Computational Solutions of Wave Problems

Kristof Cools

October 3, 2024

Contents

Acknowledgements: These lecture notes have been carefully prepared by Prof Hendrik Rogier in the period 2015-2021 and will be maintained and updated to reflect changes in course content and student input and feedback. - Prof Kristof Cools, 2022.

Chapter 1

Mathematical techniques

Generally speaking, the numerical approximation of the solution of linear partial differential equations, such as wave equations, amounts to approximating the unknown continuous solution by a linear combination of simple and known functions. The linear combination coefficients are subsequently obtained by solving a finite system of linear equations. The intention is that, as more and more of these simple functions are used, the true solution will be approximated better and better. This better accuracy comes at the cost of solving a larger linear system.
Clearly, to know what 'better' means, a norm is required to quantify the error. Also, methods for solving systems of linear equations are needed. Since direct methods, such as the LU-decomposition, have already been detailed in other courses, the main focus will be on iterative methods such as the steepest descent and conjugate gradient methods. In this context, the condition number of the linear system is of paramount importance.
In the following, 2-norms will be introduced for finite-dimensional vectors and matrices. Then, the singular value decomposition (SVD) will be introduced and its relation to the matrix norm explained. Subsequently, the condition number is introduced and some iterative solution techniques are presented. Before the matrix system can be solved, its elements must be calculated. This typically entails the evaluation of integrals over multiple dimensions, potentially containing singularities, due to the Green's function kernel. Therefore, we discuss appropriate numerical schemes, such as quadrature rules to numerically evaluate this integral in an efficient and accurate manner. The chapter ends with a general discussion on integral equations.

1.1 Vector and Matrix Norms

1.1.1 Vector Norm

Assume that $\boldsymbol{v} \in \mathbb{C}^n$, where $\mathbb{C}$ are the complex numbers. We expect a norm to be a positive number, only attaining zero for the zero vector:

$$||\boldsymbol{v}|| \geq 0, \; ||\boldsymbol{v}|| = 0 \Leftrightarrow \boldsymbol{v} = 0. \tag{1.1}$$

Moreover, the regular vector operations (multiplication by a scalar and addition) should be continuous operations in the sense that small perturbations to the operand only imply

small perturbations to the result. This continuity is guaranteed by imposing positive scalability and the triangle inequality:

$$||\alpha \boldsymbol{v}|| = |\alpha| \, ||\boldsymbol{v}|| \,, \tag{1.2}$$

$$||\boldsymbol{v} + \boldsymbol{w}|| \leq ||\boldsymbol{v}|| + ||\boldsymbol{w}|| \,. \tag{1.3}$$

A vector space equipped with a norm satisfying (1.1), (1.2) and (1.3) is said to be a normed vector space.
For $\boldsymbol{v} \in \mathbb{C}^n$, a norm known as the Euclidian norm or 2-norm can be defined

$$||\boldsymbol{v}||_2^2 = \sum_{p=1}^{n} |v_p|^2 \,. \tag{1.4}$$

This is the best known example of a norm of a vector, but other norms complying to (1.1), (1.2) and (1.3) exist. The 2-norm is actually a very special example of a norm: it is derived from an inner product

$$(\boldsymbol{x}, \boldsymbol{y}) \equiv \boldsymbol{y}^H \cdot \boldsymbol{x} \quad \rightarrow \quad ||\boldsymbol{v}||_2^2 = (v, v). \tag{1.5}$$

The superscript H means hermitian conjugation, i.e. transposing and replacing every element by its complex conjugate. For real vectors, this boils down to taking the transpose. In general, an inner product must be linear in its second argument, $(\boldsymbol{x}, \boldsymbol{x})$ must be a norm, and finally

$$(\boldsymbol{x}, \boldsymbol{y}) = (\boldsymbol{y}, \boldsymbol{x})^*, \; \forall \boldsymbol{x}, \boldsymbol{y} \in \mathbb{C}^n. \tag{1.6}$$

Check whether the inner product (1.5) satisfies these requirements.
For general norms derived from an inner product $(,)$, (1.1), (1.2) and (1.3) imply the Cauchy inequality

$$|(\boldsymbol{x}, \boldsymbol{y})| \leq ||\boldsymbol{x}|| \, ||\boldsymbol{y}|| \tag{1.7}$$

which states that taking the inner product is a continuous operation. Indeed, with

$$\tau = \frac{(\boldsymbol{x}, \boldsymbol{y})}{(\boldsymbol{y}, \boldsymbol{y})}, \tag{1.8}$$

it follows that

$$0 \leq (\boldsymbol{x} - \tau \boldsymbol{y}, \boldsymbol{x} - \tau \boldsymbol{y}) = (\boldsymbol{x}, \boldsymbol{x}) + |\tau|^2 (\boldsymbol{y}, \boldsymbol{y}) - \tau^* (\boldsymbol{x}, \boldsymbol{y}) - \tau (\boldsymbol{y}, \boldsymbol{x}), \tag{1.9}$$

$$= (\boldsymbol{x}, \boldsymbol{x}) - \frac{1}{(\boldsymbol{y}, \boldsymbol{y})} |(\boldsymbol{x}, \boldsymbol{y})|^2 \,, \tag{1.10}$$

proving (1.7).
In the remainder of this course, the 2-norm will be the most widely used norm. Therefore, if no extra qualification is present, $||\boldsymbol{v}||$ will be assumed to be the 2-norm and $(\boldsymbol{x}, \boldsymbol{y})$ will be assumed to be the Euclidian inner product.

1.1.2 Matrix Norm

Matrices can be interpreted as vectors in an $n \times m$ dimensional space. Hence, their Euclidian norm can be computed. This is called the Frobenius norm, or also the Hilbert-Schmidt norm of a matrix:

$$||\mathsf{A}||_F = \sqrt{\sum_{p=1}^{n} \sum_{q=1}^{m} |A_{p,q}|^2} = \sqrt{\mathrm{Tr}\,(\mathsf{A} \cdot \mathsf{A}^H)}. \tag{1.11}$$

Even though this definition fulfills the constraints (1.1), (1.2) and (1.3), it is not the most natural norm for a matrix. It turns out that in a lot of applications, the so-called matrix 2-norm is much more useful. It is basically defined as the largest possible 2-norm of the output vector of matrix-vector multiplication when restricted to input vectors of norm one:

$$||\mathsf{A}||_2 = \sup_{\boldsymbol{v}\neq 0} \frac{||\mathsf{A}\cdot\boldsymbol{v}||_2}{||\boldsymbol{v}||_2}. \tag{1.12}$$

This definition implies the inequalities

$$||\mathsf{A}\cdot\boldsymbol{v}||_2 \leq ||\boldsymbol{v}||_2\,||\mathsf{A}||_2\,. \tag{1.13}$$

Also

$$||\mathsf{A}\cdot\mathsf{B}||_2 = \sup_{\boldsymbol{v}\neq 0} \frac{||\mathsf{A}\cdot\mathsf{B}\cdot\boldsymbol{v}||_2}{||\boldsymbol{v}||_2} \leq ||\mathsf{A}||_2 \sup_{\boldsymbol{v}\neq 0} \frac{||\mathsf{B}\cdot\boldsymbol{v}||_2}{||\boldsymbol{v}||_2}, \tag{1.14}$$

therefore

$$||\mathsf{A}\cdot\mathsf{B}||_2 \leq ||\mathsf{A}||_2\,||\mathsf{B}||_2\,. \tag{1.15}$$

Again, these are mathematical statements about continuity, being the continuity of the matrix-vector and matrix-matrix multiplication.

1.2 Unitary matrices and the Singular Value Decomposition

1.2.1 Unitary Matrices

A complex matrix U is called unitary if multiplication by U preserves the inner product (1.5), meaning that

$$(\mathsf{U}\cdot\boldsymbol{x}, \mathsf{U}\cdot\boldsymbol{y})_2 = (\boldsymbol{x},\boldsymbol{y})_2, \quad \forall \boldsymbol{x},\boldsymbol{y}. \tag{1.16}$$

An important special case of this is

$$||\mathsf{U}\cdot\boldsymbol{x}||_2 = (\mathsf{U}\cdot\boldsymbol{x}, \mathsf{U}\cdot\boldsymbol{x})_2 = ||\boldsymbol{x}||_2\,. \tag{1.17}$$

Because of the definition of the 2-inner product (1.5), it is quickly showed that

$$\mathsf{U}^H\cdot\mathsf{U} = \mathbb{1}. \tag{1.18}$$

For real matrices, the hermitian conjugate reduces to the transpose.

From the definition of the 2-matrix norm, and with an additional unitary matrix Q, it is clear that

$$||\mathsf{U}\cdot\mathsf{A}\cdot\mathsf{Q}||_2 = \sup_{\boldsymbol{v}\neq 0} \frac{||\mathsf{U}\cdot\mathsf{A}\cdot\mathsf{Q}\cdot\boldsymbol{v}||_2}{||\boldsymbol{v}||_2} = \sup_{\boldsymbol{w}\neq 0} \frac{\left|\left|\mathsf{U}\cdot\mathsf{A}\cdot\mathsf{Q}\cdot\mathsf{Q}^H\cdot\boldsymbol{w}\right|\right|_2}{||\mathsf{Q}^H\cdot\boldsymbol{w}||_2}, \tag{1.19}$$

$$= \sup_{\boldsymbol{w}\neq 0} \frac{||\mathsf{A}\cdot\boldsymbol{w}||_2}{||\boldsymbol{w}||_2} = ||\mathsf{A}||_2\,, \tag{1.20}$$

where the substitution $\boldsymbol{v} = \mathsf{Q}^H\cdot\boldsymbol{w}$ was made. Therefore, the matrix norm is invariant under unitary transformations.

1.2.2 The Singular Value Decomposition

Every $n \times m$ matrix can be decomposed as follows:

$$\mathsf{A} = \mathsf{U}^H \cdot \mathsf{S} \cdot \mathsf{V}, \tag{1.21}$$

where U and V are unitary matrices and S is a diagonal matrix with real positive diagonal elements, usually sorted in decreasing order.

To construct the SVD of A, find the unit vector $\boldsymbol{v}_1$ such that it realizes the supremum in the matrix norm definition (1.12). Then define the unit vector $\boldsymbol{u}_1$ as

$$\sigma_1 \boldsymbol{u}_1 = \mathsf{A} \cdot \boldsymbol{v}_1, \tag{1.22}$$

$$\sigma_1 = ||\mathsf{A}||_2 . \tag{1.23}$$

Now extend $\boldsymbol{v}_1$ and $\boldsymbol{u}_1$ with their orthogonal complement to form the unitary transformation matrices

$$\mathsf{V}_1 = [\boldsymbol{v}_1, \mathsf{V}_1'], \tag{1.24}$$

$$\mathsf{U}_1 = [\boldsymbol{u}_1, \mathsf{U}_1']. \tag{1.25}$$

Applying the transformation to A yields

$$\mathsf{U}_1^H \cdot \mathsf{A} \cdot \mathsf{V}_1 = \begin{bmatrix} \sigma_1 & \boldsymbol{r}_1^H \\ 0 & \mathsf{B}_1 \end{bmatrix}. \tag{1.26}$$

It will now be shown that $\boldsymbol{r}_1 = 0$, such that the matrix in the right hand side becomes diagonal. Then, the SVD can be continued by applying the above line of reasoning on the sub-matrix B_1, and so on.

To show that $\boldsymbol{r}_1 = 0$, define the column vector

$$\boldsymbol{q} = \begin{bmatrix} \sigma_1 \\ \boldsymbol{r}_1 \end{bmatrix}. \tag{1.27}$$

Now, by computing

$$\begin{bmatrix} \sigma_1 & \boldsymbol{r}_1^H \\ 0 & \mathsf{B}_1 \end{bmatrix} \cdot \boldsymbol{q} = \begin{bmatrix} \sigma_1^2 + ||\boldsymbol{r}_1||^2 \\ \mathsf{B}_1 \cdot \boldsymbol{r}_1 \end{bmatrix}, \tag{1.28}$$

it follows that

$$\frac{\left|\left|\mathsf{U}_1^H \cdot \mathsf{A} \cdot \mathsf{V}_1 \cdot \boldsymbol{q}\right|\right|_2}{||\boldsymbol{q}||_2} = \sqrt{\frac{\left[\sigma_1^2 + ||\boldsymbol{r}_1||^2\right]^2 + ||\mathsf{B}_1 \cdot \boldsymbol{r}_1||_2^2}{\sigma_1^2 + ||\boldsymbol{r}_1||^2}} \geq \sqrt{\sigma_1^2 + ||\boldsymbol{r}_1||^2} \tag{1.29}$$

Now, because of (1.20), we know that

$$\left|\left|\mathsf{U}_1^H \cdot \mathsf{A} \cdot \mathsf{V}_1\right|\right|_2 = ||\mathsf{A}||_2 = \sigma_1, \tag{1.30}$$

which contradicts (1.29) unless $\boldsymbol{r}_1 = 0$.

It is important to note that the above construction can be done for *any* complex $n \times m$ matrix. This means that the SVD is more general than the eigenvalue decomposition

$$\mathsf{A} = \mathsf{M}^{-1} \cdot \mathsf{D} \cdot \mathsf{M}, \tag{1.31}$$

which requires that A be diagonalizable. However, for Hermitian matrices with positive eigenvalues, the SVD is identical to the eigenvalue decomposition (possibly up to a reordering of the diagonal elements and phase factors in the eigenvectors). This is because Hermitian matrices, as special cases of Normal matrices, are diagonalizable by means of a unitary similarity transformation.

Moreover, the SVD can be computed using the eigenvalue decomposition of the following two auxiliary matrices

$$\mathsf{A} \cdot \mathsf{A}^H = \mathsf{U}^H \cdot \mathsf{S}^2 \cdot \mathsf{U}, \tag{1.32}$$

$$\mathsf{A}^H \cdot \mathsf{A} = \mathsf{V}^H \cdot \mathsf{S}^2 \cdot \mathsf{V}, \tag{1.33}$$

$$\tag{1.34}$$

Because these two auxiliary matrices are Hermitian, they are always diagonalizable. Therefore, both U and V can be computed by means of the eigenvalue decomposition. The singular values are the square root of the eigenvalues of both matrices, which always yields something real and positive, albeit not strictly positive. It should be stressed, however, that the above algorithm is not the best neither the fastest algorithm for computing the SVD. Many other algorithms have been developed over the years, many of which are more efficient than the algorithm described in the above. Nevertheless, the complexity of computing the SVD of a square matrix is still $\mathcal{O}\left(N^3\right)$.

Finally, it is useful to gain a geometrical interpretation of the SVD for real matrices. For such matrices, the matrices U and V become real, hence orthogonal. Now define a set of unit vectors $\boldsymbol{v}$ that span the entire unit sphere in m dimensions. Then multiplying $\mathsf{A} = \mathsf{U}^H \cdot \mathsf{S} \cdot \mathsf{V}$ by the vectors $\boldsymbol{v}$ can be interpreted as a rotation, followed by a scaling, followed by another rotation. Since the first rotation leaves the unit sphere invariant, only the scaling and the last rotation (multiplication with U^H) influence the final result. Indeed, the scaling squeezes the unit sphere in some directions, while stretching it in other directions. The result is an m-dimensional ellipsoid. The final rotation rotates the axes of the ellipsoid away from the coordinate axes. If the smallest and largest singular values differ much, some of the vectors $\boldsymbol{v}$ end up being large while others end up with a small norm. If that is the case, recovering $\boldsymbol{v}$ from $\mathsf{A} \cdot \boldsymbol{v}$ (i.e. solving the system $\mathsf{A} \cdot \boldsymbol{v} = \boldsymbol{b}$) becomes difficult if the matrix elements are noisy (i.e. they deviate from the exact value by a small amount). This problem is the topic of the next section.

1.3 Solving Systems of Linear Equations

1.3.1 Solution sensitivity/Condition Number

When solving a system of linear equations

$$\mathsf{A} \cdot \boldsymbol{v} = \boldsymbol{b}, \tag{1.35}$$

on a real computer, both the system matrix A and the right hand side $\boldsymbol{b}$ are represented with at least some error. Even when the matrix elements are computed very accurately, the floating point format used by computers to store real numbers incurs an inevitable truncation error that is proportional to the stored number and approximately 10^{16} times smaller. In many cases, other error sources will be larger, but the important message here is that there will always be errors in practice. Therefore, it is important to be able to estimate or bound the error on the solution vector that is caused by the errors on A and $\boldsymbol{b}$.

