ϕ_e) within an element is determined by the tri-linear interpolation, and it interrelates with the potential distribution in various elements so that the potential is continuous across interelement boundaries. The potential function is expressed as $\phi_e(x, y, z) = \sum_{i=1}^8 \alpha_i(x, y, z) \phi_i$, where α_i is the interpolation function, and ϕ_i is the potential, and the index i corresponds to the node in the cube. The electric field in the voxel is obtained by $E_e = -\nabla \phi_e(x, y, z)$. The function of energy corresponding to the equation of local current density (the Ohm law, $J_e = \sigma_p E_e$) is $W_e = \frac{1}{2} \int_0^1 \int_0^1 \int_0^1 \sigma_p |E_e|^2 dx dy dz$, where σ_p is the conductivity of the phase (material) within the element that can be a constant or a 3×3 matrix. When the process of assembling over all elements of the material is carried out, the total energy is given by $W(\Phi) = \sum_{e=1}^N W_e = \frac{1}{2} \Phi^T A \Phi$, where Φ is the vector potential over

the whole image, and the global stiffness matrix is A . When the periodic boundary conditions are applied, this equation becomes $W(\Phi) = \frac{1}{2}\Phi^T A\Phi + b\Phi + C$, where b is a vector, and C is a constant. The global current density equation is satisfied when the total energy in the solution is minimum; then it requires that the partial derivative of the function W with respect to each node value of the potential be zero, i.e. $\frac{\partial W}{\partial \phi_k} = 0$.

Matrix A in the total energy function $W(\Phi)$ is symmetric and positive definite, and the approach uses the Conjugate Gradient (CG) algorithm [Press et al., 1990] to minimise the energy function, hence the potentials are computed. The electric field in the voxel (E_e) is figured out using the potentials of the voxel. Now, the conductivity σ_p and the electric field within the element are used to apply the Ohm law in order to compute $J_e = \sigma_p E_e$. The average of the electric field ($\bar{E} = \sum_{e=1}^N E_e / N$) and current density ($\bar{J} = \sum_{e=1}^N J_e / N$) over the image are calculated in order to use the Ohm law again and express the effective conductivity as $\sigma_{eff} = \bar{J} / \bar{E}$.

The effect of a static electric field to a dielectric material represented in a 3D image can be measured by computing the effective permittivity. The calculation is accomplished by making a few changes to Garbozci's scheme. Instead of using the Ohm law to calculate the local current density, the equation of the displacement field ($D_e = \epsilon_p E_e$) is applied in each element, where ϵ_p is the dielectric constant of the phase which can be a constant or a 3×3 matrix. The form of the function $W(\Phi)$ is equal to the function that is used to calculate the effective conductivity but the coefficients of the matrix A change. This is due to the use of D_e and the assembling of finite elements. All the procedure is the same until the electric field (E_e) in each voxel is computed. The electric displacement field for each cube is computed using the dielectric constant ϵ_p of the phase and the electric field corresponding to each the element. Then the averages of the electric displacement field ($\bar{D} = \sum_{e=1}^N D_e / N$) and the electric field ($\bar{E} = \sum_{e=1}^N E_e / N$) are calculated, hence the effective permittivity is expressed by the equation $\epsilon_{eff} = \bar{D} / \bar{E}$.

The mathematical justification of why Garbozci's approach works very well is due to the important properties that the real matrix A has. After using Finite Element Method and applying periodic boundary conditions, the real matrix A generated in the function $W(\Phi)$ is symmetric and positive definite. The symmetry turns out when the conductivity and the dielectric constant are constants or when they are 3×3 with values only in the diagonal. As matrix A is symmetric and positive definite, the partial derivative of the function $W(\Phi)$ can be expressed as a linear system of equation $Ax = b$ [Hackbusch, 2016]. If $\langle Ax, x \rangle > 0$ for all real vectors $x \neq 0$, then A is regular and the linear system has a unique solution. The advantage of using the CG algorithm is that its convergence is always guaranteed if only the floating-point errors do not have take over.

1.2 Physical problem

1.2.1 Complex Permittivity

The use of materials is determined by their properties such as mechanical, chemical, electrical, thermal, optical, and magnetic, and they are applied in their relevant fields in engineering. Materials with dielectric (insulators) and conductive components under the influence of an electric field may be characterised by electrical properties such as permittivity and conductivity. One important property of a dielectric material is its permittivity. It is a measure of the ability of the material to be polarised by an electric field (E). The influence of the electric field on the configuration of the electrical charges in a given material is represented by the electric displacement field (D). These fields are related to the permittivity as

$$D = \varepsilon E \quad (1.2.1)$$

where ε is absolute permittivity or permittivity.

In a static field of moderate intensities, the permittivity is only dependent on the chemical composition and the density of the material, and not on the electric field. In that case, the permittivity is often called dielectric constant. Hence equation (1.2.1) holds for a static field, and it also holds for an alternating field as long as the frequency does not exceed certain critical values. For linear, homogeneous and isotropic dielectric materials, equation (1.2.1) is fulfilled. However, it does not hold for anisotropic ones, where $D = (D_x, D_y, D_z)$ is a vector function of $E = (E_x, E_y, E_z)$ and the dielectric constant is replaced by a 3×3 matrix:

$$\left. \begin{aligned} D_x &= \varepsilon_{xx}E_x + \varepsilon_{xy}E_y + \varepsilon_{xz}E_z \\ D_y &= \varepsilon_{yx}E_x + \varepsilon_{yy}E_y + \varepsilon_{yz}E_z \\ D_z &= \varepsilon_{zx}E_x + \varepsilon_{zy}E_y + \varepsilon_{zz}E_z \end{aligned} \right\}$$

When an alternating or time varying electric field is applied to a dielectric material, there are two possible cases which depend on the frequency of the field, the temperature and the type of material. For the first case, there is no measurable phase difference between D and E , which means that the polarisation is in phase with the alternating field, then the relation $D = \varepsilon E$ is valid. In this condition, no energy is absorbed by the dielectric from the electromagnetic field. In the second case, the phase difference between D and E is noticeable. Then, the relation $D = \varepsilon E$ is not valid and there is a dissipation of energy in the dielectric which is generally called dielectric loss [Böttcher, 1952, Hippel, 1995, Scaife, 1998, Kao, 2004]. Consider a dielectric material inserted between two plates to form a capacitor. To calculate the dissipation of energy, an alternating voltage is applied to the condenser plates, leading to a periodical alternating electric field E , represented by

$$E = E_0 e^{i\omega t} \quad (1.2.2)$$

where E_0 is the amplitude, ω the circular frequency and t the time.

For the simulation of mediums which are made up of complex geometries and inhomogeneous regions composed of several materials, numerical techniques based on partial differential equations and integral equations have been developed [Booton, 1992]. When the medium consists of dielectric and conductivity compounds, the numerical solution becomes time dependent, and it is given by the complex electric potential in conductivity and non-conductivity (insulator) regions. The solution is reached by an equation which is known as the continuity equation for the current density. It can be deduced by starting with the Ampere law:

$$\nabla \times H = J + \frac{\partial D}{\partial t} \quad (1.2.3)$$

substituting current density ($J = \sigma E$) and electric displacement ($D = \epsilon_r \epsilon_0 E$) equations

$$\nabla \times H = \sigma E + \frac{\partial(\epsilon_r \epsilon_0 E)}{\partial t} \quad (1.2.4)$$

using the alternating electric field (equation (1.2.2)) and after doing some operations the equation (1.2.4) is as follows:

$$\nabla \times H = i\omega \epsilon_0 \left(\epsilon_r - i \frac{\sigma}{\omega \epsilon_0} \right) E \quad (1.2.5)$$

with

$$\epsilon^*(\omega) = \left(\epsilon_r - i \frac{\sigma}{\omega \epsilon_0} \right)$$

where $\epsilon^*(\omega)$ is the complex permittivity. The ϵ_r and ϵ_0 are the dielectric constant of the material and the dielectric constant of air, respectively. The σ is the electrical conductivity of the material. As a divergence of the Curl of a vector is zero [$\nabla \cdot (\nabla \times A) = 0$], when applying this property to equation (1.2.5), we obtain

$$\nabla \cdot \nabla \times H = \nabla \cdot (i\omega \epsilon^*(\omega) E) \Rightarrow \nabla \cdot (\epsilon^*(\omega) E) = 0$$

substituting the electric field ($E = -\nabla \phi$), hence the complex permittivity

$$\nabla \cdot \left[\left(\epsilon_r - i \frac{\sigma}{\omega \epsilon_0} \right) \nabla \phi \right] = 0 \quad (1.2.6)$$

which is a second order elliptic partial differential equations with varying complex coefficients.

When the voltage distribution (ϕ) is known, the complex permittivity of the heterogeneous medium or material can be calculated. There are several ways to do it: by using total current J and the phase difference θ [Scaife, 1998, Tuncer and Gubański, 2001, Sareni et al., 2001], energy balance [Sareni et al., 2001, 1997, Yonghong et al., 2008], and by using average values of the electric displacement \bar{D} and the electric field \bar{E} [Scaife, 1998, Landau et al., 1984].

1.2.2 Complex Conductivity

The characterisation of electrical charge transport is described by the Ohm law. The influence of a static electric field (E) on a material causes the displacement of various charged particles which gives rise to an electric current density (J). The displacement is represented by a proportional constant of the material property called electrical conductivity. For isotropic material, the relation is as follows:

$$J = \sigma E \quad (1.2.7)$$

where E is expressed in Volts/m and J in Amperes/m². When a material is anisotropic, the equation (1.2.7) is replaced by

$$\left. \begin{aligned} J_x &= \sigma_{xx}E_x + \sigma_{xy}E_y + \sigma_{xz}E_z \\ J_y &= \sigma_{yx}E_x + \sigma_{yy}E_y + \sigma_{yz}E_z \\ J_z &= \sigma_{zx}E_x + \sigma_{zy}E_y + \sigma_{zz}E_z \end{aligned} \right\}$$

where the electrical current is a vector $J = (J_x, J_y, J_z)$, and the electric field is vector $E = (E_x, E_y, E_z)$ as well, and σ is a 3×3 matrix. The conductivity is an intrinsic material property independent of the sample geometry [Rajinder, 2015, Nabighian, 1988].

In semiconductor materials the current density is able to follow the alternating current (AC) fields only at frequencies low enough. In this case, the value of conductivity would have the same magnitude that when a direct current (DC) field is applied. When AC fields are applied to the material and they are not high enough to heat the charge carriers, the electric field at frequencies for which $\omega\tau$ is comparable to unity, may no longer follow by the current density. τ represents the mean free time between collisions of charge carriers, and depends on the magnitude of the applied field. When the temperature and the field magnitude increase, τ decreases. This behaviour can be described by the complex conductivity $\sigma^*(\omega) = \sigma'(\omega) + i\sigma''(\omega)$. The real part of $\sigma^*(\omega)$ represents the in-phase conductivity where the current density is capable of following the field. The conductivity out of phase ($\pi/2$ lagging the field) is expressed by the imaginary part of $\sigma^*(\omega)$ [Kao, 2004].

The total current density is formed by two contributions: one is associated with the electromigration of the charge carriers and the second comes from the polarisation process of the material. The Ampere law (1.2.3) is taken again and using the same procedure to generate the equation (1.2.6), the resultant equation is expressed as

$$\nabla \cdot [(\sigma + i\omega\epsilon_r\epsilon_0)\nabla\phi] = 0 \quad (1.2.8)$$

where σ and ϵ are both scalars or 3×3 matrices. The complex conductivity is $\sigma^*(\omega) = \sigma + i\omega\epsilon_r\epsilon_0$ which can be computed for heterogeneous medium using the equation (1.2.8). In general, the complex effective permittivity and complex effective conductivity are calculated either using the equation (1.2.6) or (1.2.8).

1.2.3 Applications

Materials may be characterized by using its dielectric properties which establish how materials interact when an electric field is applied at various frequency ranges. This interaction is used to determine properties of the material such as moisture content, bulk density, bio-content, chemical concentration and stress-strain. The relationship between dielectric properties and other properties of the material plays an important role for research and application in food science, medicine, biology, agriculture, chemistry, electric device, defence industry, to name a few. For instance, the development of agricultural technology depends upon available data on the dielectric behaviour of agricultural products. Data on frequency dependence of the electric properties of grain and insects are needed to determine the optimum frequency range for selective dielectric heating of insects and the control of stored-grain insects with radio-frequency energy [Nelson, 1974]. Other applications that depend upon dielectric properties of grain and seed include radio-frequency treatment of hard seeds to increase germination [Nelson, 1976] and electrical measurement of moisture content in grain [Nelson, 1973]

Dielectric properties are important physical properties associated with radio frequency and microwave heating systems. Thus, for the development of production and processing of food it is critical to have available data with its dielectric properties due to the fact that the dielectric behaviour of food is affected by their heating characteristics. For example, it is crucial for the design of heating systems of food [Wang, 2005] and when choosing appropriate materials for containers and packaging [Ohlsson, 1989].

The electrical properties of biological material are of key interest for different reasons. These properties determine the pathway of current flow through the human body. It has been of fundamental importance in studies of biological effects of electromagnetic fields in which physiologic parameters can be measured. Moreover, they can be used in basic and applied studies in electrocardiography, muscle contraction and nerve transmission. For example, in cardiology, the knowledge of dielectric properties of tissues at low frequency permits the analysis of distribution of currents and potentials generated by the heart. For tissue and cell suspensions studies, their dielectric properties are related to a structural analysis of organism, mechanism of excitation and the analysis of characteristics of protein molecules, such as dipole moment, shape and hydration [Schwan, 1957, Gabriel et al., 1996a,b].

The increase in demand for textile materials (clothing, household and special applications) is motivated by the need to improve their properties. A great amount of textile materials are dielectrics. Thus, there has been a lot interest in studying their behaviour under the influence of an electric field for which it is important to determine the tangent of losses, the relative permittivity and the electric resistance. For instance, natural and synthetic textile fibres have a polymeric structure whose properties depend on the molecular structure of the polymeric molecules which constitute the fibre, the arrangement of macromolecules within the fibre and the external characteristics of the fibre [Browning, 1974]. A dielectric relaxation phenomena have been

used to study how the molecule structures form polymers [McCrum et al., 1967]. The dielectric properties have been employed to reduce the static generation in the textile industry [Morton and Hearle, 1993], to measure moisture content in textiles [Spencer-Smith and Mathew, 1936] and to observe moisture transmission through textile [Ito and Muraoka, 1993].

Petroleum drilling is an essential stage of the oilfield exploration whose expenditures represent 75% of the total exploitation cost. The largest source of trouble, waste of time and additional costs during drilling is the wellbore instability. This serious problem mainly occurs in shale (principally clay) which represent 75% of all formation industry drilled by oil and gas. The physical properties and behaviour of shale exposed to drilling fluids depend on the type and amount of clay in the shale. Wellbore stability is due to the dispersion of the clay into ultra-fine colloidal particles and this has a direct impact on the drilling fluid properties. Clay characterisation is the main parameter that allows to understand borehole stability. Clay minerals are considered as particularly active colloids, partly because of their anisotropy due to their shape (tiny platelets), and partly because of their molecular structure which represents high negative charges, mainly on their basal surface and possible negative charges on their edges. Interaction between these opposite charges strongly influences the viscosity of clay at low velocities. A method developed for shale characterisation is based on the dielectric constant which is used to quantify the swelling of clay content and to determine a specific area [Leung and Steig, 1992].

The influence of clay on the electrical response of the reservoir rock, and problems associated with its interpretation, have been major issues of investigation in the petroleum industry for many years. A new generation of logging tools that are capable of measuring electrical properties over a wide range of frequency have become available. Consequently, attention is shifting towards the possibility of using the frequency at varying electrical responses as a method of extracting information about clays present in reservoir. Al-Mjeni et al. [2002] studied the relationship between clay type and concentration, and the complex impedance and dielectric constant. In the case of the petroleum industry, complex impedance has the attraction of being a non-invasive technique, which measures the rock over a range of frequencies. The impedance value, dielectric constant and their frequency dependencies have been used as tools to estimate various properties of rock such as grain shape, permeability, porosity, water saturation and wettability.

An important petrophysical property is the wettability; it impacts on the reservoir behaviour which is reflected in the fluid saturation, multiphase flow and some parameters used to interpret well logs. For instance, the wettability data is critical to apply enhanced oil recovery methods. The dielectric response of rock is affected by its wettability because the wetting fluid tends to fill the smallest pore and form a thin continuous film over the solid surface. On the other hand, the non-wetting fluid tends to place itself principally at the centre of the large pore. As a consequence the mechanisms of polarisation in the pore space are being affected by the shape of the water phase, in particular the space charge polarisation. Discontinuities of charge concentration are created at the water-oil and oil-water interfaces when an electric

field is applied to a saturated rock. The discontinuities increase the polarisation in the medium and decrease its effective conductivity. A lot of work has been done to characterise the rock wettability by using dielectric measurements at different frequencies [Wael et al., 2007, Bona and Rossi, March, 2001, Garrouch, 2000, Bona, 1998].

Rocks are aggregates of minerals of more or less invariable composition whose properties depend primarily on the chemical composition of minerals and its macrostructure. The response of minerals under the influence of an electric field is different at distinct frequencies due to their chemical compounds. Thus, it affects the dielectric properties of rocks. Additional factors such as moisture, pressure and textural characteristics of rocks also have an influence on them. In particular, the texture has an effect on the space charge polarisation. The dielectric properties of sedimentary rocks have been studied. Models have been developed and experiments have been carried out to look into how the dielectric constant and conductivity at different frequencies are influenced by surface and geometrical effects, and scale invariance. For clay particles with surface active and plate-like, it was determined that these effects contribute to dielectric constants [Sen, 1980, Sen et al., May, 1981, Sen, December, 1981].

1.2.4 Difficulty in computing complex effective properties

The computing of complex effective permittivity and complex effective conductivity of materials plays an important role due to applications in different fields. The characterisation of materials can be carried out by a study of the response of these properties under the influence of an alternating current field. A scheme or methodology should be developed to do it. An interesting starting point is Garboczi's approach. A few modifications have to be done in order to incorporate the complex dielectric $\epsilon^*(\omega)$ which depends on the application of an alternating current field at different frequencies. The total energy function $F(x) = \frac{1}{2}x^H Ax + bx + C$ has the same form as before, where vector x represents the voltage potentials and its transpose conjugate is denoted by H ; vector b and constant C come from the periodic boundary conditions; and the global stiffness matrix A has complex numbers as coefficients, and the matrix is complex symmetric.

The fundamental difficulty in minimising the complex function $F(x)$ is that the minimisation can be done only for real function, and the matrix $x^H Ax$ is not real. However, if the matrix A is positive definite, the function can be minimised. Unfortunately, this is not the case. The attractive property is lost given that the equivalent property for a real symmetric matrix is a Hermitian matrix. Even though the complex dielectric or complex conductivity is used as a complex scalar, the generated stiffness matrix is still a complex symmetric one. The minimisation of function $F(x)$ or the solution of the system of equations $Ax = b$ where the matrix A is complex symmetric, is very demanding owing to the diagonalisation process of the matrix A can be difficult. This is possibly the key reason why no one may have been capable of developing a formulation to calculate these sort of physical properties.

1.3 Mathematical problem

The second order elliptic partial differential equation with real coefficients ($\nabla \cdot (a(x, y, z)\nabla u(x, y, z)) = 0$) has to be solved to compute the effective conductivity or effective permittivity at DC field. The conductivity or the dielectric constant coefficients are represented by $a(x, y, z)$, and the function $u(x, y, z)$ corresponds to the potentials. The solution of this equation is reasonably easy to compute given that the matrix generated after the discretisation of the equation is real symmetric.

The computing of complex effective permittivity and complex effective conductivity is much more difficult because of a second order elliptic partial differential equation with varying complex coefficients that must be resolved. The discretisation of the equation produces a complex matrix which is not necessary diagonalisable, and it makes it very challenging to solve the equation. For the development of an approach, it is important to depict the relationships (1.2.6) and (1.2.8) from the mathematical point of view. Moreover, it is essential to review the possible numerical techniques to be used in order to solve the equation. Once the solution is found, i.e. the voltage distribution is known, the physical properties can be calculated.

1.3.1 Description of the equation

A second order elliptic partial differential equation in a domain where the complex coefficients vary in space must be solved to compute the complex effective electrical properties. The domain of the equation is a 3D image and the complex coefficients are defined by the physical properties of the material components represented in the image. For example, the complex effective permittivity can be calculated by solving $\nabla \cdot [\epsilon^*(\omega)\nabla\phi(x, y, z)] = 0$ (equation (1.2.6)). While the solution of $\nabla \cdot [\sigma^*(\omega)\nabla\phi(x, y, z)] = 0$ is used for the computation of the complex effective conductivity. In general, the second order elliptic partial differential equation takes the following form:

$$\nabla \cdot [a(x, y, z)\nabla u(x, y, z)] = g(x, y, z) \quad \text{in } \Omega \quad (1.3.1)$$

$$u(x, y, z) = g_D \quad \text{on } \Gamma_D \quad (1.3.2)$$

$$\frac{\partial u(x, y, z)}{\partial n} = g_N \quad \text{on } \Gamma_N \quad (1.3.3)$$

where the complex coefficients $a(x, y, z)$ for the complex permittivity $\epsilon^*(\omega)$ are the components of the following matrix:

$$a(x, y, z) = \begin{bmatrix} \epsilon_{xx} - i\frac{\sigma_{xx}}{\omega\epsilon_0} & \epsilon_{xy} - i\frac{\sigma_{xy}}{\omega\epsilon_0} & \epsilon_{xz} - i\frac{\sigma_{xz}}{\omega\epsilon_0} \\ \epsilon_{yx} - i\frac{\sigma_{yx}}{\omega\epsilon_0} & \epsilon_{yy} - i\frac{\sigma_{yy}}{\omega\epsilon_0} & \epsilon_{yz} - i\frac{\sigma_{yz}}{\omega\epsilon_0} \\ \epsilon_{zx} - i\frac{\sigma_{zx}}{\omega\epsilon_0} & \epsilon_{zy} - i\frac{\sigma_{zy}}{\omega\epsilon_0} & \epsilon_{zz} - i\frac{\sigma_{zz}}{\omega\epsilon_0} \end{bmatrix} \quad (1.3.4)$$

where $\epsilon_{ij}, \sigma_{ij} \in \mathbb{R}$ for $i = j = \{x, y, z\}$ are dielectric constant, and conductivity, respectively. They are the physical properties of the material. ϵ_0 is the air dielectric constant. The frequency is $\omega \in \mathbb{R}$ and varies as $\omega_{min} \leq \omega \leq \omega_{max}$.

For the complex conductivity $\sigma^*(\omega)$, the coefficients are as follows:

$$a(x, y, z) = \begin{bmatrix} \sigma_{xx} + i\omega\epsilon_{xx}\epsilon_0 & \sigma_{xy} + i\omega\epsilon_{xy}\epsilon_0 & \sigma_{xz} + i\omega\epsilon_{xz}\epsilon_0 \\ \sigma_{yx} + i\omega\epsilon_{yx}\epsilon_0 & \sigma_{yy} + i\omega\epsilon_{yy}\epsilon_0 & \sigma_{yz} + i\omega\epsilon_{yz}\epsilon_0 \\ \sigma_{zx} + i\omega\epsilon_{zx}\epsilon_0 & \sigma_{zy} + i\omega\epsilon_{zy}\epsilon_0 & \sigma_{zz} + i\omega\epsilon_{zz}\epsilon_0 \end{bmatrix} \quad (1.3.5)$$

The function $u(x, y, z)$ represents the voltage potential at the node position (x, y, z) in 3D image which is the domain space Ω . The boundary of the domain $\partial\Omega$ is split into two nonempty disjoint open sets: Γ_D and Γ_N , they are Dirichlet and Neumann boundary conditions, respectively.

An important aspect to consider in order to solve the equation (1.3.1) is the condition that should be fulfilled by the coefficients ϵ_{ij} and σ_{ij} . They have to be positive definite. The sort of materials used in this study is a mixture of dielectric (electrical insulator) and conductive components. It is assumed that all the dielectric constants of the insulators have positive values. In terms of conductivity, the insulators have very low conductivities ranging between $10^{-10} (\text{Ohm.meter})^{-1}$ and $10^{-20} (\text{Ohm.meter})^{-1}$; while the components with high electrical conductivity are above $10^7 (\text{Ohm.meter})^{-1}$ [Moliton, 2007, Mitchell, 2004].

1.3.2 Complexity of the numerical solution to the equation

The solution of partial differential equations is not an easy task. There are different methods for the numerical treatment of them. The methods are built out by taking a set of discrete points to approximate the solution, and where the differential equation should be satisfied by the points. Some methods do not assume that the differential equation holds at every points; they are based on a weak formulation or a variational problem of the partial differential equation. The integrals in the variational problem have linear forms that require to use appropriate function spaces to assure the existence of the weak solution. The existence theorems for the weak solutions are valid under assumptions which are much more realistic than the assumptions for the existence theorems used to find the classical solution.

The partial differential equation is transformed into an equivalent weak form, find $u \in V$ such that $a(u, v) = l(v)$, $\forall v \in V$, where l is a continuous linear functional, $a(u, v)$ is the bilinear form or sesquilinear form. It should be proved that the sesquilinear form is bounded and V -elliptic in order to use the Lax-Milgram theorem. It guarantees that there is a solution, and that it is unique. The Galerkin method is used to produce the best approximation to the solution u of the variational problem, from a given approximating subspace in a finite-dimensional space ($V_h \subset V$). The formulation of Galerkin method is as follows: find $u_h \in V_h$ such that $a(u_h, v) = l(v)$, $v \in V_h$. When the Galerkin equation is subtracted from the variational problem, the result is $a(u - u_h, v) = 0$ for $v \in V_h$. This is the orthogonality condition which is necessary and sufficient to make u_h the best approximation to the solution u . The complete procedure will be explained in detail in chapter 2.

Krylov methods are used for this purpose. Krylov subspaces ($\mathcal{K}_m(A, x) = \text{span}(x, Ax, A^2x, \dots, A^{m-1}x)$) are built up to look for a good approximation to invariant

subspaces and eigenvectors in the Krylov spaces. When these subspaces are combined with suitable preconditioning, it makes that the performance of the iterative algorithms becomes robust to compute the solution of large and sparse linear systems. The Krylov subspace methods need a procedure to generate suitable basis vectors for $\mathcal{K}_m(A, x)$. There are two approaches to do this task: Arnoldi algorithm and Lanczos algorithm. The first one generates orthonormal basis vectors for the Krylov subspace $\mathcal{K}_m(A, r_0)$, and they are stored in an upper Hessenberg matrix H_m . If H_m is singular, then the linear system is inconsistent. Therefore, the use of Arnoldi algorithm could be a problem [Freund et al., 1991]. The second algorithm, Lanczos, produces two sequences of biorthogonal vectors $\text{span}\{v_1, v_2, \dots, v_m\} = \mathcal{K}_m(A, v_1)$ and $\text{span}\{w_1, w_2, \dots, w_m\} = \mathcal{K}_m(A^T, w_1)$. The parameter $\mu_m = \frac{w_m^T A v_m}{w_m^T v_m}$ has to be computed. Unfortunately, the process will stop because of $w_m^T v_m = 0$, even though the vectors $w_m \neq 0$ and $v_m \neq 0$. Then, the Lanczos algorithm terminates prematurely before an invariant Krylov subspace can be found. This situation is called "serious breakdown" [Freund, 1992, Freund et al., 1993].

Iterative Methods are used to solve large sparse system of equations. They have been employed to solve systems of equations with Hermitian and Non-Hermitian matrices [Freund et al., 1991, Freund and Nachtigal, 1991, Freund et al., 1993]. These methods focus on different algorithms including biconjugate gradient (BICG), generalized minimal residual method (GMRES), quasi-minimal residual (QMR) method and transpose-free quasi-minimum residual (TFQMR), and they are described in several books [Barrett et al., 1994, Kelley, 1995, Greenbaum, 1997, Saad, 2003]. However, they have an interesting mixture of advantages and disadvantages in terms of convergence and breakdown. For example, the iterations of the BICG are defined by a Galerkin condition, so this algorithm is able to show an irregular convergency behaviour with the residual norm oscillating a lot. Moreover, a breakdown or near-breakdown may happen to BICG [Freund and Nachtigal, 1991]. The GMRES algorithm uses matrix H_m generated by Arnoldi algorithm and this could be a problem if the matrix is singular [Freund et al., 1991]. Finally, the QMR algorithm employs Lanczos process to generate two biorthogonal subspaces which could have some difficulties in the generations.

In general, the processes of matrix diagonalisation, in particular for complex symmetric matrices, are very difficult. For example, Craven states that "a real symmetric matrix can be diagonalised by orthogonal transformation. This is not true, in general, for a symmetric matrix of complex elements" [Craven, 1968, pg. 341]. The difficulty is that a complex symmetric matrix is diagonalisable by complex orthogonal transformation if and only if each eigenspace of the matrix has an orthonormal basis. Craven also mentions that the proof of the diagonalisation process applied to real symmetric matrices can be used for complex symmetric matrices in the construction of the orthogonal basis vectors, where a nonzero vector is able to be normalised. However, this is not always true for the complex case, since a quasi-null vector can occur. u is defined as a quasi-null vector if $\|u\|^2 = 0$ but $u \neq 0$, where u is a complex vector and the norm is defined as $\|u\| = \sqrt{u^* u}$. He also describes some general

properties of complex symmetric matrices. Horn and Johnson describe quite well a lot of properties for Hermitian matrices but not much is said about complex symmetric matrices. In particular, they are interested in the spectral theorem for Hermitian matrices which says [Horn and Johnson, 2013, page 229]:

Theorem 1.3.1. *A matrix $A \in \mathbb{C}^{n \times n}$ is Hermitian if and only if there is a unitary $U \in \mathbb{C}^{n \times n}$ and a real diagonal $\Lambda \in \mathbb{C}^{n \times n}$ such that $A = U \Lambda U^*$. Moreover, A is real and Hermitian (that is, real symmetric) if and only if a real orthogonal $P \in \mathbb{C}^{n \times n}$ and a real diagonal $\Lambda \in \mathbb{C}^{n \times n}$ such that $A = P \Lambda P^T$.*

This theorem would be quite useful, if the system of equation had an Hermitian matrix instead of a complex symmetric matrix.

As Craven [1968] mentions, the diagonalisation process for complex symmetric matrices could have difficulties. Moreover, J. H. Wilkinson in his book **The Algebraic Eigenvalues Problem**, when writing about Complex Symmetry Matrix and NonHermitian Matrix makes the following comment [Wilkinson, 1965, page 26]: "That none of the important properties of real symmetric matrix is shared by complex symmetric matrices".

One of the most fundamental theorems of Matrix Theory is the Schur decomposition which says:

Theorem 1.3.2. *Let $A \in \mathbb{C}^{n \times n}$ be a matrix with eigenvalues $\lambda_1, \lambda_2, \dots, \lambda_n$. Then there exists a unitary matrix U and upper-triangular matrix T such that $A = UTU^*$. Also, the diagonal entries of T are equal to $\lambda_1, \lambda_2, \dots, \lambda_n$.*

A matrix U with n columns that satisfies $U^*U = I_n$ is called complex orthonormal. The complex orthogonality of U in the above expressions represent the complex symmetry of A . However, a complex symmetric matrix may not be diagonalisable and the reason is that the decomposition does not exist. This is why there are complex vectors u where $u^T u = 0$ but $u \neq 0$. For example, one can have the following complex symmetric matrix:

$$A = \begin{bmatrix} 2i & 1 \\ 1 & 0 \end{bmatrix}, \quad \text{where } i = \sqrt{-1}.$$

This matrix has just one eigenvalue, it is $\lambda = i$. The algebraic multiplicity is 2 but the geometric multiplicity is 1. The Jordan form of A is:

$$U^T A U = \begin{bmatrix} i & 1 \\ 0 & i \end{bmatrix}, \quad \text{where } U = \begin{bmatrix} i & 1 \\ 1 & 0 \end{bmatrix}.$$

Thus, A is not diagonalisable. In general the resolution of systems of equations with complex symmetric matrices can be a difficult task.

1.4 Aim of thesis

The computing of the complex effective electrical properties such as permittivity and conductivity of materials has many applications in different fields. However,

they are only generated by some models or measurements in the laboratory. This project is about the development of a computational tool to calculate the properties using a 3D image where the material is represented. The physical parameters of the different compounds of the material, and the range of frequencies for the electric field where the compounds react are also used to observe the behaviour of the material as functions of complex effective electrical properties. Basically, the development is carried out in two stages: the first one is to prove that the second order elliptic partial differential equation fulfils some mathematical conditions to make sure that the equation has a unique solution. After this, artificial and real materials represented in 3D image are used to evaluate the performance of the numerical techniques in solving the equation. They are measured in terms of numerical parameters to reach the solution of the equation. The second stage is the validation of the computational tool to calculate the complex effective properties. This is carried out by using artificial materials and their analytical equations, and real materials to run experiments in a laboratory to measure the complex permittivity and complex conductivity. Moreover, numerical experiments have to be run to compute the complex effective properties. A comparison between the analytical, numerical, and experimental results is carried out in order to observe whether a correlation exists.

The key aim of the thesis is to focus on the first stage of the development of the computational tool, which includes the proof of the existence of the solutions of the second order elliptic partial differential equation with varying complex coefficients, and the numerical approach to find the best approximation to the solution. The system of equations $Ax = b$ is generated by using the 3D image, the physical parameters of the material, Finite Element methods, and by applying the Dirichlet and the Neumann boundary conditions. The resolution of the system of equations represents a very challenging task given the contrast between the complex parameter values of the different phases of the material.

Hierarchical Matrices (\mathcal{H} -Matrices) are the numerical techniques to be used due to the fact that they are not based on Krylov methods. However, under certain circumstances they can be combined with Biconjugate Gradient (BICG) and Generalised Minimal Residual (GMRES) algorithms. In the case of \mathcal{H} -Matrices, the LU factorisation is used if the following condition $\rho(LU^{-1}A) \leq \varepsilon < 1$ is fulfilled, and it is denoted by \mathcal{H} -LU. The idea is to reach the best approximation to the solution by running a few iterations of a linear method in combination with the \mathcal{H} -LU or the GMRES algorithm using the \mathcal{H} -LU decomposition as a preconditioner. The decision depends on the spectral radius, if this is close to 0, the first combination can be used as a good one. However, if the spectral radius is close to 1, then the second option is much better.

The numerical technique \mathcal{H} -Matrices was implemented in a library developed by Dr. Ronald Kriemann at Max-Planck-Institut für Mathematik in den Naturwissenschaften in Leipzig, Germany. The library is called \mathcal{H} -Lib^{PRO} and it includes routines for Richardson, BICG, and GMRES algorithms. For this thesis, a C code was written to read the 3D image and the physical parameters, with the implementation of Finite Element Methods applying the Dirichlet and Neumann boundary condi-

tions, and to generate the system of equations when an electric field at frequency ω is applied to the material. Hence, the routines of the \mathcal{H} -Lib^{pro} are used to find the approximating solution of the system.

The matrix of the system of equations which arises from the discretisation of the second order elliptic partial differential equation has the characteristic that is not diagonalisable. There are some special techniques that help to improve this feature. One of the most efficient and popular of them is the use of preconditioners or preconditioning matrix. For the preconditioning matrix M and the linear system $Ax = b$, there exists a mapping called Richardson iteration which is applied to the linear system as $W^{-1}Ax = W^{-1}b$. The \mathcal{H} -LU iteration can be used to obtain the matrix M , in this case these iteration is equivalent to Richardson iteration.

1.5 Overview

The rest of the thesis is structured as follows:

Chapter 2 provides the framework to establish the existence and uniqueness of the second order elliptic partial differential equation. There is a description of the functional analysis tools which are required to transfer the strong formulation into the variational formulation in an appropriate function space. In the variational formulation, i.e. find $u \in V$ such that $a(u, v) = l(v) \forall v \in V$, the sesquilinear form and the continuous linear functional have to fulfil some properties in order to use the Lax-Milgram theorem to demonstrate the existence and uniqueness of the solution. Moreover, the Galerkin method is used to approximate the solution of the variational problem in finite-dimensional subspace V_h of a space V , as follows: find $u_h \in V_h$ such that $a(u_h, v_h) = l(v_h) \forall v_h \in V_h$, this is called the discrete problem.

The Lax-Milgram theorem is applied to the discrete problem in order to have one and only one solution u_h . It is also important to use Céa's lemma to show that the error $\|u - u_h\|$ is reduced to a problem of approximation theory. This error which is the distance between the solution u of the original problem and the solution u_h of the discrete problem, is up to a constant independent of the space V_h , bounded above by the distance $\inf_{v_h \in V_h} \|u - v_h\|$ between the function u and the subspace V_h . This is particularly important to define the convergence of the discrete problems. The process of construction of finite-dimensional subspaces V_h of space V is carried out using Finite Element method.

Chapter 3 shows how to build the system of equations that arises from the second order elliptic partial differential equation. This starts by explaining the representation of the physical parameters in the 3D image. Then, it described how the Finite Element method, the Dirichlet and the Neumann boundary conditions are used to build up the system of equations. The structure of the stiffness matrix used by \mathcal{H} -Matrices is depicted.

The description of the linear iterative method such as Richardson is done in chapter 4. This algorithm is used in combination with LU-factorisation of \mathcal{H} -Matrices to solve the system of equations. A brief explanation of Krylov Methods is given be-

cause they are used by GMRES algorithm. This algorithm and how it works in combination with \mathcal{H} -Matrices is illustrated in this chapter.

Different aspects related to \mathcal{H} -Matrices are described in chapter 5. This starts with some basic definitions, how \mathcal{H} -Matrices are formatted and constructed. Moreover, how the operations of matrix-vector and the matrix-matrix multiplications work, and it discusses these operations in term of computational work. There is also an explanation of the transfer of the sparse matrix generated by the Finite Element method into the \mathcal{H} -Matrices format. The application of the \mathcal{H} -Matrices to solve the system of equations is carried out using the library \mathcal{H} -Lib^{pro} for which a C code was developed. The use of the \mathcal{H} -Lib^{pro} and the implementation of the C code is described. The computation of the solutions of different systems of equations generated from 3D images of distinct artificial and real samples are showed.