To do this, perturb A and $\boldsymbol{b}$ with a contribution proportional to ϵ and make the solution $\boldsymbol{v}$ a function of ϵ:

$$[\mathsf{A} + \epsilon\mathsf{E}] \cdot \boldsymbol{v}(\epsilon) = \boldsymbol{b} + \epsilon\boldsymbol{e}. \tag{1.36}$$

Taking the derivative with respect to ϵ allows us to compute the derivative of $\boldsymbol{v}(\epsilon)$ for $\epsilon = 0$:

$$\boldsymbol{v}'(0) = \mathsf{A}^{-1} \cdot [\boldsymbol{e} - \mathsf{E} \cdot \boldsymbol{v}(0)]\,. \tag{1.37}$$

For small ϵ, this derivative can be used to approximate the difference between the solution vector and its error-free value:

$$\Delta\boldsymbol{v} = \boldsymbol{v}(\epsilon) - \boldsymbol{v}(0) \approx \epsilon\boldsymbol{v}'(0). \tag{1.38}$$

Therefore, the relative error on the solution vector can be estimated as

$$\frac{||\Delta\boldsymbol{v}||_2}{||\boldsymbol{v}(0)||_2} \approx \epsilon\frac{||\boldsymbol{v}'(0)||_2}{||\boldsymbol{v}(0)||_2} = \epsilon\frac{\left|\left|\mathsf{A}^{-1} \cdot [\boldsymbol{e} - \mathsf{E} \cdot \boldsymbol{v}(0)]\right|\right|_2}{||\boldsymbol{v}(0)||_2}, \tag{1.39}$$

$$\leq \epsilon\left|\left|\mathsf{A}^{-1}\right|\right|_2 \frac{||\boldsymbol{e}||_2 + ||\mathsf{E}||_2\,||\boldsymbol{v}(0)||_2}{||\boldsymbol{v}(0)||_2}. \tag{1.40}$$

Since

$$||\mathsf{A}||_2\,||\boldsymbol{v}(0)||_2 \geq ||\boldsymbol{b}||_2\,, \tag{1.41}$$

this finally becomes

$$\frac{||\Delta\boldsymbol{v}||_2}{||\boldsymbol{v}(0)||_2} \leq \left|\left|\mathsf{A}^{-1}\right|\right|_2 ||\mathsf{A}||_2 \left[\frac{||\epsilon\boldsymbol{e}||_2}{||\boldsymbol{b}||_2} + \frac{||\epsilon\mathsf{E}||_2}{||\mathsf{A}||_2}\right]. \tag{1.42}$$

Apparently, the quantity $\left|\left|\mathsf{A}^{-1}\right|\right|_2 ||\mathsf{A}||_2$ provides an upper bound on the sensitivity of the solution with respect to perturbations of A and $\boldsymbol{b}$. This quantity is called the 'condition number' $\kappa(\mathsf{A})$. The condition number of a matrix can be computed by means of the SVD:

$$\kappa(\mathsf{A}) = \left|\left|\mathsf{A}^{-1}\right|\right|_2 ||\mathsf{A}||_2 = \frac{\sigma_{\text{max}}}{\sigma_{\text{min}}}. \tag{1.43}$$

where the σ_{max} and σ_{min} are the largest and smallest singular values of the matrix A. **Prove this.**

The geometrical interpretation of the SVD in the previous section agrees well with this analysis: a large difference between the largest and smallest singular value results in hard-to-solve systems of linear equations.

1.3.2 Solution techniques

Linear systems such as

$$\mathsf{A} \cdot \boldsymbol{v} = \boldsymbol{b}, \tag{1.44}$$

can be solved in many ways. The two most widely used 'direct' methods are Gauss-Jordan elimination and LU decomposition. These two techniques both exhibit an

$\mathcal{O}\left(N^3\right)$ complexity, i.e. when doubling the matrix size, the computation becomes eight times slower. However, they have the significant advantage that the runtime is finite and predictable. This is desirable in commercial software packages, which require the guarantee that a solution is found. Generally speaking, LU decomposition is preferred by most people because, of all direct methods, it has the lowest factor in front of the N^3.

In this course, the focus will be on the class of methods known as iterative methods because the direct methods have mostly been covered in earlier courses. Moreover, owing to their finite and predictable runtime, the direct methods can almost always be used as a black-box algorithm, while iterative methods require much more knowledge about the matrix properties to be used successfully. Therefore, goal of this section is to cover some basic iterative solution techniques and provide a high-level overview of more general methods, such that informed decisions can be made for choosing one.

Note: There exist methods that allow the inversion of a dense N by N matrix in less than $\mathcal{O}\left(N^3\right)$ complexity. The first and most notable example of these techniques is the Strassen algorithm. It allows the multiplication of two matrices in $\mathcal{O}\left(N^{\log_2(7)}\right)$ complexity. This algorithm also allows inversion with the same order of complexity. More elaborate versions of this algorithm have been found, such as the Coppersmith-Winograd algorithm, which exhibits an asymptotic complexity of $\mathcal{O}\left(N^{2.37}\right)$.

1.3.3 Complexity of Iterative Solution Methods

In iterative methods, one typically starts out with an initial guess $\boldsymbol{v}_0$ for the solution. Next, this guess is multiplied with the matrix: $\mathsf{A} \cdot \boldsymbol{v}_0$. The difference between this result and the desired one, i.e. $\boldsymbol{b}$, is subsequently used to construct a new and hopefully improved guess $\boldsymbol{v}_1$. This process is continued until convergence is reached up to an a priori set error. It is clear that the complexity of such methods is proportional to the number of iterations P and to the complexity $C(N)$ of the matrix-vector multiplication, resulting in a complexity $\mathcal{O}\left(PC(N)\right)$. Usually, a distinction is made between dense and sparse matrices:

- Dense matrix: the number of nonzero matrix entries is $\mathcal{O}\left(N^2\right)$. This implies $C(N) = N^2$,
- Sparse matrix: the number of nonzero matrix entries is $\mathcal{O}\left(N\right)$. This implies $C(N) = N$.

Of course, all kinds of intermediary complexities $C(N)$ are possible, but these two types are the most widely encountered. It is clear that, for a comparable number of iterations P, sparse matrix systems will be more amenable to iterative solution.

However, sometimes, a dense matrix has enough structure to still allow multiplication in $\mathcal{O}\left(N\right)$ or $\mathcal{O}\left(N \log N\right)$ operations. Prominent examples of such matrices are the matrices arising from the discretization of integral equations, or the matrices associated with the discrete Fourier transform. The fast multiplication algorithms for these matrices are the fast multipole method and the fast Fourier transform respectively. Algorithms of this type have already enabled the iterative solution of dense linear systems with tens or even hundred *billion* unknowns. **Convince yourself** that multiplying matrices of such a size with a vector using the $\mathcal{O}\left(N^2\right)$ algorithm would be completely impossible with current computer hardware.

1.3.4 The Steepest Descent Method

One of the most basic iterative solution methods is the steepest descent method. For real matrices A, it requires that A is symmetric, i.e. $\mathsf{A} = \mathsf{A}^T$ and positive definite[1] (SPD):

$$\frac{\boldsymbol{x}^T \cdot \mathsf{A} \cdot \boldsymbol{x}}{\boldsymbol{x}^T \cdot \boldsymbol{x}} > 0, \; \forall \boldsymbol{x} \neq \mathbf{0}. \tag{1.45}$$

The generalization to complex matrices will be briefly discussed at the end of this section.

The steepest descent method is inspired by the fact that the solution of (1.44), i.e. $\mathsf{A}^{-1} \cdot \boldsymbol{b}$, is the minimum of the functional

$$\phi(\boldsymbol{x}) = \frac{1}{2}\boldsymbol{x}^T \cdot \mathsf{A} \cdot \boldsymbol{x} - \boldsymbol{x}^T \cdot \boldsymbol{b}. \tag{1.46}$$

Clearly, $\phi(\boldsymbol{x})$ is a quadratic function of the components of $\boldsymbol{x}$. To see where its minimum is located, compute the gradient:

$$\frac{\partial}{\partial x_n}\phi(\boldsymbol{x}) = \frac{1}{2}\boldsymbol{e}_n^T \cdot \mathsf{A} \cdot \boldsymbol{x} + \frac{1}{2}\boldsymbol{x}^T \cdot \mathsf{A} \cdot \boldsymbol{e}_n - \boldsymbol{e}_n^T \cdot \boldsymbol{b}, \tag{1.47}$$

$$= \boldsymbol{e}_n^T \cdot [\mathsf{A} \cdot \boldsymbol{x} - \boldsymbol{b}]. \tag{1.48}$$

Now, the minimum is located where all derivatives are zero, i.e. when

$$\mathsf{A} \cdot \boldsymbol{x} = \boldsymbol{b}. \tag{1.49}$$

This proves our earlier claim. The value of the functional at the minimum is equal to

$$\phi_{\text{min}} = \phi(\mathsf{A}^{-1} \cdot \boldsymbol{b}) = -\frac{1}{2}\boldsymbol{b}^T \cdot \mathsf{A}^{-1} \cdot \boldsymbol{b}. \tag{1.50}$$

From this point onwards, the steepest descent method is quite simple: start at a given point $\boldsymbol{x}_0 \in \mathbb{R}_n$ and move 'downhill', based on the knowledge of the gradient (1.48). In particular, the gradient at $\boldsymbol{x}_0$ is given by

$$\boldsymbol{\nabla}\phi(\boldsymbol{x})|_{\boldsymbol{x}=\boldsymbol{x}_0} = \mathsf{A} \cdot \boldsymbol{x}_0 - \boldsymbol{b} = -\boldsymbol{r}_0. \tag{1.51}$$

Here, $\boldsymbol{r}_0$ is the residual associated to $\boldsymbol{x}_0$. The vector in the left hand side is the direction in which $\phi(\boldsymbol{x})$ increases the fastest (in a neighborhood of $\boldsymbol{x}_0$). Taking the negative of this vector yields the steepest descent direction, hence the name of the method.

To find a new and better tentative solution $\boldsymbol{x}_1$, the steepest descent method looks in the direction of steepest descent for the value α for which $\phi(\boldsymbol{x}_0 + \alpha\boldsymbol{r}_0)$ is minimal. Expansion by means of the definition of the functional yields

$$\phi(\boldsymbol{x}_0 + \alpha\boldsymbol{r}_0) = \phi(\boldsymbol{x}_0) - \alpha\boldsymbol{r}_0^T \cdot \boldsymbol{r}_0 + \frac{1}{2}\alpha^2\boldsymbol{r}_0^T \cdot \mathsf{A} \cdot \boldsymbol{r}_0. \tag{1.52}$$

The minimum of this parabola is given by

$$\alpha = \frac{\boldsymbol{r}_0^T \cdot \boldsymbol{r}_0}{\boldsymbol{r}_0^T \cdot \mathsf{A} \cdot \boldsymbol{r}_0}, \tag{1.53}$$

[1] Remember that a matrix is positive definite if and only if all its eigenvalues are positive and if and only if all its leading principal minors are positive. A positive definite matrix can also serve as the kernel matrix of an inner product, i.e. $(\boldsymbol{x}, \boldsymbol{y})_{\mathsf{A}} \equiv \boldsymbol{y}^H \cdot \mathsf{A} \cdot \boldsymbol{x}$ satisfies all properties to be an inner product.

which leads to the new tentative solution

$$\boldsymbol{x}_1 = \boldsymbol{x}_0 + \alpha \boldsymbol{r}_0. \tag{1.54}$$

It is clear that obtaining this new guess for the solution requires exactly one matrix-vector multiplication, i.e. $\mathsf{A} \cdot \boldsymbol{r}_0$. This result can be reused to compute $\boldsymbol{r}_1$ in the next iteration. The new value of the functional is equal to

$$\phi(\boldsymbol{x}_1) = \phi(\boldsymbol{x}_0) - \frac{1}{2} \frac{\left[\boldsymbol{r}_0^T \cdot \boldsymbol{r}_0\right]^2}{\boldsymbol{r}_0^T \cdot \mathsf{A} \cdot \boldsymbol{r}_0}. \tag{1.55}$$

The positivity of the second term means that the functional is always decreasing (this requires A to be positive definite). Now it will be investigated how quickly it is decreasing.

Convergence Speed

To estimate how quickly the functional is decreasing, consider the fraction

$$\frac{\phi(\boldsymbol{x}_1) - \phi_{\text{min}}}{\phi(\boldsymbol{x}_0) - \phi_{\text{min}}} = 1 - \frac{\frac{1}{2} \frac{\left[\boldsymbol{r}_0^T \cdot \boldsymbol{r}_0\right]^2}{\boldsymbol{r}_0^T \cdot \mathsf{A} \cdot \boldsymbol{r}_0}}{\phi(\boldsymbol{x}_0) - \phi_{\text{min}}}. \tag{1.56}$$

After some simplifications, the second term becomes

$$\frac{\left[\boldsymbol{r}_0^T \cdot \boldsymbol{r}_0\right]^2}{\left[\boldsymbol{r}_0^T \cdot \mathsf{A}^{-1} \cdot \boldsymbol{r}_0\right] \left[\boldsymbol{r}_0^T \cdot \mathsf{A} \cdot \boldsymbol{r}_0\right]}. \tag{1.57}$$

To find the worst-case convergence speed, we need to find the smallest possible value this term can attain. This is equivalent to finding the supremum of

$$\left[\boldsymbol{r}_0^T \cdot \mathsf{A}^{-1} \cdot \boldsymbol{r}_0\right] \left[\boldsymbol{r}_0^T \cdot \mathsf{A} \cdot \boldsymbol{r}_0\right], \tag{1.58}$$

under the constraint that $\boldsymbol{r}_0^T \cdot \boldsymbol{r}_0 = 1$. Clearly, the upper bound $\kappa(\mathsf{A})$ holds (**prove this**), which already gives some information regarding the convergence. However, it is possible to improve this upper bound considerably.

To do this, it is useful to compute the eigenvalue decomposition $\mathsf{A} = \mathsf{U}^T \cdot \mathsf{S} \cdot \mathsf{U}$, where the diagonal elements of the diagonal matrix S are given by s_n. This yields

$$\left[\boldsymbol{r}_0^T \cdot \mathsf{A}^{-1} \cdot \boldsymbol{r}_0\right] \left[\boldsymbol{r}_0^T \cdot \mathsf{A} \cdot \boldsymbol{r}_0\right] = \left[\sum_n r_n^2 s_n\right] \left[\sum_n \frac{r_n^2}{s_n}\right]. \tag{1.59}$$

Subsequently, the optimization problem is solved using a Lagrange multiplier, by finding the points where the derivatives of the function

$$F(\boldsymbol{r}_0, \lambda) = \left[\sum_n r_n^2 s_n\right] \left[\sum_n \frac{r_n^2}{s_n}\right] + \lambda \left(\left[\sum_n r_n^2\right] - 1 \right). \tag{1.60}$$

become zero. Here r_n is the nth element in the vector $\mathsf{U} \cdot \boldsymbol{r}_0$. Setting to zero the derivative with respect to r_p yields

$$\frac{\partial}{\partial r_p} F(\boldsymbol{r}_0, \lambda) = 2 r_p s_p \left[\sum_n \frac{r_n^2}{s_n}\right] + 2 \frac{r_p}{s_p} \left[\sum_n r_n^2 s_n\right] + 2 r_p \lambda = 0. \tag{1.61}$$

Therefore, if both r_p and r_q are different from zero, the following is found:

$$(s_p - s_q)\left[\sum_n \frac{r_n^2}{s_n}\right] = \frac{s_p - s_q}{s_p s_q}\left[\sum_n r_n^2 s_n\right]. \tag{1.62}$$

Assuming that all the eigenvalues are distinct, this means that

$$\left[\sum_n r_n^2 s_n\right] = s_p s_q \left[\sum_n \frac{r_n^2}{s_n}\right], \ \forall p, q. \tag{1.63}$$

Clearly, the product $s_p s_q$ needs to be independent of p and q for this to hold. This is impossible in general. In fact, it implies that only two of the r_n can be different from zero at the extremum (**Exercise: check that having three nonzero r_n leads to a contradiction with the distinctiveness of the eigenvalues.**).