Chapter 6 describes how the computations to solve the second order elliptic partial differential equation are carried out to evaluate the numerical scheme developed in this research. Moreover, it shows how the accuracy which is related to the rank-r of the matrices has an influence on the computational work in terms of memory and time. The results of the solutions of the complex systems of linear equations generated by the samples are also showed in terms of convergence rate and frequency.

A summary of the conclusions of this research study and the guidelines for future investigations are provided in chapter 7.

Second Order Elliptic Partial Differential Equation

2.1 Introduction

This chapter describes the procedure to prove the existence of numerical solutions for the second order elliptic differential equation with varying complex coefficients (2.1.1). In general, there are several methods for the numerical treatment of partial differential equations. They consist in using information from a discrete set of points that satisfy approximately the differential equations. The more common methods are based on a weak formulation or a variational problem of the partial differential equation. The elliptic differential equation (2.1.1) is solved in the variational problem for which a sesquilinear form should be established. This form has to be defined on an appropriate space such as the Sobolev space in order to assure the existence of the weak solution. The assumptions for the variational problem will be shown to be valid in order to use the existence theorems of the weak solution.

The procedure starts deriving from the weak form of the differential equation (2.1.1). The second step is to use the Lax-Milgram theorem which guarantees that there is a solution and that it is unique. In order to apply this theorem, it has to be proved that the sesquilinear form is bounded and V -elliptic, and that the linear form is continuous in the space V . The next step is to generate the best approximation to the solution u of the variational problem using an approximating subspace V_h of V . The Galerkin method is used to carry it out defining a discrete problem related to the weak form. The last step is to approximate the solution of the discrete problem using the finite element method. Basically, this method takes in the subspace V_h piecewise functions as elements. These functions are chosen with small support in order to build a manageable linear system of equations. The solution of the system corresponds to the best approximate solution of the discrete problem.

In order to avoid difficulties with the notation of the equations (1.3.1), (1.3.4), and

(1.3.5) in chapter 1, they are rewritten as:

$$\nabla \cdot [Q(x_1, x_2, x_3) \nabla u(x_1, x_2, x_3)] = g(x_1, x_2, x_3) \quad \text{in } \Omega \quad (2.1.1)$$

$$u(x_1, x_2, x_3) = g_D \quad \text{on } \Gamma_D \quad (2.1.2)$$

$$\sum_{i=3}^3 \sum_{j=3}^3 Q_{ij} \frac{\partial u(x_1, x_2, x_3)}{\partial x_j} n_i = g_N := 0 \quad \text{on } \Gamma_N \quad (2.1.3)$$

where Q_{ij} represents the 3×3 matrices in the equations (1.3.4) or (1.3.5) and it is associated at the point (x_1, x_2, x_3) . Γ is the boundary of Ω which is split up into Γ_D and Γ_N with $\Gamma_D \cap \Gamma_N = \emptyset$. It is assumed that $\Omega \in \mathbb{R}^3$ is connected. The equation (2.1.3) is the conormal derivative to $\partial\Omega$ being n the unit outer normal vector.

2.1.1 Operators and Linear Functionals

Definition 2.1.1. Let X and Y be normed spaces with the norms $\|\cdot\|_X$ and $\|\cdot\|_Y$. Let a mapping $T : X \rightarrow Y$ be a linear operator. The operator T is said to be bounded, if there exists the finite operator norm

$$\|T\|_{Y \leftarrow X} := \sup_{\substack{x \in X \\ x \neq 0}} \frac{\|Tx\|_Y}{\|x\|_X}.$$

$L(X, Y)$ denotes the set of all bounded linear operators and it forms a linear space under the definition of addition and scalar multiplication operations. $L(X, Y)$ endowed with the norm $\|\cdot\|_{Y \leftarrow X}$ is defined as a normed space.

Definition 2.1.2. An antilinear functional f on a complex vector space V is an operator $f : V \rightarrow \mathbb{C}$ which satisfies the following property:

$$f(\alpha x + \beta y) = \bar{\alpha} f(x) + \bar{\beta} f(y)$$

for all $x, y \in V$, and arbitrary $\alpha, \beta \in \mathbb{C}$ where $\bar{\alpha}$ and $\bar{\beta}$ are complex conjugates.

Definition 2.1.3. A bounded linear functional f is a bounded operator $f : \|\cdot\|_X \rightarrow \mathbb{C}$. The corresponding dual norm is expressed as

$$\|f\|_{X'} := \sup_{\substack{x \in X \\ x \neq 0}} \frac{|f(x)|}{\|x\|}.$$

Definition 2.1.4. An antilinear functional f is called continuous at the point $u \in V$, if

$$\lim_{v \rightarrow u} f(v) = f(u),$$

or,

$$|f(u) - f(v)| \rightarrow 0 \quad \text{as} \quad \|u - v\| \rightarrow 0.$$

Let f be a linear functional that if f is continuous at $u = 0$ (and hence for every u) if and only if there exists a nonnegative constant C such that

$$|f(u)| \leq C\|u\| \quad \forall u \in V.$$

It is important to note that the set of all continuous linear functionals defined on a normed vector space V constitutes a normed space which is called the dual or conjugate space of V and it is denoted by $V' = L(V, \mathbb{C})$. For $x \in V$, $f' \in V'$ can be written as

$$\langle x, f' \rangle_{V \times V'} = \langle f', x \rangle_{V' \times V} = f'(x),$$

where $\langle \cdot, \cdot \rangle_{V \times V'}$, and $\langle \cdot, \cdot \rangle_{V' \times V}$ are called dual forms or duality pairings.

2.1.2 Hilbert space

Definition 2.1.5. Let V be a complex vector space. A scalar product is a map $(\cdot, \cdot): V \times V \rightarrow \mathbb{C}$ which satisfies the following conditions:

$$\begin{aligned} (x, x) &\geq 0 \quad \forall x \in X/\{0\}, \\ (\lambda x + y, z) &= \lambda(x, z) + (y, z) \quad \forall \lambda \in \mathbb{C}, \text{ and } x, y, z \in V, \\ (x, y) &= \overline{(y, x)} \quad \forall x, y \in V. \end{aligned}$$

Proposition 2.1.6. If (\cdot, \cdot) is a scalar product on a vector space V , then $\|x\| := (x, x)^{1/2}$ is a norm on V .

Let V be a vector space over \mathbb{C}^n . The scalar product and the norm are defined as follows:

$$(x, y) = \sum_{i=1}^n x_i \bar{y}_i, \quad \|x\| = \sqrt{\sum_{i=1}^n |x_i|^2},$$

where $x, y \in \mathbb{C}^n$.

Proposition 2.1.7. If (\cdot, \cdot) is a scalar product on a vector space V , then $|(x, y)| \leq \|x\| \|y\|$ $\forall x, y \in V$.

Definition 2.1.8. A complex Hilbert space V is a vector space over \mathbb{C} with a scalar product such that V is complete in the norm $\|x - y\| = \sqrt{(x - y, x - y)}$.

Definition 2.1.9. Let V be a complex Hilbert space. The mapping $a(\cdot, \cdot): V \times V \rightarrow \mathbb{C}$ is called a sesquilinear form if

$$a(\lambda_1 x + \lambda_2 y, z) = \lambda_1 a(x, z) + \lambda_2 a(y, z)$$

and

$$a(x, \lambda_1 y + \lambda_2 z) = \bar{\lambda}_1 a(x, y) + \bar{\lambda}_2 a(x, z)$$

where $\bar{\lambda}_1$ and $\bar{\lambda}_2$ denote the complex conjugate of λ_1 and λ_2 , respectively, and for $x, y, z \in V$ and $\lambda_1, \lambda_2 \in \mathbb{C}$.

Definition 2.1.10 (Continuity of sesquilinear forms). *A sesquilinear form $a(\cdot, \cdot) : V \times V \rightarrow \mathbb{C}$ is continuous (or bounded) if there exists a $C_{cs} < \infty$ such that*

$$|a(u, v)| \leq C_{cs} \|u\|_V \|v\|_V, \quad \forall u, v \in V. \quad (2.1.4)$$

Definition 2.1.11. *A sesquilinear form $a(\cdot, \cdot)$ is called V -elliptic if it is continuous in $V \times V$ and there is a constant $C_e > 0$ such that*

$$|a(u, u)| \geq C_e \|u\|_V^2 \quad \forall u \in V. \quad (2.1.5)$$

The sesquilinear form mapping $V \times V \rightarrow \mathbb{C}$ and a linear operator from $V \rightarrow V'$, where V' is the anti-linear dual space of V , are related according to the next lemma.

Lemma 2.1.12. *For every continuous sesquilinear form $a : V \times V \rightarrow \mathbb{C}$ there exists a unique $A \in L(V, V')$ such that*

$$a(u, v) = \langle Au, v \rangle_{V' \times V} \quad \forall u, v \in V. \quad (2.1.6)$$

Moreover,

$$\|A\|_{V \rightarrow V'} \leq C_{cs}, \quad (2.1.7)$$

with C_{cs} in (2.1.4).

Proof. See [Hackbusch, 2017, Sauter and Schwab, 2011].

Now, there is no distinction between the sesquilinear form $a : V \times V \rightarrow \mathbb{C}$ and the associated operator $A : V \rightarrow V'$. Then, given an operator $A \in L(V, V')$, one says that A is invertible if it is both injective and surjective, i.e., A^{-1} exists, where $A^{-1} \in L(V', V)$.

Lemma 2.1.13. *Let $A \in L(V, V')$ be the operator associated to a continuous sesquilinear form $a(\cdot, \cdot)$. Then the following statements (i), (ii), and (ii) are equivalent:*

(i) $A^{-1} \in L(V', V)$ exists;

(ii) $\epsilon, \epsilon' > 0$ exist such that

$$\inf\{\sup\{|a(x, y)| : y \in V, \|y\|_V = 1\} : x \in V, \|x\|_V = 1\} = \epsilon > 0, \quad (2.1.8a)$$

$$\inf\{\sup\{|a(x, y)| : x \in V, \|x\|_V = 1\} : y \in V, \|y\|_V = 1\} = \epsilon' > 0; \quad (2.1.8b)$$

(iii) the inequalities (2.1.8a) and (2.1.8c) hold:

$$\sup\{|a(x, y)| : x \in V, \|x\|_V = 1\} > 0, \quad 0 \neq y \in V. \quad (2.1.8c)$$

If one of the statements (i)-(iii) holds, then

$$\epsilon = \epsilon' = 1/\|A\|_{V \rightarrow V'}, \quad (\epsilon, \epsilon' \text{ from (2.1.8a, 2.1.8b)}). \quad (2.1.8d)$$

Proof. See [Hackbusch, 2017, Lemma 6.94].

Lemma 2.1.14. *V-ellipticity (2.1.5) implies (2.1.4) and (2.1.8a) and (2.1.8b) with $\epsilon = \epsilon' \geq C_e$ and thus $\|A^{-1}\|_{V \leftarrow V'} \leq 1/C_e$.*

Proof. See [Hackbusch, 2017, Lemma 6.97].

2.1.3 $L^2(\Omega)$, $H^1(\Omega)$, and $H_0^1(\Omega)$ spaces

Let Ω be a nonempty Lebesgue-measurable set in \mathbb{R}^n . $L^2(\Omega)$ denotes all Lebesgue measurable functions on Ω such that $f : \Omega \rightarrow \mathbb{C}$ satisfies $\int_{\Omega} |f|^2 dx \leq \infty$. Two functions $u(x)$ and $v(x)$ for all $x \in \Omega$ are not distinguishable from each other, if they differ only on a set of zero measure.

Theorem 2.1.15. *$L^2(\Omega)$ is a Hilbert space with the scalar product*

$$(u, v)_{0, \Omega} := (u, v)_{L^2(\Omega)} := \int_{\Omega} u(x) \overline{v(x)} dx \quad (2.1.9)$$

and the norm

$$\|u\|_{0, \Omega} := \|u\|_{L^2(\Omega)} := \sqrt{\int_{\Omega} |u(x)|^2 dx}. \quad (2.1.10)$$

Definition 2.1.16. *Let u and w be functions in the space $L^2(\Omega)$. w is called the weak derivative $\frac{\partial u}{\partial x_i}$ of u if*

$$\int_{\Omega} w \cdot v dx = - \int_{\Omega} u \frac{\partial v}{\partial x_i} \quad \forall v \in C_0^\infty(\Omega). \quad (2.1.11)$$

Definition 2.1.17. *Sobolev space of order 1 on Ω is called the space*

$$H^1(\Omega) = \left\{ v \in L^2(\Omega), \frac{\partial v}{\partial x_i} \in L^2(\Omega), 1 \leq i \leq n \right\}. \quad (2.1.12)$$

The Space $H^1(\Omega)$ is endowed with the the inner product:

$$(u, v)_{1, \Omega} = \int_{\Omega} \left(uv + \sum_{i=1}^n \frac{\partial u}{\partial x_i} \frac{\partial v}{\partial x_i} \right) dx, \quad (2.1.13)$$

and the corresponding norm:

$$\|v\|_{H^1(\Omega)} = \sqrt{(v, v)_{1, \Omega}} = \left(\int_{\Omega} |v|^2 dx + \int_{\Omega} |\nabla v|^2 dx \right)^{1/2}. \quad (2.1.14)$$

$H_0^1(\Omega)$ is a Sobolev space which is a subset of the functional space $H^1(\Omega)$. The functions of $H_0^1(\Omega)$ vanish on $\Gamma = \partial\Omega$ and this set refers to the Dirichlet boundary conditions. $H_0^1(\Omega)$ is defined as:

$$H_0^1(\Omega) = \overline{C_0^\infty(\Omega)}^{\|\cdot\|_{H^1}} \quad (2.1.15)$$

where $\|\cdot\|_{H^1}$ is the completion of $C_0^\infty(\Omega)$ with regard to the $H^1(\Omega)$ norm.

A particular subset of the space $H^1(\Omega)$ will be used:

$$H_\gamma^1(\Omega) = \left\{ v \in H^1(\Omega), v = 0 \text{ on } \gamma \subset \partial\Omega \right\}.$$

Let N be a positive integer. The vector $\alpha = (\alpha_1, \alpha_2, \dots, \alpha_N)$ where α_i are nonnegative integers with $1 \leq i \leq N$. Each component of the vector α is called a multi-index. The number $|\alpha| = \alpha_1 + \alpha_2 + \dots + \alpha_N$ is called the length of the multi-index α and

$$D^\alpha = \frac{\partial^{|\alpha|}}{\partial^{|\alpha_1|} x_1 \partial^{|\alpha_2|} x_2 \dots \partial^{|\alpha_N|} x_N}$$

is a $|\alpha|$ -fold partial derivative operator.

Let $k \in \mathbb{N}_0$. Let $H^k(\Omega)$ be the set of all functions $u \in L^2(\Omega)$ whose weak derivatives $D^\alpha u \in L^2(\Omega)$ for $|\alpha| \leq k$:

$$H^k(\Omega) = \left\{ u \in L^2(\Omega), D^\alpha u \in L^2(\Omega) \text{ for } |\alpha| \leq k \right\}.$$

Theorem 2.1.18. $H^k(\Omega)$ equipped with the following scalar product

$$(u, v) := \sum_{|\alpha| \leq k} (D^\alpha u, D^\alpha v)_{L^2(\Omega)}$$

and the norm

$$\|u\|_{H^k(\Omega)} := \sqrt{\sum_{|\alpha| \leq k} \|D^\alpha u\|_{L^2(\Omega)}^2}$$

forms a Hilbert space for all $k \in \mathbb{N}_0$.

Proof. See [Hackbusch, 2017, Theorem 6.23] and [Bhattacharyya, 2012, Proposition 2.15.1, Theorem 2.15.1].

Let γ be the trace operator defined as $\gamma : H^1(\Omega) \rightarrow L^2(\partial\Omega)$. The trace of the following Sobolev space

$$H^{1/2}(\partial\Omega) = \left\{ u \in L^2(\partial\Omega), \exists v \in H^1(\Omega), u = \gamma(v) \right\}$$

will be used later.

2.2 Abstract variational problem

The solution of a partial differential equation starts by transforming into the weak form. It is carried out by multiplying the equation by a test function and integrating it by parts. The weak form of partial differential equation is given by a complex Hilbert space V , a sesquilinear form $a(\cdot, \cdot)$ and a continuous linear functional f , as follows

$$\text{find } u \in V \text{ such that } a(u, v) = f(v) \text{ for all } v \in V. \quad (2.2.1)$$

2.2.1 Variational problem

The Sobolev space of order 1 (equation 2.1.12) denoted by $V = H^1(\Omega)$ is used to write the variational problem (2.2.1) in the appropriate way for the sesquilinear form $a : V \times V \rightarrow \mathbb{C}$, given a continuous antilinear functional $f : V \rightarrow \mathbb{C}$, and $a(u, \cdot) \in V'$, as follows:

$$\text{find } u \in V \text{ such that } a(u, v) = f(v) \text{ for all } v \in V. \quad (2.2.2)$$

According to Lemma 2.1.12, a unique linear operator $A : V \rightarrow V'$ exists such that $a(u, v) = \langle Au, v \rangle$. To show that the solution of the second order elliptic partial differential equation (2.1.1) exists, it has to be proved that $A^{-1} : V' \rightarrow V$ exists.

To solve the elliptic partial differential equation, it has to be proved that its sesquilinear form is V -elliptic. Let V be a complex vector space. $Q(x) \in \mathbb{C}^{3 \times 3}$ is a matrix value function in the domain Ω which is defined by a cube. The sesquilinear form is

$$a(u, v) := \int_{\Omega} \langle Q(x) \nabla u, \nabla v \rangle dx \quad \text{for all } u, v \in V. \quad (2.2.3)$$

Lemma 2.2.1. *Let θ be a complex number with $|\theta| = 1$. Take the equation (2.2.3) and assume*

$$\text{Re} \langle \theta Q(x) z, z \rangle \geq \lambda_0 |z|^2 \quad \text{for } z \in \mathbb{C}^3 \text{ and all } x \in \Omega,$$

where λ_0 is positive real constant. Let u be a function in $H_{\gamma}^1(\Omega)$ then,

$$\begin{aligned} |a(u, u)| &= |\theta a(u, u)| \quad \text{for } |\theta| = 1 \\ &\geq \text{Re} \theta a(u, u) \\ &= \int_{\Omega} \text{Re} \langle \theta Q(x) \nabla u, \nabla u \rangle dx \\ &\geq \lambda_0 \int_{\Omega} |u|^2 dx \\ &\geq C \lambda_0 \int_{\Omega} (|\nabla u|^2 + |u|^2) dx \quad \text{by Lemma 2.2.2, 2.2.3} \\ &= C \lambda_0 \|u\|_{H_{\gamma}^1(\Omega)}^2. \end{aligned}$$

The components of matrix $Q(x)$ are referenced as $\epsilon_{jk} - i \frac{\sigma_{jk}}{\omega \epsilon_0}$ where $0 < \epsilon_{jk}, \sigma_{jk}$ for $1 \leq j, k \leq 3$, $0 < \omega < \infty$, and $\epsilon_0 > 0$ is a constant. One takes $\theta = \frac{1+i}{\sqrt{2}}$ and $\bar{\theta} = \frac{1-i}{\sqrt{2}}$, and multiplies θ by matrix Q as follows

$$\begin{aligned} \theta Q_{jk} &= \theta \left(\epsilon_{jk} - i \frac{\sigma_{jk}}{\omega \epsilon_0} \right), \quad 1 \leq j, k \leq 3 \\ &= \theta \epsilon_{jk} - \theta i \frac{\sigma_{jk}}{\omega \epsilon_0} \\ &= \theta \epsilon_{jk} + \bar{\theta} \frac{\sigma_{jk}}{\omega \epsilon_0} \\ &= \frac{1+i}{\sqrt{2}} \epsilon_{jk} + \frac{1-i}{\sqrt{2}} \frac{\sigma_{jk}}{\omega \epsilon_0}, \end{aligned}$$

let $B \in \mathbb{R}^{3 \times 3}$ and assigning the real part of θQ_{jk} to matrix B as

$$\begin{aligned} (\epsilon_{jk}) \geq \lambda_\epsilon I, \lambda_\epsilon > 0 \\ (\sigma_{jk}) \geq \lambda_\sigma I, \lambda_\sigma > 0, \end{aligned} \quad \implies B \geq \frac{1}{\sqrt{2}} \left(\lambda_\epsilon + \frac{\lambda_\sigma}{\omega \epsilon_0} \right) I := \lambda_0 I > 0,$$

$$B = \operatorname{Re}(\theta Q_{jk}) = \frac{1}{\sqrt{2}} \left(\epsilon_{jk} + \frac{\sigma_{jk}}{\omega \epsilon_0} \right). \quad (2.2.4)$$

Lemma 2.2.2. For Ω bounded, $\|\cdot\|_{H^k(\Omega)}$ and

$$|u|_{k,0} := \left[\sum_{|\alpha|=k} \|D^\alpha u\|_{L^2(\Omega)}^2 \right]^{1/2}$$

are equivalent norms in $H_0^k(\Omega)$.

Proof. See [Hackbusch, 2017, Lemma 6.29].

Lemma 2.2.3. Let Ω be a bounded domain and $\Gamma_D \subset \partial\Omega$. $\mu(\Gamma_D) > 0$ implies $\|\nabla u\|_{L^2(\Omega)} \sim \|u\|_{H^1(\Omega)}$ for all $u|_{\Gamma_D} = 0$.

The V -ellipticity condition of the sesquilinear form (2.2.3) is proved given that matrix B is positive definite and using Lemma 2.2.1. This conclusion implies that the inf-sup conditions are fulfilled, hence A^{-1} exists.

2.2.2 Dirichlet boundary condition

The boundary value problem is formulated and analysed for the second order elliptic differential equation (2.1.1), starting with Dirichlet boundary condition. Assume that $g \in L^2(\Omega)$, let $u \in H^1(\Omega)$ and $v \in H^1(\Omega)$. A Dirichlet boundary condition should be satisfied on a part Γ_D of Γ , $u = \tilde{g}$ on Γ_D . The function \tilde{g} must be in the space $H^{1/2}(\Omega)$, since $\gamma(H^1(\Omega)) = H^{1/2}(\Gamma_D)$, where γ is the trace operator. For any $\tilde{g} \in H^{1/2}(\Gamma_D)$,

there exists a function $G \in H^1(\Omega)$ such that $\gamma G = \tilde{g}$. One introduces

$$w := u - G \in H_{\Gamma_D}^1(\Omega). \quad (2.2.5)$$

After multiplying the relation (2.1.1) by the test function v , integrating by parts over the space Ω , and substituting the equation (2.2.5), the transformed problem is:

$$\text{find } w \in H_{\Gamma_D}^1(\Omega) \text{ such that } a(w, v) = f(v) := g(v) + a(G, v) \text{ for all } v \in H_{\Gamma_D}^1(\Omega). \quad (2.2.6)$$

2.2.3 Neumann boundary condition

The solution u of the second order elliptic partial differential equation (2.1.1) when applying Neumann boundary condition (equation 2.1.3) is determined beginning by its weak formulation. As before, the equation (2.1.1) is multiplied by the test function v , integrated by parts over Ω , and choosing the Sobolev space $H^1(\Omega)$ for the trial function u and the test function v . Then, the weak formulation is

$$u \in H_{\Gamma_D}^1(\Omega), \quad \int_{\Omega} (Q(x) \nabla u \cdot \nabla v) dx = \int_{\Omega} g v dx + \int_{\Gamma_N} g_N v ds \quad \forall v \in H_{\Gamma_D}^1(\Omega),$$

and it can be rewritten as

$$\text{find } u \in H_{\Gamma_D}^1(\Omega) \text{ such that } a(u, v) = f(v) := g(v) + \int_{\Gamma_N} g_N v ds \text{ for all } v \in H_{\Gamma_D}^1(\Omega). \quad (2.2.7)$$

For the second order elliptic partial differential equation (2.1.1) with different boundary conditions such as Dirichlet and Neumann, the solution is calculated using the following weak form:

$$\text{find } u \in H_{\Gamma_D}^1(\Omega) \text{ such that } a(u, v) = f(v) \text{ for all } v \in H_{\Gamma_D}^1(\Omega), \quad (2.2.8)$$

with

$$a(u, v) = \int_{\Omega} (Q(x) \nabla u \cdot \nabla v) dx, \quad (2.2.9)$$

$$f(v) = \int_{\Omega} g v dx + \int_{\Gamma_N} g_N v dx + a(G, v), \quad (2.2.10)$$

where

$$H_{\Gamma_D}^1(\Omega) = \{v \in H^1(\Omega), v = 0 \text{ on } \Gamma_D\}.$$

2.3 Galerkin method

In general, the exact solution of the variational problem (2.2.1) is impossible to be found due to the fact that the space V is infinite dimensional. A very natural ap-

proach is to approximate the solution by defining a finite dimensional problem. An useful finite dimensional approximation scheme is the Galerkin method. Basically, it uses a sequence of finite dimensional subspaces $\{V_N\}_{N=1}^{\infty} \subset V$, with $V_N \subset V_{N+1}$ which fills the space V in the limit . The problem (2.2.1) is solved in each finite-dimensional space V_N .

The variational problem (2.2.1) is transformed into a discrete problem as follows:

$$\text{find } u_n \in V_N \text{ such that } a(u_n, v) = f(v) \quad \text{for all } v \in V_N. \quad (2.3.1)$$

The solution u_n is expressed as a linear combination of this basis function with unknown coefficients:

$$u_n = \sum_{j=1}^N y_j b_j. \quad (2.3.2)$$

where $V_N = \text{span}\{b_1, b_2, \dots, b_N\}$. Substituting the equation (2.3.2) into the discrete problem (2.3.1), one obtains

$$a\left(\sum_{j=1}^N y_j b_j, v\right) = f(v) \quad \text{for all } v \in V_N. \quad (2.3.3)$$

By linearity of the sesquilinear form $a(\cdot, \cdot)$ in its first component and substituting the basis function b_1, b_2, \dots, b_N for v in (2.3.3), it yields

$$\sum_{j=1}^N a(b_j, b_i) y_j = f(b_i), \quad i = 1, 2, \dots, N. \quad (2.3.4)$$

Let us define the matrix for the system of equations (2.3.4):

$$\mathbf{L} = \{L_{i,j}\}_{i,j=1}^N, \quad L_{i,j} := a(b_j, b_i), \quad (2.3.5)$$

the vector on the right-hand side of the system:

$$\mathbf{f} = \{f_i\}_{i=1}^N, \quad f_i := f(b_i), \quad (2.3.6)$$

and the unknown coefficient vector

$$\mathbf{y} = \{y_i\}_{i=1}^N. \quad (2.3.7)$$

The linear system of algebraic equations (2.3.4) can be rewritten in the matrix form as

$$\mathbf{L}\mathbf{y} = \mathbf{f}. \quad (2.3.8)$$

The solution of the linear system of equations (2.3.2) defines a unique solution $u_n \in V_N$ for the discrete problem (2.3.8).

The V -elliptic condition assures that for a continuous variational problem exists a unique solvability. For the discrete problem, this condition is also sufficient. It is showed in the following theorem.

Theorem 2.3.1. *Let V and V_N be an infinite-dimensional and a finite-dimensional spaces, respectively with $V_N \subset V$ and $\dim V_N = N < \infty$. Suppose the sesquilinear form is V -elliptic: $a(u, u) \leq C_E \|u\|_V^2$ for all $u \in V$ with $C_e > 0$. Then the matrix L in (2.3.8) is nonsingular and the Galerkin solution $u_n \in V_N$ satisfies*

$$\|u_n\|_V \leq \frac{1}{C_E} \|f\|_{V'_N} \leq \frac{1}{C_E} \|f\|_{V'}.$$

Proof. See [Hackbusch, 2017, Theorem 8.16 and Lemma 8.14].

Let u be the solution of the variational problem (2.2.1) and let u_n be the solution to the discrete problem (2.3.1). The error between these solutions $e_n = u - u_n$ shows an orthogonality property which is used to compute the best approximation to the solution u . This property is called orthogonality condition and it is obtained subtracting the equation (2.3.1) from the equation (2.2.1):

$$a(u - u_n, v) = 0 \quad \forall v \in V_N \subset V. \quad (2.3.9)$$

The error $\|u - u_n\|_V$ is estimated using the distance $d(u, V_N) = \inf_{u_n \in V_N} \|u - u_n\|_V$ between a function $u \in V$ and the subspace $V_N \subset V$. The Céa lemma is used to do this estimation applying the orthogonality condition.

Theorem 2.3.2 (Céa lemma). *Let V be a Hilbert space, $a(\cdot, \cdot) : V \times V \rightarrow \mathbb{C}$ is a bounded, V -elliptic sesquilinear form and $f \in V'$ is a antilinear continuous functional. Let $u \in V$ be the solution of the variational problem (2.2.1). Moreover, let $V_N \subset V$ be a subspace and $u_n \in V_N$ the solution of the discrete problem (2.3.1). Let C_{cs} and C_{ce} be the continuity and V -ellipticity constant of the form $a(\cdot, \cdot)$. Then,*

$$\|u - u_n\|_V \leq \frac{C_{cs}}{C_{ce}} \inf_{v \in V_N} \|u - v\|_V. \quad (2.3.10)$$

Proof. See [Hackbusch, 2017, Atkinson and Han, 2009].

Remark 2.3.3. *This lemma establishes that the approximation error e_n is based on the choice of the subspaces V_N and it does not depend on selection of its basis.*

An important aspect of the Galerkin method is the convergence. The approximation error is supposed to converge to zero in the way that the sequence of subspace $\{V_N\}_{N \in \mathbb{N}}$ converges to V . The following theorem shows the convergence of the Galerkin method and it is a consequence of Céa lemma.

Theorem 2.3.4. *Let V be a Hilbert space. Assume a sequence of subspaces $\{V_N\}$ of V . Let $a(\cdot, \cdot) : V \times V \rightarrow \mathbb{C}$ be a bounded and V -elliptic sesquilinear form, and $f \in V'$ is an antilinear continuous functional. Then*

$$\lim_{n \rightarrow \infty} \|u - u_n\|_V = 0,$$

if

$$\lim_{n \rightarrow +\infty} \text{dist}(v, V_N) = 0 \quad \text{for all } v \in V. \quad (2.3.11)$$

Proof. See [Hackbusch, 2017, Atkinson and Han, 2009].

Remark 2.3.5. The limit in (2.3.11) holds for $V_1 \subset V_2 \subset \dots \subset V_N \subset \dots$, and $\overline{\bigcup_{N \in \mathbb{N}} V_N} = V$.

2.4 Finite Element Method

The Finite Element method is a very popular numerical technique for solving elliptic boundary value problems. Basically, this is a Galerkin approximation scheme as was discussed in the previous section, where the elements in the finite dimensional approximating subspaces V_N are piecewise polynomial functions. The problem of estimation the finite element solution error can be reduced to one of estimating the approximating error $\|u - u_n\|_V \leq C\|u - \pi_n u\|_V$ using Céa lemma, where $\pi_n u$ is a function constructed by taking nodal values of u on the vertex of the finite element.

The success in using the Finite Element method is due to the fact that the selected basis functions in the subspace V_N have small support. This condition gives rise to a linear system of equations whose matrix is sparse. The sparsity brings a couple of important purposes: the matrix is less costly to be built and the linear system can usually be solved more efficiently. A particular family of elements will be used which are cubes defined in the domain $\Omega = (0, 1)^3$.

Definition 2.4.1. Let $\Omega \subset \mathbb{R}^3$ be a domain. The set $\tau := \{T_1, \dots, T_t\}$ is called an admissible tessellation if the following conditions are fulfilled:

$$T_p \text{ are open cubes ("finite elements")} \text{ for } 1 \leq p \leq t, \quad (2.4.1)$$

$$T_p \text{ are disjoint, i.e., } T_p \cap T_q = \emptyset \text{ for } p \neq q, \quad (2.4.2)$$

$$\bigcup_{p=1}^t \overline{T_p} = \overline{\Omega}, \quad (2.4.3)$$

for $p \neq q$ the set $\overline{T_p} \cap \overline{T_q}$ is either

i) empty, or

$$\text{ii) a common vertex, side or face of } T_p \text{ and } T_q. \quad (2.4.4)$$

Let τ be an admissible tessellation in the domain Ω . Let $\mathbf{x} \in \overline{\Omega}$ be a point which is called a node if \mathbf{x} is a vertex in a cube $T_p \in \tau$. The active nodes are defined by $\mathbf{x} \in \Omega \cup \Gamma_N$. The size of the domain is described by $NC \times NC \times NC$ cubes where NC is the number of cubes in the directions X , Y , and Z . Let h be side length of T_p . The total number of active nodes is computed by $N = (NC - 1)(NC + 1)^2$ and $h = 1/N$.

The active nodes are represented by the respective components x_i , y_j , and z_k where $x_i = ih$, $y_j = jh$, and $z_k = kh$ for $(0 \leq i, j \leq N)$ and $(1 \leq k \leq N - 1)$.

The linear functions in the X-direction are defined as

$$\ell_i(x) = \max \left\{ 0, 1 - \frac{|x - x_i|}{h} \right\}. \quad (2.4.5)$$

The corresponding expression is applied in the X- and Y-direction for the linear functions $\ell_j(y)$ and $\ell_k(z)$ with the components y_j and z_k , respectively. The basis functions at the active nodes can be constructed as a tensorized basis:

$$b_{(i,j,k)}(x, y, z) = \ell_i(x)\ell_j(y)\ell_k(z), \quad (2.4.6)$$

where ℓ_i , ℓ_j , and ℓ_k are defined in (2.4.5). The functions in (2.4.6) have the following property:

$$b_{(i,j,k)}(x_{i'}, y_{j'}, z_{k'}) = \delta_{((i,j,k),(i',j',k'))}. \quad (2.4.7)$$

The subspace of the piecewise linear functions is defined as:

$$V_h := \{u \in C^0(\bar{\Omega}) : u = 0 \text{ on } \Gamma_D; \text{ on each } T_p \in \tau, u \text{ is a trilinear function} \\ u(x, y, z) = \sum_{i=0}^1 \sum_{j=0}^1 \sum_{k=0}^1 a_{(i,j,k)}^p x^i y^j z^k \text{ on } T_p \text{ with } a_{(i,j,k)}^p \in \mathbb{C}\}. \quad (2.4.8)$$

This subspace is contained in the space $H_{\Gamma_D}^1(\Omega)$. Moreover, each function $u \in V_h$ is only determined by its active node values (x_i, y_j, z_k) .

The Finite Element method consists in writing the discrete problem (2.3.1) using the functions in the subspace V_h (2.4.8). Following the procedure to form a linear system of equations from the equation (2.3.2) to the equation (2.3.8) and using the basis functions defined in (2.4.8), a system of linear equations can be built as well. An important consequence of using the basis functions with small support is that the matrix \mathbf{L} is now sparse. This condition makes that matrix \mathbf{L} in (2.3.5) for three indices $L_{(i,j,k),(i',j',k')}$ only has nonzero components for the corresponding indices $|i - i'| \leq 1$, $|j - j'| \leq 1$, and $|k - k'| \leq 1$.

2.5 Numerical solution of the partial differential equation

The classical formulation of the second order elliptic partial differential equation (2.1.1) in the divergence form with Dirichlet (2.1.2) and Neumann (2.1.3) boundary conditions in the domain $\Omega \subset \mathbb{R}^3$ is considered as:

find $u \in H^1(\Omega)$ such that

$$\sum_{i=1}^3 \frac{\partial}{\partial x_i} \left[\sum_{j=1}^3 Q_{ij}(x_1, x_2, x_3) \frac{\partial u(x_1, x_2, x_3)}{\partial x_j} \right] = g(x_1, x_2, x_3) \quad \forall x_1, x_2, x_3 \in \Omega \quad (2.5.1)$$

where matrix Q is associated with the cube at the point (x_1, x_2, x_3) . This matrix represents the physical properties of material which are homogeneous within cube.

The computational domain Ω for the boundary values problem (2.5.1) is a unit cube. The domain has different subdomains with distinct material properties (several phases). In this case, only in the subdomain where the data is supposed to be smooth, the partial differential equation is valid as the strong form. The weak formulation for this problem is derived using the equations (2.2.8), (2.2.9), and (2.2.10) taking into account the matrix Q as:

$$\text{find } u \in H^1(\Omega) \text{ such that } a(u, v) = f(v) \quad \text{for all } v \in H_{\Gamma_D}^1(\Omega) \quad (2.5.2)$$

with

$$a(u, v) = \int_{\Omega} \langle Q(x_1, x_2, x_3) \nabla u(x_1, x_2, x_3), \nabla \overline{v(x_1, x_2, x_3)} \rangle dx_1 dx_2 dx_3, \quad (2.5.3)$$

$$f(v) = \int_{\Omega} g(x_1, x_2, x_3) \overline{v(x_1, x_2, x_3)} dx_1 dx_2 dx_3 + \int_{\Gamma_N} g_N \overline{v} d\Gamma \quad (2.5.4)$$

where $x = (x_1, x_2, x_3)$ and

$$H_{\Gamma_D}^1(\Omega) = \{v \in H^1(\Omega), v = 0 \text{ on } \Gamma_D\}.$$

Galerkin method is used to look for the approximation solution u_n of the weak formulation (2.5.2). The finite dimensional subspace V_h is defined by

$$V_h = \text{span}\{b_{(i,j,k)}; (i, j, k) \in \Omega \subset \mathbb{R}^3\} = \left\{ \sum_{(i,j,k) \in \Omega} v_{(i,j,k)} b_{(i,j,k)} : v_{(i,j,k)} \in \mathbb{C} \right\}.$$

Let v_h^* be vector defined by

$$v_h^* = \sum_{(i,j,k) \in \gamma_h} u_{(i,j,k)}^* b_{(i,j,k)} \quad (2.5.5)$$

where γ_h represents the points $(i, j, k) \in \partial\Gamma_D$ and $u_{(i,j,k)}^*$ is given. A subspace for the test functions is determined by

$$V_{oh} = \left\{ \sum_{(i,j,k) \in \Omega \setminus \partial\Gamma_D} v_{(i,j,k)} b_{(i,j,k)} : v_{(i,j,k)} \in \mathbb{C} \right\}. \quad (2.5.6)$$

The vector v_h^* and the space V_{oh} together form an affine space. The affine space V_{gh} is formed using the subspaces in (2.5.5) and (2.5.6), i.e., $V_{gh} = v_h^* + V_{oh}$. The weak formulation (2.5.2) is transformed into the discrete problem as

$$\text{find } u_n \in V_{gh} \text{ such that } a(u_n, v_n) = f(v_n) \quad \text{for all } v_n \in V_{oh}, \quad (2.5.7)$$

where the approximate solution is

$$u_n = \sum_{(i,j,k) \in \gamma_h} u_{(i,j,k)}^* b_{(i,j,k)} + \sum_{(i,j,k) \in \Omega \setminus \partial\Gamma_D} v_{(i,j,k)} b_{(i,j,k)}. \quad (2.5.8)$$

To construct the linear system of equation in order to find the approximate solution for the discrete problem (2.5.5), one uses the same process applied to the equations (2.3.2), (2.3.3), and (2.3.4) as follows:

$$\sum_{i=1}^N \sum_{j=1}^N \sum_{k=1}^N a \left(b_{(i,j,k)}, b_{(m,q,p)} \right) y_{(i,j,k)} = f \left(b_{(m,q,p)} \right) \quad (m, q, p) \in \Omega_h \setminus \partial\Gamma_D \quad (2.5.9)$$

where $b_{(i,j,k)}$ and $b_{(m,q,p)}$ are the basis functions corresponding to the points (i, j, k) and (m, q, p) , respectively. The best approximation solution of the discrete problem is computed by solving the linear system of equations (2.5.9), where the stiffness matrix (2.3.5) and the load vector (2.3.6) can be written as

$$L_{(m,q,p),(i,j,k)} = a \left(b_{(i,j,k)}, b_{(m,q,p)} \right) \quad (2.5.10)$$

and

$$f_{(m,q,p)} = f \left(b_{(m,q,p)} \right).$$

The details about how to build the system will be explained in chapter 3. The computation of the solution of the system will be described in chapter 4, 5, and 6.

Construction of the Complex Linear System of Equations

3.1 Introduction

The numerical solution of the second order elliptic partial differential equation (2.1.1)-(2.1.3) is based on the solution of the complex linear system of equations (2.5.9). This chapter is focused on the construction of the complex linear systems, beginning with the description of the model problem for the partial differential equation. The following section is about the building of the linear systems that will be described using the Finite Element method, Dirichlet (2.1.2) and Neumann (2.1.3) boundary conditions, and the coefficient matrix in (1.3.4).

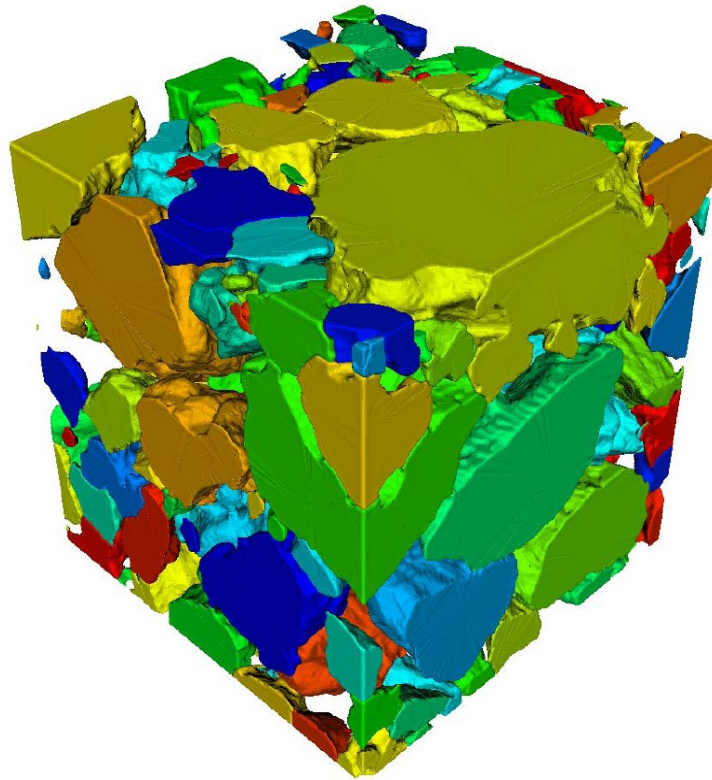
The last section is related to the details of the 3D image data that is used to build the complex linear systems. The importance of using 3D image is to capture the characteristics of the porous material that are represented in the image. This section describes the different materials, the physical properties of the materials, the distinct range of frequencies to apply an electric field on the materials, and the general scheme of the process to build and solve the complex linear system generated at different frequencies.

3.2 Model problem

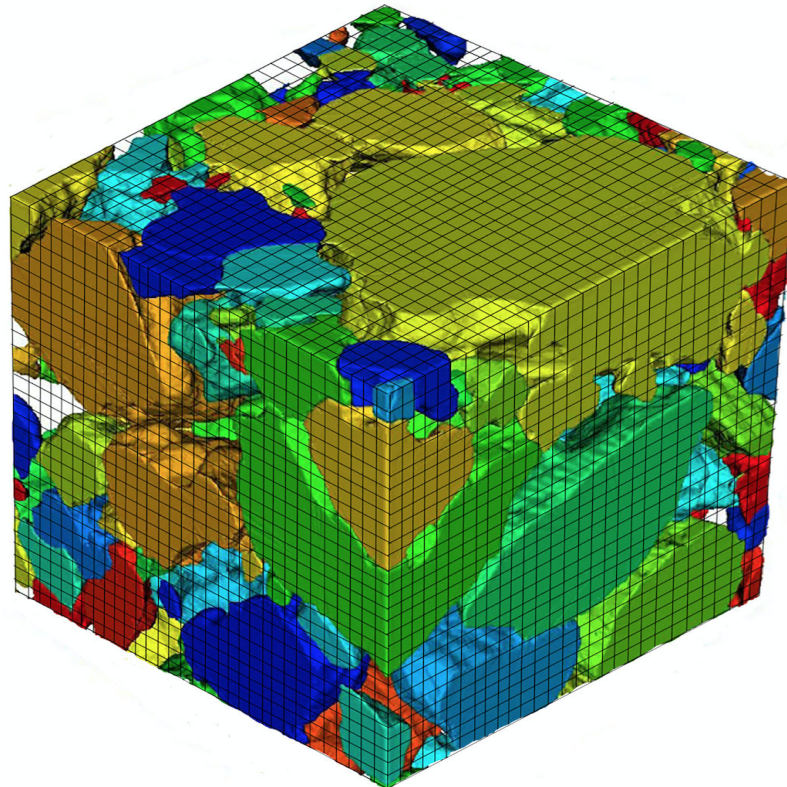
Let us start by defining the domain for the boundary value problem (2.5.1). Figure 3.2.1a shows a porous material represented in a 3D image where the different colours correspond to distinct phases. The discretisation of the 3D image is a cube with a set of grid points as can be seen in Figure 3.2.1b. The domain is chosen as a unit cube

$$\Omega_h = (0,1) \times (0,1) \times (0,1), \quad (3.2.1)$$

where h is the step size of the grid points in the domain which envelops the 3D image. Figure 3.2.1c shows the points in the expression (2.5.7) as red circle in the top and bottom of the grid. The points in the subspace V_{oh} (equation (2.5.8)) as a black point.



(a)



(b)

Figure 3.2.1: (a) A porous material. (b) Discretisation of the porous material (*cont.*)

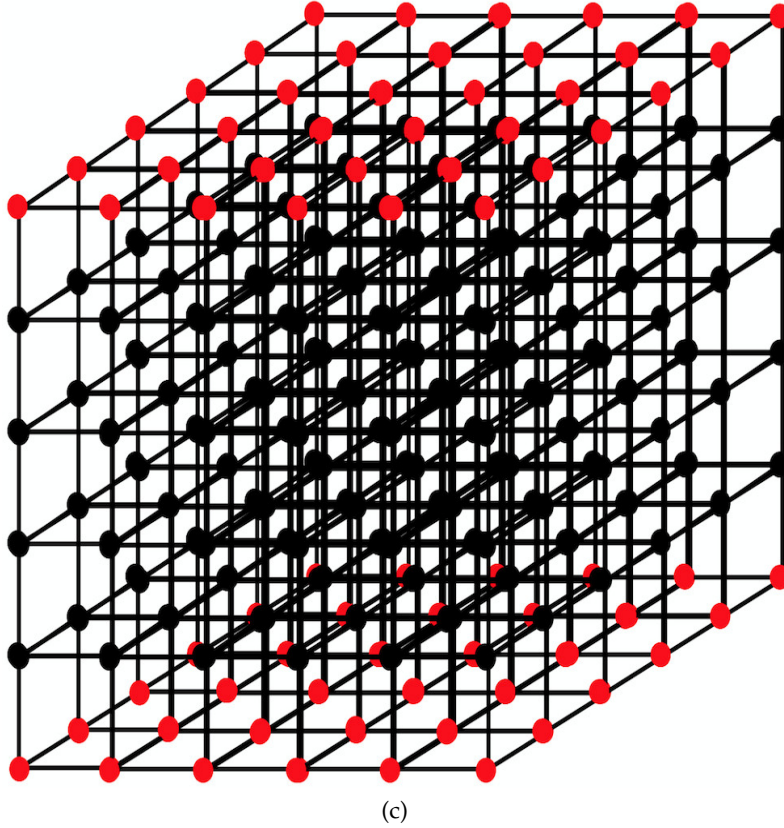


Figure 3.2.1: (c). The grid points of the discretisation where the active points are represented by the black circles and the boundary points are illustrated by the red circles.

In order to define the representation of the grid points in terms of matrices and vectors, one begins with an index set denoted by I where $I = \{0, \dots, N\} \times \{0, \dots, N\} \times \{1, \dots, N-1\}$. The elements of the set I are expressed by Greek letters, i.e., $\alpha \in I$. For example, the indices $\alpha \in I$ for active points will be triples $\alpha = (i, j, k)$.

A vector is defined in the complex vector space as $b \in \mathbb{C}^I$ which is a mapping $b : I \rightarrow \mathbb{C}$ into the field of complex numbers. The vector component value at $\alpha \in I$ is described by b_α . A vector can be written in the form

$$b = (b_\alpha)_{\alpha \in I}.$$

The complex square matrices are a mapping of the index set $I \times I \rightarrow \mathbb{C}$. These matrices are denoted by $\mathbb{C}^{I \times I}$ and represented by upper case letters. The component of the matrix A corresponding to the index pair $(\alpha, \beta) \in I \times I$ is written as $a_{\alpha\beta}$ or $a_{\alpha,\beta}$. It can sometimes be expressed as $A_{\alpha\beta}$. Matrix A with its components $a_{\alpha\beta}$ is denoted by

$$A = (a_{\alpha\beta})_{\alpha,\beta \in I}.$$

3.3 Application of the Finite Element Method

The complex linear system of equations is constructed using the Finite Element method and the boundary conditions. Let v and μ be vectors with $v = (i, j, k) \in I$ and $\mu = (i', j', k') \in I$, where I is the index set of active points. Let L be a sparse matrix with $L \in \mathbb{C}^{I \times I}$. This matrix is defined by

$$L_{v\mu} = a(b_\mu, b_v) = \int_{\Omega} \langle Q \nabla b_\mu, \nabla b_v \rangle dx_1 dx_2 dx_3 = \int_{C_{\alpha, \beta, \gamma}} \langle Q \nabla b_\mu, \nabla b_v \rangle dx_1 dx_2 dx_3, \quad (3.3.1)$$

where $C_{\alpha, \beta, \gamma}$ represents the support of the integrand. The interval for the component i' of the active point μ is denoted by J_α where α is defined as

$$\alpha = \begin{cases} \{i + 1/2\} & \text{for } i' = i + 1 \\ \{i + 1/2, i - 1/2\} & \text{for } i' = i \\ \{i - 1/2\} & \text{for } i' = i - 1, \quad \alpha, \beta \text{ correspondingly.} \end{cases} \quad (3.3.2)$$

Considering h as the size of the interval, J_i is expressed by

$$J_i = [ih - h/2, ih + h/2] \quad (3.3.3)$$

Similarly, the definition in (3.3.2) is applied for the indices β and γ where j and j' correspond to the first index, while k and k' are associated to the second one. The interval in (3.3.3) for the indices β and γ are represented by J_j and J_k , respectively. Then the intervals in the cube are written as a product and the expansion of the integral is as follows:

$$\begin{aligned} \int_{C_{\alpha, \beta, \gamma}} \langle Q \nabla b_\mu, \nabla b_v \rangle dx_1 dx_2 dx_3 &= \int_{J_\alpha} \int_{J_\beta} \int_{J_\gamma} \sum_{p, q=1}^3 Q_{pq} \frac{\partial b_\mu}{\partial x_q} \frac{\partial b_v}{\partial x_p} dx_1 dx_2 dx_3 \\ &= Q_{11} \int_{\alpha h - 1/2}^{\alpha h + 1/2} \ell'_{i'}(x_1) \ell'_i(x_1) dx_1 \int_{\beta h - 1/2}^{\beta h + 1/2} \ell_{j'}(x_2) \ell_j(x_2) dx_2 \int_{\gamma h - 1/2}^{\gamma h + 1/2} \ell_{k'}(x_3) \ell_k(x_3) dx_3 \\ &+ Q_{12} \int_{\alpha h - 1/2}^{\alpha h + 1/2} \ell'_{i'}(x_1) \ell_i(x_1) dx_1 \int_{\beta h - 1/2}^{\beta h + 1/2} \ell_{j'}(x_2) \ell'_j(x_2) dx_2 \int_{\gamma h - 1/2}^{\gamma h + 1/2} \ell_{k'}(x_3) \ell_k(x_3) dx_3 \\ &+ \dots \end{aligned}$$

The rest of terms comes from the symmetric and cross derivatives.

Let $u \in \mathbb{C}^I$ be the vector solution. The vector $b \in \mathbb{C}^I$ represents the right hand side of the system. The components of the vector v_N^* in the expression (2.5.5) are

contained in vector b . Then, the complex linear system is defined as

$$Lu = b. \quad (3.3.4)$$

3.4 Description of the 3D image data

3D images are used in several fields of science. They have been employed to study the properties of materials. The use of 3D images has allowed the structure of materials to be captured in an image which makes it an excellent tool to characterise materials. In principal, there are three steps for the study using images: take the image of the material, process the image in order to improve its quality, and simulate the physical properties of the material.

The methodology for digital rock uses modern techniques to acquire the image, for example the X-ray micro-computed tomography and the helical micro-computed tomography setup. The main idea is to image the 3D geometry of the mineral phases, and the pore-space of the rock, and then simulate computationally the physical processes in the digital image. The simulating physical processes determine the effective material properties such as elastic, transport, and electrical properties. These processes involve the use of robust numerical techniques due to the fact that the material structures can be complex. For the computing of the complex effective electrical properties, the simulation process is a challenging work for the numerical techniques. This is the reason why this thesis is only focused on solving the complex linear systems of equations generated from any material represented in a 3D image with the adequate boundary conditions. The types of materials used in the research are rocks.

The domain Ω_h is determined by a 3D image which is formed by voxels or cubes in each direction X , Y , and Z . The voxel is the three-dimensional discrete unit of the image. A natural way to define the finite elements of the domain is by the voxels.

Each voxel or cube has one piece of material, i.e., the material is homogeneous within the voxel. The physical properties of the material in the voxel are assigned to matrix Q . In the complex linear system of equations (3.3.4) where matrix L is defined in (3.3.1), matrix Q represents the matrix in the equation (1.3.4). The conductivity is expressed in Siemens/meter= $1/(\text{Ohm.meter})$, the permittivity of the free space in Faraday/meter= $\text{second}/(\text{Ohm.meter})$, and frequency in 1/second. The physical units of the complex term in the equation (1.3.4) are cancelled out and as the dielectric constant of materials (real term) do not have unit, then matrix Q has no unit. This matrix is multiplied by function u which represents the potentials, hence the complex linear system is expressed in terms of potentials.

In order to compute electric field ($\mathbf{E} = -\nabla\phi$), and in turn the electrical current density (1.2.7) and the electrical displacement field (1.2.1) within the cube, the potentials are needed. Essentially, the solution of the complex linear system of equations provides the potential values in the whole image. It makes this stage very critical.

The sample set has six types of different materials, three of them are artificial samples and the other three are rocks. The first artificial sample is a cube in the

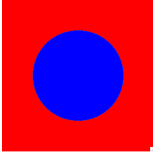
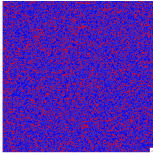
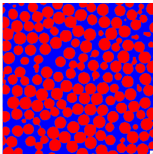
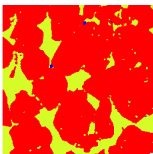
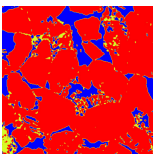
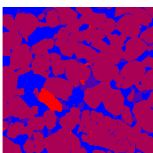
Type of Material	Image	Phase	Range of Frequency	Number of Frequency
Sphere		<ul style="list-style-type: none"> • Host • Inclusion 	$10^7 \leq \omega \leq 10^{14}$	36
Random		<ul style="list-style-type: none"> • Host • Inclusion 	$10^{-2} \leq \omega \leq 10^4$	30
Sphere Crystal		<ul style="list-style-type: none"> • Quartz • Brine 	$10^4 \leq \omega \leq 10^{11}$	34
Bentheimer		<ul style="list-style-type: none"> • Crude • Brine • Grain 	$10^4 \leq \omega \leq 10^{11}$	34
Berea		<ul style="list-style-type: none"> • Air • Clay • Grain 	$10^2 \leq \omega \leq 10^{10}$	29
Heterogeneous rock		<ul style="list-style-type: none"> • Air • Clay • Grain • Pyrite 	$10^7 \leq \omega \leq 10^{11}$	36

Table 3.1: The description of the sample set used to generate the complex linear systems of equations within a range of frequency

3D image with a sphere at its centre. The phase for the cube and the sphere are called the host and the inclusion, respectively. The values of conductivities, dielectric constants, and the range of frequency were taken from a paper published by Wu et al. [2007]. The second artificial sample is also a cube as a host and the inclusions are voxels at random positions. The physical parameters are found in a work developed by Tuner et al. [2001]. These two samples were generated using a computer code. The third artificial sample is a package of spheres whose 3D image was taken using a X-ray micro-computed tomography. The phases for this sample are quartz and brine. Telford et al. [1990] shows values of resistivity (ρ) for quartz as 4×10^{10} Ohm.meter $< \rho < 2 \times 10^{14}$ Ohm.meter . The equiva-

lence values were computed using the relation $\sigma = 1/\rho$ and this is the conductivity range: 5×10^{-15} Siemens/meter $< \sigma < 2.5 \times 10^{-11}$ Siemens/meter. It was chosen 1.25×10^{-11} S/m as the conductivity value. The dielectric constant is between 4.2 and 5. It was taken the value of 3.73. The conductivity of brine is 4 Siemens/meter [Gueguen and Palciauskas, 1994] and the dielectric constant is 80 [Gueguen and Palciauskas, 1994, Schön, 2004].

Table 3.1 shows the name and a picture of the artificial materials whose colours represent the phases, the range of frequency at where the AC potential is applied on the top and bottom of the image as boundary conditions, and the number of frequency. The values of the conductivities and dielectric constants of the phases of the first three samples are described in table 3.2. The idea of using artificial samples is not to start working with a complex medium. It is important to observe how the \mathcal{H} -Matrices behave solving the complex linear system of equations which are generated from these samples.

Phase	Conductivity (s/m)	Dielectric Constant
Sphere host	10^{-3}	1
Sphere inclusion	10^{-1}	4
Random host	10^{-12}	2
Random inclusion	10^{-10}	10
Quartz	1.25×10^{-11}	3.73
Brine	4	80
Crude	10^{-8}	2.2
Grain	10^{-12}	3.73
Air	0.539×10^{-14}	1.0006
Clay	0.3	5
Pyrite	0.75×10^{-2}	57.35

Table 3.2: The electrical property values of the materials used to construct the complex linear systems of equations.

Three porous materials are also used in this study. They are Bentheimer and Berea sandstones with three phases each of them, and a heterogeneous rock with four phases. Bentheimer sandstone has crude, brine, and grain as phases. The conductivity of crude is in the range between 10^{-9} S/m and 10^{-7} S/m [Lees, 2005]. The selected value is 10^{-8} S/m. The dielectric constant is 2.2 [Gueguen and Palciauskas, 1994, Schön, 2004]. The characterisation of grains depends on different parameters which are out of the scope of this research. This makes it difficult to establish the electrical conductivity. However, they are insulators, a conductivity value from the range of insulators was chosen and it is 10^{-12} S/m. The same reasons of conductivity also apply for the dielectric constant. One assumes that the value for the grain is equal to the dielectric constant of quartz.

There are three phase in Berea sandstone. They are air, clay, and grain. The value of conductivity of air is 0.539×10^{-14} S/m which is the conductivity average of the range between 0.295×10^{-14} S/m and 0.783×10^{-14} S/m [Pawer et al., 2009]. The

dielectric constant of air is 1.0006 [Schön, 2004]. For clay, the conductivity and dielectric constant are 0.3 S/m [Gueguen and Palciauskas, 1994] and 5 [Schön, 2004], respectively. The components of heterogeneous rock are air, clay, grain, and pyrite. Telford et al. [1990] shows a range for the conductivity of pyrite which is 2.9×10^{-5} S/m $< \sigma < 1.5$ S/m. The average value was taken which is 0.75×10^{-2} S/m. According to Rosenholtz and Smith [1936], the dielectric constant of pyrite is between 33.7 and 81. The selected value is 57.35 which corresponds to the average. The permittivity of the free space is approximately 8.85418×10^{-12} F/m [Zhang et al., 1999]. Table 3.2 shows the physical parameters used for the construction of the complex linear systems of equations.

Figure 3.4.1 illustrates the general scheme to generate complex linear systems based on the list of materials in table 3.1 and using the phases of materials where their physical properties are in table 3.2. The scheme starts reading the 3D image file, the physical parameters, and the range of frequencies. The second step is to set the initial frequency. The following step uses the Finite Element method and the Dirichlet boundary condition is applied to the top and bottom of the image. Then, the system of equation associated to the frequency ω is built. As the matrix is sparse, two different format are used to store the matrix: that Compress Row Storage (CRS) format and a matrix with six components, i.e., $L[i, j, k](\alpha, \beta, \gamma)$. The indices i, j, k correspond to the nodes in the 3D image and they are stored as the rows of the matrix, while the subindices α, β, γ are the neighbours in the image of the nodes in the rows which are represented by the columns. For the \mathcal{H} -Matrices, the \mathcal{H} -Lib^{Pro} library uses the CRS format to transform the sparse matrix into a \mathcal{H} -Matrix. After building the system of equations, \mathcal{H} -Matrices are used to solve the system and write its solution. Then, the frequency is increased and the process is repeated until the maximum frequency is reached.

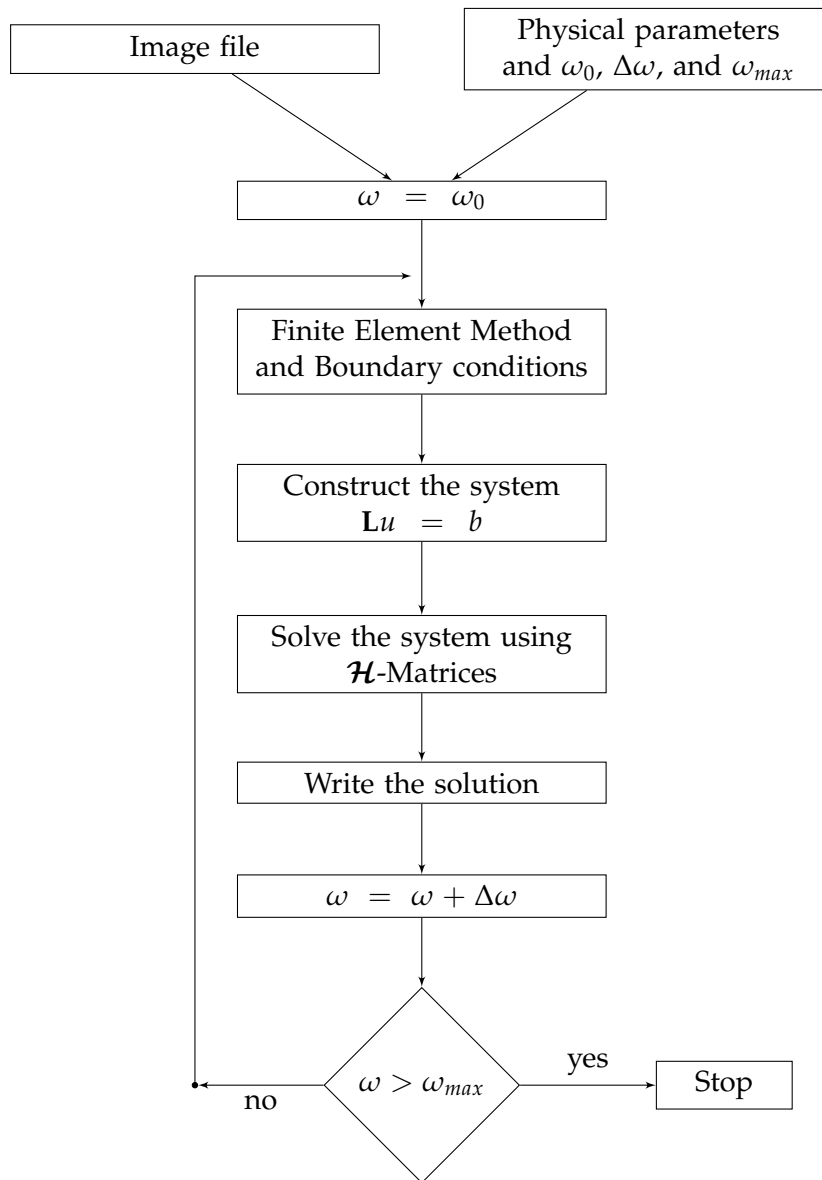


Figure 3.4.1: The construction process of the complex linear system for each frequency within a range using a 3D image, the physical properties of the material, Finite Element method and the boundary conditions. The complex linear systems are solved by using \mathcal{H} -Matrices.

Iterative Methods

4.1 Introduction

This chapter is devoted to classical linear iterative methods and semi-iterative methods that can be used in combination with \mathcal{H} -Matrices. In particular, Richardson iteration, its convergence, and the computational work for sparse matrices are described. For the semi-iteration case, these methods iterate on an affine space which is called Krylov space. There is a brief description of this space and how it is generated. The chapter also covers a short explanation of GMRES algorithm.

4.2 Iterative Methods

The iterations u^1, u^2, \dots starting from the initial value u^0 are generated by an iterative method which can be based on matrix L and vector b which are define in (3.3.4). The notation is given as

$$u^{m+1} := \Phi(u^m, b, L) \quad \text{for } m \geq 0. \quad (4.2.1)$$

Definition 4.2.1. *An iterative method is a mapping*

$$\Phi : \mathbb{C}^I \times \mathbb{C}^I \times \mathbb{C}^{I \times I} \rightarrow \mathbb{C}^I.$$

For the starting point $u^0 = y \in \mathbb{C}^I$, the sequence of iteration produced by (4.2.1) is

$$\begin{aligned} u^0(y, b, L) &:= y, \\ u^{m+1}(y, b, L) &:= \Phi(u^m(y, b, L), b, L) \quad \text{for } m \geq 0. \end{aligned} \quad (4.2.2)$$

For the convergent or divergent sequence of iteration u^m , it says that an iterative method $\Phi(\cdot, \cdot, L)$ is well defined

Definition 4.2.2. (a) *The domain of the iterative method Φ is $\mathfrak{D}(\Phi) := \{L : \Phi(\cdot, \cdot, L) \text{ well defined}\}$.*

(b) *An iteration $\Phi(\cdot, \cdot, L)$ which is totally based on the data $L \in \mathfrak{D}(\Phi)$ is called algebraic.*

Definition 4.2.3. Let b be a complex vector and $L \in \mathfrak{D}(\Phi)$, $x^* = x^*(b, L)$ is called a fixed point of the iteration corresponding to b and L , if $x^* = \Phi(x^*, b, L)$.

Definition 4.2.4. The iterative method Φ is called consistent to the system of equation (3.3.4), if for all $b \in \mathbb{C}^I$ on the right-hand side, any solution of the system is a fixed point of $\Phi(\cdot, b, L)$.

For all b and u in \mathbb{C}^I and all complex matrices $L \in \mathfrak{D}(\Phi)$, consistency means the following implication: $Lu = b \Rightarrow u = \Phi(u, b, L)$ holds.

Definition 4.2.5. Let L be a fixed complex matrix in $\mathfrak{D}(\Phi)$. An iterative method is $\Phi(\cdot, \cdot, L)$ is called convergent if for all complex vectors b , there exists a limit $x^*(b, L)$ of the iterates (4.2.2) which does not depend on the starting value $u^0 = y \in \mathbb{C}^I$.

It is important to mention that the consistency property of an iterative method is related to all matrices $L \in \mathfrak{D}(\Phi)$, and the convergence property is associated to a certain matrix $L \in \mathfrak{D}(\Phi)$.

4.3 Linear Iterative Methods

As the linear system in (3.3.4) is formed by complex linear equations, the iterative methods which are linear in u and b can be used to solve this sort of system.

Definition 4.3.1. An iterative method is called linear if $\Phi(x, b)$ is linear in (x, b) . This can be expressed as

$$\Phi(x, b, L) = M[L]x + N[L]b \quad (4.3.1)$$

where $M = M[L]$ and $N = N[L]$ are two complex matrices. $M[L]$ is called the iteration matrix of $\Phi(\cdot, \cdot, L)$.

Note that if the matrices M and N in (4.3.1) are explicit functions of L , then $\Phi(\cdot, \cdot, L)$ is called algebraic where $L \in \mathfrak{D}(\Phi)$.

Linear iterations can be represented in their second form using the consistency theorem described by Hackbusch [2016]. The form is as follows

$$x^{m+1} := x^m - N[L](Lx^m - b) \quad \text{for } m > 0 \quad (4.3.2)$$

where $N[L] = N$ is called the matrix of the second form of Φ . From the algorithmic point of view the equation (4.3.2) can be expressed by

$$W\delta = Lx^m - b \quad \text{with } \delta = x^m - x^{m+1}, \quad (4.3.3)$$

where $W = N^{-1}$. The correction term $Lx^m - b$ is called the defect of x^m . This term multiplied by N plus x^m updates x^{m+1} . The defect of the m -th iterate x^m is denoted by $d^m = Lx^m - b$.

Remark 4.3.2. All linear and consistent iterations are represented by the second order form (4.3.2) with some $N \in \mathbb{C}^{I \times I}$.

The set of all linear and consistent iterations is denoted by

$$\mathcal{L} := \{\Phi : \mathbb{C}^I \times \mathbb{C}^I \times \mathbb{C}^{I \times I} \rightarrow \mathbb{C}^I \text{ consistent linear iteration}\}. \quad (4.3.4)$$

Let M be the iteration matrix. A necessary and sufficient convergence criterion for matrix M can be based on its spectral radius.

Definition 4.3.3. *The spectral radius of a matrix M is the largest absolute value of the matrix eigenvalues:*

$$\rho(M) := \max\{|\lambda| : \lambda \in \sigma(M)\}. \quad (4.3.5)$$

Theorem 4.3.4. *A linear iteration (4.3.1) with the iteration matrix $M[L]$ is convergent if and only if*

$$\rho(M) < 1. \quad (4.3.6)$$

$\rho(M)$ is called the convergence rate of the iteration $\Phi(\cdot, \cdot, L)$.

Proof. See [Hackbusch, 2016, Theorem 2.16]. $\rho(M)$ can be also called convergence speed and iteration speed.

Theorem 4.3.5. *The spectral radius for all matrix $M \in \mathbb{C}^{I \times I}$ and any matrix norm is the following limit:*

$$\rho(M) = \lim_{m \rightarrow \infty} \|M^m\|^{1/m}.$$

Proof. See [Hackbusch, 2016, Theorem B.27].

Corollary 4.3.6. (a) *If the iteration method (4.3.1) is convergent, the iterates converge to $(I - M)^{-1}Nb$.*

(b) *If the iteration is convergent, the matrix L and $N[L]$ are regular.*

(c) *If the iteration is also consistent, then iterates u^m converge to the unique solution $u = L^{-1}b$.*

Proof. See [Hackbusch, 2016, Corollary 2.17].

Let x be the solution of the system $Lx = b$. For the iterates x^m , the iteration error of x^m is

$$e^m := x^m - x. \quad (4.3.7)$$

For the iteration x^m , the defect is $d^m = Lx^m - b$. Substituting the equation (4.3.7) in the system equation, and doing some operations, it is found a relation between the iteration error and the defect which is:

$$Le^m = d^m.$$

As the solution is not known, it is not possible to use the error iteration to stop the iterative process.

Theorem 4.3.7. Let $\|\cdot\|$ be a corresponding matrix norm. A sufficient condition to estimate the convergence of an iteration is

$$\|M\| < 1 \quad (4.3.8)$$

for the iteration matrix. If the iteration is consistent, the estimated error holds:

$$\|e^{m+1}\| \leq \|M\| \|e^m\|, \quad \|e^m\| \leq \|M\|^m \|e^0\|. \quad (4.3.9)$$

Proof. See [Hackbusch, 2016, Theorem 2.19].

$\|M\|$ is called contraction number of the iteration. An iteration is named monotonically convergent with respect the norm $\|\cdot\|$, since $\|e^{m+1}\| < \|e^m\|$. The term "convergence" and "monotone convergence" are indistinguishable, if the norm fulfils that $\rho(M) = \|M\|$.

Using theorem 4.3.7 and the condition (4.3.9) for each $\varepsilon > 0$ there is some m_0 such that $m \geq m_0$ implies that $\rho(M) \leq \|M^m\|^{1/m} \leq \rho(M) + \varepsilon$ and $\|e^m\| \leq \|M\|^m \|e^0\| = (\rho(M) + \varepsilon)^m \|e^0\|$.

Remark 4.3.8. A suitable measure for evaluating (asymptotically) the convergence speed is $\rho(M)$. The proof is based on Theorem 4.3.5, for each $\varepsilon > 0$ there is some m_0 such that $M \leq m_0$ implies that $\rho(M) \leq \|M^m\|^{1/m} \leq \rho(M) + \varepsilon$ and $\|e^m\| \leq (\rho(M) + \varepsilon)^m \|e^0\|$.

From the equation (4.3.2) it can be seen that matrix $N[L]$ transforms the defect d^m into the correction of $x^m - x^{m+1}$. Therefore, one may consider $N[L]$ as an approximate inverse of L , i.e., $N[L] \approx L^{-1}$. In order to approximate x^m to the solution x , the iteration error with a factor $\varepsilon < 1$ fulfils the following relation

$$\|e^m\|_2 \lesssim \varepsilon. \quad (4.3.10)$$

Assume $x^0 = 0$. Then $\|e^0\| = \|x\|$ and $\|d^0\| = \|b\|$ hold. As the quantities $\|e^m\|_2$ and $\|d^m\|_2$ are in different scale, one normalises the iteration error with respect to the vector in (4.3.10) as follows

$$\frac{\|e^m\|_2}{\|e^0\|_2} = \frac{\|e^m\|_2}{\|x\|_2} \lesssim \varepsilon, \quad (4.3.11)$$

and the defect is normalised w.r.t the vector b as

$$\frac{\|d^m\|_2}{\|d^0\|_2} = \frac{\|d^m\|_2}{\|b\|_2} \leq \varepsilon. \quad (4.3.12)$$

When x^m tends to the solution x , the normalised iteration error declines. On the other hand, when the normalised defect decrease, then $N \approx L^{-1}$. The relations (4.3.11) and (4.3.12) show how the iterative process approximates to the solution. However, since the solution is unknown for our application, the defect is used for the stopping criterion.

4.3.1 Storage, computation work and efficacy

The finite element discretisation in chapter 3 generates sparse matrices. The number of nonzero entries $s(I)$ per row is bounded by $n = \#I$, where I is the index set of active nodes. The number is defined by

$$s(I) := \max_{v \in I} \#\{\mu \in I : L_{v\mu} \neq 0\}.$$

According to this property, the storage cost is $\mathcal{O}(n)$. The matrix-vector multiplication and most of the operations which are carried out by one step of an iterative method also have a computational cost of $\mathcal{O}(n)$.

The algorithmic interpretation of the second normal form (4.3.3) requires computation for any iteration. The defect has to be computed in each iteration, it means that the matrix-vector multiplication is carried out. Considering L as a complex sparse matrix where n is equal to the matrix size and $s(n)$ represents the number of nonzero components of L , one obtains the following expression:

$$s(n) \leq C_L \tag{4.3.13}$$

where C_L is a constant with respect to n , but may depend on matrix L . In our particular case, the finite element discretisation generates a matrix L where each row has 27 components, it makes $C_L = 27$. Assuming (Chapter 3, page 37, $L_{v\mu}$) the matrix-vector product can be performed in $2C_L n$ operations.