Finally, knowing that only r_p and r_q are different from zero, the extremum can be readily computed because $r_q = \sqrt{1 - r_p^2}$ and

$$\left[\sum_n r_n^2 s_n\right]\left[\sum_n \frac{r_n^2}{s_n}\right] = \left[r_p^2 s_p + (1 - r_p^2) s_q\right]\left[\frac{r_p^2}{s_p} + \frac{(1 - r_p^2)}{s_q}\right]. \tag{1.64}$$

The supremum lies at the point $r_p = \frac{1}{\sqrt{2}}$, which leads to

$$\left[\sum_n r_n^2 s_n\right]\left[\sum_n \frac{r_n^2}{s_n}\right] = \frac{1}{4}\frac{(\frac{s_p}{s_q} + 1)^2}{\frac{s_p}{s_q}}. \tag{1.65}$$

Now the only thing left is to optimize with respect to p and q. Since the function

$$\frac{(x+1)^2}{x}, \tag{1.66}$$

is invariant under the substitution $x \to \frac{1}{x}$, we can safely replace the optimization problem with maximizing $\frac{s_p}{s_q}$. Obviously, the maximal value is the condition number $\kappa(\mathsf{A})$, which leads to the result:

$$\sup_{||\boldsymbol{r}_0||=1}\left[\boldsymbol{r}_0^T \cdot \mathsf{A}^{-1} \cdot \boldsymbol{r}_0\right]\left[\boldsymbol{r}_0^T \cdot \mathsf{A} \cdot \boldsymbol{r}_0\right] = \frac{1}{4}\frac{(\kappa(\mathsf{A}) + 1)^2}{\kappa(\mathsf{A})} \tag{1.67}$$

Therefore, the convergence rate

$$\frac{\phi(\boldsymbol{x}_1) - \phi_{\min}}{\phi(\boldsymbol{x}_0) - \phi_{\min}} \leq 1 - 4\frac{\kappa(\mathsf{A})}{(\kappa(\mathsf{A}) + 1)^2} = \left(\frac{\kappa(\mathsf{A}) - 1}{\kappa(\mathsf{A}) + 1}\right)^2. \tag{1.68}$$

is finally obtained. **Convince yourself that this result still holds when the eigenvalues are not all distinct.**

Number of iterations

In practice, one often wishes to reduce

$$R_k = \frac{\phi(\boldsymbol{x}_k) - \phi_{\min}}{\phi(\boldsymbol{x}_0) - \phi_{\min}} \tag{1.69}$$

below a given error threshold ϵ. Here $\boldsymbol{x}_k$ is the kth tentative solution. To estimated how many iterations are required, the convergence result (1.68) can be used to show that

$$R_k \leq \left(\frac{\kappa(\mathsf{A}) - 1}{\kappa(\mathsf{A}) + 1} \right)^{2k}. \tag{1.70}$$

Therefore, whenever k is such that

$$\left(\frac{\kappa(\mathsf{A}) - 1}{\kappa(\mathsf{A}) + 1} \right)^{2k} \leq \epsilon, \tag{1.71}$$

the computation will surely have converged. For large $\kappa(\mathsf{A})$, this means that convergence is assured if

$$k \geq \frac{1}{2} \frac{\ln(\epsilon)}{\ln\left(\frac{\kappa(\mathsf{A})-1}{\kappa(\mathsf{A})+1} \right)} \approx -\frac{1}{4} \kappa(\mathsf{A}) \ln(\epsilon), \tag{1.72}$$

i.e. the number of iterations is proportional to the condition number. It is worthwhile to point out that the lower bound for k is a sufficient condition for convergence, but not a necessary one. For example, if $\boldsymbol{x}_0$ and $\boldsymbol{b}$ are both proportional to an eigenvector of A, the steepest descent method converges in exactly one iteration! In practice, though, this almost never happens and (1.72) provides a good indication of how many iterations will be necessary.

1.3.5 The Conjugate Gradient Method

The steepest descent method, as explained in the previous subsection, has a strict recipe for generating the new direction in which to minimize the functional $\phi(\boldsymbol{x})$. However, there is no reason to believe that this recipe is the best possible, and one might replace it by any other method for generating new directions (general search directions method). Of course, it is clear that certain methods will be inferior to the steepest descent choice. An example of such a bad method is when the search directions are parallel to the level curves of the functional (see slides for a graphical representation).

However, it is now also known that better options than the steepest descent method exist. The conjugate gradient (CG) method, which will be introduced in this subsection, is such a method. The CG method is motivated by the desire to make the subsequent search directions linearly independent from each other, such that stepping in a certain direction and then backtracking along that same direction is avoided. The fact that this is possible without jeopardizing the computational complexity is one of the surprises hidden in the CG method!

The CG method can be derived as follows: Assume we have a (not necessarily good) initial guess $\boldsymbol{x}_0$ for the solution, and a set of search directions $\boldsymbol{p}_n$, $\forall n \in [1, k]$. Now consider the optimization problem

$$\inf_{\boldsymbol{y}} \phi(\boldsymbol{x}_0 + \sum_{n=1}^{k} [\boldsymbol{y}]_n \, \boldsymbol{p}_n), \tag{1.73}$$

which has solution $\boldsymbol{y}_k$. The vector

$$\boldsymbol{x}_k = \boldsymbol{x}_0 + \sum_{n=1}^{k} [\boldsymbol{y}_k]_n \, \boldsymbol{p}_n. \tag{1.74}$$

realizes the minimum of the functional, restricted to the space $\boldsymbol{x}_0 + \text{span}\{\boldsymbol{p}_1, \ldots, \boldsymbol{p}_k\}$. Computing the vector $\boldsymbol{y}_k$ is a difficult task if nothing is known about the search directions $\boldsymbol{p}_n$. However, if A-conjugacy (i.e. A-orthogonality) is imposed, i.e.

$$\boldsymbol{p}_n^T \cdot \mathsf{A} \cdot \boldsymbol{p}_m = 0, \ \forall n \neq m, \tag{1.75}$$

then the minimization problem can be recursively solved. Indeed:

$$\begin{aligned}
\inf_{\boldsymbol{y}} \phi(\boldsymbol{x}_0 + \sum_{n=1}^{k} [\boldsymbol{y}]_n \boldsymbol{p}_n) &= \inf_{\boldsymbol{y}} \Bigg[\phi(\boldsymbol{x}_0 + \sum_{n=1}^{k-1} [\boldsymbol{y}]_n \boldsymbol{p}_n) \\
&\qquad + [\boldsymbol{y}]_k (\boldsymbol{x}_0 + \sum_{n=1}^{k-1} [\boldsymbol{y}]_n \boldsymbol{p}_n)^T \cdot \mathsf{A} \cdot \boldsymbol{p}_k + \phi([\boldsymbol{y}]_k \boldsymbol{p}_k) \Bigg], \\
&= \inf_{\boldsymbol{y}} \left[\phi(\boldsymbol{x}_0 + \sum_{n=1}^{k-1} [\boldsymbol{y}]_n \boldsymbol{p}_n) + [\boldsymbol{y}]_k \boldsymbol{x}_0^T \cdot \mathsf{A} \cdot \boldsymbol{p}_k + \phi([\boldsymbol{y}]_k \boldsymbol{p}_k) \right], \\
&= \inf_{\boldsymbol{y}} \phi(\boldsymbol{x}_0 + \sum_{n=1}^{k-1} [\boldsymbol{y}]_n \boldsymbol{p}_n) \\
&\qquad + \inf_{[\boldsymbol{y}]_k} \left[[\boldsymbol{y}]_k^2 \frac{1}{2} \boldsymbol{p}_k^T \cdot \mathsf{A} \cdot \boldsymbol{p}_k - [\boldsymbol{y}]_k \boldsymbol{p}_k^T \cdot \boldsymbol{r}_0 \right].
\end{aligned}$$

Clearly, the first term results from optimization over the space $\boldsymbol{x}_0 + \text{span}\{\boldsymbol{p}_1, \ldots, \boldsymbol{p}_{k-1}\}$, and can, therefore, be considered to have been performed in the previous iteration. This also means that, if A-conjugacy is imposed, the first $k-1$ elements of $\boldsymbol{y}_k$ stay the same in iteration k. The second minimization problem is one-dimensional and can be preformed with minimal effort:

$$[\boldsymbol{y}]_k = \frac{\boldsymbol{p}_k^T \cdot \boldsymbol{r}_0}{\boldsymbol{p}_k^T \cdot \mathsf{A} \cdot \boldsymbol{p}_k}. \tag{1.76}$$

Now, there is only one ingredient left to add before completing the algorithm: a method for generating A-conjugate search directions. One way to do this would be to generate k random vectors and to A-orthogonalize them using, for example, the Gram-Schmidt process. However, this would require $\mathcal{O}\left(k^2 N\right)$ (where N is the size of the matrix A) operations, which turns out not to be optimal. Indeed, it has been proved that there always exists a search direction of the form

$$\boldsymbol{p}_k = \boldsymbol{r}_{k-1} + \beta_k \boldsymbol{p}_{k-1}, \tag{1.77}$$

that is A-orthogonal to all previous search directions. Here, $\boldsymbol{r}_{k-1} = \boldsymbol{b} - \mathsf{A} \cdot \boldsymbol{x}_{k-1}$. By A-conjugacy, the factor β_k can be determined to be

$$\beta_k = -\frac{\boldsymbol{p}_{k-1}^T \cdot \mathsf{A} \cdot \boldsymbol{r}_{k-1}}{\boldsymbol{p}_{k-1}^T \cdot \mathsf{A} \cdot \boldsymbol{p}_{k-1}}. \tag{1.78}$$

This final piece of knowledge completes the algorithm and leads to an iterative solution method that, at any time, minimizes the functional over a set of A-conjugate search directions.

Convergence Speed

It is clear that this algorithm terminates after N iterations, because the minimization is then performed over the entire vector space. However, in practice, this turns out not always to be true. This is due to the fact that the computer representation of real numbers is not exact, but a truncated approximation. The small errors, made in each step, accumulate and lead to the loss of exact convergence after N iterations.

Fortunately, a rigorous analysis of these errors has also shown that the convergence behaviour still obeys the estimate

$$\frac{\phi(\boldsymbol{x}_k) - \phi_{\text{min}}}{\phi(\boldsymbol{x}_0) - \phi_{\text{min}}} \leq \left(\frac{\sqrt{\kappa(\mathsf{A})} - 1}{\sqrt{\kappa(\mathsf{A})} + 1} \right)^{2k}. \tag{1.79}$$

This behavior is very similar to that of the steepest descent method, but with the square root $\sqrt{\kappa(\mathsf{A})}$ instead of $\kappa(\mathsf{A})$. Therefore, one expects the number of iterations to be around (or smaller than)

$$k \approx -\frac{1}{4}\sqrt{\kappa(\mathsf{A})}\ln(\epsilon). \tag{1.80}$$

Clearly, this is a large improvement over the steepest descent method.

Computational Complexity Examples

To determine which solution technique (direct of iterative) is best used to solve a given problem, it is often instructive to look at how the complexity of the solution technique scales with the size N of the matrix. For example, take the finite element matrix that is generated by discretizing the one-dimensional Helmholtz equation (see the FEM part of the lectures). When the mesh that is used for the discretization is approximately uniform, it can be shown that the condition number scales as $\mathcal{O}\left(N^2\right)$. Moreover, the matrix is sparse, which means that the complexity of a matrix-vector multiplication is $\mathcal{O}\left(N\right)$. Then the iterative solution complexities are

- steepest descent: $\mathcal{O}\left(N^3\right)$,
- conjugate gradient: $\mathcal{O}\left(N^2\right)$.

Clearly, conjugate gradient has a better asymptotic complexity than steepest descent. It is useful to keep in mind that is need not be the case. Indeed, if we were to solve a type of system for which the condition number scales as $\mathcal{O}\left(1\right)$, the two methods would have approximately the same run time.

When solving the one-dimensional wave equation by means of LU decomposition (without frontal methods), the complexity is $\mathcal{O}\left(N^3\right)$. Therefore, conjugate gradient still outperforms LU decomposition. However, if a frontal method is used, the complexity of the direct method is $\mathcal{O}\left(N\right)$, which means that the frontal method is better than both steepest descent and conjugate gradient.

The situation changes for higher-dimensional Helmholtz equations. For example, in two dimensions, the condition number scales as $\mathcal{O}\left(N\right)$, which means that the complexity of the conjugate gradient solution is $\mathcal{O}\left(N^{\frac{3}{2}}\right)$. On the other hand, frontal methods become less efficient in higher dimensions, having an $\mathcal{O}\left(N^{\frac{3}{2}}\right)$ complexity as well.

For three-dimensional problems, the iterative solution exhibits a better asymptotic complexity than the frontal method. From the above, it is clear that the choice of solution strategy heavily depends on many characteristics of the problem being solved.

1.3.6 GMRES

The conjugate gradient method has a simple interpetation in terms of a convex minimisation problem and is very efficient because new search directions can be computed in a very small number of floating point operations. Unfortunately, it can only be shown to work for symmetric positive definite systems. In next section we discuss some strategies for using GC in a broader context.

Nevertheless, it is good to have at our disposal an iterative solver that works for any solvable linear system and that is guaranteed to converge. In this section we will discuss the GMRES method, which offers these advantages. Even though in general GMRES solution could require more operations than solution by GC, for well conditioned systems, where the number of iterations required for iterative solvers to reach a solution remains small and independent of the system dimension, the complexity of these two methods is equal.

From the exposition below it will be clear that GMRES can be considered a deterministic solution methodology for linear system (just like solution by LU factorisation, or the pivot method, or Gaussian substitution). It can also be considered an iterative method because often times high quality approximations of the solution are already reached within the first few iterations.

As always, we are interested in finding the solution to the linear system $Ax = b$ where A can now be any square system. It is clear that solving the system is equivalent to solving the minimisation problem

$$\min_{x \in \mathbb{C}^n} \|Ax - b\| \tag{1.81}$$

We will solve this problem by subsequently considering the minimisation over the so-called Krylov subspaces

$$\mathcal{K}_k = \text{span}\{b, Ab, A^2b, ..., A^{k-1}b\} \tag{1.82}$$

Let $K_k \in \mathbb{C}^{n \times k}$ be the matrix build from the columns $(A^s b)_{s=0}^{k-1}$, i.e.

$$K_k = \left(b|Ab|...|A^{k-1}b\right) \tag{1.83}$$

A generic vector in $\mathcal{K}_k$ can be written as $K_k y_k$ with $y_k \in \mathbb{C}^k$ and so our minimisation problems can be written as

$$\min_{y_k \in \mathbb{C}^k} \|AK_k y - b\| \tag{1.84}$$

It turns out we can recursively build orthonormal bases $(q_k)_{s=1}^k$ for $\mathcal{K}_k$ through a process called Arnoldi iteration (closely related to Gram-Schmidt orthogonalisation). More precisely we can find the factorisation

$$A = Q_n H_n Q_n^* \tag{1.85}$$

or $AQ_n = Q_n H_n$, with Q_n a unitary matrix whose first k columns provide the sought after bases for $\mathcal{K}_k$ and H_n a matrix in so-called Hessenberg form, meaning that all its

entries under the first sub-diagonal are zero, i.e. $h_{pq} = 0$ for $p > q + 1$. Because of this special structure, we can write the recursive identities

$$AQ_k = Q_{k+1}\bar{H}_k \tag{1.86}$$

with $\bar{H}_k = (h_{pq,})_{p=1,q=1}^{p=k+1,q=k} \in \mathbb{C}^{(k+1)\times k}$ the (non-square) sub-matrix from H_n retaining entries in the first $k + 1$ rows and the first k columns. Since the columns of Q_k hold an orthonormal basis for $\mathcal{K}_k$, the minimisation problem (1.84) can just as well be written as

$$\min_{c_k \in \mathbb{C}^k} \|AQ_k c_k - b\| \tag{1.87}$$

or, making use of the partial Arnoldi factorisation (1.86),

$$\min_{c_k \in \mathbb{C}^k} \|Q_{k+1}\bar{H}_k c_k - b\| \tag{1.88}$$

Making use of the unitarity of Q_{k+1} and the choice $q_1 = b/\|b\|$, we finally get

$$\min_{c_k \in \mathbb{C}^k} \|\bar{H}_k c_k - \|b\| e_1^{(k+1)}\| \tag{1.89}$$

with $e_1^{(k+1)} = (1, 0, 0, ..., 0) \in \mathbb{C}^{k+1}$. This least squares problem can be solved by QR decomposition. It is important to note that given the QR decomposition for $\bar{H}_{k-1}$, available from the previous iteration, the QR decomposition of $\bar{H}_k$ can be efficiently computed (details follow below).