The second computation in (4.3.3) is to solve the system $W\delta = d$ for which it is required to consume only $\mathcal{O}(n)$ operations. The constant in $\mathcal{O}(n)$ is related to C_L in (4.3.13) so as to obtain a formulation which expresses the number of arithmetic operations per iteration step of the iterative method Φ as follows:

$$Work(\Phi, L) \leq C_\Phi C_L n, \tag{4.3.14}$$

where $Work(\Phi, L)$ is the amount of work carried out by the Φ iteration applied to $Lu = b$. C_Φ only depends on the iteration Φ but not on L , and it is called the cost factor of the iteration. $C_L n$ describes the sparsity degree of L .

The last computation may arise for the iterative methods which require data before they start the iterates. This cost is denoted by $Init(\Phi, L)$. The effective cost at the m -th iteration is:

$$Work(\Phi, L) + Init(\Phi, L) / m. \tag{4.3.15}$$

The performance of two iteration Φ and Ψ for the some amount of work can be measured in terms of the speed or the convergence rate. The measurement could be the amount of work that has to be done to reduce the error by a fix factor. $1/e$ is chosen as the factor given that it involved the natural logarithm. The remark 4.3.8 shows that the convergence rate $\rho(M)$ is a useful parameter to describe the error reduction per iteration step. The asymptotic error reduction after m iteration steps is $\rho(M)^m$. The condition $\rho(M)^m \leq 1/e$ has to be fulfilled for which $m \geq -1/\log(\rho(M))$ is chosen. It makes that the convergence holds: $\rho(M) < 1 \Leftrightarrow \log(\rho(M)) < 0$. Hence,

the (asymptotic) number of the iteration steps for an error reduction by the factor $1/e$ is defined as:

$$It(\Phi) := -1/\log(\rho(M)). \quad (4.3.16)$$

The corresponding amount of work for the error reduction by $1/e$ is given by the multiplication of $It(\Phi)$ by equation (4.3.14): $It(\Phi) \text{Work}(I, L) \leq It(\Phi)C_\Phi C_L n$. The effective amount of work is a quantity that measures the amount of work by $1/e$ in the unit $C_L n$ arithmetic operations and it is given by

$$Eff(\Phi) := It(\Phi)C_\Phi = -C_\Phi/\log(\rho(M)). \quad (4.3.17)$$

4.4 Richardson Iteration

The selection of suitable parameters for the numerical methods is not an easy task. They depend on the spectral properties of matrices or they will make the computations more expensive. If there are less parameters to be tuned using the methods, it contributes to have less difficulties in handling the computational problems. This is one of the reasons why the linear iterative methods are used by the more modern computing techniques. A useful linear iterative method comes from the second norm form (4.3.2) taking $N = I$ where I is the identity matrix. The resulting linear iteration is called Richardson iteration and it is expressed as follows:

$$x^{m+1} = x^m - \Theta(Lx^m - b) \quad \text{with } \Theta \in \mathbb{C}. \quad (4.4.1)$$

This iteration is denoted by Φ_Θ^{Rich} .

Proposition 4.4.1. (a) $\Phi_\Theta^{Rich} \in \mathcal{L}$ is algebraic and the domain $\mathfrak{D}(\Phi_\Theta^{Rich})$ includes all sets of complex matrices.

(b) The iteration matrix of the Richardson iteration is

$$M_\Theta^{Rich} := I - \Theta L.$$

(c) The Richardson method is independent of the ordering of indices.

Proof. See [Hackbusch, 2016, Proposition 3.5].

The computation work for the Richardson iteration is expressed as:

$$\text{Work}(\Phi_\Theta^{Rich}, L) \leq (2C_L + 2)n.$$

This expression is used to evaluate $x^{m+1} = x^m - \Theta(Lx^m - b)$ using the defect $d := Lx^m - b$ and $x^{m+1} = x^m - \Theta d$.

4.5 Krylov Methods

A brief description of semi-iteration with polynomials should be done before introducing the Krylov spaces. In this case the basic iteration is the complete sequence

$$U_m := (u^0, u^1, \dots, u^m) \in (\mathbb{C}^I)^{m+1}. \quad (4.5.1)$$

The main idea is that using the sequence U_m one can find a better result than one produced by u_m .

Definition 4.5.1. A semi-iterative method is a mapping

$$\Sigma : \bigcup_{m=1}^{\infty} (\mathbb{C}^I)^{m+1} \rightarrow \mathbb{C}^{m+1}.$$

A new sequence $y^m := \Sigma(U_m)$ with $m = 0, 1, 2, \dots$ produces a semi-iterative sequence. It can be proved that in many cases the sequence $\{y^m\}$ converges faster than $\{u^m\}$ [Hackbusch, 2016].

A semi-iterative method Σ is said to be consistent if the following expression holds for all solution of $Lu = b$:

$$u = \underbrace{\Sigma(u, u, \dots, u)}_{m+1 \text{ arguments}} \quad \text{for } m = 0, 1, 2, \dots$$

Definition 4.5.2. All sequence of the semi-iteration $\{y^m\}$ generated from an arbitrary starting iterates $y^0 = u^0$ and has the asymptotic convergence rate ρ , if this is the smallest number in the following expression:

$$\overline{\lim}_{m \rightarrow \infty} (\|y^m - u\| / \|y^0 - u\|)^{1/m} \leq \rho,$$

where $u = L^{-1}b$.

Σ is called a linear semi-iteration if $y^m = \Sigma(U_m)$ is a linear combination

$$y^m = \sum_{j=0}^m \zeta_{mj} u^j$$

where coefficients $\zeta_{mj} \in \mathbb{C}$ with $m \in \mathbb{N}_0$ and $i \leq j < m$. It can be observed that the linear semi-iterative method is consistent if and only if

$$\sum_{j=0}^m \zeta_{mj} = 1 \quad \text{for all } m = 0, 1, 2, \dots$$

where the initial condition $y^0 = u^0$ is satisfied by Σ .

The relation between the linear semi-iteration Σ and a family of polynomials or a sequence of polynomials is established by Theorem 8.4 in [Hackbusch, 2016, chapter 8, p. 177]. A consequence of this theorem is that a linear semi-iterative method is uniquely described by the family of associated polynomials $p_m(x) := \sum_{j=0}^m \zeta_{mj} x^j$.

Definition 4.5.3. Let X be a complex matrix with $X \in \mathbb{C}^{I \times I}$ and let $v \in \mathbb{C}^I$ be a vector. The

Krylov space associated with matrix X and vector v is defined as

$$\mathcal{K}_m(X, v) := \text{span}\{v, Xv, X^2v, \dots, X^{m-1}v\} \quad \text{for } m \in \mathbb{N},$$

where $\mathcal{K}_0(X, v) := \{0\}$.

GMRES method is characterised by $u^m \in u^0 + \mathcal{K}_m(L, r^0)$ which the minimal residual $r^m = b - Lu^m$:

$$\|r^m\|_2 = \min \left\{ \|b - Lu^m\|_2 : u^m \in \mathcal{K}_m(L, r^0) \right\}. \quad (4.5.2)$$

4.5.1 Arnoldi algorithm

The Arnold procedure is used to yield an orthonormal basis of Krylov subspace. Let $\{v^1, \dots, v^m\}$ be any basis of $\mathcal{K}_m(L, r^0)$. The orthogonal condition $u^m \in u^0 + \mathcal{K}_m(L, r^0)$ and $r^m \perp \mathcal{K}_m(L, r^0)$ for Krylov space establishes the minimisation of the iterates u^m and its residual r^m . The generation of a suitable basis can be carried out ordering the vectors v^k such that $\mathcal{K}_m(L, r^0) = \text{span}\{v^1, \dots, v^m\}$ where $m \leq \text{deg}_L(r^0)$, i.e., $\mathcal{K}_{m+1}(L, r^0) = \text{span}\{\mathcal{K}_m(L, r^0), v^{m+1}\}$. The basis should be orthonormal in order to have stability in generating the vectors. In terms of computational work, the generation has to be as small as possible.

The Arnoldi method proceeds as follows: given the Krylov subspace $\mathcal{K}_m(L, r^0)$ at the m -th iteration, a vector v^{m+1} is computed such that $\mathcal{K}_{m+1}(L, r^0) = \text{span}\{v^1, \dots, v^{m+1}\}$. The generation of v^{m+1} is produced taking the vector $v^m \in \mathcal{K}_m(L, r^0) \setminus \mathcal{K}_{m-1}(L, r^0)$ to calculate the product Lv^m which forms vector v^{m+1} by the following normalisation process:

$$w^{m+1} := Lv^m - \sum_{i=1}^m h_{im} v^i \quad (4.5.3)$$

where

$$h_{im} = \langle Lv^m, v^i \rangle = (v^i)^* Lv^m, \quad (4.5.4)$$

and the asterisk denotes a conjugate transpose. The new basis vector is defined as

$$v^{m+1} = w^{m+1} / \|w^{m+1}\|_2.$$

The vector v^{m+1} is orthogonal to $\{v^1, \dots, v^m\}$ if and only if the coefficients h_{im} $1 \leq i \leq m$ are defined as in (4.5.4). If $Lv^m \notin \text{span}\{v^1, \dots, v^m\}$, then $w^{m+1} \neq 0$ and the generation continues. On the other hand, if $Lv^m \in \text{span}\{v^1, \dots, v^m\}$, the coefficients in (4.5.4) forces $w^{m+1} = 0$, and in this case the process ends. The complete algorithm is:

From the expression (4.5.3), the following equation can be obtained:

$$LV^m = \sum_{i=1}^{m+1} h_{im} v^i \quad \text{for } i = 1, \dots, m \quad (4.5.5)$$

Algorithm 4.5.1 Arnoldi method

```

 $w^0 := r^0; \quad h_{0,-1} := \|r^0\|_2; \quad m := 0;$ 
while  $h_{m,m-1} \neq 0$  do
   $v^m := w^m / h_{m,m-1};$ 
  for  $i := 1$  do  $m$ 
     $h_{im} := \langle Lv^m, v^i \rangle;$ 
  end for
   $w^{m+1} := Lv^m - \sum_{i=1}^m h_{im} v^i;$ 
   $h_{m+1,m} := \|w^{m+1}\|_2;$ 
   $m := m + 1;$ 
end while

```

where the vectors generated are collected in a matrix as

$$V_m = [v^1, v^2, \dots, v^m] \in \mathbf{C}^{I \times m}. \quad (4.5.6)$$

The coefficients h_{im} in (4.5.5) where $h_{ik} := 0$ for $i > k + 1$, form two Hessenberg matrices:

$$H_m = (h_{ik})_{1 \leq i \leq m} \in \mathbf{C}^{m \times m}$$

$$\tilde{H}_{m+1} = (h_{ik})_{1 \leq i \leq m+1, 1 \leq k \leq m} \in \mathbf{C}^{(m+1) \times m}.$$

Using the equation (4.5.5), one can derive the expression:

$$V_{m+1}^H LV^m = \tilde{H}_{m+1},$$

where V_{m+1}^H is the Hermitian transport matrix in the iteration $m + 1$.

Considering the vector sequence in (4.5.6) with $u^m \in u^0 + \mathcal{K}_m(L, r^0)$, the iterate u^m can be expressed as $u^m = V_m z^m$ where $z^m \in \mathbf{C}^m$ is a vector to be determined. The residual is defined as $r^m = r^0 - LV_m z^m$ where r^0 is chosen as $r^0 = \|r^0\|_2 v_1 = \|r^0\|_2 V_{m+1} e^1$ and where e^1 is the first unit column vector. Since matrix V_{m+1} is orthogonal, $V_{m+1} V_{m+1}^H$ is the orthogonal projection onto $\mathcal{K}_m(L, r^0)$. Hence, $\text{range}(LV_m) \subset \mathcal{K}_{m+1}(L, r^0)$ implies

$$LV_m = (V_{m+1} V_{m+1}^H)(LV_m) = V_{m+1} \tilde{H}_{m+1}.$$

Then, the residual can be expressed as

$$r^m = V_{m+1} \left[\|r^0\|_2 e^1 + \tilde{H}_{m+1} z^m \right]. \quad (4.5.7)$$

4.6 Generalised Minimal Residual Method

The use of Krylov methods and GMRES algorithm was proposed by Saad and Schultz [1986] to be used for general matrices. The algorithm determines over all possible

vectors in the affine space $u^0 + \mathcal{K}_m(L, r^0)$ to minimise the residual in (4.5.7). The orthogonal condition of V_{m+1} and the Euclidean norm are used to compute the minimisation by the following equation:

$$\|r^m\|_2 = \|(\|r^0\|_2 \mathbf{e}^1 + \tilde{H}_{m+1} z^m)\|_2,$$

where $z^m \in \mathbb{C}^m$. This is a least square problem in which the QR factorisation can be applied as $\tilde{H}_{m+1} = Q_{m+1} R_{m+1}$ and $R_{m+1} z^m = -\|r^0\|_2 Q_{m+1}^H \mathbf{e}^1$ has to be solved. The QR factorisation is usually carried out with Given rotations given that as \tilde{H}_{m+1} is a Hessenberg matrix. The factorisation is rather low-cost. The m -th GMRES step costs $\mathcal{O}(mn)$. The total amount of the operations that produces m steps is $\mathcal{O}(m^2n)$. The storage cost is $\mathcal{O}(mn)$.

The complex systems of linear equations were solved using two types of evaluations. The first test was the use of the Richardson iteration in combination with the \mathcal{H} -Matrices. The second one was utilising the GMRES algorithm together with the \mathcal{H} -Matrices.

Hierarchical Matrices

5.1 Introduction

The solution of the complex linear system of equations (3.3.4) consists in finding the inverse matrix of the sparse matrix which is usually dense. The dense matrices are expensive in terms of storage and operations. There have been researchers working on developing techniques to deal with these sort of matrices. Panel clustering methods [Hackbusch, 1990, Hackbusch and Novak, 1989] are used to approximate kernel functions using Taylor expansion so that some sub-matrices can be approximated by low rank matrices. In 1998 W. Hackbusch extended the idea which is based on the panel clustering methods. He introduced a new format of matrices which has a hierarchical tree structure [Hackbusch, 1999]. This class of matrices is called Hierarchical matrices or \mathcal{H} -matrices by Hackbusch.

There are two basic considerations for the \mathcal{H} -matrices. The first one is related to an efficient treatment of integral operators that can be done using separable kernel functions. The second observation is that the kernel functions in the form of an integral operator are used to represent the inverse of an elliptic partial operator. The approximate approach is based on computing the inverse of the finite element discretisation of the operator. For application purposes, the sparse discretisation of the operator is used instead of the kernel functions.

In general the \mathcal{H} -matrix technique approximates certain blocks of a given matrix by low rank matrices. The block decomposition for building \mathcal{H} -matrices and their arithmetic operations are described in the first three sections. LU decomposition and \mathcal{H} -LU decomposition for sparse matrices are covered in the next section. \mathcal{H} -LU decomposition with a moderate accuracy is enough for building a fast iteration which defines as the \mathcal{H} -LU iteration. This iteration method is described in the following section. The last section describes the scheme of how the complex systems are solved. The results of the solution of the complex linear systems generated by the six samples using \mathcal{H} -matrices are described in chapter 6.

5.2 \mathcal{H} -Matrices Construction

For the building of \mathcal{H} -matrices it is very convenient to use a tree structure for searching the appropriate block decomposition. The index set I was defined for the basis functions in (2.4.6). The vector blocks in \mathbb{C}^I are provided from the cluster trees $T(I)$. The block cluster tree $T(I \times I)$ produces all sizes of different matrix blocks.

5.2.1 Cluster Trees

The different partitions of the index set I into disjoint subsets are described by the tree $T(I)$. The set of these partitions is called cluster trees. The elements of the partitions are named vertices. The clusters are denoted by τ or σ . τ is a leaf, if τ cannot be decomposed. The set of leaves is represented by $\mathcal{L}(T(I))$. On the other hand, if τ is decomposed into subsets τ_1, \dots, τ_s , they are called sons of τ . The set of sons is denoted by

$$S(\tau) = \{\tau_1, \dots, \tau_s\}.$$

The following conditions are required:

$$I = \text{root}(T(I)), \quad \tau \neq \emptyset \quad \text{for all } \tau \in T(I),$$

$$\#S(\tau) > 1 \text{ and } \bigcup_{\sigma \in S(\tau)} \sigma = \tau \text{ disjoint union for all } \tau \in T(I)/\mathcal{L}(T(I)).$$

A conclusion is that $\tau \subset I$ holds for all $\tau \in T(I)$.

A recommendation for practical purpose is to avoid blocks too small. One fixes some $n_{min} \in \mathbb{N}$ and requires

$$\#\tau \leq n_{min} \quad \text{if and only if } \tau \in \mathcal{L}(T(I)).$$

Each index $\nu \in I$ associated with the basis functions b_ν in the subspace V_{gh} , the support of these functions is denoted by

$$\Omega_\nu := \text{supp}(b_\nu) \quad \nu \in I.$$

The generalisation of Ω_ν for the cluster $\tau \in T(I)$ is given by

$$\Omega_\tau := \bigcup_{\nu \in \tau} \Omega_\nu, \quad \tau \in T(I). \quad (5.2.1)$$

Dealing with the supports to construct a cluster tree is very difficult. It is more convenient to use the nodal points. Each index $\nu \in I$ is connected with a nodal point $\xi_\nu \in \mathbb{R}^3$. For the construction, it is considered a correspondence between a subset $\tau \subset I$ and a set of nodal points as

$$X_\tau := \{\xi_\nu : \nu \in \tau\} \subset \mathbb{R}^3. \quad (5.2.2)$$

One considers 3-dimensional cuboids as

$$Q = [a_1, b_1] \times [a_2, b_2] \times [a_3, b_3],$$

that contains X_τ . The bounding box of X_τ is the smallest cuboid Q where $X_\tau \subset Q$ and it is denoted by

$$Q_{min}(X_\tau).$$

Let $\xi_{v,\mu}$ denote the components of $\xi_v \in X_\tau$ with $1 \leq \mu \leq 3$. Then, $Q_{min}(X_\tau) = [a_1, b_1] \times [a_2, b_2] \times [a_3, b_3]$ holds with $a_\mu := \min_{v \in \tau} \{\xi_{v,\mu}\}$ and $b_\mu := \max_{v \in \tau} \{\xi_{v,\mu}\}$.

A binary cluster tree can be built by using the following steps:

- (a) Let $\tau := I$ be the root to which $Q_I := Q_{min}(X_I)$ is associated.
- (b) The box Q_τ is split into two boxes Q_τ^1 and Q_τ^2 generating two subsets of clusters represented by $\tau_k := \{i \in \tau; \xi_i \in Q_\tau^k\}$ for $k = 1, 2$. The clusters τ_k are the sons of τ and are used for the decomposition at step (b). The subboxes Q_τ^k have to satisfy $X_{\tau_k} \subset Q_{\tau_k} \subset Q_\tau^k$.
- (c) The decomposition process is carried out until the condition $\#\tau \leq n_{min}$ is fulfilled.

For a regular geometric partition which is the case using a 3D image, the Regular Geometric Bisection method is used to determine Q_τ^k and τ_k . Let Q_τ be the box with $Q_\tau = [a_1, b_1] \times [a_2, b_2] \times [a_3, b_3]$. Let j be an index that corresponds to the largest side length $b_j - a_j$ and the set $m_j := \frac{a_j - b_j}{2}$. The division of the box Q_τ is as follows:

$$\begin{aligned} Q_\tau^1 &:= [a_1, b_1] \times \cdots \times [a_j, m_j] \times \cdots \times [a_3, b_3], \\ Q_\tau^2 &:= [a_1, b_1] \times \cdots \times (m_j, b_j) \times \cdots \times [a_3, b_3] \end{aligned}$$

and the set $Q_{\tau_k} := Q_\tau^k$ for $k = 1, 2$. After the l steps of the decomposition process, all boxes are similar and their volumes are defined by $Vol(Q_\tau) := 2^{-l}(Q_I)$. For the case where X_τ is completely contained in Q_τ^1 , then $\tau_1 := \tau$ and $\tau_2 := \emptyset$ hold. For this situation, τ_2 is not considered as a son and τ has only one son $\tau_1 = \tau$.

5.2.2 Block Cluster Tree

The rows and columns of matrices in $\mathbb{C}^{I \times I}$ correspond to the product index set $I \times I$. The construction of the block decomposition of $I \times I$ is the tree $T(I \times I)$ which is called block cluster tree. This is obtained from the product of cluster tree $T(I)$.

Definition 5.2.1 (Block cluster tree). *Let $T(I)$ be a cluster tree with the index set I . A block cluster tree is built as follows*

- (a) $I \times I$ is the root.
- (b) The recursion starts with the block $b = \tau \times \sigma$ for $\tau = I$ and $\sigma = I$.

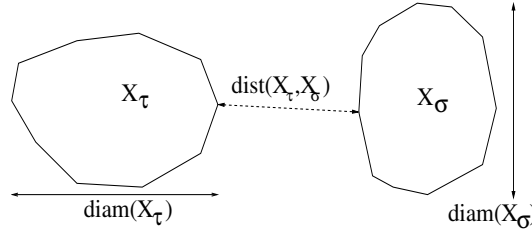


Figure 5.2.1: Supports X_τ and X_σ . Image after Hackbusch [2015].

(1b) The set of sons of $b = \tau \times \sigma$ is defined by

$$S(b) := \begin{cases} \emptyset & \text{if } S_{T(I)}(\tau) = \emptyset \text{ or } S_{T(I)}(\sigma) = \emptyset, \\ \{\tau' \times \sigma' : \tau' \in S_{T(I)}(\tau), \sigma' \in S_{T(I)}(\sigma)\} & \text{otherwise.} \end{cases}$$

(2b) Apply recursively (1,2b) to all sons of (b) if $S(b) \neq \emptyset$.

Let $T(I \times I)$ be a block cluster tree. Let P be a set of blocks $b \subset I \times I$. P is called the block partition or the partition of $I \times I$ if

$$P \subset T(I \times I), \text{ for all } b, b' \in P \text{ with } b \neq b' \Rightarrow b \cap b' = \emptyset, \\ \bigcup_{b \in P} b = I \times I.$$

5.2.3 Admissible Blocks

In order to construct an optimal partition, the admissibility condition should be satisfied. This is used to define a balance between the storage requirement of an \mathcal{H} -matrix and its approximation. This is also utilised to determine the blocks that can be approximated by low rank matrices.

The set of nodal points in (5.2.2) is redefined as

$$X_\tau := \bigcup_{v \in \tau} X_v \subset \mathbb{R}^3. \quad (5.2.3)$$

The diameter of the cluster and the distance between two clusters can be defined by

$$\text{diam}(\tau) := \max\{\|x' - x''\| : x', x'' \in X_\tau\}, \quad \tau \in I, \quad (5.2.4a)$$

$$\text{dist}(\tau, \sigma) := \min\{\|x - y\| : x \in X_\tau, y \in X_\sigma\}, \quad \tau, \sigma \subset I. \quad (5.2.4b)$$

Definition 5.2.2 (η -admissibility of a block). Let τ and σ be clusters with $\tau, \sigma \subset I$ which are associated with the support X_τ and X_σ . Then the block $b = \tau \times \sigma$ is called η -admissible if

$$\min\{\text{diam}(\tau), \text{diam}(\sigma)\} \leq \eta \text{dist}(\tau, \sigma), \quad (5.2.5)$$

where $\eta > 0$.

For a partition P not all its blocks b are admissible. For example, in a diagonal block $b = \tau \times \tau$ even though $\text{diam}(\tau)$ is positive and $\eta > 0$, as the $\text{dist}(\tau, \tau) = 0$, then the condition (5.2.5) can not be held. For these sort of blocks, it is not expected to find low rank approximation. However, the full representation of the matrix block $M|_b$ is used, and it is required for the blocks $b = \tau \times \sigma$ to fulfil $\min\{\#\tau, \#\sigma\} \leq \eta_{\min}$. This condition implies that $b = \mathcal{L}(T(I \times I))$. The requirement of the blocks is combined in the following definition:

$$\text{adm}^*(b) := \left\{ \begin{array}{ll} \text{true} & \text{if (5.2.5) holds or if } b \in \mathcal{L}(T(I \times I)) \\ \text{false} & \text{otherwise.} \end{array} \right\}. \quad (5.2.6)$$

Let P be a partition with $P \subset T(I \times I)$, it is called admissible if $\text{adm}^*(b)$ holds for all $b \in P$.

The minimal admissible partition $P \subset T(I \times I)$ is the coarsest partition such as the $\text{adm}^*(b)$ holds for all $b \in P$. For the construction of the partition P an equivalent formulation $P = \mathcal{L}(T')$ with $T' = T(I \times I; P)$ is used. T' and P are constructed as follows:

$$T' := \emptyset; \quad P := \emptyset; \quad \text{MinAdmPart}(T', P, I \times I),$$

where the recursion is defined in the following procedure:

Procedure $\text{MinAdmPart}(T', P, I \times I)$; (5.2.7)
begin $T' := T' \cup \{b\}$;
 if $\text{adm}^*(b)$ **then** $P := P \cup \{b\}$ **else for all** $b' \in S(b)$ **do** $\text{MinAdmPart}(T', P, b')$
end;

Definition 5.2.3 (near and far field). *Let $P \subset T(I \times I)$ be an admissible partition. Then P^- and P^+ are the 'near-field' and the 'far-field', respectively. They are defined by*

$$\begin{aligned} P^- &:= \{b \in P : b \in \mathcal{L}(T(I \times I))\}, \\ P^+ &:= P \setminus P^-. \end{aligned} \quad (5.2.8)$$

5.3 Low-Rank Matrices

The representation of submatrices can be done using rank- r matrices. They are the bases to build the blocks of \mathcal{H} -Matrices. Let I be the index set. If $M \in \mathbb{C}^{I \times I}$ satisfies $\text{rank}(M) \leq r$, there are two matrices in the full-matrix format called factors, A and B such that

$$M = AB^H \quad \text{with} \quad \begin{cases} A \in \mathbb{C}^{I \times \{1, \dots, r\}}, \\ B \in \mathbb{C}^{I \times \{1, \dots, r\}}, \\ r \in \mathbb{N}_0. \end{cases} \quad (5.3.1)$$

For $1 \leq l \leq r$, let $a^{(l)}$ and $b^{(l)}$ be complex vectors which represent the r columns of the matrices A and B , respectively. Then, the M in (5.3.1) can be written as

$$M = \sum_{l=1}^r a^{(l)} b^{(l)H}. \quad (5.3.2)$$

This expression is equivalent to (5.3.1). The number r in (5.3.2) is called the representation rank, even if $\text{rank}(M) < r$.

The matrix representation (5.3.1) is ensured if M is a rank- r matrix. Then, the factors in (5.3.1) exist and they are given explicitly. For $M \in \mathbb{C}^{I \times I}$, let $P(M) := \{(A, B) \in \mathbb{C}^{I \times r} \times \mathbb{C}^{I \times r} : M = AB^H\}$ be the set of all pair (A, B) that represents M and it is denoted by $\mathcal{R}(r, I)$ or \mathcal{R}_r . In contrast, the notation for matrices in the full format is $\mathcal{F}(I \times I) := \{M \in \mathbb{C}^{I \times I} : M \text{ stored in the full format}\}$. The full format means that M is described in the standard way by its entries, i.e., M_{ij} with $i, j \in I$. The set of full matrix blocks for $b = \tau \times \sigma$ with $\tau, \sigma \subset I$ is denoted by $\mathcal{F}(b)$.

Remark 5.3.1. The storage sizes for the matrices $M \in \mathcal{F} \cap \mathbb{C}^{I \times I}$ and $M \in \mathcal{R}_r \cap \mathbb{C}^{I \times I}$ are $(\#I)^2$ and $2r\#I$, respectively.

Remark 5.3.2. Assume $M \in \mathcal{R}(r, I)$ and $\tau, \sigma \subset I$. Then $M|_{\tau \times \sigma} \in \mathcal{R}(r, \tau, \sigma)$ holds for all submatrices of M . The restriction $M \mapsto M|_{\tau \times \sigma}$ does not require any arithmetical cost.

The arithmetic operations of matrices from \mathcal{R}_r are cheaper than those which involve fully populated matrices. Let $r > 0$ and $x \in \mathbb{C}^I$. The matrix $M \in \mathcal{R}_r \cap \mathbb{C}^{I \times I}$ and the factors A and B^H in (5.3.1). The matrix-vector multiplication is expressed as

$$Mx = A(B^H x), \quad (5.3.3)$$

where the first product $B^H x$ costs $r(2\#I - 1)$ operations and the second one costs $\#I(2r - 1)$ operations. The total operations is $N_{MV} = 4r\#I - \#I - r$.

Let $M' \in \mathcal{R}_{r'} \cap \mathbb{C}^{I \times I}$ and $M'' \in \mathcal{R}_{r''} \cap \mathbb{C}^{I \times I}$ be matrices given by the form in (5.3.1) with $r', r'' > 0$. The matrix-matrix addition is represented by

$$M = M' + M'' = AB^H \quad \text{with} \quad \begin{cases} A := [A' A''] \in \mathbb{C}^{I \times \{1, \dots, r' + r''\}}, \\ B := [A' B''] \in \mathbb{C}^{I \times \{1, \dots, r' + r''\}}, \end{cases} \quad (5.3.4)$$

which means that $M \in \mathcal{R}(r' + r'', I)$ where the matrix $[A' A'']$ is the agglomeration of A' and A'' . According to definition 1.9 in [Hackbusch, 2015, page 13], the addition does not require arithmetical operations but the representation rank or the storage cost increases.

The matrix-matrix multiplication of two matrices $M' \in \mathcal{R}_{r'} \cap \mathbb{C}^{I \times I}$ and $M'' \in \mathcal{R}_{r''} \cap \mathbb{C}^{I \times I}$ with $r', r'' > 0$ is given by

$$M' = A' B'^H, \quad M'' = A'' B''^H \quad \text{with} \quad \begin{cases} A' \in \mathbb{C}^{I \times \{1, \dots, r'\}}, & B' \in \mathbb{C}^{I \times \{1, \dots, r'\}}, \\ A'' \in \mathbb{C}^{I \times \{1, \dots, r''\}}, & B'' \in \mathbb{C}^{I \times \{1, \dots, r''\}}. \end{cases}$$

The product $M := M' \cdot M'' = AB^H$ has two possible representations:

- (1) For $M \in \mathcal{R}_{r'}$ with $A := A'$ and $B := B'' A''^H B'$, the computational cost is $N_{MM} = 42r'r''\#I - r'\#I - r'r''$.
- (2) For $M \in \mathcal{R}_{r''}$ with $A := A' B'^H A''$ and $B := B''^H$, the cost is $N_{MM} = 4r'r''\#I - r''\#I - r'r''$.

For the application of \mathcal{R} -Matrices, there is a particular problem that has to be faced. This is the approximation of a rank- s matrix $M \in \mathcal{R}(s, I)$ by a rank- r matrix $M' \in \mathcal{R}(r, I)$ for $s > r$. The approximation is carried out using the following mapping:

$$\mathcal{T}_{r \leftarrow s}^{\mathcal{R}} : \mathcal{R}(s, I) \rightarrow \mathcal{R}(r, I) \quad (5.3.5)$$

which is denoted by $M' = \mathcal{T}_{r \leftarrow s}^{\mathcal{R}}(M)$ and it is called truncation to rank r . When the rank of the matrix M is omitted, the truncation can be written as

$$\mathcal{T}_r^{\mathcal{R}}(M) : \mathcal{T}_{r \leftarrow \text{rank}(M)}^{\mathcal{R}}(M),$$

and $M' = \mathcal{T}_r^{\mathcal{R}}(M)$.

Since the representation becomes larger (5.3.4), it is reasonable to consider truncating the sum to a smaller rank $r < r' + r''$. The truncated addition is represented by

$$M' \oplus_r M'' := \mathcal{T}_{r \leftarrow r' + r''}^{\mathcal{R}}(M' + M''), \quad (5.3.6)$$

and it is called formatted addition.

Let $M \in \mathcal{C}^{I \times I}$ be a matrix with the Singular Value Decomposition (SVD)

$$M = \sum_{i=1}^{\#I} \sigma_i u_i v_i^H$$

where $\{u_i\}_{i=1}^{\#I}$ and $\{v_i\}_{i=1}^{\#I}$ are orthonormal vectors, and $\sigma_1 \geq \sigma_2 \geq \dots \geq 0$ are the singular values. The closest rank- r matrix R with $r \leq \#I$ is expressed by

$$R = \sum_{i=1}^r \sigma_i u_i v_i^H.$$

The relative error $\varepsilon > 0$ of the approximation of a matrix M by a rank- r matrix R is measured by $\|M - R\|_2 \leq \varepsilon \|M\|_2$. To find the best matrix R , the condition that has to be satisfied is $\|M - R\|_2 = \sigma_{r+1} / \sigma_1$. This means that the rank can be expressed as a function of the relative error as

$$r(\varepsilon) := \min\{r \in \mathbb{N}_0 : \sigma_{r+1} \leq \varepsilon \sigma_1\}. \quad (5.3.7)$$

The truncation to generate low-rank matrices is performed by using SVD and omitting the small singular values.

5.4 \mathcal{H} -Matrices

5.4.1 Format and Storage

A simple structure for the block partition is introduced considering the index set $I = \{1, \dots, n\}$, where $n = 2^p$ with $p \in \mathbb{N}_0$ and I_p indicates the dependence of I on p . The matrix format \mathcal{H}_p can be constructed recursively with respect to p . For $p = 0$, the matrix $M \in \mathbb{C}^{I \times I}$ is defined by $\mathcal{H}_0(r)$ as the set of 1×1 full representation matrices. For $p > 0$, it is assumed that the format of the matrix $\mathcal{H}_{p-1} \in \mathbb{C}^{I_{p-1} \times I_{p-1}}$ is known. A matrix $M \in \mathbb{C}^{I_p \times I_p}$ can be represented by blocks matrices as follows

$$M = \begin{bmatrix} M_{11} & M_{12} \\ M_{21} & M_{22} \end{bmatrix}, \quad M_{11}, M_{22} \in \mathcal{H}_{p-1}, \quad M_{12}, M_{21} \in \mathcal{R}_{p-1}(r), \quad (5.4.1)$$

where $\mathcal{R}_{p-1}(r) := \mathcal{R}_{p-1}(r, I_{p-1})$ is the set of rank- r matrices. The set of \mathcal{H}_p is formed by the set of matrices defines in (5.4.1). This indicates that the local rank r should be selected in such a way that a reasonable accuracy can be reachable. For instance, with $r = 1$ and using \mathcal{R}_{p-1} is the representation of $\mathcal{R}_{p-1}(1)$, the recursive structure of the format \mathcal{H}_p can be denoted by

$$\mathcal{H}_p = \begin{bmatrix} \mathcal{H}_{p-1} & \mathcal{R}_{p-1} \\ \mathcal{R}_{p-1} & \mathcal{H}_{p-1} \end{bmatrix}. \quad (5.4.2)$$

The block partition for $p = 0, 1, 2, 3$ are showed in the figure 5.4.1. Let $\tau, \sigma \subset I$ be nonempty index subsets and let $M \in \mathbb{C}^{I \times I}$ be any matrix. The matrix block corresponding to the block $b = \tau \times \sigma$ is described as $M|_b := (M_{\alpha\beta})_{\alpha \in \tau, \beta \in \sigma} \in \mathbb{C}^{\tau \times \beta}$.

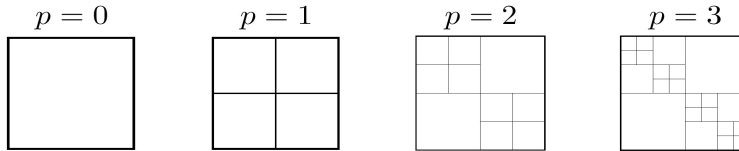


Figure 5.4.1: Block partitions. Image after Hackbusch [2015].

An important quality to be established using the format \mathcal{H}_p is the number of blocks. Let $N_{bl}(p)$ be the number of blocks in $\mathcal{H}_p(r)$. Setting $N_{bl}(0) = 1$ and using the recursion (5.4.2) can be proved that $N_{bl}(p) = 2 + 2N_{bl}(p-1)$ for $p > 0$. The recursive equation is solved by

$$N_{bl} = 3n - 2.$$

Hackbusch [2016, page 447] shows that using recursive arguments and the requirement that all matrix blocks $M|_b$ of $M \in \mathcal{H}_1$ are \mathcal{R}_1 matrices, the storage cost is

$$S_p = n + 2n \log_2 n.$$

All details related to the different operations of \mathcal{H}_p and \mathcal{R}_p matrices can be found in

[Hackbusch, 2015, chapter 3]. The structural form and the cost of the operations are briefly described:

(a) **Matrix-Vector Multiplication.** For $n = 2^p$, let $M \in \mathcal{H}_p$ and $x \in \mathbb{C}^{I_p}$. For $p \geq 1$, matrix M is taken as in (5.4.1) and x is decomposed as $x = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$ with $x_1, x_2 \in \mathbb{C}^{I_{p-1}}$. The product Mx requires the following computations: $y_{11} := M_{11}x_1$, $y_{12} := M_{12}x_2$, $y_{21} := M_{21}x_1$, $y_{22} := M_{22}x_2$, and the sums $y_{11} + y_{12}$ and $y_{21} + y_{22}$. The total cost of the operation is $N_{MV} = 4n \log_2 n - n + 2$.

(b) **Matrix addition.** Let $A, B \in \mathcal{H}_p$ be matrices that use the block structure (5.4.2), the structure of the sum is:

$$A + B = \begin{bmatrix} \mathcal{H}_{p-1} & \mathcal{R}_{p-1} \\ \mathcal{R}_{p-1} & \mathcal{H}_{p-1} \end{bmatrix} + \begin{bmatrix} \mathcal{H}_{p-1} & \mathcal{R}_{p-1} \\ \mathcal{R}_{p-1} & \mathcal{H}_{p-1} \end{bmatrix} = \begin{bmatrix} \mathcal{H}_{p-1} + \mathcal{H}_{p-1} & \mathcal{R}_{p-1} + \mathcal{R}_{p-1} \\ \mathcal{R}_{p-1} + \mathcal{R}_{p-1} & \mathcal{H}_{p-1} + \mathcal{H}_{p-1} \end{bmatrix},$$

where the required operation is $17n \log_2 n + 39n - 38$.

(c) **Matrix-matrix multiplication.** There can be three kinds of matrix-matrix products. However, one is only showed. For $n = 2^p$, and $A, B \in \mathcal{H}_p$, the product $A \cdot B$ has the following form:

$$\begin{aligned} & \begin{bmatrix} \mathcal{H}_{p-1} & \mathcal{R}_{p-1} \\ \mathcal{R}_{p-1} & \mathcal{H}_{p-1} \end{bmatrix} \cdot \begin{bmatrix} \mathcal{H}_{p-1} & \mathcal{R}_{p-1} \\ \mathcal{R}_{p-1} & \mathcal{H}_{p-1} \end{bmatrix} \\ &= \begin{bmatrix} \mathcal{H}_{p-1} \cdot \mathcal{H}_{p-1} + \mathcal{R}_{p-1} \cdot \mathcal{R}_{p-1} & \mathcal{H}_{p-1} \cdot \mathcal{R}_{p-1} + \mathcal{R}_{p-1} \cdot \mathcal{H}_{p-1} \\ \mathcal{R}_{p-1} \cdot \mathcal{H}_{p-1} + \mathcal{H}_{p-1} \cdot \mathcal{R}_{p-1} & \mathcal{R}_{p-1} \cdot \mathcal{R}_{p-1} + \mathcal{H}_{p-1} \cdot \mathcal{H}_{p-1} \end{bmatrix}. \end{aligned}$$

The total number of operations for the combination of the different matrix products is given by

$$\begin{aligned} N_{H \cdot H}(p) &= \frac{25}{2}np^2 + \frac{39}{2}np - 31n + 31, \\ N_{H \cdot R}(p) &= N_{R \cdot H}(p) = 4n \log_2 n - n + 2, \\ N_{R \cdot R} &= 3n - 1. \end{aligned}$$

(d) **Matrix inversion.** Let $M \in \mathcal{H}_p$ be a matrix, the approximation of its inversion is defined as an inversion mapping $inv : D_p \subset \mathcal{H}_p \rightarrow \mathcal{H}_p$ recursively. The inv of M for $p = 0$ is given by $inv(M) := M^{-1}$ where this is the exact inverse of the 1×1 -matrix M . The inverse of M defined on $D_{p-1} \subset \mathcal{H}_{p-1}$, using the block structure in (5.4.2) is as follows:

$$M^{-1} = \begin{bmatrix} M_{11}^{-1} + M_{11}^{-1}M_{12}S^{-1}M_{21}M_{11}^{-1} & -M_{11}^{-1}M_{12}S^{-1} \\ -S^{-1}M_{21}M_{11}^{-1} & S^{-1} \end{bmatrix}, \quad (5.4.3)$$

where $S := M_{22} - M_{21}M_{11}^{-1}M_{12}$ which is the Schur complement. The expression (5.4.3) and the algorithm that are used to compute the inverse, require M_{11} to be

regular. The approximate inverse of a matrix M from \mathcal{H}_p requires the following amount of operations:

$$N_{inv}(p) = \frac{25}{2}n \log_2^2 n + \frac{55}{2}n \log_2 n - 69n + 70.$$

In real applications the cluster must be admissible (see definition below), this is not the case for the model format. One has to generalize the definition of \mathcal{H} -Matrices.

Definition 5.4.1 (Hierarchical Matrix). *Let I be an index set, $T(I \times I)$ is a block cluster tree, and P is a partition as in (5.2.8). Furthermore, a local rank distribution function is given by*

$$r : P \rightarrow \mathbb{N}_0. \quad (5.4.4)$$

Then, the set $\mathcal{H}(r, P) \subset \mathbb{C}^{I \times I}$ of hierarchical matrices is formed by all matrices $M \in \mathbb{C}^{I \times I}$ with $\text{rank}(M|_b) \leq r(b)$ for all $b \in P^+$.

It is important to note that $M|_b \in \mathcal{R}_r(b)$ (in (5.3.2)) is required for all blocks $b \in P^+$, i.e., the factors A_b and B_b should be explicitly available in order to have the representation $M|_b = A_b \cdot B_b^H$. The matrix blocks $M|_b$ associated to the small blocks $b \in P^-$ are implemented as matrices such that $M|_b \in \mathcal{F}(b)$.

Remark 5.4.2. (a) *For the function in (5.4.4), the common choice is a constant $r \in \mathbb{N}_0$.*

Then, one says that the hierarchical matrix has the local rank r .

(b) *For the adaptive choice of the local ranks can be used a variable rank $r(b)$.*

The local rank is used to approximate the admissible block in $T(I \times I)$ up to a predefined accuracy $\varepsilon \geq 0$. This is ensured by the admissibility condition. In principles, the rank r should be chosen such that the error (5.3.7) is bounded by an accuracy $\varepsilon > 0$ which is fixed.

5.4.2 Matrix-Vector Multiplication

Let $P \subset T(I \times I)$ be a partition. Let $M \in \mathcal{H}(r, P)$, $x \in \mathbb{C}^I$, and $y \in \mathbb{C}^I$ be a matrix, and vectors, respectively. The matrix-vector product is established by the additive form $y := y + Mx$ which is carried out using a recursively procedure. The recursion is applied to any block $b = \tau \times \sigma \in T(I \times I)$. The vector y may be initialised as $y := 0$ and calling the procedure *MVM* produces the form and its description is as follows:

```

procedure MVM( $y, M, x, b$ );
if  $b = \tau \times \sigma \in P$  then  $y|_\tau := y|_\tau + M|_b \cdot x|_\sigma$ 
else for all  $b' \in S(b)$  do MVM( $y, M, x, b'$ );

```

If $b \in P^-$, then $M|_b$ is represented as a full matrix and the product $M|_b \cdot x|_\sigma$ in the second line of *MVM* is the standard matrix-vector multiplication. If $b \in P^+$, $M|_b \in \mathcal{R}_r(b)$ holds, the product $M|_b \cdot x|_\sigma$ is performed as in (5.3.3).

The matrix-Vector multiplication of hierarchical matrices requires one multiplication and one addition per matrix entry. The number of arithmetical operations N_{MV} can be bounded by the storage cost $S_{\mathcal{H}}(r, P) \leq N_{MV} \leq 2S_{\mathcal{H}}(r, P)$. This result comes from Lemma 7.17 in [Hackbusch, 2016, page 189]

5.4.3 Matrix-Matrix Multiplication

The matrix-matrix product of hierarchical matrices requires addition, and combination of addition and multiplication of low rank matrices.

Definition 5.4.3 (\mathcal{H} -matrices addition). *Let $r, r_1, r_2 : P \rightarrow \mathbb{N}_0$ be the local ranks. Let $M_1 \in \mathcal{H}(r_1, P)$ and $M_2 \in \mathcal{H}(r_2, P)$ be matrices with the same partition P . Then the formatted matrix addition \oplus_r is defined by*

$$\oplus_r : \mathcal{H}(r_1, P) \times \mathcal{H}(r_2, P) \rightarrow \mathcal{H}(r, P)$$

with $M_1 \oplus_r M_2 := \mathcal{T}_{r \leftarrow r_1 + r_2}^{\mathcal{H}}(M_1 + M_2)$.

The operation of addition $M|_b := M_1|_b \oplus_r M_2|_b$ for any block $b \in T(I \times I)$ is carried out by the following procedure

```

procedure Add( $M, M_1, M_2, b, r$ );           { $M|_b := M_1|_b \oplus_r M_2|_b$ }
{output :  $M \in \mathcal{H}(r_1, P)$ ,
  input :  $M_1 \in \mathcal{H}(r_1, P), M_2 \in \mathcal{H}(r_2, P), b \in T(I \times I, P), r \in \mathbb{N}_0$ }
if  $b \notin P$  then for all  $b' \in S_{T(I \times I)}(b)$  do Add( $M, M_1, M_2, b', r$ )
else           { $b \in P$  holds.  $r_1, r_2$  are the local ranks of  $M_1, M_2$ }
if  $b \in P^+$  then
     $M|_b := \mathcal{T}_{r(b) \leftarrow r_1(b) + r_2(b)}^{\mathcal{R}}(M_1|_b + M_2|_b)$ 
else  $M|_b := M_1|_b + M_2|_b$ ; {addition of full matrices, since  $b \in P^-$ }

```

The result of calling the procedure *Add*($M, M_1, M_2, I \times I, r$) is $M := M_1 \oplus_r M_2$. If $r \geq r_1 + r_2$, $\mathcal{T}_{r(b) \leftarrow r_1(b) + r_2(b)}^{\mathcal{R}}$ is the identity, and the blockwise addition is exact.

For the combination of the arithmetical operation of rank matrices, let $b = \tau \times \rho \in P$ be the block contained in the matrix M . Let $M' \in \mathbb{C}^{\tau \times \sigma}$ and $M'' \in \mathbb{C}^{\sigma \times \rho}$ be matrices for some $\rho \in T(I)$. Then the operation $M|_{\tau \times \rho} \leftarrow M|_{\tau \times \rho} \oplus_r (M'|_{\tau \times \sigma} \odot_r M''|_{\sigma \times \rho})$ is performed by the following procedures where \odot_r represents the truncated product

to rank r .

```

procedure MMR( $M, M', M'', \tau, \sigma, \rho$ );
begin
  if  $\tau \times \sigma \in P'$  or  $\sigma \times \rho \in P''$  then
    begin  $Z := M'|_{\tau \times \sigma} M''|_{\sigma \times \rho}$ ;
      if  $\tau \times \rho \subset b \in P^+$  then  $Z := \mathcal{T}_r^{\mathcal{R}}(Z)$            {for a suitable  $b \in T$ }
    end else           {the else case corresponds to  $\tau \times \sigma \notin P'$  and  $\sigma \times \rho \notin P''$ }
    begin  $Z|_{\tau \times \rho} := 0$ ;
      for all  $\tau' \in S(\tau), \sigma' \in S(\sigma), \rho' \in S(\rho)$ 
        do MMR( $Z, M', M'', \tau', \sigma', \rho'$ )           {recursion}
      end;
    if  $\tau \times \rho \in P^{-1}$  then  $M|_{\tau \times \rho} := M|_{\tau \times \rho} + Z$  else  $M|_{\tau \times \rho} := \mathcal{T}_r^{\mathcal{R}}(M|_{\tau \times \rho} + Z)$ 
  end;

```

If $b \in P^+$, the operation result is approximated by $\mathcal{T}_{r, \text{pairw}}^{\mathcal{T}}(M|_{\tau \times \rho} + M'|_{\tau \times \sigma} M''|_{\sigma \times \rho}) \in \mathcal{R}(r, \tau, \rho)$. If $b \in P^-$, the result is computed exactly and is represented as a full matrix.

For the multiplication of \mathcal{H} -matrices, one considers $M = M' M''$ with $M', M'' \in \mathbb{C}^{I \times I}$. Let $T(I \times I)$ be the cluster tree which is the block structure of M' and M'' . The multiplication algorithm is based on using the substructure hierarchical matrices defined as follows:

$$M := M' M'' = \begin{bmatrix} M'_{11} & M'_{12} \\ M'_{21} & M'_{22} \end{bmatrix} \begin{bmatrix} M''_{11} & M''_{12} \\ M''_{21} & M''_{22} \end{bmatrix} = \begin{bmatrix} M_{11} & M_{12} \\ M_{21} & M_{22} \end{bmatrix}, \quad (5.4.5a)$$

where

$$M_{ij} = M'_{i1} \cdot M''_{1j} + M'_{i2} \cdot M''_{2j} \quad \text{for } i, j = 1, 2 \quad (5.4.5b)$$

with $M \in \mathcal{H}(r, P)$, $M' \in \mathcal{H}(r', P')$, and $M'' \in \mathcal{H}(r'', P'')$.

The multiplication can be computed recursively by performing the products of submatrices M' and M'' , which are represented by four submatrices (5.4.5). Taking the submatrix products as $\hat{M} = \hat{M}' \hat{M}''$, if one of the submatrix \hat{M}' or \hat{M}'' does not have a hierarchical structure but belongs to \mathcal{R} or \mathcal{F} , the product can be evaluated. On the other hand, if the submatrices contain hierarchical structures, they have to be divided recursively until one of the following criterion applies:

- 1a) $\hat{M}'' = M''|_b$ for $b \in P^+$, i.e., $\hat{M}'' = \sum a_i b_i^H \in \mathcal{R}_r(b)$. The product is reduced to r matrix-vector multiplication $\hat{M}' a_i$.
- 1b) $\hat{M}'' = M''|_b$ for $b \in P^-$, i.e., $\hat{M}'' \in \mathcal{F}$. The same procedure is applied as in 1a) where the columns of \hat{M}'' are the vectors a_i .
- 2) $\hat{M}' = M'|_b$ for $b \in P$. $\hat{M}^H = \hat{M}''^H \hat{M}'^H$ can be treated as before.

- 3) The block $\hat{M} = M|_b$ for $b = \tau \times \rho \in P$ is contained in the target matrix M . The other matrices are $\hat{M}' \in \mathbf{C}^{\tau \times \rho}$ and $\hat{M}'' \in \mathbf{C}^{\rho \times \sigma}$ for some $\rho \in T(I)$. Then the operation $M|_{\tau \times \rho} \leftarrow M|_{\tau \times \rho} \oplus_r (M|'_{\tau \times \sigma} \odot_r M|''_{\sigma \times \rho})$ is performed by the following procedure.

```

procedure MM( $M, M', M'', \tau, \sigma, \rho$ );
if  $\tau \times \sigma \notin P'$  and  $\sigma \times \rho \notin P''$  and  $\tau \times \rho \notin P$  then
    for all  $\tau' \in S_{T(I)}(\tau), \sigma' \in S_{T(I)}(\sigma), \rho' \in S_{T(I)}(\rho)$  do
        MM( $M, M', M'', \tau', \sigma', \rho'$ )
    else if  $\tau \times \rho \notin P$  then { $\tau \times \sigma \in P'$  or  $\sigma \times \rho \in P''$  hold}
    begin  $Z := M'|_{\tau \times \sigma} M''|_{\sigma \times \rho};$   $M|_{\tau \times \rho} := \mathcal{T}_r^{\mathcal{H}}(M|_{\tau \times \rho} + Z)$ 
    end else MM( $M, M', M'', \tau, \sigma, \rho$ ); { $\tau \times \rho \in P$ }

```

The components M' and M'' are the input parameters and M is an input/output parameter. τ , σ , and ρ are parameters that must satisfy $\tau \times \sigma \in T(I \times I, P')$, $\sigma \times \rho \in T(I \times I, P'')$, and $\tau \times \rho \in T(I \times I, P)$.

The following computational cost of the matrix-matrix multiplication considering $n := \#I \rightarrow \infty$. The standard construction leads to the $\text{depth}(T(I \times I, P)) = \mathcal{O}(\log n)$ and $\#P = \mathcal{O}(n)$. Then, one obtains the following expression of the number of arithmetic operations of the matrix-matrix multiplication

$$N_{MM}(P, r', r'') \leq \mathcal{O}(rn \log(n)(\log(n) + r^2)), \quad (5.4.6)$$

where $r := \max\{r', r'', n_{\min}\}$. The details of the analysis is described in [Hackbusch, 2016, section 7.8.3].

5.5 \mathcal{H} -LU Decomposition

The LU decomposition of \mathcal{H} -matrices produces an approximation of the exact LU factors in $A = LU$ with a selectable accuracy which is controlled by an appropriate local rank. This factorisation is called \mathcal{H} -LU decomposition and allows to build a fast iteration in order to solve systems of equations. The matrix operations of \mathcal{H} -matrices such as addition and multiplication, and also the matrix-vector multiplication can be described without reference to an ordering of the index set. However, in the case of the \mathcal{H} -LU decomposition an ordering of the index set must be explicitly fixed. It is important to note that different orderings lead to distinct \mathcal{H} -LU decomposition. Let I be an index set and let $T(I)$ be a tree. For all clusters $\tau \in T(I)$, the orderings of indices in I are identified by a pair of integers $(\alpha_{(\tau)}, \beta_{(\tau)})$ with

$$\tau = \{i_k \in I : \alpha_{(\tau)} \leq k \leq \beta_{(\tau)}\}.$$

For two clusters $\tau, \sigma \in T(I)$ with $\tau \cap \sigma = \emptyset$ which are ordered, $\tau < \sigma$ holds if $i < j$ for all $i \in \tau$ and $j \in \sigma$. For a partition P and let $M \in \mathcal{H}(r, p)$ be a hierarchical matrix. All blocks $b = \tau \times \sigma \in P$ with $\tau \neq \sigma$ belong completely to the upper (U) and lower (L) triangular part of the partition. The diagonal blocks $\tau \times \tau \in P$ are in P^- and their corresponding matrix blocks $M|_{\tau \times \tau} \in \mathcal{F}(\tau \times \tau)$.

The hierarchical triangular matrices L and U formats for the LU decomposition are defined as

$$\begin{aligned} \mathcal{H}_L(r, p) &:= \{L \in \mathcal{H}(r, p) : L_{i_\alpha, i_\beta} = 0 \text{ for } \alpha < \beta, L_{i_\alpha, i_\alpha} = 1 \text{ for } 1 \leq \alpha < \#I\}, \\ \mathcal{H}_U(r, p) &:= \{U \in \mathcal{H}(r, p) : U_{i_\alpha, i_\beta} = 0 \text{ for } \alpha > \beta\}. \end{aligned} \quad (5.5.1)$$

Note that $U_{i_\alpha, i_\alpha} \neq 0$ for all i_α is required for the solvability of a system $LUx = b$.

The triangular matrices can be expressed by block-triangular matrices as follows:

$$\begin{aligned} \text{off-diagonal blocks: } & L|_{\tau \times \sigma} = 0 \text{ for } \tau < \sigma \text{ and } U|_{\tau \times \sigma} = 0 \text{ for } \tau > \sigma, \\ \text{diagonal blocks: } & L|_{\tau \times \tau} = I \text{ and } U|_{\tau \times \tau} \in \mathcal{F}(\tau \times \tau) \text{ for } \tau \times \tau \in P, \end{aligned}$$

where $U|_{\tau \times \tau}$ is no longer triangular. The advantage of the block-triangular decomposition is that it may be well defined even if the standard LU decomposition does not exist.

The solution of the system $Ax = b$ given the factorisation $A = LU$ is carried out by solving the systems $Ly = b$ and $Ux = y$. The first system is treated by the procedure *Forward_Substitution* where the input data are L, I , and b , and the output is y . These parameters require that $\tau \in T(I \times I, p)$, and $y, b \in \mathbb{C}^I$, and L satisfies (5.5.1) with $P \subset T(I \times I)$. The procedure *Forward_Substitution*(L, I, y, b) is called with $\tau = I$ and the input vector b is overwritten. The solution $y|_\tau$ of $L|_{\tau \times \tau} y|_\tau = b|_\tau$ is computed by the procedure which is showed in the appendix B.1 of LU decomposition.

The second system $Ux = y$ is solved using the procedure *Backward_Substitution* where the input parameters are τ, y , and U (satisfying (5.5.1)), and the output is the vector $x \in \mathbb{C}^I$. This procedure is similar to the *Forward_Substitution* procedure and it is described in the appendix B.1. The vector y in this procedure is overwritten.

The complete solution of the system $LUx = b$ is performed by the following procedure:

```

procedure Solve_LU( $L, U, I, x, b$ );           { $L, U, I, b$  input ;  $x$  output }
begin  $x := b$ ;
        Forward_Substitution( $L, I, x, x$ );
        Backward_Substitution( $U, I, x, x$ );
end;

```

The hierarchical LU factors in $A = LU \in \mathbb{C}^{I \times I}$ are generated by four subtasks.

For example, taking the two sons of the index set I , the matrices have the structure

$$\begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} = \begin{bmatrix} L_{11} & 0 \\ L_{21} & L_{22} \end{bmatrix} \begin{bmatrix} U_{11} & U_{12} \\ 0 & U_{22} \end{bmatrix}.$$

The subtasks are the following:

- (i) Compute L_{11} and U_{11} as factors of the LU decomposition of A_{21} ,
- (ii) Use $L_{11}U_{12} = A_{12}$ to compute U_{12} ,
- (iii) Use $L_{21}U_{11} = A_{21}$ to compute U_{21} ,
- (iv) Compute L_{22} and U_{22} as LU decomposition of $L_{22}U_{22} = A_{22} - L_{21}U_{12}$.

The procedures $Forward_M(L_{11}, U_{12}, A_{12}, \tau_1, \tau_2)$ and $ForwardT_M(U_{22}, L_{22}, A_{22}, \tau_1, \tau_2)$ are used to solve the problems (ii) and (iii), respectively. They are shown in the appendix B.2 of LU decomposition and their descriptions are in [Hackbusch, 2016, section 7.6.3]. The right-hand side in the LU decomposition in the problem (iv) can be solved by using the usual formatted multiplication.

The LU factors of $L_{11}U_{11} = \dots$ and $L_{22}U_{22} = \dots$ have to be determined. A recursion is defined and it is performed until the leaves are reached, and the usual LU decomposition for the full matrices is applied.

The desired LU factors of A are produced by calling the procedure $LU_Decomposition(L, U, A, I)$. In general, the problem $L|_{\tau \times \tau} U|_{\tau \times \tau} = A|_{\tau \times \tau}$ for $\tau \in T(I \times I, P)$ is solved using $LU_Decomposition(L, U, A, \tau)$.

```

procedure  $LU\_Decomposition(L, U, A, \tau)$ ;
if  $\tau \times \tau \in P$  then compute  $L_{\tau \times \tau}$  and  $U_{\tau \times \tau}$  as LU factors of  $A|_{\tau \times \tau}$ 
else for  $i = 1$  to  $\#S(\tau)$  do
  begin  $LU\_Decomposition(L, U, A, \tau[i])$ ;
    for  $j = i + 1$  to  $\#S(\tau)$  do
      begin  $ForwardT\_M(U, L, A, \tau[j], \tau[i])$ ; (5.5.2)
         $Forward\_M(L, U, A, \tau[i], \tau[j])$ ;
      for  $r = i + 1$  to  $\#S(\tau)$  do
         $A|_{\tau[j] \times \tau[r]} := A|_{\tau[j] \times \tau[r]} \ominus L|_{\tau' \times \sigma[i]} \odot U|_{\sigma[i] \times \sigma[j]}$ 
      end
    end
  end;

```

The cost of $Forward_Substitution(L, I, y, b)$ and $Backward_Substitution(U, \tau, x, y)$ can be verified and estimated by Lemma D.18 in [Hackbusch, 2016]. The combination of both costs give the cost of the LU decomposition:

$$N_{LU}(r, P) \leq 2S_{\mathcal{H}}(r, P),$$

where $S_{\mathcal{H}}(r, P)$ is the storage cost of matrices in $\mathcal{H}(r, P)$ and it is described in Lemma D.17 [Hackbusch, 2016].

The costs of solving the systems $LX = Z$ and $XU = Z$ are compared with a standard multiplication of \mathcal{H} -matrices in which one obtains

$$N_{\text{Forward}_M}(r, P) + N_{\text{Forward}_T_M}(r, P) \leq N_{MM}(P, r, r),$$

where $N_{MM}(P, r, r)$ is in (5.4.6). The procedure in (5.5.2) that produces the LU decomposition does not require more operations than the matrix-matrix multiplication:

$$N_{LU \text{ decomposition}}(r, P) \leq N_{MM}(P, r, r).$$

5.5.1 Sparse Matrices

The inverse of the finite element matrix from a general elliptic differential operator can be well approximated by \mathcal{H} -matrices as is proved in [Hackbusch, 2016]. The finite element matrix in (2.5.10) is stored in sparse format. In terms of the storage cost and the cost of matrix-vector multiplication, this format is more convenient. However, the main reason to convert the matrix L into the $\mathcal{H}(r, P)$ format is to apply the hierarchical matrix operations such as LU decomposition. The transfer is based on the following Lemma:

Lemma 5.5.1. *Let $\mathcal{H}(r, P) \in \mathbb{C}^{I \times I}$ be an arbitrary \mathcal{H} -matrix format, and P based on the admissibility condition (5.2.5). Furthermore, let $\text{dist}(\tau, \sigma)$ be defined by (5.2.3) and (5.2.4b). Then any finite element matrix is in $\mathcal{H}(r, P)$ for any $r \in \mathbb{N}_0$.*

Proof. See [Hackbusch, 2015, Lemma 9.4.a].

The transfer of a sparse matrix into the \mathcal{H} -format is performed by the next two steps:

- (i) Set $M|_b := 0$ for $b \in P^+$,
- (ii) Copy the block matrix $M|_b$ for $b \in P^-$ as a full matrix.

One tries to optimise the \mathcal{H} -LU decomposition for sparse matrices using the following idea: find a permutation P such that the LU decomposition of PAP^H is sparser than for A . The details of sparsity pattern will be described in section 5.5.2. The LU decomposition of a matrix $A \in \mathbb{C}^{I \times I}$ is determined by the order of the index set I , which is obtained from the cluster tree $T(I)$ (*Forward_Substitution Procedure*). Hence, distinct cluster trees are needed for different permutations. The construction of cluster trees for sparse matrices will be introduced in 5.5.3.

The inverse process of a sparse finite element matrix yields a fully populated matrix. The inverse matrix can be well approximated by \mathcal{H} -LU decomposition where the factors L and U are represented by hierarchical triangular matrices (5.5.1) in $\mathcal{H}(r, P)$ [Hackbusch, 2015, section 9.2.8], [Grasedyck et al., 2009]

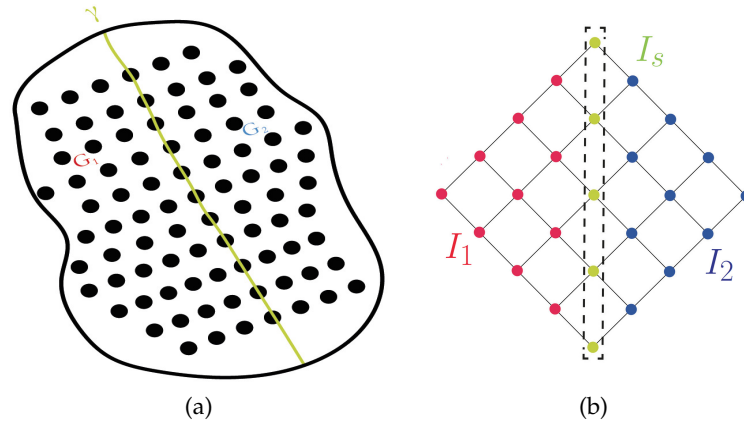


Figure 5.5.1: (a) The graphs G_1 and G_2 and the separator γ . (b) The graphs represented in the graph matrix.

5.5.2 The \mathcal{H} -LU Decomposition for Sparse Matrices

The sparse matrix data structure is not sufficient to compute LU decomposition to solve sparse linear systems of equations. This is due to the fact that the fill-in effects should be minimised during the factorisation process. In other words, given a matrix A find a permutation matrix such that the computation of the nonzero in the decomposition PAP^T is minimum.

The index ordering is important to control fill-in. This has a connection to graph theory given that the permutation of a square matrix A with nonzero diagonal entries into block triangular form is identical to finding a strong connected components of a direct graph $G(A)$ (Appendix A). There are many ways to find the components.

A popular method to reorder the indices to reduce the fill-in is called nested dissection method. The domain is described by a graph where its vertices are represented by nodes or points that are in the index set I . The main idea is to separate the graph in three subgraphs. Two of them G_1 and G_2 are disconnected and their vertices are in the subsets $I_1, I_2 \in I$, respectively. The third graph γ contains the vertices which connect two subgraphs and it is named separator (Fig. 5.5.1a). These vertices are in the index subset I_s .

The matrix graph of $G(A)$ (Definition A.0.1) is showed in figure 5.5.1b where I_1 and I_2 are the index subsets, and I_s is the index subset of the separator. The index subsets represent the vertices of the graph in the figure 5.5.1a. In terms of I , the formulation of the nested dissection method is

$$I = I_1 \dot{\cup} I_2 \dot{\cup} I_s \quad \text{with} \quad \#I_1 \approx \#I_2, \quad \#I_s \ll \#I, \quad (5.5.3a)$$

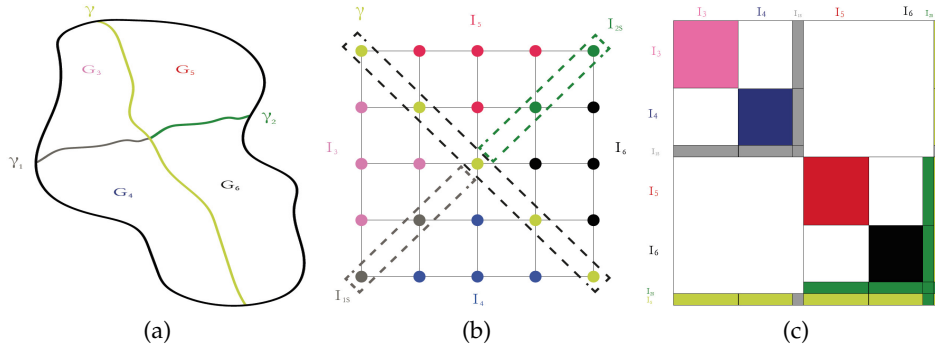


Figure 5.5.2: The two level recursion: (a) The subgraphs G_3 , G_4 , G_5 , and G_6 , and the separators γ , γ_1 , and γ_2 . (b) The graph matrix of the subgraphs. (c) The corresponding block matrices of the \mathcal{H} -matrix.

$$A = \begin{bmatrix} A_{11} & A_{12} & A_{1s} \\ A_{21} & A_{22} & A_{2s} \\ A_{s1} & A_{s2} & A_{ss} \end{bmatrix} = \begin{array}{|c|c|c|} \hline \text{red} & \text{white} & \text{green} \\ \hline \text{white} & \text{blue} & \text{green} \\ \hline \text{green} & \text{green} & \text{green} \\ \hline \end{array}, \quad (5.5.3b)$$

where

$$A_{ii} := A|_{I_i \times I_i}, \quad (5.5.3c)$$

$$A_{is} := A|_{I_i \times I_s}, \quad A_{si} := A|_{I_s \times I_i}, \quad \text{for } i = 1, 2,$$

$$A_{ss} := A|_{I_s \times I_s}.$$

The process of separation can be applied recursively in the graph G_1 and G_2 where they are divided by the separators γ_1 and γ_2 into four subgraphs: G_3 , G_4 , G_5 , and G_6 (Fig 5.5.2a). The subgraphs and the separators are represented in the graph matrix in figure 5.5.2b. The corresponding block matrices are showed in figure 5.5.2c. To ensure that the recursive process is continued, the subblock matrices $A_{ii} := A|_{I_i \times I_i}$ for $i = 1, 2$ must fulfil (5.5.3a) or be sufficiently small.

As the nested dissection method is based on domain separation, it gives a major advantage to parallelise the \mathcal{H} -LU decomposition [Grasedyck et al., 2008, 2009]. This is an important aspect to be considered because the complex linear systems of equations can be large. The relationship between voxels of a 3D image and the size of the linear system is given by $(NV_x + 1) * (NV_y + 1) * (NV_z - 1) = size$, where NV_x , NV_y , and NV_z are the number of voxel in the directions X , Y , and Z , respectively. For example, for a image of $1000 \times 1000 \times 1000$ voxels, the size of the linear system is 10,000,998,999 equations.

5.5.3 Construction of the Cluster Trees and Admissibility Condition

Let I be the index set which is split into three subsets. Let $\xi_i \in \mathbb{C}^3$ be the nodal points with $i \in I$. The binary decomposition of the set I into the sets \hat{I}_1 and \hat{I}_2 is produced by the partition of the 3-dimensional cuboid (minimal box). The construction of the index sets I_1, I_2 and I_s is carried out in such way that they satisfy the condition (5.5.3a). This is as follows: the set $I_1 := \hat{I}_1$ and it does not change, the sets I_2 and I_s are built by

$$I_s := \{i \in \hat{I}_2 : \text{there are } A_{ij} \neq 0 \text{ or } A_{ji} \neq 0 \text{ for some } j \in I_1\}, \quad I_2 := \hat{I}_2 \setminus I_s.$$

One assumes that exists a domain $\Omega \subset \mathbb{R}^3$. The index sets I_1 and I_2 correspond to two subdomains $\Omega_1, \Omega_2 \subset \mathbb{R}^3$, respectively. The indices of the set I_s are in a 2-dimensional plane which represent the separator γ .

Let $T(I)$ be the cluster tree which contains '3-dimensional' clusters $T_d(I)$ and '2-dimensional' clusters $T_{d-1}(I)$. They are defined as:

- (a) $I \in T_d(I)$,
- (b) if $\tau \in T_d(I)$, the sons τ_1, τ_2 belong to $T_d(I)$, whereas $\tau_s \in T_{d-1}(I)$,
- (c) all successors of $\tau \in T_{d-1}(I)$ belong to $T_{d-1}(I)$.

The usual decomposition rule halves the volume $T - d(I)$ or the area $T - d - 1(I)$ independent of d . However, the diameter is halved after 3 steps for $d = 3$ and after 2 steps for $d = 2$. This would lead to a tree where the clusters of $T - d - 1$ may have smaller diameter than the clusters of $T - d$ although they belong to the same level. Therefore, one has to change the rule so that also clusters of $T - d - 1$ obtain a halved diameter after 3 steps and not 2. For this purpose, each cluster of $T - d - 1$ which has been twice decomposed before remains unchanged in the third step, i.e., the cluster and its only son are equal.

The admissibility condition in (5.2.6) does not work if one wants to build the block structure for sparse matrices. The zero blocks generated in (5.5.3b) (represented by white the colour in the matrix) are characterised by

$$\tau' \times \tau'' \text{ with } \tau' \neq \tau'' \text{ and } \tau', \times \tau'' \in S(\tau) \cap T_d(I) \text{ for some } \tau \in T_d(I). \quad (5.5.4)$$

The blocks $b = \tau' \times \tau''$ do not fulfil this condition due to the fact that the support sets $X_{\tau'}$ and $X_{\tau''}$ touch at the separator γ , and as a consequence the $\text{dist}(\tau', \tau'')$ vanishes. In this case, the decomposition process of b does not make sense. In order to satisfy the expression (5.5.4), the admissibility condition is modified as follows:

$$\text{adm}^{**}(\tau' \times \tau'') := [\text{adm}^*(\tau' \times \tau'') \text{ or } \tau' \times \tau'' \text{ satisfies (5.5.4)}].$$

The procedure of the minimal admission partition defined in (5.2.7) can be applied for $P \subset T(I \times I)$ using adm^{**} . The partition P is divided into P^- and P^+ . A suitable ternary partition is given by $P = P^0 \cup P^- \cup P^+$ where $P^0 := \{b \in P \text{ satisfies (5.5.4)}\}$ and $P \setminus P^0$ is separated into $P^- \cup P^+$.

5.5.4 Algebraic LU Decomposition

There is an advantage generated by the cluster tree $T(I)$ of the \mathcal{H} -matrix which represents a finite element matrix. For a matrix $A \in \mathcal{H}(r, P)$ that satisfies $A|_b = 0$ for all $b \in P^0$, its \mathcal{H} -LU decomposition performed by the procedure (5.5.2) produces the factors $L, U \in \mathcal{H}(r, P)$ such that $L|_b = U|_b = 0$ for $b \in P^0$, i.e., they contain many zero blocks (Fig. 5.5.3).

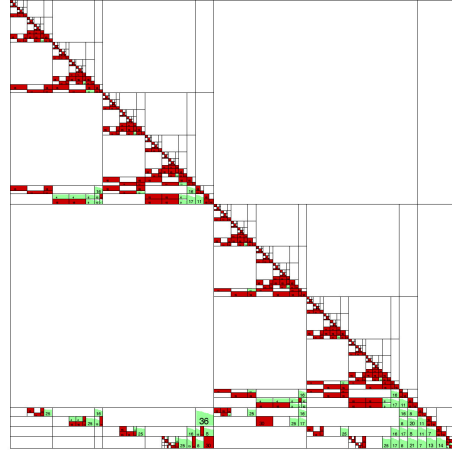


Figure 5.5.3: The factor L where the white colour represents the zero blocks. Image after Hackbusch [2016].

5.6 \mathcal{H} -LU Iteration

The hierarchical matrix operations are approximations. The accuracy of the \mathcal{H} -matrix technique is based on the local rank. The error is not characterised by the machine precision; it depends on the selection of the local rank. There are two schemes to be considered in order to use the hierarchical matrices:

- (a) For a smaller local rank, the performance of the \mathcal{H} -matrices in terms of storage and computational costs is decreased. The accuracy is low ($\varepsilon < 1$). However, it may be sufficient to produce a \mathcal{H} -LU decomposition that allows to carry out by several steps of a \mathcal{H} -LU iteration (equation (5.6.1)).
- (b) For a larger local rank, the reachable accuracy is high but the computational work rises given that the local rank increases logarithmically, i.e., $r \sim \log(1/\varepsilon)$ for $\varepsilon \ll 1$. However, the iteration method requires one or two steps.

The iteration $\Phi_{\mathcal{H}\text{-LU}}$ can be defined combining the linear iteration (4.3.2) and the \mathcal{H} -LU decomposition as

$$x^{m+1} := x^m - W^{-1}(Lx^m - b), \quad (5.6.1)$$

where $N = W^{-1}$ and $W = LU$. This method has the following properties:

- (i) For a \mathcal{H} -LU decomposition that approximates the LU factors of any matrix, the error $I - W^{-1}A$ is regulated by the local rank.
- (ii) The procedure (5.5.2) is used to compute the inverse of the matrix $W = LU$.
- (iii) The graph $G(L)$ has the data required to build the matrix W . In this case, the iteration is called algebraic.
- (iv) If $W > 0$ and L as defined in the section 4.2, then the iteration is positive definite.