The Arnoldi iteration

Assuming a decomposition of the form $A = Q_n\bar{H}_nQ_n$ exists, with Q_k unitary and $\bar{H}_k$ Hessenberg, a algorithm follows by writing down the that the left and right hand sides in the equivalent expressions $AQ_k = Q_{k+1}\bar{H}_k$ match up column per column. In particular, it should hold that

$$Aq_k = \sum_{s=1}^{k+1} h_{s,k}q_s = \sum_{s=1}^{k} h_{s,k}q_s + h_{k+1,k}q_{k+1} \tag{1.90}$$

This is equivalent to finding the decomposition of Aq_k in the orthonormal basis $(q_s)_{s=1}^{k+1}$ where the subset $(q_s)_{s=1}^{k}$ has already been constructed during a previous step. This leads to the following algorithm:

$q_{k+1} \leftarrow Aq_k$
for $s \in \{1, 2, ..., k\}$ **do**
  $h_{s,k} \leftarrow q_s^* q_{k+1}$
  $q_{k+1} \leftarrow q_{k+1} - h_{s,k}q_s$
end for
$h_{k+1,k} \leftarrow \sqrt{q_{k+1}^* q_{k+1}}$
$q_{k+1} \leftarrow q_{k-1}/h_{k+1,k}$

Note that this algorithm is essentially Gram-Schmidt; we are subtracting contributions from Aq_k along q_s with $q = 1, .., k$ and normalising the remainder.

Recursive QR decomposition of $\bar{H}_k$

The least squares problem (1.89) can be solved by QR decomposition of $\bar{H}_k = P_k \bar{R}_k$ where P_k is a unitary $k \times k$ matrix and $\bar{R}$ is a $k \times n$ upper triangular matrix, whose action on a vector can be efficiently computed through back substitution. Fortunately, also P_k can be efficiently constructed, assuming we already have at our disposal the QR decomposition $\bar{H}_{k-1} = P_{k-1}\bar{R}_{k-1}$. We know $\bar{H}_k$ is in Hessenberg form and thus can be written as

$$\bar{H}_k = \begin{pmatrix} \bar{H}_{k-1} & h_k \\ 0 & h_{k+1,k} \end{pmatrix} \tag{1.91}$$

with $h_k \in \mathbb{C}^k$ and $h_{k+1,k} \in \mathbb{C}$. We now use the induction hypothesis $\bar{H}_{k-1} = P_{k-1}\bar{R}_{k-1}$ with $P_{k-1} \in \mathbb{C}^{k\times k}$ unitary to write

$$\bar{H}_k = \begin{pmatrix} P_{k-1}\bar{R}_{k-1} & P_{k-1}P_{k-1}^* h_k \\ 0 & h_{k+1,k} \end{pmatrix} = \begin{pmatrix} P_{k-1} & \\ & 1 \end{pmatrix} \begin{pmatrix} \bar{R}_{k-1} & P_{k-1}^* h_k \\ 0 & h_{k+1,k} \end{pmatrix} \tag{1.92}$$

Remember that $\bar{R}_{k-1}$ is of size $k \times (k-1)$ and is upper triangular, and thus can be written in the format

$$\bar{R}_{k-1} = \begin{pmatrix} R_{k-1} \\ 0 \end{pmatrix} \tag{1.93}$$

with $R_{k-1} \in \mathbb{C}^{(k-1)\times(k-1)}$ square and upper triangular. Introducing the notation $P_{k-1}^* h_k =: (d_k, a)^T$ with $d_k \in \mathbb{C}^{k-1}$, $a \in \mathbb{C}$ and $b = h_{k+1,k}$, we can rewrite (1.92) as

$$\bar{H}_k = \begin{pmatrix} P_{k-1} & \\ & 1 \end{pmatrix} \begin{pmatrix} R_{k-1} & d_k \\ 0 & a \\ 0 & b \end{pmatrix} \tag{1.94}$$

The rightmost factor can be reduced to upper triangular form if we can find a unitary transformation that zeros the entry in position $(k+1, k)$ without causing fill-in in the (strictly) lower triangle. A Givens rotation does the trick:

$$\bar{H}_k = \begin{pmatrix} P_{k-1} & \\ & 1 \end{pmatrix} \begin{pmatrix} I_{k-1} & & \\ & c & -s \\ & s & c \end{pmatrix} \begin{pmatrix} R_{k-1} & d_k \\ 0 & \sqrt{a^2+b^2} \\ 0 & 0 \end{pmatrix}, \tag{1.95}$$

where $c = a/\sqrt{a^2+b^2}$ and $s = b/\sqrt{a^2+b^2}$. From this representation we can read off P_k and $\bar{R}_k$:

$$P_k = \begin{pmatrix} P_{k-1} & \\ & 1 \end{pmatrix} \begin{pmatrix} I_{k-1} & & \\ & c & -s \\ & s & c \end{pmatrix}, \tag{1.96}$$

$$\bar{R}_k = \begin{pmatrix} R_{k-1} & d_k \\ 0 & \sqrt{a^2+b^2} \\ 0 & 0 \end{pmatrix}. \tag{1.97}$$

Least squares solution in $\mathcal{K}_k$

Now that we have the QR decomposition of $\bar{H}_k$, it is straightforward to find the solution to the minimisation problem (1.89), which is the equivalent least square solution to the over-determined linear system $\bar{H}_k c_k = \|b\| e_1$:

$$\begin{pmatrix} R_k \\ 0 \end{pmatrix} c_k = \|b\| P_k^* e_1^{(k+1)}. \tag{1.98}$$

This system can be solved for c_k by backward substitution.

1.3.7 General systems

The conjugate gradient method can only be applied to symmetric positive definite matrices. When this constraint is not satisfied, or when the matrix is not square, additional tricks are needed. For example, when a system is overdetermined (has more equations than unknowns), one can apply the conjugate gradient method to solve

$$\mathsf{A}^T \cdot \mathsf{A} \cdot \boldsymbol{x} = \mathsf{A}^T \cdot \boldsymbol{b}. \tag{1.99}$$

This system is square and symmetric. Also, the matrix $\mathsf{A}^T \cdot \mathsf{A}$ is positive definite if the rank of A is equal to the number of unknowns. The solution to this system is the least-square solution, i.e. the vector $\boldsymbol{x}$ that leads to the smallest 2-norm error on the right hand side:

$$||\mathsf{A} \cdot \boldsymbol{x} - \boldsymbol{b}||^2 = -\boldsymbol{x}^T \cdot \mathsf{A}^T \cdot \boldsymbol{b} + \boldsymbol{b}^T \cdot \boldsymbol{b} + \boldsymbol{x}^T \cdot \mathsf{A}^T \cdot \mathsf{A} \cdot \boldsymbol{x} - \boldsymbol{b}^T \cdot \mathsf{A} \cdot \boldsymbol{x}. \tag{1.100}$$

Equating the derivatives to zero yields equation (1.99).

Alternatively, if the system is underdetermined, the matrix $\mathsf{A}^T \cdot \mathsf{A}$ has singular values that are zero. Therefore, a different system must be constructed. Solve

$$\mathsf{A} \cdot \mathsf{A}^T \cdot \boldsymbol{y} = \boldsymbol{b}, \tag{1.101}$$

and compute $\boldsymbol{x} = \mathsf{A}^T \cdot \boldsymbol{y}$ after obtaining $\boldsymbol{y}$. This solution is one of an infinitely large set of solutions, since the solution to the original problem is not unique. However, obviously, the solution to (1.101) is uniquely determined. The condition that makes it unique is the fact that this solution is the minimum norm solution, i.e. the solution to $\mathsf{A} \cdot \boldsymbol{x} = \boldsymbol{b}$ that has the smallest 2-norm. This is proved by means of a Lagrange multiplier argument, to minimize $||\boldsymbol{x}||^2$ under constraint of satisfying the linear system:

$$\text{minimize } \frac{1}{2} ||\boldsymbol{x}||^2 - \sum_n \lambda_n \hat{\boldsymbol{e}}_n^T \cdot (\mathsf{A} \cdot \boldsymbol{x} - \boldsymbol{b}) \tag{1.102}$$

This yields

$$\boldsymbol{x} - \sum_n \lambda_n \mathsf{A}^T \cdot \hat{\boldsymbol{e}}_n = 0, \tag{1.103}$$

$$\mathsf{A} \cdot \boldsymbol{x} = \boldsymbol{b}, \tag{1.104}$$

The first equation is clearly stating that $\boldsymbol{x}$ has the form $\mathsf{A}^T \cdot \boldsymbol{y}$, with $\boldsymbol{y} = \sum_n \lambda_n \hat{\boldsymbol{e}}_n$. Therefore, the vector $\boldsymbol{y}$ contains the Lagrange multipliers associated to the minimization problem.

Finally, when the system is square but not symmetric positive definite, both of the above methods can be used. However, keep in mind that the condition number is squared in the process (**prove this!**), leading to many more iterations when the condition number is large.

1.3.8 Preconditioning

If a symmetric positive definite matrix A is very ill-conditioned, iterative solution methods will lead to very long computation times. However, if one also has an easily invertible symmetric positive definite matrix K that is 'close' to A, it is possible to use

this to speed up the convergence. The mathematically more precise statement of K being close to A is

$$\frac{\lambda_{\max}\left(\mathsf{K}^{-1}\cdot\mathsf{A}\right)}{\lambda_{\min}\left(\mathsf{K}^{-1}\cdot\mathsf{A}\right)} < \kappa(\mathsf{A}). \tag{1.105}$$

where $\lambda_{\max}$ and $\lambda_{\min}$ are the largest and smallest eigenvalue of $\mathsf{K}^{-1}\cdot\mathsf{A}$ (**Prove that all eigenvalues of this matrix are real and positive**). When criterium (1.105) holds, K can be used as a preconditioner, i.e. it speeds up convergence. If the reverse inequality holds, using the matrix K as a preconditioner will generally slow down convergence rather than speed it up.

To show how this can be done, it should be noted that one cannot simply apply the conjugate gradient method to $\mathsf{K}^{-1}\cdot\mathsf{A}$, since it is not a symmetric matrix. However, for positive symmetric definite matrices, it is possible to define a 'square root' by means of the Cholesky decomposition $\mathsf{K}=\mathsf{L}\cdot\mathsf{L}^T$ (other square roots exist, such as a symmetric one based on the eigenvalue decomposition, but those are beyond the scope of this course). Using the Cholesky decomposition, the linear system can be rewritten into the symmetric form

$$\mathsf{L}^{-1}\cdot\mathsf{A}\cdot\mathsf{L}^{-T}\cdot\boldsymbol{y}=\mathsf{L}^{-1}\cdot\boldsymbol{b}, \tag{1.106}$$

with $\boldsymbol{x}=\mathsf{L}^{-T}\cdot\boldsymbol{y}$. The conjugate gradient method can now be used on this symmetrized system. The condition number of the preconditioned system is

$$\kappa\left(\mathsf{L}^{-1}\cdot\mathsf{A}\cdot\mathsf{L}^{-T}\right) = \frac{\lambda_{\max}\left(\mathsf{L}^{-1}\cdot\mathsf{A}\cdot\mathsf{L}^{-T}\right)}{\lambda_{\min}\left(\mathsf{L}^{-1}\cdot\mathsf{A}\cdot\mathsf{L}^{-T}\right)}, \tag{1.107}$$

$$= \frac{\lambda_{\max}\left(\mathsf{L}^{-T}\cdot\mathsf{L}^{-1}\cdot\mathsf{A}\right)}{\lambda_{\min}\left(\mathsf{L}^{-T}\cdot\mathsf{L}^{-1}\cdot\mathsf{A}\right)}, \tag{1.108}$$

$$= \frac{\lambda_{\max}\left(\mathsf{K}^{-1}\cdot\mathsf{A}\right)}{\lambda_{\min}\left(\mathsf{K}^{-1}\cdot\mathsf{A}\right)} < \kappa(\mathsf{A}). \tag{1.109}$$

The last inequality holds under assumption (1.105), and proves that the convergence is faster with a good preconditioner.

Finally, it is worth knowing that, when using the symmetrized system (1.106) in the conjugate gradient method, there is no need to compute the Cholesky decomposition of K. Indeed, all the occurrences of the Cholesky factors L and L^T can be cleverly converted to factors K. This converted form of the conjugate gradient method is usually called the preconditioned conjugate gradient method.

1.4 Numerical Integration

A commonly used definition of integration is the Riemann integral[2], which involves limit operation of a sum. In the limit, the sum becomes infinitely long, meaning that it is impossible to realize this definition on a computer. Therefore, other approaches need to be found. An extremely widespread approach is to use so-called quadrature rules.

[2]This notion can be extended to the Lebesgue integral, which does not distinguish between functions that differ only on a set of μ-measure zero, being functions that are equal almost everywhere (a.e). This excludes a set of points capable of being enclosed in intervals whose total length is arbitrarily small.

Such rules consist of a set of N points x_n and weights w_n, which have the property that

$$\int_a^b f(x)\mathrm{d}x \approx \sum_n^N w_n f(x_n). \tag{1.110}$$

1.4.1 Gaussian Quadrature

Usually, the quadrature rule is constructed such that it is exact for a certain set of functions $f(x)$. For example, the so-called Gauss-Legendre quadrature rules have the property that the N-point rule is exact for the integration over the range $[-1, 1]$ for all polynomials of degree $2N - 1$ (or lower). We say that the degree of algebraic precision of the N-point Gauss-Legendre quadrature rule equals $2N - 1$. Because smooth functions can be approximated very accurately by means of polynomials, this is of great practical importance. Of course, when the function that needs to be integrated is not smooth (for example, if it contains an integrable singularity or other non-smooth behavior), other quadrature rules need to be applied. If a Gauss-Legendre quadrature rule is used nonetheless, the result may converge to the correct value for larger and larger N, but this convergence will be (very) slow.

There are a number of ways to handle this situation. If the non-smooth behavior can be isolated in a multiplicative function $g(x)$, it is possible to construct N-point 'Gaussian' quadrature rules that exactly integrate $g(x)$ times all polynomials of degree $2N-1$. For example, Gauss-Jacobi quadrature rules implement this scheme for $g(x) = (x+1)^\alpha(1-x)^\beta$, which can handle certain singularities at the edge of the integration domain. Quadrature rule for interior singularities can also be computed. Of course, this type of rules can also be constructed for smooth $g(x)$ such as e^{-x^2} (Gauss-Hermite quadrature) or e^{-x} (Gauss-Laguerre). The main advantage of these rules is that they achieve the same accuracy with a reduced number of points if the integrand contains the factor $g(x)$, or behaves likewise.

1.4.2 The $\tanh\sinh$ or Double-Exponential Rule

If the non-smoothness cannot be isolated in a multiplicative factor, but it is limited to the edges of the integration domain, a neat trick can be used to transform away the problem. Essentially, when integrating over the range $[-1, 1]$, the substitution $x = \tanh\left(\frac{\pi}{2}\sinh(u)\right)$ is performed. When the integration range is $[a, b]$, the formula is easily adapted to

$$\int_a^b f(x)\mathrm{d}x = \frac{b-a}{2}\int_{-1}^{1} f\left(\frac{b-a}{2}t + \frac{a+b}{2}\right)\mathrm{d}t, \tag{1.111}$$

$$= \frac{b-a}{2}\int_{-\infty}^{+\infty} f\left(\frac{b-a}{2}\tanh\left(\frac{\pi}{2}\sinh u\right) + \frac{a+b}{2}\right)J(u)\mathrm{d}u, \tag{1.112}$$

where the Jacobian is given by

$$J(u) = \frac{\mathrm{d}}{\mathrm{d}u}\left[\tanh\left(\frac{\pi}{2}\sinh u\right)\right] = \frac{\pi}{2}\frac{\cosh u}{\cosh^2\left(\frac{\pi}{2}\sinh u\right)}. \tag{1.113}$$

The integral (1.112) is now discretized uniformly as

$$\int_a^b f(x)\mathrm{d}x = h\frac{b-a}{2}\sum_{n=-N}^{N} f\left(\frac{b}{1+e^{-\pi\sinh u_n}}+\frac{a}{1+e^{\pi\sinh u_n}}\right)J(u_n) \quad (1.114)$$

where $u_n = nh$. There are two parameters in this integration rule: N and h. Both influence the accuracy. In general, the product Nh must be large enough, and the value of h must be small enough. For large u, the Jacobian has the behavior

$$J(u) \approx \pi e^{u-\frac{\pi}{2}e^u}, \quad (1.115)$$

which explains the name 'double-exponential rule'. Because the Jacobian is such a quickly decreasing function of u, it compensates almost all possible integrable singularities.

Remark: for numerical stability, it is best to evaluate the argument of $f(.)$ as

$$\begin{aligned}\frac{b-a}{2}\tanh\left(\frac{\pi}{2}\sinh u\right)+\frac{a+b}{2} &= \frac{b}{2}\left[1+\tanh\left(\frac{\pi}{2}\sinh u\right)\right]+\frac{a}{2}\left[1-\tanh\left(\frac{\pi}{2}\sinh u\right)\right],\\ &= \frac{b}{1+e^{-\pi\sinh u}}+\frac{a}{1+e^{\pi\sinh u}}. \quad (1.116)\end{aligned}$$

1.4.3 Variable Singularities Inside the Integration Range

If singularities are located inside the integration domain, it is possible to split the integration domain such that the singularities are at the edges of the sub-domains. Then the methods introduced in the above can be applied once more. Alternatively, if a function $h(x)$ can be found that has the same singular behavior as the integrand $f(x)$, and an analytical expression H can be found for the integral of $h(x)$, the integral can be computed as

$$\int_a^b f(x)\mathrm{d}x = H + \int_a^b f(x) - h(x)\mathrm{d}x. \quad (1.117)$$

Since the singular behavior is now removed from the numerical part of the integral, the latter can be computed using the standard quadrature rules. Such techniques can typically be used for meromorphic functions, i.e. functions that are entire, up to poles at a set of isolated points. For such functions, a Laurent series can be constructed around the singular point(s), and the singular terms of the Laurent series can then be integrated analytically (in the principal value sense). An example is

$$I = \text{P.V.}\int_{-\frac{1}{2}}^{1}\frac{1}{\sin(e^x-1)}\mathrm{d}x = \ln(2)+\int_{-\frac{1}{2}}^{1}\frac{1}{\sin(e^x-1)}-\frac{1}{x}\mathrm{d}x. \quad (1.118)$$

In this case, the remaining integral's integrand is smooth.

1.4.4 Generalized Gaussian Quadrature

Finally, in certain cases, it is possible to construct an N-point quadrature rule tailored to a set of $2N$ functions $f_n(x)$. Essentially, the construction proceeds by imposing the equality

$$\int_a^b f_n(x)\mathrm{d}x = \sum_p^N w_p f_n(x_p),\ \forall n \in [0, 2N-1], \quad (1.119)$$

and then solving the nonlinear system of equations. Since there are $2N$ variables and $2N$ equations, one can expect the existence of a solution. However, in the general case the nodes x_p could lie outside of the integration domain or could even be complex, which can make the integration rule unstable. Moreover, one is not guaranteed anymore that the integration weights w_n are positive, also leading to potential instabilities. To top it all off, it is not even known in general whether a solution exists, or whether it is unique if a solution is found.

A quite complicated theory concerning so-called Chebyshev systems has succeeded in finding criteria for answering these questions, but that is beyond the scope of this course. An important example of a set of functions for which these criteria provide existence and uniqueness is the set $\{\mathbb{P}_{N-1}, \mathbb{P}_{N-1}\ln(x)\}$ for the integration domain $[0,1]$. $\mathbb{P}_{N-1}$ is the set of polynomials of degree at most $N-1$. Therefore, a generalized Gaussian quadrature rule exists for the evaluation of integrals of the type

$$\int_0^1 p(x) + q(x)ln(x)\mathrm{d}x = \sum_n^N w_n \left[p(x_n) + q(x_n)ln(x_n)\right]. \tag{1.120}$$

This has practical importance in the evaluation of boundary integral equation matrix elements with the two-dimensional Green function $H_0^{(2)}(x)$.

1.5 Integral Equations

1.5.1 Fredholm Integral Equations

Let us define the linear integral operator

$$\int_a^b K(t,s)f(s)\mathrm{d}s, \tag{1.121}$$

which maps a function $f(s)$ onto another function. The bivariate function $K(t,s)$ is called the integration kernel and is usually considered to be bounded for all $s,t \in [a,b]$ (extensions exist). In this case the following integral equation is a Fredholm integral equation of the first kind:

$$g(t) = \int_a^b K(t,s)f(s)\mathrm{d}s, \forall t \in [a,b]. \tag{1.122}$$

Here, $g(t)$ is a known function, and this equation needs to be solved for $f(s)$. A so-called Fredholm integral equation of the second kind also exists:

$$g(t) = \int_a^b K(t,s)f(s)\mathrm{d}s - \sigma f(t), \forall t \in [a,b]. \tag{1.123}$$

Under the assumption of bounded $K(t,s)$, it can be proved that the equation of the second kind either has a unique solution or the homogeneous equation (i.e. with $g(t) = 0$) has one or more nonzero solutions. In the latter case, the integral equation can only be solved if $g(t)$ is orthogonal to all functions $\psi(t)$ that are solutions to the adjoint homogeneous equation. In a more mathematical form, if the following holds:

$$\int_a^b \overline{K(s,t)}\psi(s)\mathrm{d}s - \overline{\sigma}\psi(t) = 0 \;\rightarrow\; \int_a^b \psi(t)g(t)\mathrm{d}t = 0 \tag{1.124}$$

then the equation can still be solved (non-uniquely). This is known as the Fredholm alternative. Otherwise, no solutions exist.

A further result exists if the integral operator is self-adjoint: $\overline{K(s,t)} = K(t,s)$. In this case, an 'eigenvalue decomposition' holds:

$$\int_a^b K(t,s)f(s)\mathrm{d}s = \sum_p \lambda_p \phi_p(t) \int_a^b \overline{\phi_p(s)} f(s)\mathrm{d}s. \tag{1.125}$$

Here, the functions $\phi_p(t)$ form an orthonormal basis of eigenfunctions. **Note:** in case the operator is not self-adjoint, the singular value decomposition can be computed instead of the eigenvalue decomposition.

Now, if the eigenvalue decomposition of the integral operator is known, it is possible to predict when the Fredholm integral equation of the second kind has either a nonunique or nonexistent solution: when $\lambda_p = \sigma$. Indeed, in that case:

$$\int_a^b K(t,s)\phi_p(s)\mathrm{d}s - \lambda_p \phi_p(t) = 0, \tag{1.126}$$

such that a solution to the homogeneous equation exists. Therefore, Fredholm integral equation of the second kind is uniquely solvable if σ is not an eigenvalue of the integral operator with kernel $K(t,s)$. The solution is given by

$$f(s) = \sum_p \frac{\phi_p(s)}{\lambda_p - \sigma} \int_a^b \overline{\phi_p(t)} g(t)\mathrm{d}t. \tag{1.127}$$

Conditioning

It can be proved that the sequence of eigenvalues λ_p converges to zero. Therefore, by the eigenvalue decomposition (1.125), the integral operator with kernel $K(t,s)$, applied to a superposition of eigenvectors $\sum_p c_p \phi_p(t)$, decreases the magnitude of all but a finite number of the eigenvector coefficients c_p. Therefore, it is intuitively clear that the coefficients c_p are less easily retrievable from the output than from the input. This effect is very similar to that of a high condition number and makes Fredholm integral equations of the first kind hard to solve.

Fortunately, the situation for Fredholm equations of the second kind is much better. There, the integral operator shrinks most of the eigenvector coefficients, but the addition of σ times the input preserves most of the information. Therefore, these equations are typically much easier to solve (for σ away from eigenvalues).

An alternative interpretation is in terms of smoothing. The integral operator with bounded kernel $K(t,s)$ always has a smoothing effect (This would not necessarily be true if the boundedness constraint is removed. Indeed, the Dirac delta distribution could be considered unbounded and is not a smoothing operation). This smoothing effect is essentially a low-pass filter that removes information from a signal.

Singular Kernels

The theory of Fredholm integral equations of the second kind can be extended. In general, the integral operator defined by $K(t,s)$ should be a so-called compact operator. This allows some unbounded (singular) kernels to be treated within the same framework. Examples of such kernels include some Green's functions, or sometimes

integrations over infinite domains. However, there also exist examples (such as the D operator in the representation formulas from the integral equations lecture, or simply the Laplacian $\boldsymbol{\nabla}^2$) of operators that do not fit into the Fredholm framework. For these operators, the mathematical properties will be considered beyond the scope of this course.

1.5.2 Volterra Integral Equations

Volterra integral equations can be considered to be special cases of Fredholm integral equations, i.e. when

$$K(t,s) = 0,\ \forall s > t. \tag{1.128}$$

This property makes the upper integration bound in the Fredholm integral equation dependent on t. For example, in the equation of the first kind:

$$g(t) = \int_a^t K(t,s) f(s) \mathrm{d}s. \tag{1.129}$$

Such equations typically arise in the time domain, where the value of a certain field depends only on the fields on earlier times. The equation is somewhat similar to a lower triangular matrix system, which are easily solvable by means of forward substitution. However, when solving a discretized version of (1.129), one always has to be extremely careful that this scheme is stable. Indeed, though the continuous system may be stable due to, for example, energy conservation, this is not necessarily the case in the discretized system. This can introduce instabilities that invalidate the solution after a number of steps.

Analogous to Fredholm equations of the second kind, we can also study Volterra equations of the second kind. It can be proved that the integral operator in (1.129) has no eigenvalues, which means that the Volterra equations of first and second kind can always be solved uniquely. For an equation of the second kind, the solution can be found in the form of a Neumann series (geometric series involving an operator). This is in contrast to the Fredholm case, where this is only possible for sufficiently small integral operators. Finally note that a Volterra equation of the first kind involving a kernel with non-zero finite diagonal values can be transformed into a Volterra equation of the second kind by taking a derivative w.r.t. the free variable.

1.5.3 The Nystrom Method

To solve Fredholm integral equations of the first or second kind, various techniques exist. The Nystrom method is one of the simplest methods that can be used, but it yields surprisingly good results for many problems. Let us apply the Nystrom method for the Fredholm integral equations of the second kind (1.123). Using our knowledge of quadrature rules, we can replace the integral with a finite summation:

$$g(t) = \sum_{p=1}^{N} w_p K(t, x_p) f(x_p) - \sigma f(t). \tag{1.130}$$

Subsequently, this equation is imposed in N points y_q, leading to the equation

$$g(y_q) = \sum_{p=1}^{N} w_p K(y_q, x_p) f(x_p) - \sigma f(y_q), \forall q \in [1, N]. \tag{1.131}$$

The problem with this equation is that there are $2N$ unknown function values in this equation, i.e. $f(x_p)$ and $f(y_q)$. This problem can be resolved by taking $x_p = y_p$. Then the following system of linear equations is obtained

$$\sum_{p=1}^{N} \left[w_p K(x_q, x_p) - \sigma \delta_{p,q} \right] f(x_p) = g(x_q). \tag{1.132}$$

If the kernel $K(t, s)$ is self-adjoint, the problem can be symmetrized as follows:

$$\sum_{p=1}^{N} \left[\sqrt{w_q} K(x_q, x_p) \sqrt{w_p} - \sigma \delta_{p,q} \right] \left(\sqrt{w_p} f(x_p) \right) = \sqrt{w_q} g(x_q). \tag{1.133}$$

This makes the problem amenable to iterative solution with, for example, the conjugate gradient method (it the matrix is also positive definite). For eigenvalue computation, the symmetrized form is also important to make use of certain optimizations. Finally, after finding the values $f(x_p)$, the other function values can be computed as

$$f(t) = \frac{1}{\sigma} \left[\sum_{p=1}^{N} w_p K(t, x_p) f(x_p) - g(t) \right]. \tag{1.134}$$

For Kernels with a singularity of the form $\frac{1}{t-s}$, one can extract the singularity as follows:

$$\int_a^b K(t, s) f(s) \mathrm{d}s = \int_a^b K(t, s) \left[f(s) - f(t) \right] \mathrm{d}s + f(t) \int_a^b K(t, s) \mathrm{d}s. \tag{1.135}$$

Assuming that the last integral can be calculated by analytical or other means, this removes the problematic singularity and allows the Nystrom method to be used even for such a kernel.

Chapter 2

Finite elements

2.1 Introduction

Finite elements are used both to perform time domain and frequency domain computations. In this course, we restrict ourselves to the frequency domain. Finite element techniques compute unknowns (e.g. the electric field) in the bulk of the domain of interest, and thus can take into account local variations in material parameters. Hence, finite element methods are especially suited for simulations of strongly inhomogeneous methods.

The finite element method is an exact solution method since it makes no approximations on the wave equations except due to discretizations. This means that the solution becomes more accurate for finer discretizations. Approximate techniques, such as Born approximations, physical optics solutions or parabolic equation techniques do not have this feature and can only provide an approximate solution, albeit sometimes with very high accuracy. In those cases where these approximations hold, the dedicated but inexact techniques tend to be much more efficient, which explains their ongoing popularity.

The finite element method is an old numerical solution technique. It was pioneered for static field problems, such as mechanical strength simulations in construction engineering. The technique starts by subdividing simulation space into a large number of cells, called elements. Different element shapes can be chosen, such as triangular, rectangular, ... In each element, the field is represented by a finite number of degrees of freedom (DoFs). This means that the field is expanded into a linear combination of trial or shape functions with some unknown amplitudes and phases. Different shape functions can be implemented, such as piecewise constant, piecewise linear and globally continuous functions.

The wave equation can often be transformed into an equivalent optimization problem. In these cases, it can be proven that solving the wave equation is equivalent to finding the function that is a stationary point of some functional related to the equation (cfr. the solution of a positive definite symmetric linear system). Sometimes, more than one functional can be related to the equation. Different functionals yield different finite element methods for the same equation.

Instead of seeking for a stationary point to the functional among all possible functions, the stationary point is sought in the space spanned by the shape functions. The optimization procedure is then carried out by varying the DoFs instead of varying the

test functions in the functional. The outcome of this procedure is the optimal set of DoFs. The function corresponding to these DoFs is then the best approximation of the real solution within the space spanned by our shape functions.

2.2 General theory

In this section, we will convert a general linear operator equation into a finite elements form. Therefore, two different formalisms exist. The most general approach consists of constructing a weak-form formulation by taking the inner product of both the left-hand and right-hand sides of the operator equation with an arbitrary wave function. This approach is very general and the reader will notice the similarity with the Method of Moments introduced in the course Electromagnetism II. In some cases, this weak-form formalism is identical to calculating the stationary function of a functional. The advantage of this approach is that discretisation errors of order ϵ will only produce errors in the functional of order ϵ^2 or higher.

2.2.1 Weak solution of a linear operator equation

Consider a general linear operator equation of the form

$$Df(\mathbf{r}) = g(\mathbf{r}), \quad f(\mathbf{r}) \in \mathcal{D}(D), \tag{2.1}$$

where $f(\mathbf{r})$ is an unknown response to known excitation $g(\mathbf{r})$ in the domain $\mathcal{D}(D)$. The operator $\mathcal{D}$ is assumed to be linear, i.e.:

$$D(af_1(\mathbf{r}) + bf_2(\mathbf{r}) = aDf_1(\mathbf{r}) + bDf_2(\mathbf{r}) \in \mathcal{D}(D). \tag{2.2}$$

For many configurations, it is impossible to analytically construct a solution $f(\mathbf{r})$ that exactly satisfies (2.1) in every point of the domain $\mathcal{D}(D)$. In that case, instead of searching for a *strong* solution that exactly pointwise satisfies the operator equation in the whole of $\mathcal{D}(D)$, a *weak* solution can be constructed by rewriting (2.1) in the following *weak form*:

$$(Df(\mathbf{r}), w(\mathbf{r})) = \int_\Omega w(\mathbf{r})Df(\mathbf{r})d\mathbf{r} = \int_\Omega w(\mathbf{r})g(\mathbf{r})d\mathbf{r} = (w(\mathbf{r}), g(\mathbf{r})). \tag{2.3}$$

By taking the inner product of the operator with an arbitrary test or weighting function $w(\mathbf{r})$, the equality in (2.1) is enforced in an *average sense* instead of in a pointwise way. Taking the inner product of two functions means computing their product and integrating this product over their domain of definition. This inner product for functions obeys all the rules we discussed in the linear algebra lesson, such as scalability, symmetry, and positiveness. In this chapter, the linear operator equations under consideration will mainly be differential equations with an appropriate suitable set of boundary conditions.