It is important to notice that the matrix L in (5.6.1) is the matrix of the complex linear system in (3.3.1), while the matrix L in the expression $W = LU$ corresponds to \mathcal{H} -LU decomposition.

The approximation error can be computed as

$$\|I - NL\|_2 \leq \varepsilon < 1 \quad (5.6.2)$$

for $W \approx L$ or $N = W^{-1} \approx L^{-1}$. According to the theorem B.25 in [Hackbusch, 2016] $\|A\|_2 = \rho(A)$ for all normal matrix $A \in \mathcal{C}^{I \times I}$, the computation of the norm matrix in (5.6.2) implies the corresponding estimation using the spectral radius, i.e.,

$$\rho(I - NL) \leq \varepsilon < 1. \quad (5.6.3)$$

For example, using the inequality (5.6.2) with accuracy $\varepsilon = 1/10$, $\Phi_{\mathcal{H}\text{-LU}}$ enhances the result by one decimal for each step. Even though the accuracy can be seen as fast convergence, the approximation of the inverse may be still rough.

The matrix operations that are performed by the \mathcal{H} -matrix technique are almost linear work $\mathcal{O}(n \log^\beta n)$. As the LU factors are approximated using a local rank r , the effective amount of work for the iteration $\Phi_{\mathcal{H}\text{-LU}}$ is expressed as:

$$Eff(\Phi_{\mathcal{H}\text{-LU}}) = \mathcal{O}(r^\alpha \log^\beta n),$$

where $\alpha, \beta > 0$ and they are integers. Hence, it may be more convenient to choose a smaller local rank.

5.7 \mathcal{H} -Matrices for Solving Complex Linear Systems of Equations Using \mathcal{H} -Lib^{Pro}

The construction of the complex system of equations is based on the following information: a 3D image file where the material is represented; the electrical conductivity and the dielectric constant of the phases in the material; the electric constant of air; and a frequency (see tables 3.1 and 3.2). The Finite Element method is used where the voxel of the image constitutes the finite element. A phase in a finite element is characterised by its physical properties.

The following scheme describes how a complex system of equations is built:

- (1) A 3D image file of the material needs to be read, as well as a file with the physical parameters of the phases, and frequency or a range of frequency, the absolute residual reduction and the number of iteration for the solver.
- (2) The local stiffness matrix is constructed using the physical parameters and the frequency.
- (3) The Dirichlet boundary condition is applied at the top and at the bottom of the image, while the Neumann boundary condition is applied on the other faces of the image.
- (4) The global stiffness matrix (L) is created by assembling the local stiffness matrices. Vector b is established by the Neumann boundary condition.

The Hierarchical matrix numerical technique is implemented by Kriemann [2008a,b] in a software library (<http://www.hlibpro.com>) named \mathcal{H} -Lib^{Pro}. The software package is equipped with different sets of functions to use the hierarchical structures, convert dense or sparse matrices in \mathcal{H} -matrices, to carry out the arithmetical operations and inversion and \mathcal{H} -LU decomposition, to use various direct and iterative methods for solving linear systems of equations. The functions are grouped as follows: initialisation and finalising the use of the library, admissibility condition, clusters, cluster tree, block clusters, creation and manipulation of real and complex vectors, and of real and complex matrices, algebraic operations, different solvers, different forms of the input/output data, and accuracy management. The functions are used to write codes in C Language or C++. In the link <http://www.hlibpro.com> one can find the description of all functions, code samples, and how to install and use the library.

A complex system of equations is solved using the \mathcal{H} -Lib^{Pro} as follows:

- (1) The library is initialised, and the accuracy and the output format of the solvers are established.
- (2) The global stiffness matrix is converted in \mathcal{H} -matrix and the vector in the right side of the system is stored in a data structure to be used by the \mathcal{H} -Lib^{Pro} library.
- (3) The \mathcal{H} -LU decomposition is carried out using the corresponding functions to do it. The \mathcal{H} -Lib^{Pro} library provides a function that it is used to compute the inversion error.
- (4) The Richardson method and GMRES use the \mathcal{H} -LU decomposition in order to solve the complex system of equations. The output of the iteration process is: number of the iteration, the defect, and the convergence rate.
- (5) The library is finalised.

Two codes were implemented to compute the solutions of the complex systems of linear equations. The first code computes the \mathcal{H} -LU decomposition which in combination with the Richardson method or GMRES are used to solve the complex system

of equations. A 3D image sample is used to build the complex system corresponding to one frequency. The purpose of this code is to evaluate the accuracy used for solving a complex system. The code is showed in section C.1 in appendix C.

The second code was developed to solve complex systems of equations generated by an interval of frequency $[\omega_1, \omega_n]$ and using only one \mathcal{H} -LU decomposition. Let ω_k be the frequency at the centre of the interval. The complex system of equations associated to the frequency ω_k is constructed, and its \mathcal{H} -LU decomposition is calculated. The solution of the system is computed using the \mathcal{H} -LU decomposition first in combination with the Richardson method, and the second one with GMRES algorithm. The same \mathcal{H} -LU decomposition is utilised to solve the rest of the complex systems generated by the subintervals $[\omega_1, \omega_{k-1}]$ and $[\omega_{k+1}, \omega_n]$.

For the solution of each complex system, the iteration process is stopped when the maximum number of iterations or the absolute residual reduction are reached. However, the convergence rate is an important parameter to observe in order to measure the convergence of the process. The results will be showed in terms of the convergence rate. In appendix C, section C.2 shows this code.

Numerical Results

6.1 Introduction

The application of the Hierarchical matrix technique to solve the complex system of linear equations with a symmetric complex matrix is tested running numerical experiments. The data available to run the experiments is based on the six material samples represented in 3D images of 128 cubes, the physical parameters of the phases in the materials, and the interval of frequency (see tables 3.1 and 3.2). The accuracy for the \mathcal{H} -matrix operations, the absolute residual reduction, and the maximal number of iterations as the stop criterion for the solvers are also used as input data. As the image size is the same for all samples, the matrix sizes are the same in terms of dimension and memory, i.e., 2113407×2113407 , and 1301.92 MB , respectively. The tests were executed on an iMac computer with an Intel Core i7 at 3.5 GHz and 32 GB of memory.

To make the discussion of the numerical results comprehensive and clearer, let us clarify that for each frequency a complex system of linear equations is generated. When one refers to a frequency, the complex system associated to this frequency is implicit. Moreover, the \mathcal{H} -LU decomposition calculated for this complex system is related to the frequency as well.

In general, one takes the first frequency of a sample and the input data, and uses the code in section C.1 to be run with different accuracy. The goal is to have an idea of the accuracy values which is established by taking into account the inversion error. The selected accuracy value will be used as a starting point for some frequencies in the interval. The full interval of frequency is split into different smaller subintervals. The frequency at the centre of the subinterval is chosen to generate the complex system of linear equations and to compute the \mathcal{H} -LU decomposition. The complex system of equations generated by this frequency and the complex systems of the rest of the frequencies in the subinterval are solved using the \mathcal{H} -LU decomposition. In other words, one \mathcal{H} -LU decomposition is used to solve all the complex systems generated in the subinterval.

6.2 Artificial Samples

The accuracy is a parameter to be determined, used by the \mathcal{H} -matrices to solve the complex systems of linear equations. For the sphere sample, the first frequency ($\omega = 1.005221_{10} + 7 = 1.005221 \times 10^7$) is chosen to compute different \mathcal{H} -LU decompositions. The computations are carried out using various values of accuracies, and the absolute residual reduction equals $1.0_{10} - 6$ as one of parameters to stop the iteration process.

Table 6.1 illustrates the results of the \mathcal{H} -LU decomposition for the different accuracy values. The accuracy is represented by ε in (5.3.7) that it is used to generate the rank matrices. As is expected, when the accuracy decreases the amount of memory and the time of computation of the \mathcal{H} -LU decomposition increases. The inversion error is calculated by $\|I - WA\|_2$ where $W \approx A^1$ and $A \in \mathcal{C}^{I \times I}$. The results show that there is a reduction in the inversion error. Moreover, it is observed that for the calculations of the solutions, the defect reaches the absolute residual reduction in a few iterations. The convergence rate which is based on theorem 4.3.5, it is computed by $(\|d^m\| / \|d^0\|)^{\frac{1}{m}}$ of the m -th iterates where d^m is the defect. For $d^0 = Ax^0 - b$, the \mathcal{H} -Lib^{Pro} initialises $x^0 = 0$. It can see from the table that the results of the convergence rate reached to values much less than one. The accuracy value equals $1.0_{10} - 7$ was selected to run the numerical experiments used for the sphere sample. This was done because the inversion error is less than one, and the convergence just took two steps.

Accuracy	\mathcal{H} -LU (MB)	\mathcal{H} -LU (sec)	Inversion Error	Iterations	$\ b - Ax\ _2$	Convergence rate
$1.0_{10} - 5$	21979.12	782.1	$1.2718_{10} + 1$	3	$1.6270_{10} - 7$	$2.3438_{10} - 3$
$1.0_{10} - 6$	24686.12	1103.2	5.9287	2	$1.5126_{10} - 7$	$1.6024_{10} - 4$
$1.0_{10} - 7$	27423.56	1554.3	$6.1616_{10} - 1$	2	$1.4241_{10} - 7$	$4.8229_{10} - 4$

Table 6.1: The solutions of the complex system of linear equations generated from the sphere sample at the first frequency ($\omega = 1.005221_{10} + 7$) with an absolute residual reduction = $1.0_{10} - 6$ and for different accuracy values.

The first test of the sphere sample consists in generating the complex system of linear equations in each frequency of the interval. Only one \mathcal{H} -LU decomposition is computed to solve all the complex systems and it is calculated from the complex system generated in the first frequency. The results are showed in figure 6.2.1 where the convergence rate vs the frequency is plotted. It was observed that the convergence rate of the solution of the complex system associated to the first frequency is the lowest value. As was expected, it is due to the fact that the \mathcal{H} -LU decomposition was computed at this frequency. The solutions of the complex systems created at the rest of the frequencies produced convergence rates less than one, which fulfils the equation (4.3.6).

For the second numerical experiment of the sphere, the interval of frequency is divided into a half-subinterval (from the first frequency to its left side) and five subintervals. They are consecutively denoted by the colour frequencies: green, blue,

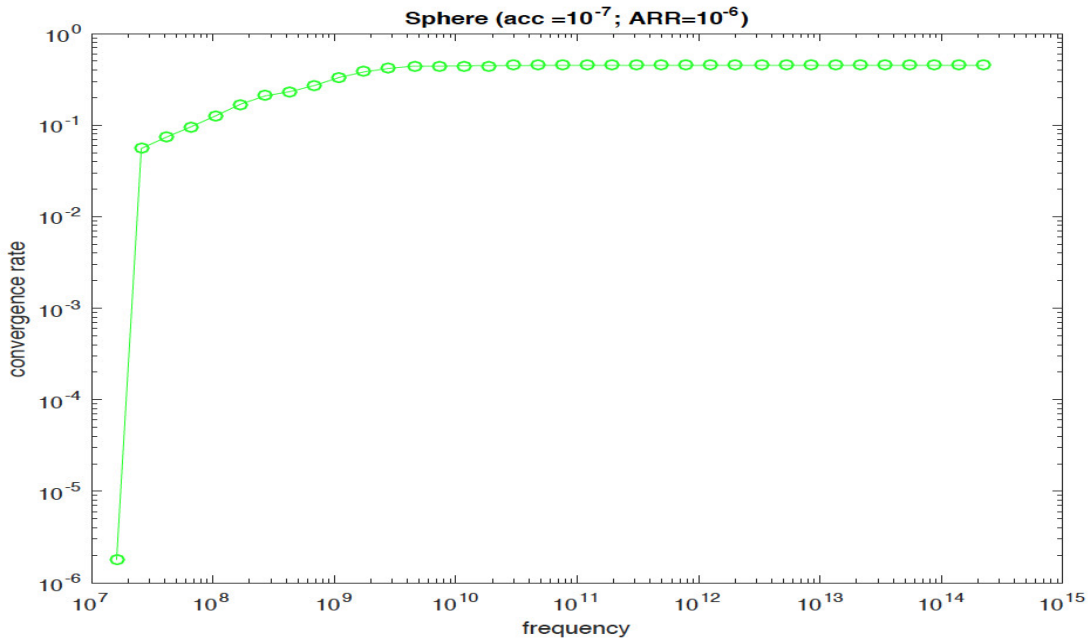


Figure 6.2.1: The convergence behaviour of the solutions of the complex systems of linear equations generated by the interval of frequencies from the sphere sample is plotted vs the frequency. Only one \mathcal{H} -LU decomposition at the first frequency is computed and it is used to solve all complex systems.

red, cyan, black, and yellow. There are intersections in all the divisions of the interval, except for the right side of the last subinterval. In the first frequency of the green half-subinterval is where the \mathcal{H} -LU decomposition is calculated, and it is used to solve all the complex systems of linear equations generated by all frequencies in this half-subinterval. In each of the following subintervals, the \mathcal{H} -LU decomposition is computed for the complex system associated to the frequency at the centre of the subinterval. This \mathcal{H} -LU decomposition is used to solve the complex systems generated by the rest of the frequency in the subinterval and the complex system associated to the central frequency.

The results of the complex systems generated for the second test are presented in Figure 6.2.2a. The convergence rate vs the frequency in terms of colour subintervals are plotted. The smallest convergence values correspond to the complex systems generated by the frequencies where the \mathcal{H} -LU decompositions were computed. The solution in the first frequency is reached for a convergence value equal to $1.78_{10} - 6$, whereas for the frequencies in the other subintervals their values are below $1.0_{10} - 6$. For the complex systems associated to the frequencies where the \mathcal{H} -LU decompositions were not calculated, the iteration method arrives at the solutions yielding convergence values that are below some fixed constant $c < 1$ and can be grouped into three ranges. For the values are bounded away from one, they correspond to the solutions of the complex systems in the subinterval of frequencies green, blue, red, and cyan. For the convergence values in the yellow and black subintervals, the

ranges are from $4.1_{10} - 3$ to $1.0_{10} - 2$, and from $4.2_{10} - 5$ to $6.8_{10} - 3$, respectively.

Figure 6.2.2a shows there is an intersection of frequency between the cyan and black subintervals. The convergence values of the frequencies in the black subinterval belong to the range of convergence values of the cyan frequencies. This also happens with an intersection of two values between the black and yellow subintervals where the two convergence values (frequencies on the right side of the yellow subinterval) belong to the range of black subinterval. These situations can be solved just by taking the lowest convergence values. Moreover, it is observed that for the higher frequencies ($1.0_{10} + 11 \leq \omega \leq 1.0_{10} + 15$) the solutions produce lower convergence rates compare to the values in the lower frequencies ($1.0_{10} + 7 \leq \omega \leq 1.0_{10} + 11$).

Due to the fact that the \mathcal{H} -LU decomposition is expensive to compute in terms of memory and time (see table 6.1), for the third numerical experiment three subintervals, namely the blue, the cyan, and the yellow from figure 6.2.2a are taken out. The green half-subinterval keeps the values of frequencies, and the red and the black subintervals become bigger covering the frequencies of the subintervals that were previously eliminated.

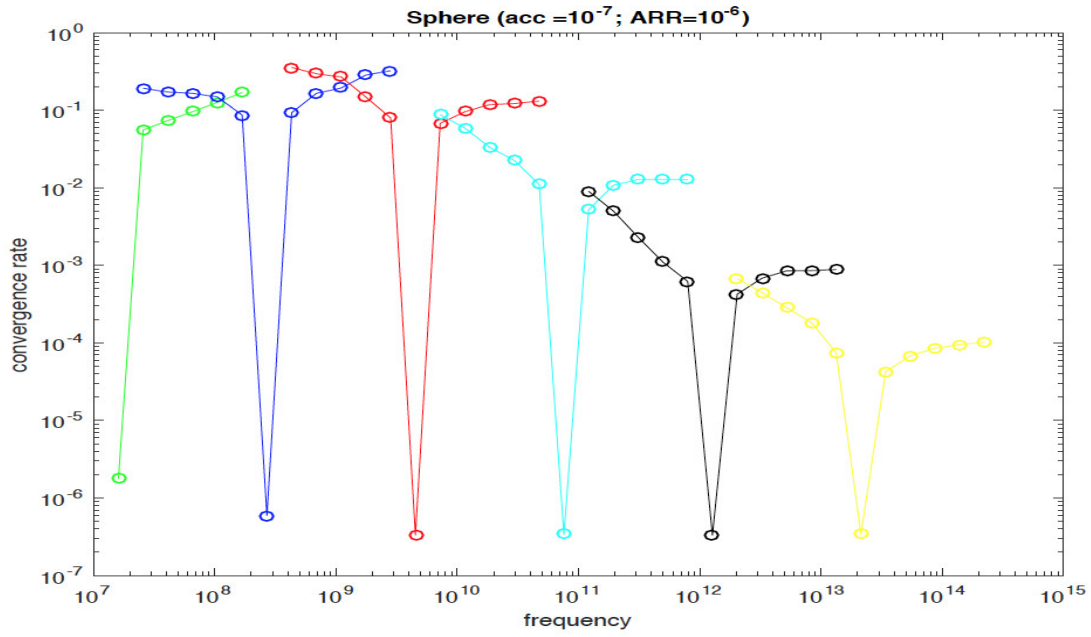
The new test is performed using the new three groups of subintervals. Figure 6.2.2b shows that the convergence speeds of the solutions of the complex systems of the frequencies in the green and the red subintervals are from $5.0_{10} - 2$ to $6.4_{10} - 1$. The convergence values of the first four frequencies in the red subinterval are a bit bigger than the convergence values of the same frequencies in figure 6.2.2a. For the black subinterval, the range of convergence speed is between $4.2_{10} - 4$ and $1.0_{10} - 2$. It can be noticed that for frequencies below $1.0_{10} + 11$ the convergence values are bigger compared to the values above $1.0_{10} + 11$. The behaviour of the convergence speed is similar to the result in figure 6.2.2a. However, the computations of the solutions of the complex systems grouped in three subintervals are less expensive because only three \mathcal{H} -LU decompositions were calculated instead of five.

The complex systems of linear equations generated by random voxel sample and its interval of frequency were solved using the same scheme of test applied for the sphere sample. The starting point is to establish the accuracy value that is calculated from the complex system generated by the first frequency in the interval, i.e., $1.005221_{10} - 2$. The absolute residual reduction is $1.0_{10} - 6$ and the computations are carried out for three accuracy values.

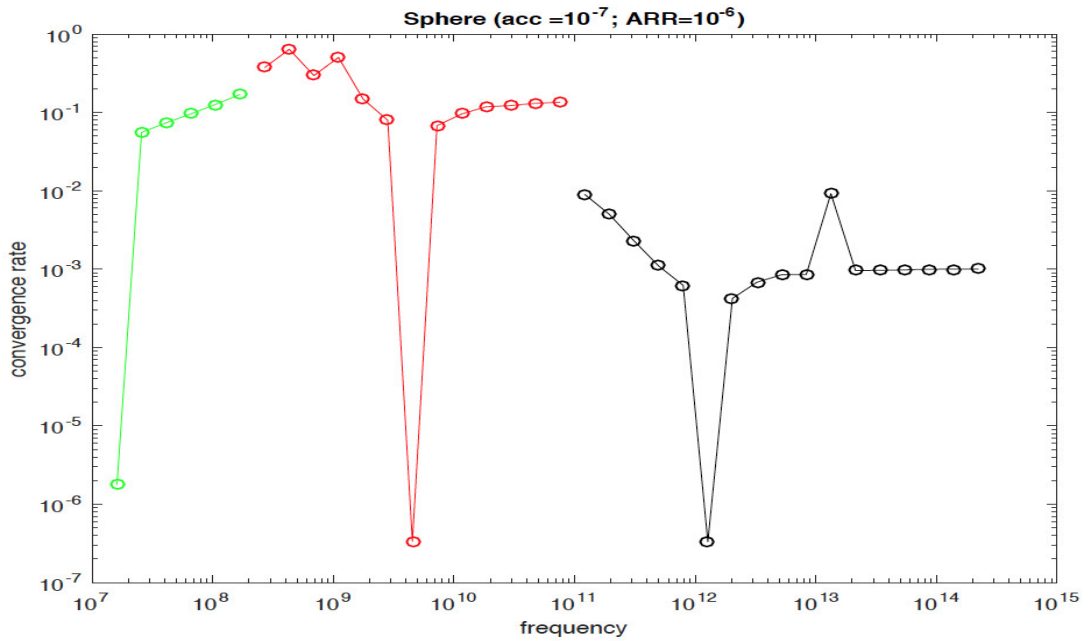
Accuracy	\mathcal{H} -LU (MB)	\mathcal{H} -LU (sec)	Inversion Error	Iterations	$\ b - Ax\ _2$	Convergence rate
$1.0_{10} - 5$	21996.43	778.6	$9.1770_{10} - 1$	2	$9.2095_{10} - 8$	$4.4878_{10} - 5$
$1.0_{10} - 6$	24709.87	1116.1	$8.3771_{10} - 2$	2	$8.4468_{10} - 8$	$1.4225_{10} - 4$
$1.0_{10} - 7$	27455.34	1489.0	$1.9837_{10} - 2$	1	$2.2089_{10} - 8$	$1.7825_{10} - 6$

Table 6.2: The solutions of the complex system of linear equations generated from the random voxel sample at the first frequency ($\omega = 1.005221_{10} - 2$) with an absolute residual reduction = $1.0_{10} - 6$ and for different accuracy values.

Table 6.2 presents the results of the solution of the complex systems of linear



(a)



(b)

Figure 6.2.2: (a) The convergence of the solutions of the complex systems of equations from the sphere sample is plotted vs the frequency. The different subintervals are represented by distinct colours. The lowest peaks of convergence rate in each subinterval correspond to the frequency where the \mathcal{H} -LU decomposition is computed. For the rest of the frequencies in each subinterval, its corresponding \mathcal{H} -LU decomposition is used to solve the complex systems of equations. (b) The convergence rate vs frequency are plotted for three subintervals of frequency.

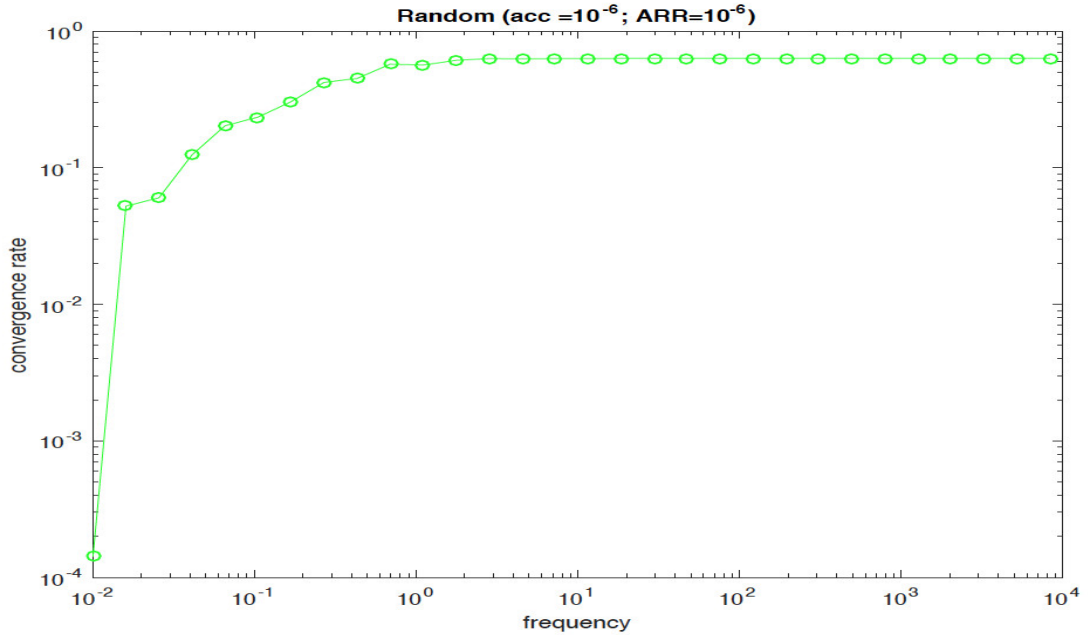


Figure 6.2.3: The convergence behaviour of the solutions of the complex systems of linear equations generated by the interval of frequencies from the random voxel sample is plotted vs the frequency. Only one \mathcal{H} -LU decomposition at the first frequency is computed and it is used to solve all complex systems.

equations in terms of memory and time of the \mathcal{H} -LU decomposition, and convergence parameters. It is evident that the accuracy value and the inversion error decline, whereas the amount of memory and the time to compute the \mathcal{H} -LU decomposition rise. The iteration process reaches the absolute residual reduction in just one or two iteration. The accuracy and the absolute residual reduction selected to test the random voxel sample in the full interval are for both $1.0_{10} - 6$.

For the first numerical test, the \mathcal{H} -LU decomposition is computed for the complex system of linear equations generated at the first frequency of the interval, and using the accuracy and the absolute residual reduction values chosen. This \mathcal{H} -LU decomposition is used to solve all the complex systems generated in the whole interval of frequency. Figure 6.2.3 plots the convergence rate vs frequencies. The results show that the convergence speeds corresponding to the complex systems solved without calculating the \mathcal{H} -LU decomposition are above $5.0_{10} - 2$ and bounded away from one. The smallest convergence value is $1.422_{10} - 4$ which the result of the complex system associated to the first frequency.

The interval of frequency of the random voxel sample is divided to run the second test. The split process of the interval is the same that was used by the sphere sample. The first group of frequency is formed by the first six frequencies in the interval where the \mathcal{H} -LU decomposition is computed using the complex system generated at the first frequency value. These frequencies belong to a half-subinterval which is in green colour. The rest of the frequencies in the interval are grouped in four subin-

tervals. They are represented by the colours blue, red, cyan, and black. The \mathcal{H} -LU decomposition is calculated using the complex system yielded by the frequency at the centre of the subinterval. This sample has four and a half subintervals

Figure 6.2.4a gives the convergence iterations that are produced by the solutions of the complex systems generated by the different groups of frequencies. The convergence rates related to the five frequencies where the \mathcal{H} -LU decompositions were computed are the lowest values. For the frequency ω_1 in the green half-subinterval, the convergence speed is $1.422_{10} - 4$. The convergence values of the complex systems associated to the frequencies ω_7 , ω_{13} , ω_{19} , and ω_{25} are below $1.0_{10} - 5$. On the other hand, the results of convergence iterations of the solutions of the complex systems generated by the frequencies which are not at the centre of the subintervals may be collected in three levels. The first level of convergence rate involved the green half-subinterval, and the blue and the red subintervals with values between $4.0_{10} - 2$ and $5.020_{10} - 1$. The convergence speeds produced in the cyan subinterval of frequencies are in the second level where the range is from $4.042_{10} - 3$ to $6.0_{10} - 2$. The last level corresponds to the black subinterval of frequencies with convergence rates below $9.5_{10} - 3$ and above $4.982_{10} - 4$. There is a convergence value corresponding to the first frequency in the cyan subinterval that could be in the first level of convergence speed. In this case, the level of convergence of the cyan frequencies would change a bit. Similar situation happens with the first two frequencies in the black subinterval, they are in an intersection with the cyan subinterval. Once again, these two convergence values may be in the second level and the third lever makes shorter.

For the third numerical experiment using the random voxel sample, the frequencies in the green half-subinterval and the black subinterval are kept. The frequencies of the blue (ω_7) and the cyan (ω_{19}) subintervals where the \mathcal{H} -LU decompositions are computed for the second test are incorporated in the red subinterval. The frequency at the centre of this subinterval is taken to generated the complex system of linear equations and it is used to calculate the \mathcal{H} -LU decomposition. All the complex systems are solved using this \mathcal{H} -LU decomposition. The results of the convergence iteration vs the frequency are showed in figure 6.2.4b where there are now two levels of convergence rates. One groups the convergence values of the green and red frequencies, and the second level corresponds to the black subinterval.

A comparison of the convergence speed between the results in figures 6.2.4a and 6.2.4b shows that they are similar. However, for the third test only three \mathcal{H} -LU decompositions were computed against five for the second test. As a consequence, the computational cost of the third test is cheaper compared to the second one.

The last artificial sample is the sphere crystal. The first numerical test consisted in splitting the range of frequencies in six subintervals. The frequency at the centre of each subinterval is chosen to generate the complex system of linear equations and it is used to compute the \mathcal{H} -LU decomposition. This is used to solve all the complex systems of equations generated for the frequencies in the rest of the subinterval. The computations were carried out in each subinterval using different values of accuracies and absolute residual reduction to compute the \mathcal{H} -LU decomposition. Table 6.3 shows the results where (n) in the first column represents the frequency number (ω_n)

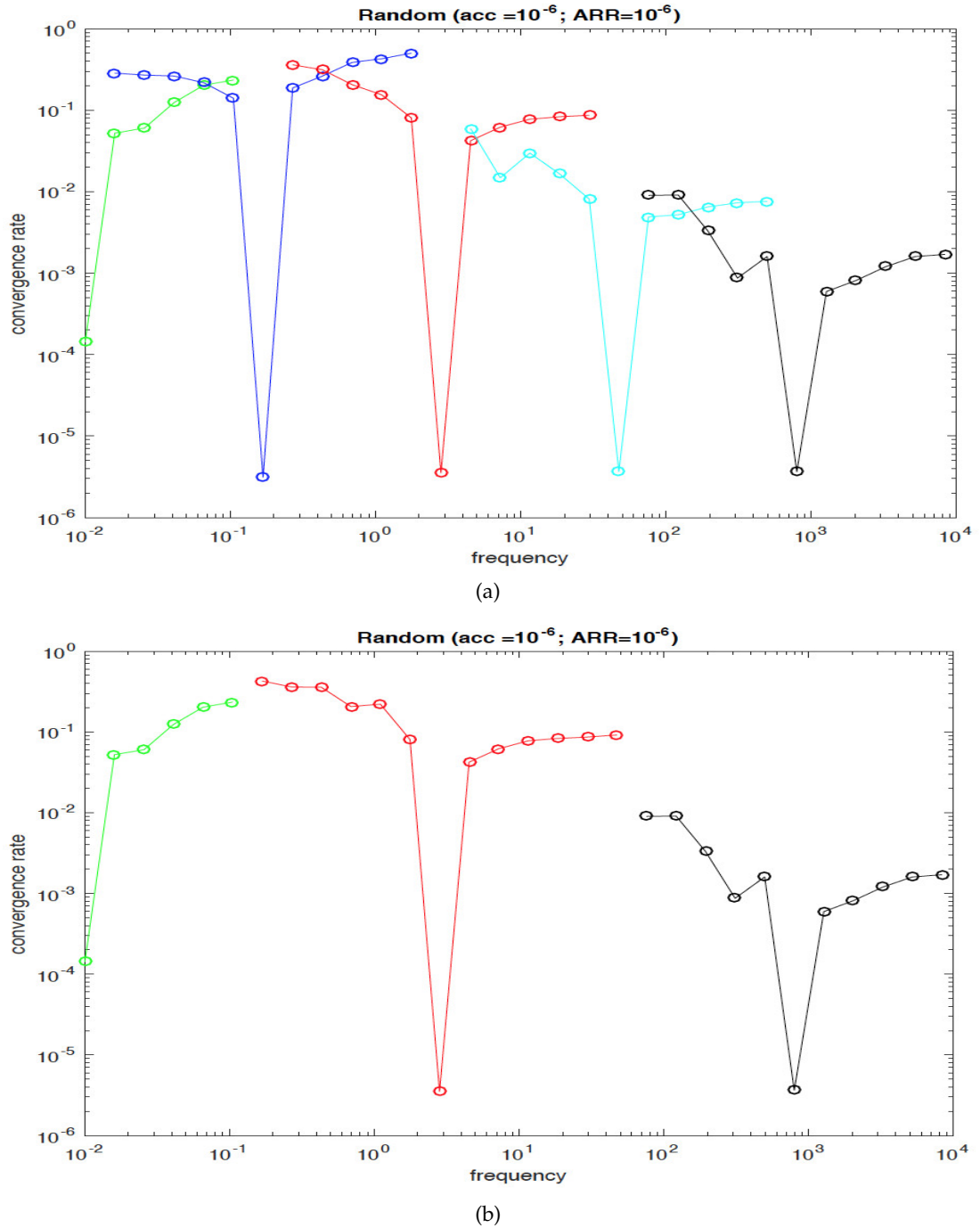


Figure 6.2.4: (a) The convergence of the solutions of the complex systems of equations from the random voxel sample is plotted vs the frequency. The different subintervals are represented by distinct colours. The lowest peaks of convergence rate in each subinterval correspond to the frequency where the \mathcal{H} -LU decomposition is computed. For the rest of the frequencies in each subinterval, its corresponding \mathcal{H} -LU decomposition is used to solve the complex systems of equations. (b) The convergence rate vs the frequency are plotted for three subintervals of frequency.

in the interval of frequency. The accuracy values of the first and third subinterval are a bit smaller (one order of magnitude) than the rest of the subintervals, whereas for the absolute residual reduction there is a difference of two orders of magnitude between the first three subintervals and the last three ones. There are no major contrasts with the \mathcal{H} -LU memory amount and the inversion error. The maximum number of iterations is two and the convergence speed is below of a constant that is less than 1.

Num. of Freq (ω_n)	Accuracy	Absolute Res. Reduction	\mathcal{H} -LU (MB)	Inversion Error	Iters.	Convergence rate
5	$1.0_{10} - 8$	$1.0_{10} - 3$	21492.08	$1.2075_{10} - 2$	2	$1.1778_{10} - 3$
10	$1.0_{10} - 7$	$1.0_{10} - 3$	20613.32	$4.3709_{10} - 2$	2	$1.1855_{10} - 3$
15	$1.0_{10} - 8$	$1.0_{10} - 3$	23699.39	$2.4439_{10} - 3$	1	$8.7949_{10} - 6$
20	$1.0_{10} - 7$	$1.0_{10} - 5$	22954.02	$1.0591_{10} - 2$	2	$8.0408_{10} - 6$
25	$1.0_{10} - 7$	$1.0_{10} - 5$	24491.53	$7.6301_{10} - 3$	1	$1.3406_{10} - 6$
30	$1.0_{10} - 7$	$1.0_{10} - 5$	25919.64	$9.5750_{10} - 3$	1	$9.1746_{10} - 7$

Table 6.3: The solutions of the complex system of linear equations generated from the sphere crystal sample at the frequencies which are at the centre of each subinterval. The frequencies are represented by a number within the interval.

The complex systems of equations generated in each subinterval are solved using its corresponding \mathcal{H} -LU decomposition described in table 6.3. The results of convergence speed vs frequency are shown in figure 6.2.5a. The lowest values plotted in the graph are the convergence rate values in table 6.3. The convergence iterations of the first two subintervals are bigger than the convergence rates of the other four subintervals, with three or four orders of magnitude. However, the convergence rates of the solutions of the complex systems generated by the frequencies which are not at the centre of the subintervals are around the same order of magnitude, except for a piece of the convergence value of the first frequency of the second subinterval. This can be solved using the results of the same frequency in the first subinterval.

In order to investigate an improvement of the results for the first subinterval of frequency, the second test of the sphere crystal is based on solving the same complex systems of equations within the subinterval, including the complex system used to compute the \mathcal{H} -LU decomposition. The computations of two \mathcal{H} -LU decompositions were calculated using the same absolute residual reduction but with smaller accuracy values. They are $1.0_{10} - 9$ and $1.0_{10} - 10$.

Figure 6.2.5b shows the results of the convergence rate vs the frequency in the first subinterval for three values of accuracies. For the complex system generated by the frequency at the centre of the subinterval, the difference of the convergence values using the \mathcal{H} -LU decompositions with different accuracy values of $1.0_{10} - 8$ and $1.0_{10} - 8$ is small, whereas for the accuracy values $1.0_{10} - 8$ and $1.0_{10} - 10$ the difference of their convergence iterations is one order of magnitude. On the other hand, for the complex systems produced by the rest of the frequencies in the subinterval, the convergence rates are similar for the three accuracy values. The computations demonstrate that there is no improvement of the results with respect to the accuracy values. A comparison between the curves of the first subinterval and the curves of

the other subintervals in figure 6.2.5b demonstrates that there are no considerable changes in the results.

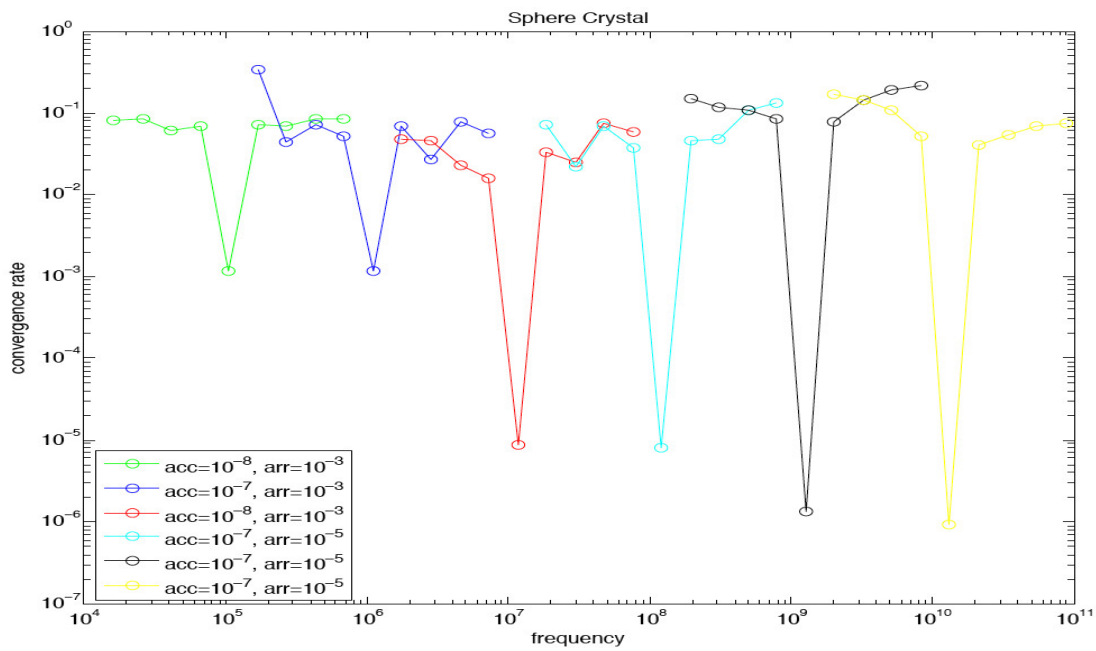
The computation costs of solving the complex systems of equations produced by the sphere crystal sample is improved by dividing the interval of frequency in only three subintervals. The first subinterval is kept as it was used for the first numerical test, i.e., from ω_1 to ω_9 and the \mathcal{H} -LU decomposition was computed at the number of frequency ω_5 . The second and third subintervals are formed by the numbers of frequencies from ω_{10} to ω_{20} , and from ω_{21} to ω_{36} , respectively. The \mathcal{H} -LU decomposition for each subinterval was computed at the frequencies ω_{15} and ω_{25} . Figure 6.2.6 gives the results of the convergence speed vs the frequency. It is noticed that the convergence rates are similar to the results in figure 6.2.5a but the computational cost is cheaper. This is due to the fact that only three \mathcal{H} -LU decompositions were calculated instead of six.