2.2.2 Rayleigh-Ritz procedure: Variational principle

Let us now consider a *functional* $J(f(\mathbf{r})) \in \mathcal{D}(D)$, that is, an expression that takes a well-defined value for each potential solution function $f(\mathbf{r})$ for (2.1), chosen from the family of admissible functions. We are now interested in the stationary functions $f_0(\mathbf{r})$ of this functional $J(f(\mathbf{r})) \in \mathcal{D}(D)$, as these are the points for which a small variation

$\delta f_0(\mathbf{r})$ will result in an even smaller variation in the value of the functional $\delta J(f_0(\mathbf{r}))$, being in the order $||\delta f_0||^2$ or higher. Crudely stated, this means that an error in f_0 of order ϵ (say 10%) results in an error on $J(f_0)$ of order ϵ^2 (say 1%).

Let now the linear operator $\mathcal{D}$ be selfadjoint in the domain $\mathcal{D}(D)$, meaning that

$$(Df, w) = (f, Dw), \quad \forall f, w \in \mathcal{D}(D) \tag{2.4}$$

In that case, the stationary function f_0 of the functional

$$J(f) = \frac{1}{2}(Df, f) - (f, g) \tag{2.5}$$

will be the weak-form solution of (2.1), being the solution of (2.3).

We will prove this property by extending the concept of stationary points of a function to stationary functions of a functional $J(f)$. For example, the functional $J(f)$ has a local minimum at admissible function $f_0(\mathbf{r})$ when δJ is positive for all possible admissible variations δf. The latter means that δf must be chosen that $f_0 + \delta f$ remains an admissible function. We will express stationarity of the functional by considering small variations

$$f(\mathbf{r}) = f_0(\mathbf{r}) + \epsilon\eta(\mathbf{r}) \tag{2.6}$$

around the stationary admissible function $f_0(\mathbf{r})$, with ϵ a small real parameter and $\eta(\mathbf{r})$ an arbitrary admissible function. The stationarity condition now requires that, after inserting these functions in $J(f)$, the resulting δJ is of order ϵ^2 or higher. Inserting (2.6) into (2.5) yields, making use of bilinearity

$$J(f) = \frac{1}{2}(Df_0, f_0) - (f_0, g) + \epsilon\left[\frac{1}{2}(D\eta, f_0) + \frac{1}{2}(Df_0, \eta) - (\eta, g)\right] + \epsilon^2\frac{1}{2}(D\eta, \eta). \tag{2.7}$$

Stationarity requires that the term in ϵ be zero, or, equivalently, that $\frac{dJ}{d\epsilon}\big|_{\epsilon=0} = 0$, or that

$$\frac{1}{2}(D\eta, f_0) + \frac{1}{2}(Df_0, \eta) - (\eta, g) = 0. \tag{2.8}$$

Making use of the symmetry of the inner product and of the selfadjointness of the linear operator $\mathcal{D}$, being (2.4), we obtain:

$$(Df_0, \eta) - (g, \eta) = 0, \tag{2.9}$$

which, indeed, yields the weak-form formulation (2.3), with the admissible function η acting as a testing function. Therefore, (2.9) is called the *Euler equation* of the functional. It is important to note that often, this Euler equation is insufficient to ensure stationarity, since $f_0(\mathbf{r})$, in order to be an *admissible* function, must satisfy some boundary conditions. Such boundary conditions, complementing (2.9), are then called *natural* boundary conditions. In certain cases, it is not enough to only apply boundary conditions to the stationary admissible function $f_0(\mathbf{r})$. In that case, for the variations in (2.6) to be admissible, we must also impose boundary conditions on the arbitrary admissible test functions $\eta(\mathbf{r})$. Such boundary conditions are then called *essential* boundary conditions.

2.3 Application to 1D wave equations

By means of example, let us construct a weak form for the one-dimensional wave equation

$$\left(\frac{d^2}{dx^2} + k^2\right) f = -g, \quad 0 \leq x \leq L, \tag{2.10}$$

and show its similarity to finding the stationary function of the functional

$$J(f) = \frac{1}{2}\int_0^L \left(\frac{df}{dx}(x)\right)^2 - k^2 f^2(x) dx - \int_0^L f(x)g(x)dx. \tag{2.11}$$

The approach will be different when completing the differential equation with its proper set of boundary conditions. Imposing Neumann boundary conditions will result in different admissibility conditions compared to applying Dirichlet boundary conditions

2.3.1 Neumann problem

We first complete the boundary value problem by adding the Neumann boundary conditions to (2.10):

$$\frac{df}{dx}(0) = \frac{df}{dx}(L) = 0 \tag{2.12}$$

Weighting both sides of (2.10) with test function $w(x)$ then leads to

$$\int_0^L w(x)\left[\frac{d^2 f(x)}{dx^2}\, dx + k^2 \int_0^L w(x) f(x)\right] dx = -\int_0^L w(x)g(x)\, dx, \tag{2.13}$$

after which integration by parts results in

$$\left[w(x)\frac{df}{dx}\right]_{x=0}^{x=L} - \int_0^L \frac{dw}{dx}\frac{df}{dx}\, dx + k^2 \int_0^L w(x)f(x)\, dx = -\int_0^L w(x)g(x)\, dx, \tag{2.14}$$

where the first term cancels out thanks to the *natural* boundary conditions (2.12), hence

$$-\int_0^L \frac{dw}{dx}(x)\frac{df}{dx}(x)\, dx + k^2 \int_0^L w(x)f(x)\, dx = -\int_0^L w(x)g(x)\, dx. \tag{2.15}$$

We now show that this weak-form formulation is the Euler equation yielding the stationary function f_0 of the functional (2.11), on which, in addition we impose the boundary conditions (2.12). The proof is as follows:

$$\begin{aligned} J(f_0(x) + \epsilon\eta(x)) = {} & J(f_0(x)) \\ & + \epsilon\left[\int_0^L \frac{d\eta}{dx}\frac{df_0}{dx}\, dx - k^2\int_0^L \eta(x)f_0(x)\, dx - \int_0^L \eta(x)g(x)\, dx\right] \\ & + \frac{\epsilon^2}{2}\int_0^L \left(\frac{d\eta}{dx}\right)^2 - k^2\eta^2(x)dx. \end{aligned} \tag{2.16}$$

where, indeed, the term in ϵ equals zero when (2.15) is satisfied.

As an exercise, verify that the operator appearing in (2.10) is indeed selfadjoint. Note that the proof requires that both functions $f_0(x)$ and $\eta(x)$ appearing in the inner product must both satisfy boundary conditions, which is a stronger requirement than the *natural* boundary condition, which only constrains $f_0(x)$.

2.3.2 Dirichlet problem

Let us now apply the Dirchlet boundary conditions to (2.10):

$$f(0) = f(L) = 0. \tag{2.17}$$

When constructing the weak-form formulation as in the previous subsection, the first term now only cancels out if we impose the *essential* boundary conditions (2.17) also on the test functions, hence

$$w(0) = w(L) = 0, \tag{2.18}$$

which also adds a restriction on the basket of admissible test functions $\eta(x)$, applied when perturbing the functional around its stationary function. With these restrictions imposed by the boundary conditions, we again obtain both (2.15) and (2.11), proving again equivalence between the weak-form and the Rayleigh-Ritz approach, in case of the *essential* Dirichlet boundary conditions.

2.4 Application to 2D wave equations

The construction of a weak form of the two-dimensional wave equations is left as an exercise.

1. Prove that the operator is self-adjoint.
2. What is now the functional to be minimized by the Rayleigh Ritz approach?
3. Which are the natural and essential boundary conditions?

2.5 Application to the Maxwell equations

Let us now cast Maxwell's equations into a weak form. Therefore, we start from both curl laws

$$\nabla \times \mathbf{E}(\mathbf{r}) = -j\omega\overline{\overline{\mu}} \cdot \mathbf{H}(\mathbf{r}) \tag{2.19}$$

$$\nabla \times \mathbf{H}(\mathbf{r}) = j\omega\overline{\overline{\epsilon}} \cdot \mathbf{E}(\mathbf{r}) + \mathbf{J}(\mathbf{r}) \tag{2.20}$$

To derive the E-field formulation (EFF), we eliminate the magnetic field and obtain

$$\nabla \times \overline{\overline{\mu}}^{-1} \cdot (\nabla \times \mathbf{E}) - \omega^2\overline{\overline{\epsilon}} \cdot \mathbf{E} = -j\omega\mathbf{J} \tag{2.21}$$

Taking the inner product of both sides of this equation with an arbitrary test function $\mathbf{w}(\mathbf{r})$ yields

$$\int_V \mathbf{w} \cdot \nabla \times \overline{\overline{\mu}}^{-1} \cdot (\nabla \times \mathbf{E})\; dV - \omega^2 \int_V \mathbf{w} \cdot \overline{\overline{\epsilon}} \cdot \mathbf{E}\, dV = -j\omega \int_V \mathbf{w} \cdot \mathbf{J}\, dV. \tag{2.22}$$

Making use of Green's theorem, we obtain the following weak-form E-field formulation

$$\int_V (\nabla \times \mathbf{w}) \cdot \overline{\overline{\mu}}^{-1} \cdot (\nabla \times \mathbf{E})\, dV - \omega^2 \int_V \mathbf{w} \cdot \overline{\overline{\epsilon}} \cdot \mathbf{E}\, dV = -j\omega \int_V \mathbf{w} \cdot \mathbf{J}\, dV + \oint_S (\mathbf{n} \times \mathbf{w}) \cdot \overline{\overline{\mu}}^{-1} \cdot (\nabla \times \mathbf{E})\, dS. \tag{2.23}$$

The latter surface integral determines which boundary conditions on S are natural and which are essential. To clearly see this, we reintroduce the magnetic field into the boundary integral:

$$\int_V (\nabla \times \mathbf{w}_i) \cdot \overline{\overline{\mu}}^{-1} \cdot (\nabla \times \mathbf{E})\, dV - \omega^2 \int_V \mathbf{w}_i \cdot \overline{\overline{\epsilon}} \cdot \mathbf{E}\, dV = \\ -j\omega \int_V \mathbf{w}_i \cdot \mathbf{J}\, dV + j\omega \oint_S \mathbf{w}_i \cdot (\mathbf{n} \times \mathbf{H})\, dS. \tag{2.24}$$

From this, we derive that the natural boundary condition corresponds to a perfectly magnetic conducting surface S, where $\mathbf{n} \times \mathbf{H} = \mathbf{0}$, whereas a perfect conducting surface S is an essential boundary condition, requiring $\mathbf{n} \times \mathbf{E} = \mathbf{0}$ *and* $\mathbf{n} \times \mathbf{w} = \mathbf{0}$ on S. This then leads to the following final weak-form formulation, without boundary term:

$$\int_V (\nabla \times \mathbf{w}) \cdot \overline{\overline{\mu}}^{-1} \cdot (\nabla \times \mathbf{E})\, dV - \omega^2 \int_V \mathbf{w} \cdot \overline{\overline{\epsilon}} \cdot \mathbf{E}\, dV = -j\omega \int_V \mathbf{w} \cdot \mathbf{J}\, dV. \tag{2.25}$$

With these appropriate boundary conditions, the operator in (2.21) is self-adjoint, and the former weak-form equation is then the Euler equation yielding the stationary point of the functional

$$J(\mathbf{E}) = \frac{1}{2} \int_V \left[(\nabla \times \mathbf{E}) \cdot \overline{\overline{\mu}}^{-1} \cdot (\nabla \times \mathbf{E}) - \omega^2 \mathbf{E} \cdot \overline{\overline{\epsilon}} \cdot \mathbf{E} \right] dV + j\omega \int_V \mathbf{E} \cdot \mathbf{J}\, dV \tag{2.26}$$

As an exercise, derive the dual magnetic field formulation (MFF) and prove that also the natural and essential boundary conditions change correspondingly.

2.6 Discretization of the weak-form formulation

The Lax-Milgram theorem, which we will not discuss here, proves, for certain conditions on the linear operator (such as coercivity, which is related to positive definiteness of the linear operator), the existence and uniqueness of the solution of the weak-form equation when (2.3) holds for all admissible test functions $w(\mathbf{r})$. Our aim now will be to construct an, albeit approximate, solution for the weak-form equation in a finite-dimensional subspace of the space of admissible functions. This will make the problem manageable for solution on a computer. Specifically, we will look for a solution in an N-dimensional subspace that we construct by means of a finite set of basis, expansion or shape functions $b_j(\mathbf{r})), j = 1, \ldots, N$. Hence, given linearity, a candidate solution in that subspace may be represented as:

$$f(\mathbf{r}) = \sum_{j=1}^{N} f_j b_j(\mathbf{r}), \tag{2.27}$$

where the expansion coefficients $f_j, j = 1, \ldots, N$ provide N degrees of freedom to approximate the solution of the weak-form problem (2.3). They may be determined by imposing the weak form equation (2.3) in a set of N test functions $w_i(\mathbf{r})), i = 1, \ldots, N$, also spanning an N-dimensional test space. The end result of this discretization process is a matrix system, consisting of N equations for N unknowns:

$$\sum_{j=1}^{N} f_j (w_i(\mathbf{r}), Db_j(\mathbf{r})) = (w_i(\mathbf{r}), g(\mathbf{r})) \quad i = 1, \ldots, N. \tag{2.28}$$

Hence, the steps that must be taken to solve a general linear operator equation by the finite elements method are the following

1. Cast the linear operator equation (2.1) into a weak-form formulation, or construct a functional having the operator equation (2.1) in its weak form as Euler equation.
2. Partition the simulation space into a set of cells, called the finite element mesh or, in short, the finite elements.
3. Choose the basis/expansion/shape functions to represent the unknown function and the weighting/test functions to impose equality in the weak sense.
4. Calculate the interactions $(w_i(\mathbf{r}), Db_j(\mathbf{r}))$, being all the elements $Z_{i,j}$ of the system matrix $\overline{\overline{Z}}$, and the weighted excitations $g_i = (w_i(\mathbf{r}), g(\mathbf{r}))$, forming the vector $\mathbf{g}$.
5. Solve the matrix system $\overline{\overline{Z}}\mathbf{f} = \mathbf{g}$ for the unknown expansion coefficients $\mathbf{f}$.

Some important remarks are in order here.

1. The partitioning of the simulation space into a cells should result in a high-quality mesh. Meshes of bad quality will result in system matrices $\overline{\overline{Z}}$ with large condition number and poor solution accuracies.
2. The accuracy of the solution will also depend on the finite-dimensional subspace spanned by the set of admissible functions, applied as approximation for the infinite subspace. Therefore, the choice of basis/expansion/shape functions, to represent the unknown function, and weighting/test functions, to impose the weak-form equality, is of paramount importance. In this respect, in the next section, we provide guidelines to choosing the correct function spaces.
3. When discretizing the weak form of a differential equation and applying a finite element mesh, only local interactions occur (in contrast to discretized integral equations that produce global interactions due to the Green's function kernels). Hence, a sparse interaction matrix $\overline{\overline{Z}}$ is obtained. Therefore, for efficiency reasons, quite often one will implement technique that never stores the complete system matrix. Instead, the calculation of the interaction matrix and the inversion of the matrix system are combined into one single step. Either an iterative solution technique is implemented in which, during each iteration, a matrix-vector product between the interaction matrix and a search vector is evaluated. When a direct solution method is preferred, then a frontal method can be implemented. The technique combines the calculation of the matrix elements with a frontal elimination process, during which expansion coefficients for which all interactions have been calculated are immediately eliminated from the matrix equation. More details will be provided later on in the course.

2.6.1 Partitioning the simulation space — finite element mesh

In the discretization process, our first task consists of partitioning the simulation space into a high-quality mesh. This means that the mesh cells should be chosen small enough, such that expansion functions are able to follow (potentially rapid) field variations. The reader might remember from the section on the Method of Moments in the

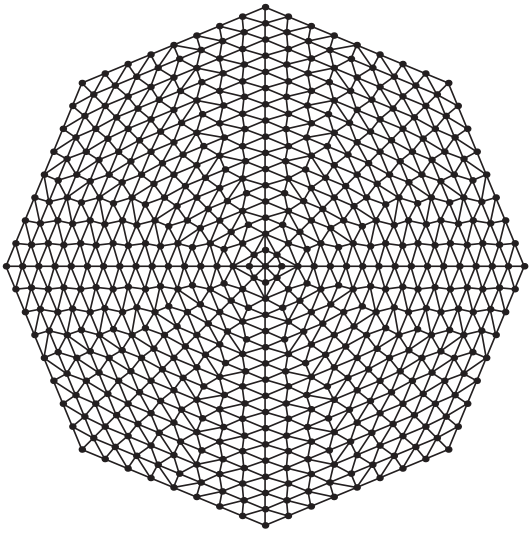

Figure 2.1: Example of a 2D finite element mesh. The octagone is discretized such that each elementary cell is a triangle.

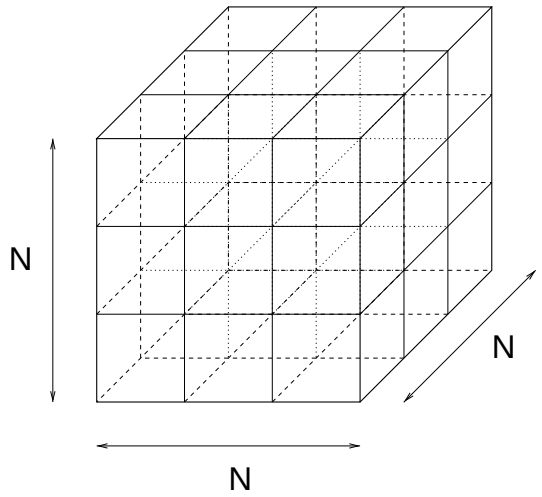

Figure 2.2: Example of a 3D finite element mesh. The cube is discretized such that each elementary cell is a brick element.

course Electromagnetism II, that at least 10 cells per wavelength are needed to guarantee an accurate approximation. However, in finite element or finite difference scheme, one must be careful about a phenomenon called "grid dispersion". Grid dispersion means that the propagation speed in the mesh will start to differ from the propagation speed of the continuous problem (being the speed of light when we consider Maxwell's equations). In the discretized problem, one finds that the propagation speed depends on the size of the cells with respect to wavelength, approaching the propagation speed of the continuous problem when cell size goes to zero. This also means that, in 2D or 3D propagation speed will become direction dependent, hence the discretized problem

acts as an anisotropic medium due to the mesh. Moreover, spurious reflections will occur at transitions between small and large cells, as the propagation speeds will differ in the two kinds of cells. The problem of grid dispersion is especially important for large simulation domains, as even a small difference in propagation speed will be noticeable when propagation over larger distances occurs.

Hence, a high-quality mesh will be sufficiently fine, with cell sizes of the order of $\lambda/20$ down to $\lambda/30$. At some locations, where rapid field variations occur, one may opt to further refine the mesh, but this should be done gradually, to avoid spurious reflections. Moreover, long sharp triangles, in 2D, or tetrahedrons, in 3D, should be avoided in order not to introduce a too large anisotropy in the discretized problem.

2.6.2 Construction of the expansion functions

After creating the finite element mesh, the next step consists of finding a suitable representation of the field within each cell by means of a set of expansion functions. The choice of finite-dimensional subspace spanned by these basis functions also determines the accuracy by which the discretized problem approximates the continuous operator equation (2.1). To develop a finite element method that pairs accuracy with efficiency in terms of memory resources and CPU time, we must bear in mind the following:

1. Computational efficiency dictates the use of simple expansion functions, preferably polynomials for which interaction integrals may be calculated analytically or exactly by means of quadrature rules with sufficiently high degree of algebraic accuracy, and with a support that is limited to a few cells of the finite element mesh, such that only local interactions occur, resulting in a sparse interaction matrix.

2. The smoothness conditions imposed on the expansion functions should be such that all integrals exist. Therefore, the expansion functions must belong to the correct *Sobolev space*, this is the space of quadratic-integrable functions in the Lebesgue sense. Depending on the weak-form formulation and/or the functional to be discretised, besides the function itself, also some of its derivatives up to a certain order must be quadratic-integrable functions in the Lebesgue sense. This will define the specific Sobolev space the expansion functions must belong to. Let us now study a few examples to understand how to choose the correct Sobolev space providing the required smoothness to be able to evaluate the integrals and/or functional.

   In the **one-dimensional case**, the governing weak-form equation is (2.15), whereas the relevant functional is (2.11). We immediately see that, for the second integral to exist in both expressions, the solution $f(x)$ should be expanded into functions belonging to the class of square-integrable functions in the Lebesgue sense over the interval $[0, L]$.

$$\mathcal{L}^2([0,L]) = \left\{ f : [0,L] \to \mathbb{C} \;\middle|\; \int_0^L |f(x)|^2 \, dx < +\infty \right\}. \tag{2.29}$$

   In Sobolev space terminology, the expansion functions must thus be chosen in

$$\mathcal{H}^0([0,L]) = \left\{ f \in \mathcal{L}^2([0,L]) \right\}. \tag{2.30}$$

Yet, we must further restrict the space of admissible functions for the first integral to exist in expressions (2.15) and (2.11), since also the first derivative should be square-integrable. Therefore, the pertinent Sobolev space from which to choose the expansion functions is

$$\mathcal{H}^1([0,L]) = \left\{ f \in \mathcal{L}^2([0,L]) \;\middle|\; \frac{df}{dx} \in \mathcal{L}^2([0,L]) \right\}. \tag{2.31}$$

Similarly, for the **three-dimensional Maxwell equations**, from the governing weak-form equation (2.25) and the relevant functional (2.26), we immediately find the relevant Sobolev space for the expansion functions:

$$\mathcal{H}(\mathrm{rot};V) = \left\{ \mathbf{E} \in \left(\mathcal{L}^2(V)\right)^3 \;\middle|\; \nabla \times \mathbf{E} \in \left(\mathcal{L}^2(V)\right)^3 \right\}. \tag{2.32}$$

3. The expansion functions must enforce (dis)continuity relations of the field components, imposed by the physics of the problem, in a natural manner. For example, Maxwell's equations require continuity of the tangential components of the electric field **E** and magnetic field **H** at the interfaces of the cells of the finite element mesh, whereas the normal components of **E** and **H** will be discontinuous at cells' interfaces with different permittivities and permeabilities, respectively. Therefore, the Sobolev space $\mathcal{H}(\mathrm{rot};V)$ in (2.32) indeed provides the correct degree of smoothness. Similarly, when discretizing **D**, **B** or a current density **J**, we must require continuity of the normal field component, allowing the tangential component to be discontinuous. Therefore, suitable basis functions for these fields should belong to the Sobolev space $\mathcal{H}(\mathrm{div};V)$. The Sobolev space

$$\mathcal{H}^1(V) = \left\{ \phi \in \mathcal{L}^2(V) \;\middle|\; \nabla\phi \in \left(\mathcal{L}^2(V)\right)^3 \right\}. \tag{2.33}$$

of functions with continuous derivatives in all directions yields a too high degree of smoothness. The fact that expansion functions with $x-$, $y-$ and $z-$components chosen in $\mathcal{H}^1$ cannot represent jumps in the normal components of **E** and **H** (or jumps in the tangential components of **D**, **B** and **J**) will reduce the accuracy of the solution.

2.6.3 Application to the 1D wave equation

Consider the 1D wave equation (2.10), represented by its weak form (2.15) or functional (2.11). We first partition the interval $[0, L]$ into a set of cells or elements, being $n+1$ intervals by defining n internal nodes. Next we choose a set of shape functions or basis functions N_j belonging to the Sobolev space $\mathcal{H}^1([0,L])$, defined by (2.30). In addition, all shape functions should be valid candidate solutions in that they fulfill the boundary conditions. For example, for homogenous Dirichlet boundary conditions, they must disappear at the endpoints of the interval. The expansion is then written as

$$\phi(x) = \sum_{j=1}^{n} \phi_j N_j(x) \tag{2.34}$$

To determine the DoFs ϕ_j, we insert the expansion into the functional (2.11), which becomes a function of the DoFs,

$$J(f) = J(\phi_1, \phi_2, \ldots, \phi_n), \tag{2.35}$$

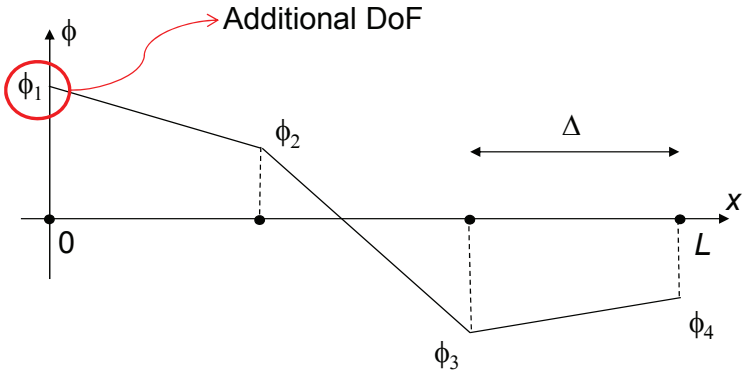

Figure 2.3: Piecewise linear approximation of a function $f(x)$ inside the interval $[0, L]$, in case of Neumann boundary conditions.

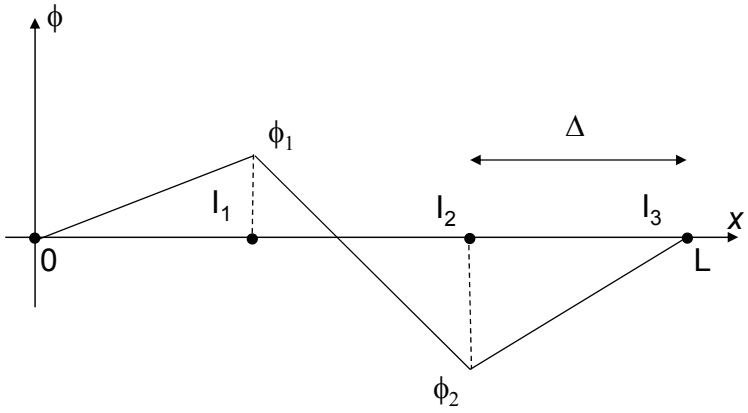

Figure 2.4: Piecewise linear approximation of a function $f(x)$ inside the interval $[0, L]$, in case of Dirichlet boundary conditions.

and, as the approximate solution to the wave equation, we look for the stationary point of this function w.r.t. these DoFs:

$$\frac{\partial}{\partial \phi_i} J(\phi_1, \phi_2, \ldots, \phi_n) = 0, \quad i = 1, \ldots, n. \tag{2.36}$$

Note that we limit our search for the stationary function of the functional to the space of functions spanned by the shape functions N_i. Specifically, we look for the optimal expansion coefficients ϕ_i, $i = 1, \ldots, n$. Within the set of admissible functions spanned by the basis functions N_i, a stationary point is characterized by requiring that all derivatives w.r.t. the expansion coefficients ϕ_i disappear. The stationarity can be seen to be equivalent to the solution of the linear system

$$\sum_{j=1}^{n} \phi_j \int_0^L \left(\frac{\partial N_i}{\partial x}(x) \frac{\partial N_j}{\partial x}(x) - k^2 N_i(x) N_j(x) \right) dx = \int_0^L N_i(x) g(x)\, dx, \quad i = 1, \ldots, n. \tag{2.37}$$

Let us now solve the eigenvalue problem, being the sourceless problem with $g(x) \equiv 0$, in the cavity $[0, L]$ w.r.t. the Dirichlet boundary conditions. These essential boundary conditions must be explicitly enforced in the points $x = 0$ and $x = L$, by not introducing any degrees of freedom at these points, as shown in Fig. 2.4. After discretization, we end up with the following $n-$dimensional eigenvalue problem

$$\sum_{j=1}^{n} S_{i,j}\,\phi_j = k^2 \sum_{j=1}^{n} T_{i,j}\,\phi_j, \quad i = 1, \ldots, n. \tag{2.38}$$

with

$$S_{i,j} = \int_0^L \frac{\partial N_i}{\partial x}(x)\frac{\partial N_j}{\partial x}(x)\,dx \tag{2.39}$$

and

$$T_{i,j} = \int_0^L N_i(x)N_j(x)\,dx. \tag{2.40}$$

The eigenvalues k^2 of this problem will approximate the eigenfrequencies of the cavity. More eigenfrequencies can be approximated accurately by increasing the number of shape functions. This will, of course, infer a higher computational cost.

2.6.4 Application to the Maxwell's equation

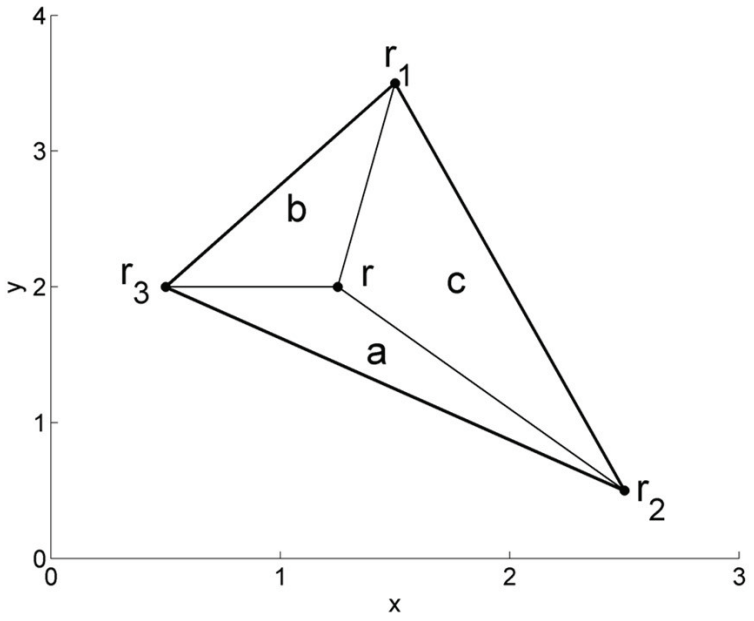

Figure 2.5: Description of a point inside a triangle by the barycentric coordinates L_1, L_2, L_3.

To solve the Maxwell's equations via the Finite Element Method, we resort to the weak-form formulation (2.25) or the functional (2.26). First, we expand the simulation

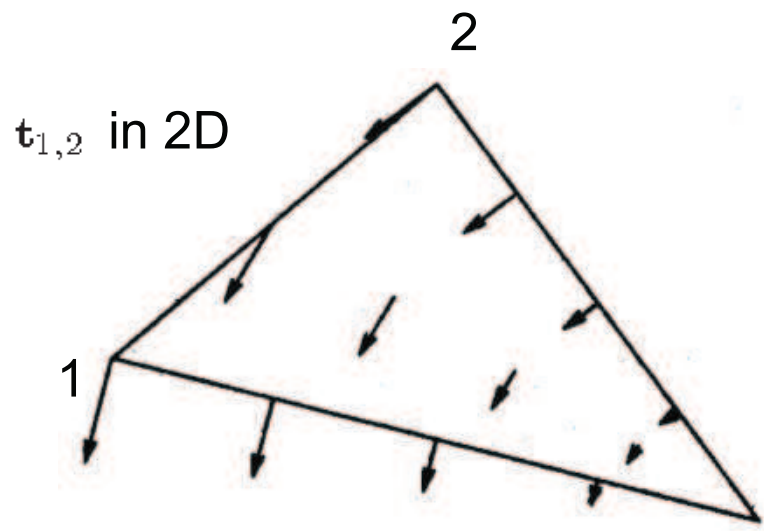

Figure 2.6: 2D edge element with DoF along edge 3 = segment $t_{1,2}$ defined on a triangle.

domain into triangles, in case of a two-dimensional problem, or into tetrahedrons, in case of a 3D problem. In case of a 2D mesh consisting of triangles, a high-quality mesh is obtained by using the Delaunay triangulation algorithm, which replaces two long sharp triangles by two better-shaped triangles. Generating a 3D mesh is more complex and we will not discuss this problem here. Next, we must choose a set of expansion functions for the electric field $\mathbf{E}$, belonging to the Sobolev space $\mathcal{H}(\text{rot}; X)$, with X being a surface S in 2D or a volume V in 3D. The applied first-order basis functions are called "edge elements". Their degree of freedom correspond to a constant tangential field component $\mathbf{E}_t$ along the edge of a triangle (in 2D) or tetrahedron (in 3D). Furthermore, an edge element will be constructed such that along all other edges of the triangle or tetrahedron, the tangential field component $\mathbf{E}_t$ is zero. In addition, the field values increase from zero in one corner point up to a maximum value at the edge opposite to that corner point. To express the expansion functions mathematically in each triangle or tetrahedron, we make use of barycentric coordinates. In 2D, the barycentric coordinates L_1, L_2, L_3 are tailored to the triangle as follows (Fig. 2.5):

$$L_1 = \frac{\text{Surface area of triangle a}}{\text{Surface area complete triangle A}} \tag{2.41}$$

$$L_2 = \frac{\text{Surface area of triangle b}}{\text{Surface area complete triangle A}} \tag{2.42}$$

$$L_3 = \frac{\text{Surface area of triangle c}}{\text{Surface area complete triangle A}} \tag{2.43}$$

each point within the triangle is then described by

$$\mathbf{r} = L_1\mathbf{r}_1 + L_2\mathbf{r}_2 + L_3\mathbf{r}_3 \tag{2.44}$$

with $\mathbf{r}_1, \mathbf{r}_2, \mathbf{r}_3$ the coordinates of the corner points and with $L_1 + L_2 + L_3 = 1$, since $a + b + c = A$. An edge element associated to edge i of triangle n is then described by

$$\mathbf{w}_{\text{t},n,i} = L_{i+1}\nabla_{xy}L_{i+2} - L_{i+2}\nabla_{xy}L_{i+1} \tag{2.45}$$

where subscripts should be taken modulo 3.