6.3 Rock Samples

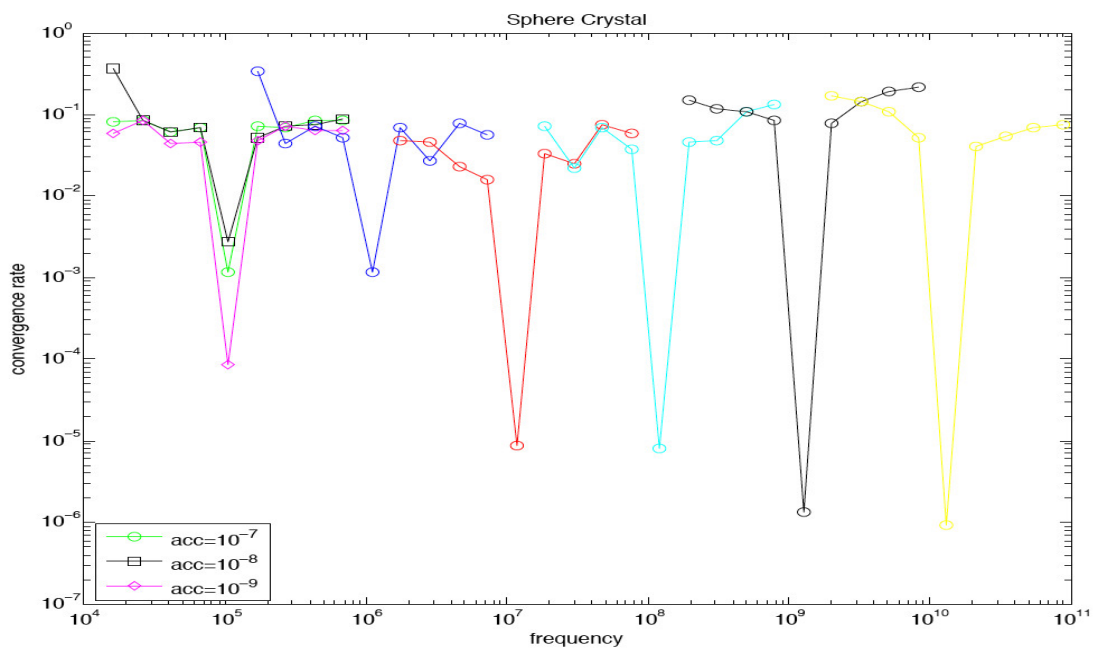
This section is related to the discussion of the results obtained from the rock samples. The description of the samples in terms of type of rocks, the phases in each rock, the physical parameters of each phase, the range of frequency where the electric field is applied, and a 3D image of the sample is found in chapter 3. The rock samples are: Bentheimer, Berea, and a heterogeneous rock. It is important to mention that a frequency within a subinterval is taken to generate the complex system of linear equations that is used to compute the \mathcal{H} -LU decomposition. The procedure applied to choose this frequency is the same that was employed by the computations for the artificial samples. This is the frequency which is at the centre of the subinterval. The \mathcal{H} -LU decomposition calculated in each subinterval is used to solve the complex systems of linear equations produced by frequencies within the corresponding subinterval.

The first numerical test for the heterogeneous rock sample is based on splitting the interval of frequency into one and a half subinterval, and five subintervals. According to the approach mentioned, the \mathcal{H} -LU decomposition is computed using the complex systems of linear equations generated by the frequencies ω_1 , ω_7 , ω_{13} , ω_{19} , ω_{25} , and ω_{31} . These frequencies are within the half subinterval and the five subintervals, respectively. Two values of accuracies were used to calculate the \mathcal{H} -LU decompositions, $1.0_{10} - 6$ and $1.0_{10} - 5$, whereas the absolute residual reduction is $1.0_{10} - 4$ for both computations.

The convergence speed and the frequency are plotted in figure 6.3.1a. The graph shows that the lowest convergence values of the solutions of the complex systems correspond to the frequencies where the \mathcal{H} -LU decompositions were calculated. These results were expected. For the half-subinterval, the convergence speed is below $1.0_{10} - 1$, whereas for the blue subinterval it is close to $1.0_{10} - 4$. The convergence rates for the rest of the subintervals are around $1.0_{10} - 5$. For the frequencies associated to the complex systems which are solved using the \mathcal{H} -LU decomposition,



(a)



(b)

Figure 6.2.5: (a) The convergence iteration of the solutions of the complex systems of equations from the sphere crystal sample is plotted vs the frequency. The different colours represent distinct subintervals of frequency. The lowest peaks of convergence rate in each subinterval are associated to the frequency where the \mathcal{H} -LU decomposition is calculated. For the rest of frequencies in each subinterval, its corresponding \mathcal{H} -LU decomposition is used to solve the complex systems of equations. (b) The different convergence rates are computed in the first subinterval using three accuracy values.

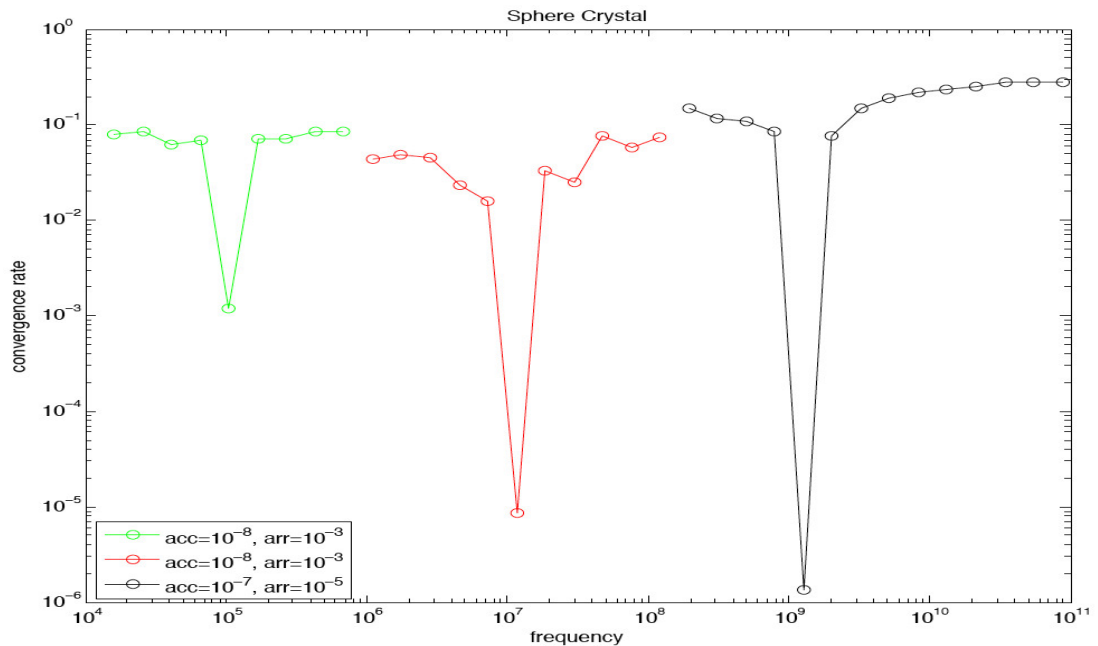


Figure 6.2.6: The convergence speed vs the frequency are plotted for three subintervals of frequency.

the convergence rates of their solutions are bounded away from one and $1.0_{10} - 1$, except for the last subinterval (yellow one) where most of the convergence rates are less than $1.0_{10} - 1$.

The second numerical experiment is carried out to reduce the computational work. The subdivisions of the frequencies made for the first test are rearranged as follows: the half-subinterval (green one) is kept, the blue subinterval and the first half (left side) of the cyan subinterval are grouped with the red subinterval, i.e., the second new subinterval goes from frequency number 4 to frequency number 18. The third new subinterval is formed by the second half (right side including the frequency at the centre) of the cyan subinterval, the black and yellow subintervals, i.e., number of frequency from 19 to 36. The \mathcal{H} -LU decompositions are calculated in the second and third subintervals using the accuracy value and the absolute residual reduction equal $1.0_{10} - 6$ and $1.0_{10} - 4$, respectively.

The results of the convergence rate vs the frequency represented by the colours green, red and yellow are grouped, and they are showed in figure 6.3.1b. The lowest convergence speeds correspond to the frequencies associated to the complex systems where the \mathcal{H} -LU decompositions were calculated. The rest of the convergence values in the graph are below of some fixed constant $c < 1$. These results are similar to the results obtained in the first numerical test. However, the computational cost is much less than the cost of the first test. That is due to the fact that only three \mathcal{H} -LU decompositions were computed instead of six to produce similar results.

The convergence results of the heterogeneous rock are interesting. Even though this sample, and the sphere and random samples have a different complexity in

their structures and distinct numbers of phases, each of them uses one accuracy value and one absolute residual reduction to compute the \mathcal{H} -LU decomposition for the whole interval of frequency. On the other hand, the sphere crystal has a less complex structure than the heterogeneous rock with only two phases. However, the computational work was more demanding due to the fact that it had to use different accuracy values and distinct absolute residual reductions for each subinterval.

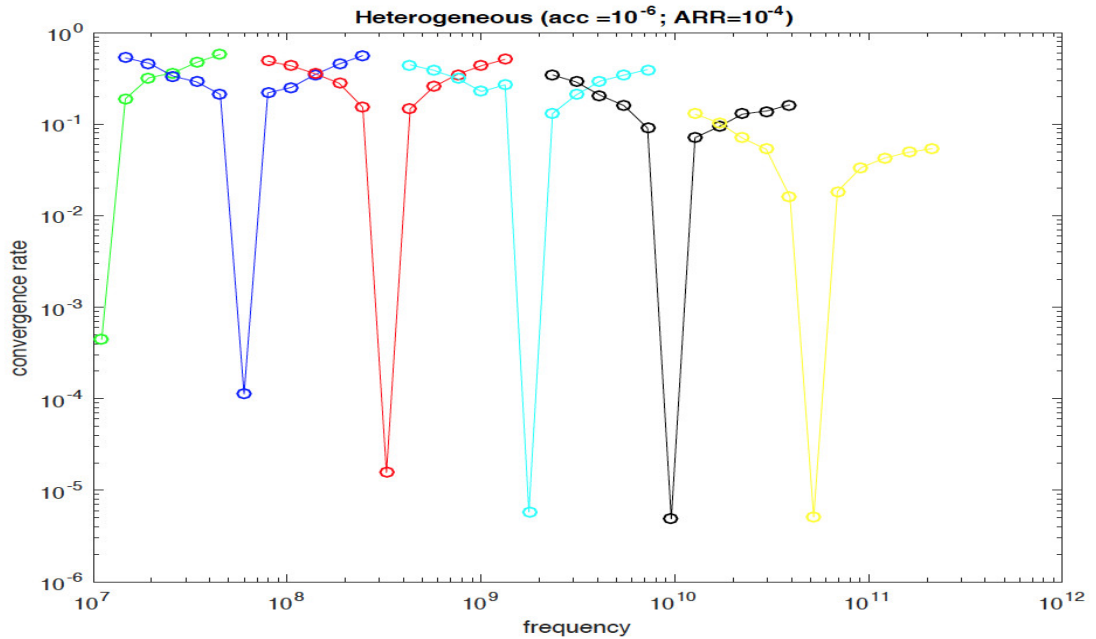
In general the major problem in solving the complex systems of linear equations is the contrast between the coefficients of the different phases in the samples. The frequency is the only parameter that can be modified by the coefficients of the phases in these samples, in particular in the heterogeneous rock. This is why it is important to carry out experiments that can give a better range of frequency.

The Bentheimer rock is the second sample used to be analysed. For the first evaluation, the interval of frequency is divided into six subintervals. They are described by colours: green, blue, red, cyan, yellow, and black. The \mathcal{H} -LU decomposition is calculated in each subinterval using the complex system of linear equations associated to the frequency which is at the centre of the subinterval. The accuracy value and the absolute residual reduction are $1.0_{10} - 6$ and $1.0_{10} - 1$, respectively, for all the subdivisions of frequencies. The \mathcal{H} -LU decomposition is used to solve all the complex systems generated by the frequencies within the subinterval.

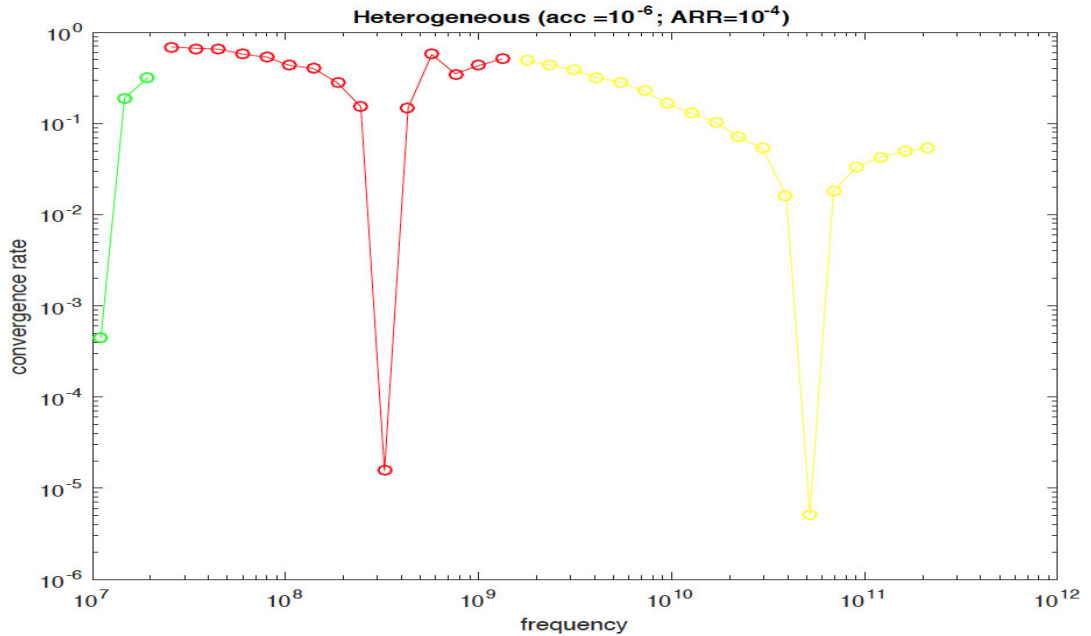
Figure 6.3.2a shows the results of the solutions of the complex systems where the convergence speed and the frequency are plotted. For the frequency associated to the complex systems where \mathcal{H} -LU decompositions were computed, in the first three subintervals the convergence rates are around $1.0_{10} - 2$, whereas for the higher frequencies, i.e., above $1.0_{10}8$, the convergence speeds are between $1.7_{10} - 4$ and $7.7_{10} - 6$. For the rest of the frequencies, the complex systems associated to them were solved using the \mathcal{H} -LU decompositions corresponding to each subinterval. The results show that their convergence rates of the solutions of the complex systems are in a range from $4.0_{10} - 2$ to $6.5_{10} - 1$. The absolute residual reduction used in the test may be decreased taking in each subinterval a smaller accuracy value to compute the \mathcal{H} -LU decomposition, as it was done for the sphere crystal.

The second evaluation of the Bentheimer sample consists in reducing the number of subintervals of frequencies from 6 to 3, and compute the solutions of the complex systems only using three \mathcal{H} -LU decompositions. Figure 6.3.2a shows the subintervals that have to be merged. The frequencies from the first frequency in the green subinterval to the left side of the red subinterval forms the first new subinterval, i.e., it takes the first 15 frequencies. The second new subinterval is constructed taking the frequencies from the centre of the red subinterval to the centre of the cyan subinterval, i.e., from the frequency number 16 to 25. The third new subinterval is just the black subinterval.

The convergence speed vs the frequency are plotted in figure 6.3.2b. The graph shows that the results of the convergence values where the \mathcal{H} -LU decomposition were calculated are similar to the results in figure 6.3.2a. For the frequencies in the green and cyan subintervals, the convergence rates are bounded away from one and $1.0_{10} - 1$. The results of the black subinterval are the same results of the black subin-



(a)



(b)

Figure 6.3.1: (a) The convergence of the solutions of the complex systems of equations from the heterogeneous rock sample is plotted vs the frequency. The different subintervals are represented by distinct colours. The lowest peaks of convergence rate in each subinterval correspond to the frequency where the \mathcal{H} -LU decomposition is computed. For the rest of the frequencies in each subinterval, its corresponding \mathcal{H} -LU decomposition is used to solve the complex systems of equations. (b) The convergence rate vs the frequency are plotted for three subintervals of frequency.

terval in figure 6.3.2a. As the number of calculations of the \mathcal{H} -LU decompositions is reduced, then the computational cost is decreased producing similar results.

The last rock sample that was employed to assess the numerical scheme is Berea. Two evaluations were carried out. For the first one, the interval of frequency is splitting into six subintervals identified by colours. For the second test, the interval is divided into two subintervals, and they are represented by colours as well. Different values of accuracies and absolute residual reductions were used to compute the \mathcal{H} -LU decomposition in each subinterval.

The results of the first test are shown in figure 6.3.3a. The solutions of the complex systems of linear equations generated by the frequencies of each subinterval that were used to calculate the \mathcal{H} -LU decompositions have the lowest values of the convergence speeds. For the first five subintervals, the convergence rates are between $1.0_{10} - 3$ and $6.9_{10} - 5$, whereas for the last subinterval the convergence value reached is $4.989_{10} - 6$ (frequency ω_{25}). For the complex systems associated to the frequencies where the \mathcal{H} -LU decompositions were used to solve them, the convergence speeds of their solutions are bounded away from one, except for the convergence rate at the frequency ω_8 (the blue subinterval) which is below $1.0_{10} - 1$.

In the second evaluation of the Berea sample, the first subinterval is formed by the numbers of frequencies from 1 to 12, i.e., the frequencies in the green and the blue subintervals, and the first five frequencies in the left side of the red subinterval in figure 6.3.3a. The last three subintervals of frequencies in the same figure (the cyan, the black, and the yellow subintervals) are grouped in the second new subinterval. The number of frequency goes from 13 to 29. The values of accuracies and absolute residual reductions to compute the \mathcal{H} -LU decompositions are different for each subinterval. The two \mathcal{H} -LU decompositions are calculated using the complex systems generated by the frequencies ω_7 and ω_{20} in the first and the second subintervals, respectively.

Figure 6.3.3b describes the results of the complex systems of linear equations of the two subintervals of frequencies of the second test. In the first subinterval, the \mathcal{H} -LU decomposition was computed using the complex system produced by the frequency ω_7 and the accuracy value of $1.0_{10} - 9$. The convergence speed of the solution of this complex system is $3.613_{10} - 5$. On the other hand, the frequency ω_7 corresponds to the frequency ω_4 in the blue subinterval of the first test (figure 6.3.3a). The \mathcal{H} -LU decomposition calculated at the frequency ω_4 was employed the accuracy value equal $1.0_{10} - 10$, where the convergence rate of the solution is $1.194_{10} - 4$. It is observed that there is one order of magnitude between the convergence rates of the frequencies ω_7 and ω_4 , even though the accuracy value used at the frequency ω_4 is smaller compared to the one used at the frequency ω_7 . The rest of frequencies in the first subinterval, the convergence speeds are similar to the results in figure 6.3.3a (the first two and an half subintervals). However, the convergence rate associated to the frequency ω_1 is reduced a bit in comparison with the convergence value of the frequency ω_1 in figure 6.3.3a. Moreover, there is a small difference between the results of the convergence of the frequency ω_8 in both figures.

The accuracy value of $1.0_{10} - 7$ was used to calculate the \mathcal{H} -LU decomposition

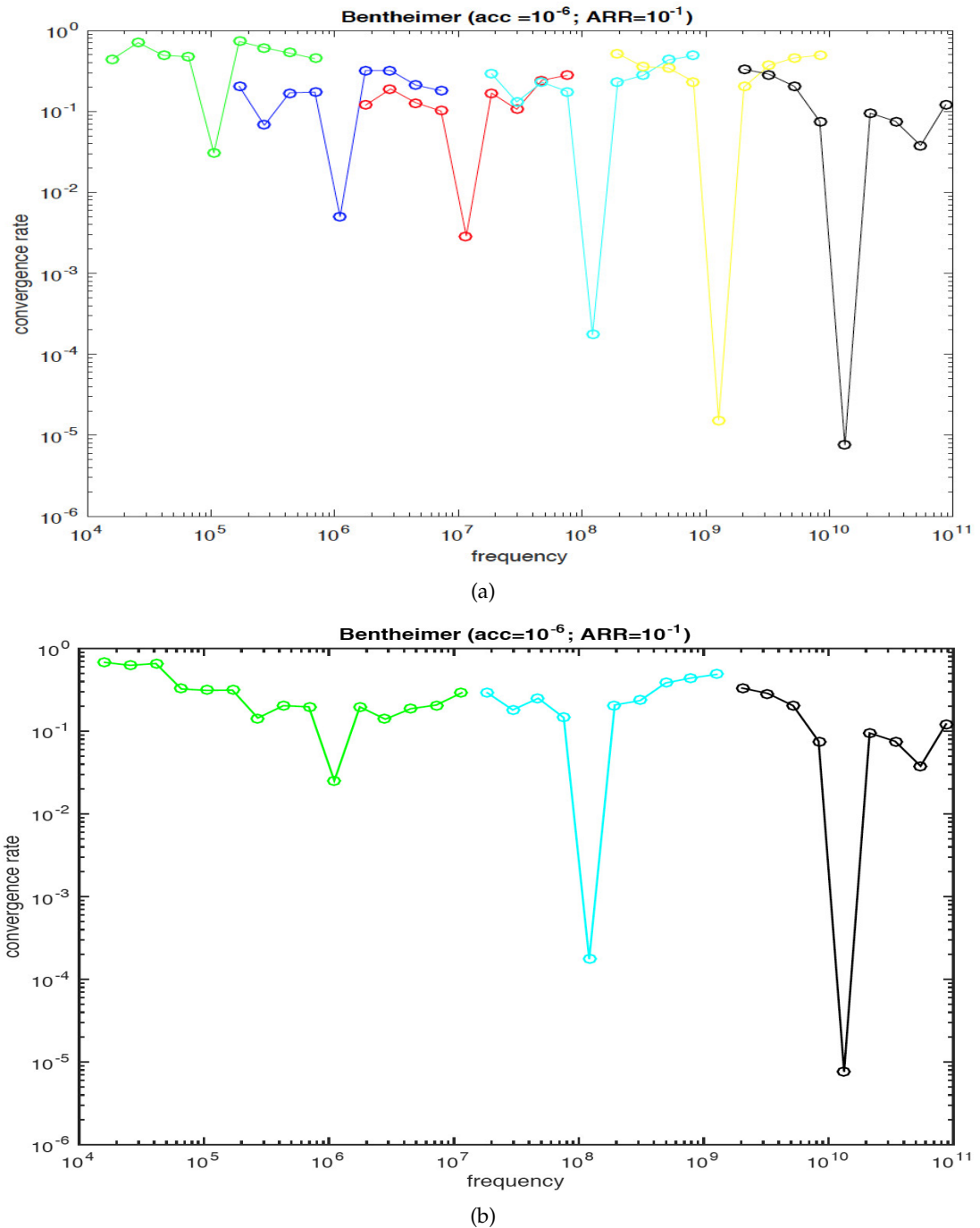
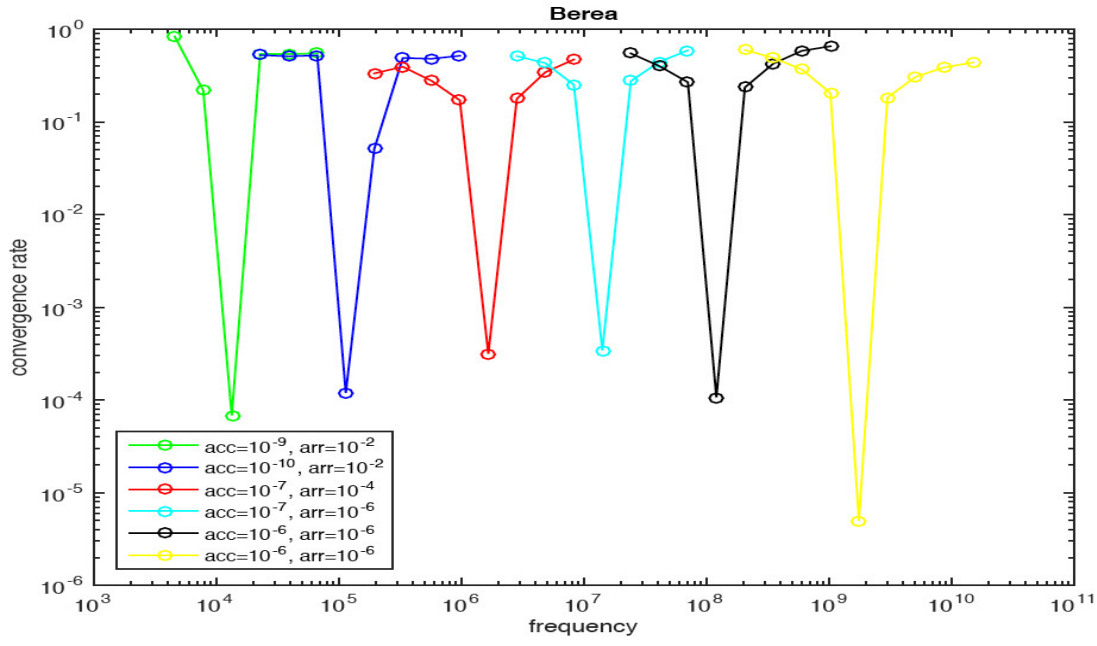
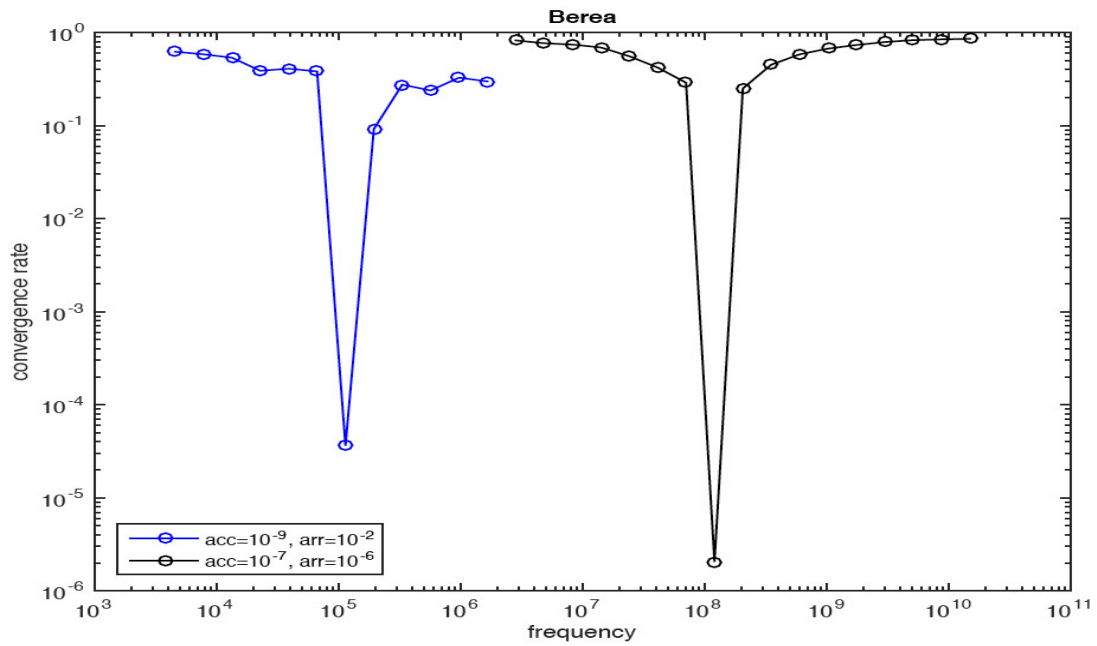


Figure 6.3.2: (a) The convergence of the solutions of the complex systems of equations from the Bentheimer sample is plotted vs the frequency. The different subintervals are represented by distinct colours. The lowest peaks of convergence rate in each subinterval correspond to the frequency where the \mathcal{H} -LU decomposition is computed. For the rest of the frequencies in each subinterval, its corresponding \mathcal{H} -LU decomposition is used to solve the complex systems of equations. (b) The convergence rate vs the frequency are plotted for three subintervals of frequency.



(a)



(b)

Figure 6.3.3: (a) The convergence of the solutions of the complex systems of equations from the Berea sample is plotted vs the frequency. The different subintervals are represented by distinct colours. The lowest peaks of convergence rate in each subinterval correspond to the frequency where the \mathcal{H} -LU decomposition is computed. For the rest of the frequencies in each subinterval, its corresponding \mathcal{H} -LU decomposition is used to solve the complex systems of equations. (b) The convergence rate vs the frequency are plotted for three subintervals of frequency.

employing the complex system generated by the frequencies ω_{20} in the second subinterval in figure 6.3.3b. This frequency ω_{20} is the same one in the black subinterval in figure 6.3.3a associated to the complex system used to compute the \mathcal{H} -LU decomposition, but using $1.0_{10} - 6$ as the accuracy value. The results show that the convergence values in the second test is $2.056_{10} - 6$, while in the first evaluation of the scheme is $1.068_{10} - 4$. The improvement of the two orders of magnitude is due to the fact that the accuracy value is smaller in the second test. For the frequencies in the second subinterval in figure 6.3.3b where the \mathcal{H} -LU decomposition was used to solve the complex systems, the convergence speeds of the solutions of the complex systems produced by these frequencies are below some fixed constant $c < 1$. For the second test of scheme only two \mathcal{H} -LU decompositions were computed in comparison with the six \mathcal{H} -LU decompositions in the first evaluation. This shows that the computational work is reduced in the second test.

Conclusions and Future work

7.1 Conclusions

This research presents two important aspects to compute the complex effective permittivity and the complex effective conductivity of materials represented in a 3D image. The first aspect consists in the proof of the existence and uniqueness of the solution of the second order elliptic partial differential equation with varying complex coefficients that allows to calculate the electric potentials. The second aspect is related to the description, the implementation, and the assessment of the numerical scheme to solve the complex system of linear equations that arises from the partial differential equation. The existence and uniqueness of the solution of the second order elliptic partial differential equation were demonstrated by using basic tools of functional analysis. To satisfy the conditions of the solution, it must be proved that the complex parameter matrix Q of each phase in the material has to be positive definite.

The computation of the solution of complex systems of linear equations could be a difficult task. In the case of a complex system with a Hermitian matrix, there are algorithms relatively powerful to find the solutions due to the properties of this type of matrix. For systems with complex symmetric matrices, the computations of their solutions can be a demanding work. These sort of complex systems were produced in this thesis. They were solved due to the fact that the hierarchical matrices are stored in a hierarchical structure. An important property used by \mathcal{H} -matrices is the connections between the nodes that come from the application of the Finite Element method and the form they are represented in, which is in a graph matrix. In particular, for the \mathcal{H} -LU decomposition the L and U factors are represented by hierarchical triangular matrices.

The numerical scheme developed in this thesis is based on using the \mathcal{H} -matrix technique in combination with the Richardson method and the GMRES algorithm. The \mathcal{H} -LU decomposition is computed using a complex system of linear equations generated by a given frequency within a subinterval. All the complex systems produced in a subinterval of frequency are solved combining its corresponding \mathcal{H} -LU decomposition with the Richardson or with the GMRES algorithm. Even though the computations to solve the complex systems were carried out using both combinations, the majority of the results generated by Richardson and the \mathcal{H} -LU decomposi-

tion could not fulfil the convergence condition (Theorem 4.3.4). The results showed in this study were computed by \mathcal{H} -LU decomposition and the GMRES algorithm, as a good option for solving the complex systems produced by the samples used for the assessment of the scheme. The results are measured in terms of the convergence speed vs the frequency. The convergence rates of the solutions of all the complex systems are below some fixed constant $c < 1$; this is the condition to assure the convergence of the iterative process.

The complexity in solving the complex systems of linear equations generated by the set of samples does not depend on the physical structure of the material represented in a 3D image. The major difficulty comes from the contrast of the physical parameters of the phases in the material. For example, the heterogeneous rock sample has more complex physical structure than the sphere crystal sample. From the results of convergence rates it can be observed that distinct values of accuracies and absolute residual reductions had to be used in the different subintervals to solve the complex systems produced by the sphere crystal sample. On the other hand, the complex systems generated by the heterogeneous rock were solved using only one accuracy value and one absolute residual reduction for all the subintervals.

The most expensive cost of the numerical scheme is given by the calculation of the \mathcal{H} -LU decomposition in each subinterval. The results show that the computational cost is reduced merging the subintervals where less \mathcal{H} -LU decompositions are computed. For instance, the number of subintervals of the samples sphere, sphere crystal, Bentheimer and the heterogeneous rock were decreased from 6 to 3 subintervals, whereas for the random sample from 4 to 2 subintervals. For the berea sample, the reduction of the subintervals was from 6 to 2. Moreover, the results of convergence rates of the solutions of the complex systems generated by the frequencies in the first subinterval of the sphere crystal sample demonstrates that the computations of the \mathcal{H} -LU decompositions with smaller accuracy values are more expensive but the results did not improve. The convergence rates computed by using the different \mathcal{H} -LU decompositions are similar.

The scheme developed in this research has demonstrated to be a robust numerical tool to solve the complex system of linear equations that arises from the second order elliptic partial differential equation with varying complex coefficients. The solution of the complex system represents the electric potentials that are necessary to compute the complex effective permittivity and the complex effective conductivity of the material described in a 3D image. However, there is still a lot of work to do in the computing of these electrical properties to be used to characterise materials.

7.2 Future Work

The major goals of this thesis were to prove the existence and uniqueness of the solution of the second order elliptic differential equation with varying complex coefficients, and the development of the numerical scheme to solve the equation. These goals were fulfilled and they are the basis used to compute the complex effective per-

mittivity and the complex effective conductivity of the materials represented in 3D image. A primary aim of the next step in the research will be focused on calculating the electrical properties from the vector solution to compare the numerical results with analytical models and experimental results from real samples.

For the artificial materials, there are two options: construct materials formed by distinct sort of geometric objects and use a computed tomography scan to produce 3D images, or create 3D images using different types of inclusions. These samples are associated to analytical models such as a package of spheres or 3D image that consists of spheres and ellipsoids with randomly varying the sizes and orientations. The 3D image, the physical parameters of the inclusions, and the numerical scheme are used to compute the complex effective electrical properties. A comparison can be done between the computational and the analytical results. This will be the initial task.

The second phase of the research will be to work with real materials. In this case, it is important to have the necessary equipments or collaborate with a research group to carry out experiments to measure the complex conductivity and the complex permittivity of different materials at distinct range of frequencies. After obtaining the experimental results, the next step will be to use a computed tomography scan and the procedures of filtering and segmentation of images to generate the 3D images of the materials employed in the experiments. The following stage will be to calculate the complex effective electrical properties using the numerical scheme, the 3D images, and boundary conditions. The numerical and experimental results are used to make the comparisons. They will allow to validate how powerful the scheme is developed. This is the real proof for the scheme.

A weakness of the numerical scheme is the size of the 3D image that is used. An additional aspect of the research will be to improve the scheme to use bigger sizes of images. In principle, this is not a major problem, except for the fact that it would have to work with the parallel version of \mathcal{H} -Lib^{Pro} library implemented by Dr. Kriemann. A few minor changes have to be done in the code to incorporate the new version of the library.

The main difficulty in solving the second order elliptic differential equation with varying complex coefficients is given by the disparity between the real value and the imaginary value of the complex parameter matrix Eqs. (1.3.4) and (1.3.5). As part of the future study to improve the numerical scheme, Multi-grid methods will be considered for being implemented to solve complex systems of linear equations. That phase of the research will start implementing Geometric Multi-grid. Moreover, the development will be for sequential and parallel codes. An additional aspect of the investigation will combine the use of Multi-grid, linear methods, and GMRES algorithm.

In general further research will be focused on having experimental results of different types of materials and the implementation of modern efficient algorithms to make the numerical scheme of computing the complex effective electrical properties robust. This computational tool will be used to carry out more calculations, and hence less experiments. In this sense, the major contribution will be to save time and

money.

Graph and Matrix Graph

Let V be a finite non-empty set which is called the vertex set. Let v and w be vertices. The edge from v to w is denoted by $e = (v, w)$. A pair set (V, E) with the property $E \subset V \times V$ is called graph where $v \in V$ and $e \in E$.

A path of a graph (V, E) is defined by a sequence of vertices (v_0, v_1, \dots, v_m) in V with $m \in \mathbb{N}_0$ and the edges $(v_{i-1}, v_i) \in E$ for all $1 \leq i \leq m$. A path is the connection from v_0 to v_m where m is the path length.

For arbitrary vertices $v, w \in V$, a graph is called connected if there is a path from v to w .

In a graph $G = (V, E)$ for all $v, w \in V$ where all the edges $(v, w) \in E$ start at v and end at w is called direct graph. On the other hand, the set of reversed edges is described as $E^T = \{(w, v) : (v, w) \in E\}$. The graph (V, E^T) is named indirect graph. Any direct graph G can be converted into a corresponding indirect $G_{sym} := (V, E) \cup (V, E^T)$. The graph G is called weakly connected if G_{sym} is strongly connected.

Definition A.0.1 (Matrix graph). *Let I be an index set. The matrix graph $G(M)$ corresponding to a matrix $M \in \mathbb{C}^{I \times I}$ is defined by*

$$V = I, \quad E = \{(i, j) \in I \times I : M_{ij} \neq 0\}.$$

Remark A.0.2. *The matrix graph of a symmetric matrix M satisfies $G(M) = G_{sym}(M)$.*

LU Decomposition Procedures

B.1 The Forward and Backward substitution procedures

```

procedure Forward_Substitution( $L, \tau, y, b$ );
if  $\tau \times \tau \in P$  then for  $j := \alpha(\tau)$  to  $\beta(\tau)$  do
    begin  $y_j := b_j$ ; for  $i := j + 1$  to  $\beta(\tau)$  do  $b_i := b_i - L_{ij}y_j$  end
else for  $j := 1$  to  $\#S(\tau)$  do
begin Forward_Substitution( $L, \tau[j], y, b$ );
    for  $i := j + 1$  to  $\#S(\tau)$  do  $b_{|\tau[i]} := b_{|\tau[i]} - L_{|\tau[i] \times \tau[j]} \cdot y_{|\tau[j]}$ 
end;

```

```

procedure Backward_Substitution( $U, \tau, x, y$ );
if  $\tau \times \tau \in P$  then for  $j := \beta(\tau)$  downto  $\alpha(\tau)$  do
    begin  $x_j := y_j / U_{jj}$ 
        for  $i := \alpha(\tau)$  to  $j - 1$  do  $y_i := y_i - U_{ij}x_j$ 
    end
else for  $j := \#S(\tau)$  downto  $1$  do
begin Backward_Substitution( $U, \tau[j], x, y$ );
    for  $i := 1$  to  $j - 1$  do  $y_{|\tau[i]} := y_{|\tau[i]} - U_{|\tau[i] \times \tau[j]} \cdot x_{|\tau[j]}$ 
end;

```

B.2 The Forward matrix and Forward transpose Matrix procedures

```

procedure Forward_M( $L, X, Z, \tau, \sigma$ );
if  $\tau \times \sigma \in P^-$  then                                {column-wise forward substitution}
    for all  $j \in \sigma$  do Forward_Substitution( $L, \tau, X_{\tau,j}, Z_{\tau,j}$ )
else if  $\tau \times \sigma \in P^+$  then
begin                {let  $Z|_{\tau \times \sigma} = AB^H$  according to (5.3.1) with  $A \in \mathbb{C}^{\tau \times \{1, \dots, r\}}$ }
    for  $j = 1$  to  $r$  do Forward_Substitution( $L, \tau, A'_{\tau,j}, A_{\tau,j}$ );
     $X|_{\tau \times \sigma} :=$  rank- $r$  representation by  $A'B^H$ 
end else
for  $i = 1$  to  $\#S(\tau)$  do for  $\sigma' \in S(\sigma)$  do
begin Forward_M( $L, X, Z, \tau[i], \sigma'$ );  { $\ominus, \odot$ : operation with truncation}
    for  $j = i + 1$  to  $\#S(\tau)$  do
         $Z|_{\tau[j] \times \sigma'} := Z|_{\tau[j] \times \sigma'} \ominus L|_{\tau[j] \times \tau[i]} \odot X|_{\tau[i] \times \sigma'}$ 
end;

```

```

procedure ForwardT_M( $U, X, Z, \tau, \sigma$ );
if  $\tau \times \sigma \in P^-$  then
    for all  $i \in \tau$  do Forward_SubstitutionT( $U, \sigma, X_{i,\sigma}, Z_{i,\sigma}$ )
else if  $\tau \times \sigma \in P^+$  then
begin                { $Z|_{\tau \times \sigma} = AB^H$  according to (5.3.1) with  $B \in \mathbb{C}^{\{1, \dots, r\} \times \sigma}$ }
    for  $j = 1$  to  $r$  do Forward_SubstitutionT( $U, \sigma, B'_{i,\sigma}, A_{i,\sigma}$ );
     $X|_{\tau \times \sigma} :=$  rank- $r$  representation by  $AB'^H$ 
end else
for  $j = 1$  to  $\#S(\sigma)$  do for  $\tau' \in S(\tau)$  do
begin ForwardT_M( $U, X, Z, \tau', \sigma[j]$ );
    for  $i = 1$  to  $j - 1$  do
         $Z|_{\tau' \times \sigma[i]} := Z|_{\tau' \times \sigma[i]} \ominus X|_{\tau' \times \sigma[i]} \odot U|_{\sigma[i] \times \sigma[j]}$ 
end;

```

\mathcal{H} -Lib^{Pro} Code

C.1 \mathcal{H} -LU decomposition at one frequency

```

1 #include <stdio.h>
2 #include <stdlib.h>
3 #include <complex.h>
4 #include <math.h>
5 #include <hlib-c.h>
6
7 #include "GlobalVar.h"
8 #include "GetMem.h"
9 #include "PhyscalPar.h"
10 #include "CDFFile.h"
11 #include "FEM.h"
12 #include "CRSFormat.h"
13
14 int main(int argc, char *argv[])
15 {
16     double h;
17     int i;
18     char par_fname[256];
19     char image_fname[256];
20
21     if (argc == 1 || argc == 2)
22     {
23         printf("\nThere is not parameter file and/or image file \n\n");
24         return(EXIT_FAILURE);
25     }
26     else
27     if (argc > 3)
28     {
29         printf("\nFormat: Solver input-file-name image.nc \n\n");
30         return(EXIT_FAILURE);
31     }
32     /* The HLibPro library starts here */
33     hlib_init( & info ); CHECK_INFO;
34     hlib_set_verbosity(3);
35     hlib_set_abs_eps(0.0);
36     acc = hlib_acc_fixed_eps(fileacc);
37     /* Create the 3x3 parameter matrix and the sparse system of equations */
38     Generation3X3ParMatrix( fnphases , phases );
39     BuildMatrixCRSFormat( NNx, NNy, NNz, Csigma );
40     /* Transform of the matrix from CRS format into sparse matrix */
41     Rows = RowPtrComp; Cols = RowPtrComp; Nnz = MatrixComp; Sym = 1;
42     S = hlib_matrix_import_ccrs( Rows, Cols, Nnz, RowPtr, Collnd, Coeffs, Sym, & info );
43     CHECK_INFO;

```

```

44 /* Create memory and set up the vector x */
45 x = hlib_matrix_col_vector(S,&info); CHECK_INFO;
46 ConvertMatrixtoHM();
47 LUFactorisation();
48 IterativeSolvers();
49 FreeHlibStructure();
50 hlib_done(&info); CHECK_INFO;
51 FreeMemoryPhysPar();
52 FreeMemoryCSRFormat();
53 return 0;
54 }
55
56 void ConvertMatrixtoHM()
57 {
58 /* solve with LU decomposition and nested dissection */
59 printf("\n## LU decomposition with algebraic nested dissection\n");
60 printf("converting sparse matrix to H-matrix\n");
61 start = hlib_walltime();
62 ct = hlib_clt_build_alg_nd( S, HLIB_ALG_AUTO, 40, &info ); CHECK_INFO;
63 adm = hlib_admcond_alg( HLIB_ADM_AUTO, 2.0, S,
64                       hlib_clt_perm_i2e( ct, &info ),
65                       hlib_clt_perm_i2e( ct, &info ),
66                       &info ); CHECK_INFO;
67 bct = hlib_bct_build( ct, ct, adm, &info ); CHECK_INFO;
68 A = hlib_matrix_build_sparse( bct, S, acc, &info ); CHECK_INFO;
69 printf("done in %.1f seconds\n", (total_time = hlib_walltime()-start));
70 printf("size of H-matrix = %.2f MB\n", ((double) hlib_matrix_bytesize(A,
71 &info)) / (1024.0 * 1024.0)); CHECK_INFO;
72 }
73
74 void LUFactorisation()
75 {
76 printf("LU factorising H-matrix ... \n");
77 start = hlib_walltime();
78 LU = hlib_matrix_factorise_inv( A, acc, &info ); CHECK_INFO;
79 printf("done in %.1f seconds\n", hlib_walltime()-start);
80 total_time = total_time + (hlib_walltime()-start);
81 printf("size of LU factor = %.2f MB\n", ((double) hlib_matrix_bytesize(
82 A,&info)) / (1024.0 * 1024.0)); CHECK_INFO;
83 /* apply permutations to compare with S */
84 PLU = hlib_perm_linearoperator( hlib_clt_perm_i2e( ct, &info ),
85                                LU,
86                                hlib_clt_perm_e2i( ct, &info ),
87                                &info ); CHECK_INFO;
88 startInvApp = hlib_walltime();
89 printf("\n inversion error = %.4e\n",
90        hlib_linearoperator_norm_inv_approx((hlib_linearoperator_t) S,
91                                           PLU, &info ) ); CHECK_INFO;
92 printf("done in %.1f seconds\n", hlib_walltime()-startInvApp);
93 }
94
95 void IterativeSolvers()
96 {
97 int max_steps = 10;
98 double absolute_residual_reduction;
99 double relative_residual_reduction = 0.0;
100 int initialise_start_value = 1;
101 int use_exact_residual = 1;
102 int GMRESReStart = 20;
103 int ii;
104 double end;
105 const char *solvers[] = {"Linear Iteration", "GMRES"};
106 const char *solvername;

```

```

107 absolute_residual_reduction = AbsResRed;
108 for(ii=0;ii < 2;ii++)
109 {
110     solvername = solvers[ii];
111     solver = NULL;
112     x2 = hlib_vector_copy(x,NULL);
113     if (strcmp(solvername,"Linear Iteration") == 0)
114         solver = hlib_solver_linear_iteration(NULL);
115     else if (strcmp(solvername,"GMRES") == 0)
116         solver = hlib_solver_gmres(GMRESReStart,NULL);
117     hlib_solver_initialise_start_value(solver,initialise_start_value,NULL);
118     hlib_solver_use_exact_residual(solver,use_exact_residual,NULL);
119     hlib_solver_stopcrit(solver,max_steps,absolute_residual_reduction,
120                         relative_residual_reduction,NULL);
121     printf( "\n## solving with %s\n",solvername);
122     start = hlib_walltime();
123     hlib_solver_solve(solver,(hlib_linearoperator_t) S,x2,b,PLU,
124                     & solve_info,&info); CHECK_INFO;
125     end = hlib_walltime();
126     if (solve_info.converged )
127         printf( "converged in %.1f seconds and %u steps with rate %.2e,
128                |r| = %.2e\n",end - start, solve_info.steps,
129                solve_info.conv_rate,solve_info.res_norm );
130     else if (solve_info.failed )
131         printf( "FAILED in %.1f seconds and %u steps with rate %.2e,
132                |r| = %.2e\n",end - start, solve_info.steps,
133                solve_info.conv_rate,solve_info.res_norm );
134     else
135         printf("not converged in %.1f seconds and %u steps\n",
136                end-start,solve_info.steps);
137 }
138 }
139 }

```

Code/HM-OneFreq.c

C.2 \mathcal{H} -LU decomposition for a range of frequencies

```

1 #include <stdio.h>
2 #include <stdlib.h>
3 #include <complex.h>
4 #include <math.h>
5 #include <hlib-c.h>
6
7 int main(int argc, char *argv[])
8 {
9     double h;
10    int i,NW;
11    char par_fname[256];
12    char image_fname[256];
13
14    if (argc == 1 || argc == 2)
15    {
16        printf("\nThere is not parameter file and/or image file \n\n");
17        return(EXIT_FAILURE);
18    }
19    else
20    if (argc > 3)
21    {
22        printf("\nFormat: Solver input-file-name image.nc \n\n");
23        return(EXIT_FAILURE);
24    }

```

```

25 /* The HLibPro library starts here */
26 hlib_init( & info );
27 hlib_set_verbosity(3);
28 hlib_set_abs_eps(0.0);
29 acc = hlib_acc_fixed_eps(fileacc);
30 /* Create the 3x3 parameter matrix and the sparse system of equations */
31 Generation3X3ParMatrix(fnphases, phases);
32 BuildMatrixCRSFormat(NNx,NNy,NNz,Csigma);
33 /* Transform of the matrix from CRS format into sparse matrix */
34 Rows = RowPtrComp; Cols = RowPtrComp; Nnz = MatrixComp; Sym = 1;
35 S = hlib_matrix_import_ccrs(Rows, Cols, Nnz, RowPtr, ColInd, Coeffs,
36                             Sym,&info); CHECK_INFO;
37 /* Create memory and set up the vector x */
38 x = hlib_matrix_col_vector(S,&info);;
39 ConvertMatrixtoHM();
40 LUFactorisation();
41 IterativeSolvers(NWHLU,NWHLU);
42 SetUpMatVecZero();
43 hlib_matrix_free(S,&info); CHECK_INFO;
44 hlib_vector_free(x,&info); CHECK_INFO;
45 for(NW = NWini;NW <= NWfin;NW++)
46     if (NW != NWHLU)
47     {
48         Generation3X3ParMatrix(fnphases, phases ,NW);
49         BuildMatrixCRSFormat(NNx,NNy,NNz,Csigma);
50         Rows = RowPtrComp; Cols = RowPtrComp; Nnz = MatrixComp; Sym = 1;
51         S = hlib_matrix_import_ccrs(Rows, Cols, Nnz, RowPtr, ColInd, Coeffs,
52                                     Sym,&info);CHECK_INFO;
53         x = hlib_matrix_col_vector(S,&info); CHECK_INFO;
54         IterativeSolvers(NW,NWHLU);
55         SetUpMatVecZero();
56         hlib_matrix_free(S,&info); CHECK_INFO;
57         hlib_vector_free(x,&info); CHECK_INFO;
58     }
59 FreeHlibStructure();
60 hlib_done(&info); CHECK_INFO;
61 FreeMemoryPhysPar();
62 FreeMemoryCSRFormat();
63 return 0;
64 }
65
66 void IterativeSolvers(int NW,int NWini)
67 {
68     int max_steps = 50;
69     double absolute_residual_reduction;
70     double relative_residual_reduction = 0.0;
71     int initialise_start_value = 1;
72     int use_exact_residual = 1;
73     int GMRESReStart = 20;
74     unsigned ii;
75     double end;
76     double AuxAbsResRed;
77     const char *solvers[] = {"Linear Iteration","GMRES"};
78     const char *solvername;
79
80     if (NW != NWini)
81         AuxAbsResRed = ComputeAbsResRed(AbsResRed);
82     else
83         AuxAbsResRed = AbsResRed;
84     absolute_residual_reduction = AuxAbsResRed;
85
86     for(ii=0;ii < 2;ii++)
87     {

```

```

88     solvername = solvers[ ii ];
89     solver = NULL;
90     x2      = hlib_vector_copy(x,NULL);
91     if (strcmp(solvername, "Linear Iteration") == 0)
92         solver = hlib_solver_linear_iteration(NULL);
93     else if (strcmp(solvername, "GMRES") == 0)
94         solver = hlib_solver_gmres(GMRESReStart,NULL);
95     hlib_solver_initialise_start_value(solver, initialise_start_value ,NULL);
96     hlib_solver_use_exact_residual(solver , use_exact_residual ,NULL);
97     hlib_solver_stopcrit(solver , max_steps , absolute_residual_reduction ,
98                         relative_residual_reduction ,NULL);
99     printf( "\n## solving with %s\n", solvername);
100    start = hlib_walltime();
101    hlib_solver_solve(solver ,(hlib_linearoperator_t) S,x2,b,PLU,
102                    & solve_info,&info); CHECK_INFO;
103    end=hlib_walltime();
104    if (solve_info.converged )
105        printf( "converged in %.1f seconds and %u steps with rate %.2e,
106              |r| = %.2e\n",end - start , solve_info.steps ,
107              solve_info.conv_rate , solve_info.res_norm );
108    else if (solve_info.failed )
109        printf( "FAILED in %.1f seconds and %u steps with rate %.2e,
110              |r| = %.2e\n",end - start , solve_info.steps ,
111              solve_info.conv_rate , solve_info.res_norm );
112    else
113        printf("not converged in %.1f seconds and %u steps\n",
114              end-start , solve_info.steps);
115    }
116 }

```

Code/HM-IntervalFreq.c

Bibliography

- W. Abdullah, J. S. Buckley, A. Carnegie, J. Edwards, E. Fordham, A. Graue, T. Habashy, H. Hussain, B. Montanan, and M. Ziauddin. Fundamentals of wettability. *Oilfield Review (Schlumberger)*, 19(2):44–61, 2007. (cited on pages xii and 3)
- R. Al-Mjeni, F. Günzel, X. D. Jing, C. A. Grattoni, and R. W. Zimmerman. The influence of clay fraction on the complex impedance of shaly sands. In *Proceedings of the 2002 International Symposium of the Society of Core Analysts*, volume SCA2002-29, pages 1–12. SPWA-Society of Core Analysts, 2002. (cited on page 10)
- H. Andrä, N. Combaret, J. Dvorkin, E. Glatt, J. Han, M. Kabel, Y. Keehm, F. Krzikalla, M. Lee, C. Madonna, M. Marsh, T. Mukerji, E. H. Saenger, R. Sain, N. Saxena, S. Ricker, A. Wiegmann, and X. Zhan. Digital rock physics benchmarks-part II: Computing effective properties. *Computers & Geosciences*, 50:33–43, 2013. (cited on page 2)
- C. H. Arns, M. A. Knackstedt, W. V. Pinczewski, and W. B. Lindquist. Accurate estimation of transport properties from microtomographic images. *Geophysical Research Letters*, 28(17):3361–3364, 2001. (cited on page 2)
- C. H. Arns, M. A. Knackstedt, W. V. Pinczewskiz, and E. J. Garboczi. Computation of linear elastic properties from microtomographic images: Methodology and agreement between theory and experiment. *Geophysics*, 67(5):1396–1405, 2002. (cited on page 2)
- K. Atkinson and W. Han. *Theoretical Numerical Analysis: A Functional Analysis Framework*, volume 39. Springer, Heidelberg, Germany, third edition, 2009. (cited on pages 29 and 30)
- R. Barrett, M. Berry, T.F. Chan, J. Demmel, J. Donato, J. Dongarra, V. Eijkhout, R. Pozo, C. Romine, and H. Van der Vorst. *Templates for the Solution of Linear System: Building Blocks for Iterative Methods*. SIAM, Philadelphia, USA, 1994. (cited on page 14)
- J. G. Berryman. Mixture theories for rock properties. In *Physics and Phase Relations, A Handbook of Physical Constants, Am. Geophys*, pages 205–228, 1995. (cited on page 3)
- P. Bhattacharyya. *Distributions: Generalised Functions with Applications in Sobolev Spaces*. Walter de Gruyter GmbH & Co. KG, Berlin, Germany, 2012. (cited on page 24)

- S. Boggs. *Principles of Sedimentology and Stratigraphy*. Pearson, Essex, England, fifth edition, 2001. (cited on pages xii and 4)
- R. B. Bohn and E. J. Garboczi. User manual for finite element and finite difference programs: A parallel version of nistir-6269. Technical Report NISTIR 6997, National Institute of Standards and Technology, Gaithersburg, USA, June 2003. (cited on page 2)
- N. Bona. Characterization of rock wettability through dielectric measurements. *Revue de L'Institut Francais du Pétrole*, 53(6):80–88, November-December 1998. (cited on page 11)
- N. Bona and E. Rossi. Electrical measurements in the 100 hz to 10 ghz frequency range for efficient rock wettability determination. *SPE journal*, 1(6):1–9, March, 2001. (cited on page 11)
- R. C. Booton. *Computational methods for electromagnetics and microwaves*. Wiley-Interscience, USA, 1992. (cited on page 7)
- C. J. Böttcher. *Theory of Electric Polarisation*. Elsevier Publishing Co., Amsterdam, The Netherlands, first edition, 1952. (cited on page 6)
- D. R. Browning. *Fibres in Chemistry*. The English University Press for the Schools Council, 1974. (cited on page 9)
- T. C. Choy. *Effective Medium Theory: Principles and Applications*. International Series of Monographs on Physics. Oxford University Press, New York, USA, first edition, September 1999. (cited on page 3)
- B. D. Craven. Complex symmetric matrices. *Journal of Australian Mathematics Society*, 10:341–354, 1968. (cited on pages 14 and 15)
- J. Dvorkin. Accuracy and relevance of digital rock results: Successes and failures. Technical report, INGRAIN Digital Rock Physics Lab, Houston, Texas, US, November 2009. URL <http://www.ingrainrocks.com/media/files/file/ValidationJDa.pdf>. (cited on page 2)
- J. Dvorkin, N. Derzhi, E. Diaz, and Q. Fan. Relevance of computational rock physics. *Geophysics*, 76(5):E141–E153, 2011. (cited on page 2)
- P. Øren and S. Bakke. Process based reconstruction of sandstones and prediction of transport properties. *Transport in Porous Media*, 46:311–343, 2002. (cited on page 2)
- R. Freund. Conjugate Gradient-Type methods for linear systems with complex symmetric coefficient matrices. *SIAM Journal on Scientific and Statistical Computing*, 13: 425–448, 1992. (cited on page 14)
- R. Freund and M. Nachtigal. QMR: a quasi-minimal residual method for non-hermitian linear systems. *SIAM Journal on Scientific and Statistical Computing*, 60: 315–339, 1991. (cited on page 14)

- R. Freund, G. Golub, and N. Nachtigal. Iterative solution of linear system. *Acta Numerica*, 1:57–100, 1991. (cited on page 14)
- R. Freund, M. Gutknecht, and N. Nachtigal. An implementation of the look-ahead lanczos algorithm for non-Hermitian matrices. *SIAM Journal on Scientific and Statistical Computing*, 14:137–158, 1993. (cited on page 14)
- C. Gabriel, S. Gabriel, and E. Corthout. The dielectric properties of biological tissues: I. literature survey. *Physics in Medicine and Biology*, 41:2231–2249, 1996a. (cited on page 9)
- C. Gabriel, S. Gabriel, and E. Corthout. The dielectric properties of biological tissues: II. measurements in frequency range 10 Hz to 20 GHz. *Physics in Medicine and Biology*, 41:2251–2269, 1996b. (cited on page 9)
- E. J. Garboczi. The use of computer simulation to interpret and understand electrical measurements. In R. A. Gerhardt, M. A. Alim, and S. R. Taylor, editors, *Electrically based microstructural characterisation II*, volume 500, pages 291–301, Boston MA, USA, 1998a. Material Research Society. (cited on page 2)
- E. J. Garboczi. Finite element and finite difference programs for computing the linear electric and elastic properties of digital images of random materials. Technical Report NISTIR 6269, National Institute of Standards and Technology, Maryland, USA, December 1998b. URL <ftp://ftp.nist.gov/pub/bfrl/garbocz/FDFEMANUAL/>. (cited on pages 2 and 4)
- E. J. Garboczi, D. P. Bentz, and N. S. Martys. Methods of the physics of porous media. In W. Po-Zen, editor, *Digital images and computer modelling*, volume 35, pages 1–41. Academic Press, San Diego, CA, USA, 1999. (cited on page 2)
- A. Garrouch. Effect of wettability and water saturation on the dielectric constant of hydrocarbon rocks. In Society of Petrophysicist and Well-Log Analysts, editors, *SPWLA 41 Annual Logging Symposium*. SPWLA, 2000. Paper NN. (cited on page 11)
- L. Grasedyck, W. Hackbusch, and R. Kriemann. Performance of \mathcal{H} -LU preconditioning for sparse matrices. *Computational Methods in Applied Mathematics*, 8:336–349, 2008. (cited on page 70)
- L. Grasedyck, R. Kriemann, and S. Le Borne. Domain decomposition based \mathcal{H} -LU preconditioning. *Numerische Mathematik*, 112:565–600, 2009. (cited on pages 68 and 70)
- A. Greenbaum. *Iterative Methods for Solving Linear Systems*. SIAM, Philadelphia, USA, 1997. (cited on page 14)
- Y. Gueguen and V. Palciauskas. *Introduction to the Physics of Rocks*. Princeton University Press, USA, 1 edition, 1994. (cited on pages 40 and 41)

- W. Hackbusch. The Panel Clustering Algorithm. In J. R. Whitman, editor, *MAFELAP 1990*, London, 1990. Academic Press. (cited on page 53)
- W. Hackbusch. A sparse matrix arithmetic based on \mathcal{H} -matrices. part I: Introduction to \mathcal{H} -matrices. *Computing*, 62:89–108, 1999. (cited on page 53)
- W. Hackbusch. *Hierarchical Matrices: Algorithms and Analysis*, volume 49 of *Springer Series in Computational Mathematics*. Springer International Publishing, 2015. (cited on pages xii, 56, 58, 60, 61, and 68)
- W. Hackbusch. *Iterative Solution of Large Sparse Systems of Equations*, volume 95 of *Applied Mathematical Sciences*. Springer International Publishing, Switzerland, 2016. (cited on pages xii, 5, 44, 45, 46, 48, 49, 60, 63, 65, 67, 68, 72, and 73)
- W. Hackbusch. *Elliptic Differential Equations: Theory and Numerical Treatment*, volume 18. Springer-Verlag, Germany, second edition, 2017. (cited on pages 22, 23, 24, 26, 29, and 30)
- W. Hackbusch and Z. Novak. On the fast matrix multiplication in the Boundary Element Method by Panel Clustering. *Numerische Mathematik*, 54:453–491, 1989. (cited on page 53)
- A. R. Hippel. *Dielectric Materials and Applications*. Artech House, London, UK, second edition, 1995. (cited on page 6)
- R. A. Horn and Ch. R. Johnson. *Matrix Analysis*. Cambridge University Press, New York, USA, second edition, 2013. (cited on page 15)
- INGRAIN. Examples of validation for porosity, permeability, electrical and elastic properties.f. Technical report, INGRAIN Digital Rock Physics Lab, July 2009. URL http://www.ingrainrocks.com/media/files/user/VALIDATION-V2_July_2009.pdf. (cited on page 2)
- H. Ito and Y. Muraoka. Water transport along textile fibres as measured by and electrical capacitance technique. *Textile Research Journal*, 63(7):414–420, 1993. (cited on page 10)
- K. C. Kao. *Dielectric Phenomena in Solids*. Elsevier Academic Press, San Diego, California, USA, first edition, 2004. (cited on pages 6 and 8)
- Y. Keehm. *Computational Rock Physics: Transport Properties in Porous Media and Applications*. PhD thesis, Department of Geophysics, 2003. (cited on page 2)
- T. C. Kelley. *Iterative Methods for Linear and Nonlinear Equations*. SIAM, Philadelphia, USA, 1995. (cited on page 14)
- R. Kriemann. HLIBpro User Manual 9/2008. Technical Report Technical report 9, Max Planck Institute for Mathematics in the Sciences, 2008a. (cited on page 74)

- R. Kriemann. HLIBpro C Language Interface 10/2008. Technical Report Technical report 10, Max Planck Institute for Mathematics in the Sciences, 2008b. (cited on page 74)
- L. Landau, L. Pitaevskii, and E. Lifshitz. *Electrodynamics of Continuous Media*. Elsevier Butterworth-Heinemann, Oxford, second edition, 1984. (cited on page 7)
- F. Lees. *Lees' Loss Prevention in the Process Industries*. Butterworth-Heinemann: Hazard Identification, Assessment and Control, USA, 3 edition, 2005. (cited on page 40)
- P. Leung and R. Steig. Dielectric constant measurement: A new, rapid method to characterize shale at the wellsite. In *IADC/SPE Drilling conference*, number SPE23887. SPE, New Orleans, LA, USA, 18-21 Feb. 1992. (cited on page 10)
- M. Madadi, A. C. Jones, C. H. Arns, and M. A. Knackstedt. 3D imaging and simulation of elastic properties of porous materials. *Computer Simulations*, pages 65–73, July/August 2009. (cited on page 2)
- D. Makarynskaa, B. Gurevicha, R. Cizc, C. H. Arns, and M. A. Knackstedt. Finite element modelling of the effective elastic properties of partially saturated rocks. *Computers & Geosciences*, 34:647–657, 2008. (cited on page 2)
- N. McCrum, B. Read, and Williams G. *Anelastic and dielectric effects in polymeric solids*. John Wiley & Sons, London, 1967. (cited on page 10)
- B. S. Mitchell. *An Introduction to Materials Engineering and Science*. Wiley-Interscience, New Jersey, 2004. (cited on page 13)
- A. Moliton. *Basic Electromagnetism and Materials*. Springer-Verlag, New York, 2007. (cited on page 13)
- W. Morton and J. Hearle. *Physical properties of textile fibres*. Textile Institute, Manchester, England, third edition, 1993. (cited on page 10)
- M. N. Nabighian, editor. *Electromagnetic Methods in Applied Geophysics: Theory*, volume 1 of *Investigations in geophysics*. Society of Exploration Geophysicists, The United States of America, 1988. Book 3. (cited on page 8)
- S. O. Nelson. Electrical properties of agricultural products-A critical review. *Transactions of the ASAE*, 16:384–400, 1973. (cited on page 9)
- S. O. Nelson. Possibilities for controlling insects with microwaves and lower frequency rf energy. *IEEE Transactions on Microwave Theory and Techniques*, MTT-22: 1303–1305, 1974. (cited on page 9)
- S. O. Nelson. Use of microwave and lower frequency rf energy for improving alfalfa seed germination. *Journal of Microwave Power*, 11:271–277, 1976. (cited on page 9)

- T. Ohlsson. Dielectric properties and microwave processing. In A. G. Medina R. P. Singh, editor, *Food Properties and Computer-aided Engineering of Food Processing System*, pages 73–92. Kluwer Academic Publishers, Amsterdam, 1989. (cited on page 9)
- S. D. Pauer, P. Murugave, and D. M. Lal. Effect of relative humidity and sea level pressure on electrical conductivity of air over indian ocean. *Journal of Geophysical Research*, 114:1–8, 2009. (cited on page 40)
- W. H Press, B. P. Frennery, S. A. Teukolsky, and W. T. Vetterling. *Numerical Recipes*. Cambridge University Press, 1990. (cited on page 5)
- P. Rajinder. *Electromagnetic, Mechanical, and Transport Properties of Composite Materials*. Surfactant Science. CRC Press, Boca Raon, FL., USA, first edition, 2015. (cited on page 8)
- G. M. Richa. *Preservation of Transport Properties Trends: Computational Rock Physics Approach*. PhD thesis, Department of Geophysics, 2010. (cited on page 2)
- C. Ringstad, E. Westphal, A. Mock, M. Hammadi, A. Ratrou, and Z. Z. Kalam. Elastic properties of carbonate reservoir rocks using digital rock physics. In *75th EAGE Conference & Exhibition incorporating SPE EUROPEC 2013*, London, UK, June 2013. (cited on page 2)
- J. L. Rosenholtz and D. T. Smith. The dielectric constant of mineral powders. *The American Mineralogist*, 21:115–120, 1936. (cited on page 41)
- Y. Saad. *Iterative Methods for Sparse Linear Systems*. SIAM, Philadelphia, USA, 2nd edition, 2003. (cited on page 14)
- Y. Saad and M. Schultz. A generalized minimal residual algorithm for solving non-symmetric linear systems. *SIAM Journal on Scientific and Statistical Computing*, 7(3): 856–869, 1986. (cited on page 51)
- R. Sain. *Numerical Simulation of Pore-Scale Heterogeneity and Its Effects on Elastic, Electrical and Transport Properties*. PhD thesis, Department of Geophysics, 2010. (cited on page 2)
- A. Sakellariou, C. H. Arns, A. P. Sheppard, R. M. Sok, H. Holger, A. Limaye, A. C. Jones, T. J. Senden, and M. A. Knackstedt. Developing a virtual materials laboratory. *Materials Today*, 10(12):44–51, 2007. (cited on page 1)
- B. Sareni, A. Krähenbühl, A. Beroual, and C. Brosseau. Effective dielectric constant of random composite materials. *Journal of Applied Physics*, 81(5):2375–2383, 1997. (cited on page 7)
- B. Sareni, A. Krähenbühl, and A. Beroual. Complex effective permittivity of a lossy composite material. *Journal of Applied Physics*, 80(8):4560–4566, 2001. (cited on page 7)

- S. A. Sauter and Ch. Schwab. *Boundary Element Methods*, volume 39. Springer, Heidelberg, Germany, 2011. (cited on page 22)
- B. K. P. Scaife. *Principles of Dielectrics*. Monographs on the Physics and Chemistry of Materials. Oxford University Press, New York, the United States of America, November 1998. Book 45. (cited on pages 6 and 7)
- J. H. Schön. *Physical Properties of Rocks: Fundamentals and Principles of Petrophysics*, volume 65 of *Developments in Petroleum Science*. Pergamon, Hungary, 1 edition, 2004. (cited on pages 40 and 41)
- H. P. Schwan. Electrical properties of tissue and cell suspensions. *Advances in Biological and Medical Physics*, 5:147–209, 1957. (cited on page 9)
- N. Seleznev and A. Boyd. Dielectric mixing laws for fully or partially saturated carbonated rocks. In Society of Petrophysicist and Well-Log Analysts, editors, *SPWLA 45 Annual Logging Symposium*. SPWLA, June 2004. Paper CCC. (cited on page 4)
- N. Seleznev, T. Habashy, A. Boyd, and M. Hizem. Formation properties derived from a multifrequency dielectric measurement. In Society of Petrophysicist and Well-Log Analysts, editors, *SPWLA 47 Annual Logging Symposium*. SPWLA, June 2006. Paper VVV. (cited on page 4)
- P. Sen, C. Scala, and H. Cohen. A self-similar model for sedimentary rocks with applications to the dielectric constant of fused glass beads. *Geophysics*, 46(5):781–795, May, 1981. (cited on page 11)
- P. N. Sen. The dielectric and conductivity response of sedimentary rocks. *SPE*, (9379), 1980. (cited on page 11)
- P. N. Sen. Relation of certain geometrical features to the dielectric anomaly of rocks. *Geophysics*, 46(12):1714–1720, December, 1981. (cited on page 11)
- A. H. Sihvola. *Electromagnetic Mixing Formulas and Applications*, volume 47 of *Electromagnetic Waves Series*. The Institution of Engineering and Technology, London, United Kingdom, first edition, 2008. (cited on page 3)
- J. Spencer-Smith and J. Mathew. 22-A rapid method of determining the moisture content of textiles. *Journal of the Textile Institute Transactions*, 27(9):T219–T228, 1936. (cited on page 10)
- J. Sun, J. Zhao, X. Liu, H. Chen, L. Jiang, and J. Y. Zhang. Pore-scale analysis of electrical properties in thinly bedded rock using digital rock physics. *Journal of Geophysics and Engineering*, 11(5):1–11, 2014. (cited on page 2)
- W. M. Telford, L. P. Geldart, and R. E. Sheriff. *Applied Geophysics*. Cambridge University Press, USA, 2 edition, 1990. (cited on pages 39 and 41)

- E. Tuncer and S. Gubański. Dielectric relaxation in dielectric mixtures: application of finite element method and its comparison with dielectric mixture formulas. *Journal of Applied Physics*, 89(12):8092–8100, 2001. (cited on page 7)
- E. Tuner, Y. V. Serdyuk, and S. M. Gubanski. Comparing dielectric properties of binary composite structures obtained with different calculation tools and methods. In *Electrical Insulation and Dielectric Phenomena 2001*. IEEE, 2001. ISBN 0-7803-7053-8. (cited on page 39)
- A. Wael, J. Buckley, A. Carnegie, and J. Edwards. Fundamentals of wettability. *Oilfield Review*, 19(2):44–61, 2007. (cited on page 11)
- S. O. Wang. Temperature-dependent dielectric properties of selected subtropical and tropical fruit and associated insect pests. *Transactions of the ASAE*, 48(5):1873–1881, 2005. (cited on page 9)
- S. Wei, Y. Yonggan, X. Deqing, and Z. Yunsheng. Finite volume numerical analysis of the thermal property of cellular concrete based on two and three dimensional x-ray computerized tomography images. In *4th International Conference on the Durability of Concrete Structures*, West Lafayette, IN, USA, July 2014. (cited on page 2)
- J. H. Wilkinson. *The Algebraic Eigenvalue Problem*. Oxford University Press, London, first edition, 1965. (cited on page 15)
- D. Wu, J. Chen, and C. Liu. Numerical evaluation of effective dielectric properties of three-dimensional composite materials with arbitrary inclusions using a finite-difference time-domain method. *Journal Applied Physics*, 102:1–8, 2007. (cited on page 39)
- C. Yonghong, C. Xiaolin, K. Wu, S. Wu, C. Yu, and Y. Meng. Modelling and simulation for effective of two-phase disordered composite. *Journal of Applied Physics*, 103(3):2034111–1:034111–8, 2008. (cited on page 7)
- X. Zhan, L. Schwartz, W. Smith, N. Toksöz, and D. Dale Morgan. Pore scale modeling of rock properties and comparison to laboratory measurements. In *2009 SEG Annual Meeting*, Houston, Texas, US, October 2009. Society of Exploration Geophysicists. (cited on page 2)
- K. Zhang, D. Li, and K. Zhang. *Electromagnetic Theory for Microwaves and Optoelectronics*. Springer, New York, 1 edition, 1999. (cited on page 41)