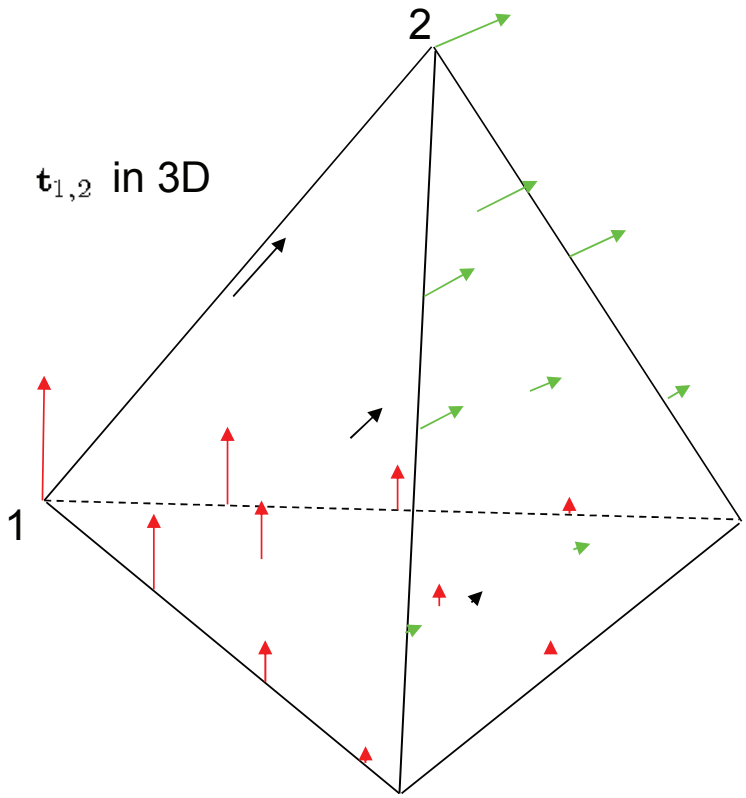

Figure 2.7: 3D edge element with DoF along edge segment $t_{1,2}$ defined on a tetrahedron.

In a similar fashion, we may define the barycentric coordinates in 3D, within a tetrahedron of the 3D finite element mesh. In formulas, the barycentric coordinate L_1 is given as proportions of tetrahedron volumes, by

$$L_1 = \frac{1}{6V} \det \begin{vmatrix} 1 & x & y & z \\ 1 & x_2 & y_2 & z_2 \\ 1 & x_3 & y_3 & z_3 \\ 1 & x_4 & y_4 & z_4 \end{vmatrix} \tag{2.46}$$

$$V = \frac{1}{6} \det \begin{vmatrix} 1 & x_1 & y_1 & z_1 \\ 1 & x_2 & y_2 & z_2 \\ 1 & x_3 & y_3 & z_3 \\ 1 & x_4 & y_4 & z_4 \end{vmatrix} \tag{2.47}$$

$$\nabla L_1 = \frac{1}{6V} \begin{pmatrix} 1 & \mathbf{u}_x & \mathbf{u}_y & \mathbf{u}_z \\ 1 & x_2 & y_2 & z_2 \\ 1 & x_3 & y_3 & z_3 \\ 1 & x_4 & y_4 & z_4 \end{pmatrix}, \tag{2.48}$$

where each point within the tetrahedron is described by

$$\mathbf{r} = L_1\mathbf{r}_1 + L_2\mathbf{r}_2 + L_3\mathbf{r}_3 + L_4\mathbf{r}_4 \tag{2.49}$$

with $\mathbf{r}_1, \mathbf{r}_2, \mathbf{r}_3, \mathbf{r}_4$ the coordinates of the corner points and with $L_1 + L_2 + L_3 + L_4 = 1$. An edge element associated to edge i of tetrahedron n is then described by

$$\mathbf{w}_{\mathrm{t},n,i} = L_{i+1}\nabla L_{i+2} - L_{i+2}\nabla L_{i+1} \tag{2.50}$$

where subscripts should be taken modulo 4.

We now replace the local numbering per triangle or per tetrahedron by a global number scheme and introduce the expansion in edge elements

$$\mathbf{E}(\mathbf{r}) = \sum_{j=1}^{n} E_{t,j}\mathbf{w}_t, \tag{2.51}$$

spanning a finite dimensional subspace of $\mathcal{H}(\mathrm{rot}; X)$. Note the vectorial character of each expansion function, whereas the DoF is scalar, being the value of the tangential field along edge j. Introducing this expansion in the weak-form (2.25), or after expressing stationarity of the functional (2.26), we obtain

$$\begin{aligned}\sum_{j=1}^{n} E_{t,j} \int_V (\nabla \times \mathbf{w}_\mathrm{t}) \cdot \overline{\overline{\mu}}^{-1} \cdot (\nabla \times \mathbf{w}_\mathrm{t})\, dV \\ -\omega^2 \sum_{j=1}^{n} E_{t,j} \int_V \mathbf{w}_\mathrm{t} \cdot \overline{\overline{\epsilon}} \cdot \mathbf{w}_\mathrm{t}\, dV = -j\omega \int_V \mathbf{w}_\mathrm{t} \cdot \mathbf{J}\, dV.\end{aligned} \tag{2.52}$$

This matrix system, with a sparse interaction matrix, can then be solved efficiently by a direct frontal elimination technique or an iterative solver.

2.6.5 Applying absorbing material as a boundary condition

By applying absorbing material at the outer boundaries of the simulation domain, one can try to minimize reflections at the perfect magnetically or electrically conducting walls. To obtain good absorption, we should take into account two conflicting aspects. On the one hand, one wants to apply very lossy materials to obtain maximal absorption with very thin additional sheets inside the simulation domain, to keep the number of additional unknowns as small as possible. On the other hand, one should keep the contrast between the absorber and the medium in which it is deployed small, to minimize reflections at the medium-absorber interface.

To unite these apparently conflicting requirements, Jean-Pierre Bérenger introduced the concept of the "perfectly matched layer" (PML). As explained in the chapter on FDTD, the absorbing character of these layers was originally obtained by modifying Maxwell's equations through the splitting of electric and magnetic veld into split-form equations within the PML. An identical result is obtained by applying coordinate stretching in the original Maxwell equations. In the frequency domain, one can prove that a PML is equivalent to a special type of dispersive (frequency-varying), uniaxial anisotropic material with material parameters

$$\overline{\overline{\mu}}(r) = \mu_i \begin{bmatrix} \kappa_z + \frac{\sigma_z}{j\omega\epsilon_0} & 0 & 0 \\ 0 & \kappa_z + \frac{\sigma_z}{j\omega\epsilon_0} & 0 \\ 0 & 0 & \left(\kappa_z + \frac{\sigma_z}{j\omega\epsilon_0}\right)^{-1} \end{bmatrix} \tag{2.53}$$

and

$$\bar{\bar{\epsilon}}(r) = \epsilon_i \begin{bmatrix} \kappa_z + \frac{\sigma_z}{j\omega\epsilon_0} & 0 & 0 \\ 0 & \kappa_z + \frac{\sigma_z}{j\omega\epsilon_0} & 0 \\ 0 & 0 & \left(\kappa_z + \frac{\sigma_z}{j\omega\epsilon_0}\right)^{-1} \end{bmatrix} \tag{2.54}$$

where (μ_i, ϵ_i) denote the material parameters of medium i from which the fields are incident onto the PML. Note that the wave impedance of the PML equals that of the neighboring medium, being $Z_i = \sqrt{\frac{\mu_i}{\epsilon_i}}$, such that no reflections occur at the interface. The parameter σ_z, with the z-direction perpendicular to the interface between medium i and the PML, generates the absorption of waves propagation through the PML. Moreover, $\kappa_z = 1$ yields the original Bérenger medium, whereas a material with $\kappa_z \geq 1$ also absorbs evanescent waves.

Chapter 3

Integral equations

3.1 Introduction

Integral equations are very versatile tools to solve linear field problems. We will focus only on frequency domain problems. Their main advantage is that they take into account the radiation boundary condition in a rigorous manner, without resorting to approximations such as PMLs, as proposed in Section 2.6.5.

Boundary integral equations, and their discretizations, also called boundary elements, are especially suited to homogeneous or piecewise homogeneous objects. In this case, they have the additional advantage that the currents and fields acting as unknowns must only be determined on the boundary, in contrast to the complete volume of the cavity or scatterer, as in finite elements or in finite differences approaches. For inhomogeneous objects, one can resort to volume integral equations. Even though the unknowns for these configurations are the fields defined in the whole object, integral methods are still popular owing to the perfect satisfaction of the radiation condition. Boundary and volume integral equations can also be solved in the time-domain. There, they give rise to Volterra-like equations.

Both in boundary and volume integral equations, the fundamental unknowns are sources that generate a field, computed through convolution with the Green function of the solution domain's medium. In a first step, we need to compute this Green function of the background medium. This fundamental solution for an impulse source will take into account the properties of that background medium, including boundary conditions, such as the Silver-Müller radiation condition. In practice, Green functions can only be calculated in an analytical or computationally efficient manner for homogeneous media and for multi-layered metal/dielectric background media. Second, we make use of field equivalence or of the Huygens principle to derive a representation formula that yields the fields in a certain subregion based on the evaluation of tangential fields or (virtual) currents on a boundary interface. Next, we obtain a boundary integral equation by the expressing boundary conditions of the fields at interfaces between subregions. In a final step, we discretize this integral equation by the Method of Moments, yielding a matrix system that can be solved by a computer. Typically, integral equations give rise to dense system matrices. The direct solution of dense matrix systems is typically limited to a few ten thousand unknowns. Therefore, to solve large problems, we resort to iterative techniques and fast solution schemes, such as the Fast Multipole Method.

3.2 Green functions

3.2.1 Scalar wave equation

The Green function corresponds to the impulse response of the wave equation

$$\nabla^2 G(\mathbf{r}, \mathbf{r}') + k^2 G(\mathbf{r}, \mathbf{r}') = -\delta(\mathbf{r} - \mathbf{r}'). \tag{3.1}$$

Since the wave equation we are solving is of second order, there are two linearly independent solutions. We pick the solution that exhibits the correct radiating behavior at infinity.

For the scalar Helmholtz equation in homogeneous spaces, the Green function is easily derived in closed form. In a 1D free-space environment, we have that $\mathbf{r} = z\mathbf{u}_z$, $\nabla = \frac{\partial}{\partial z}\mathbf{u}_z$, the source can be interpreted as a plane wave and the problem to be solved is that of, for example, a transmission line problem. The Green function becomes

$$G(z, z') = \frac{e^{-jk|z-z'|}}{2jk}. \tag{3.2}$$

Its behavior close to the source point is given by

$$G(z, z') = \frac{1}{2jk}. \tag{3.3}$$

Hence, the Green function is not singular in the origin. Nevertheless, it is not smooth in the origin either, as there is a discontinuity in its first derivative, due to the jump condition enforced by the Dirac source. In a 2D free-space environment, we have that $\mathbf{r} = \boldsymbol{\rho} = x\mathbf{u}_x + y\mathbf{u}_y$, $\nabla = \frac{\partial}{\partial x}\mathbf{u}_x + \frac{\partial}{\partial y}\mathbf{u}_y$ and the source can be interpreted as a line source. The Green function becomes

$$G(\boldsymbol{\rho}, \boldsymbol{\rho}') = -\frac{j}{4} H_0^{(2)}\left(k\left|\boldsymbol{\rho} - \boldsymbol{\rho}'\right|\right). \tag{3.4}$$

This 2D Green function, representing cylindrical wave propagation, exhibits a logarithmic singularity in the origin, being

$$\lim_{\boldsymbol{\rho}\to\boldsymbol{\rho}'} G(\boldsymbol{\rho}, \boldsymbol{\rho}') \sim -\frac{j}{4} - \frac{1}{2\pi}\left[\gamma + \ln\left(\frac{k\left|\boldsymbol{\rho} - \boldsymbol{\rho}'\right|}{2}\right)\right]. \tag{3.5}$$

At large distances, it decays as the inverse of the square root. This 2D Green function, representing cylindrical wave propagation, exhibits a logarithmic singularity in the origin, being

$$\lim_{k|\boldsymbol{\rho}-\boldsymbol{\rho}'|\to\infty} G(\boldsymbol{\rho}, \boldsymbol{\rho}') \sim \sqrt{\frac{2}{\pi k\left|\boldsymbol{\rho} - \boldsymbol{\rho}'\right|}} e^{-j\frac{\pi}{4}}\ e^{-jk\left|\boldsymbol{\rho}-\boldsymbol{\rho}'\right|}. \tag{3.6}$$

In a 3D free-space environment, the solution is given by

$$G(\mathbf{r}, \mathbf{r}') = \frac{e^{-jk|\mathbf{r}-\mathbf{r}'|}}{4\pi|\mathbf{r} - \mathbf{r}'|}. \tag{3.7}$$

The scalar three-dimensional Green function, pertinent, e.g., to an acoustical field problem, exhibits a $1/r$ singularity close to the origin

$$\lim_{\mathbf{r}\to\mathbf{r}'} G(\mathbf{r}, \mathbf{r}') \sim \frac{1}{4\pi|\mathbf{r} - \mathbf{r}'|}. \tag{3.8}$$

In the three-dimensional case, this function is also the solution to the Laplacian equation, or the static solution, obtained by taking the limit for k to zero. In 1D and 2D, this is not the case. At large distances, also a $1/r$ decay is observed, in correspondence to the radiation condition.

3.2.2 Vectorial wave equation

In Electromagnetics, the sources generating the Green function will be Hertzian dipoles or "point currents", which are vector impulse sources oriented along an arbitrary orientation $\mathbf{u}$. To account for all possible orientations, the Green function will be a 3×3 tensor, found by solving the vector wave equation

$$\nabla \times \left[\nabla \times \overline{\overline{G}}(\mathbf{r}, \mathbf{r}') \right] - k^2 \overline{\overline{G}}(\mathbf{r}, \mathbf{r}') = -\overline{\overline{I}} \delta(\mathbf{r} - \mathbf{r}'), \tag{3.9}$$

with $\overline{\overline{I}}$ the unit dyadic. In 3D, the Green dyadic is related to the scalar Green function by applying the following differential operator

$$\overline{\overline{G}}(\mathbf{r}, \mathbf{r}') = - \begin{pmatrix} 1 + \frac{1}{k^2} \frac{\partial^2}{\partial x^2} & \frac{1}{k^2} \frac{\partial^2}{\partial x \partial y} & \frac{1}{k^2} \frac{\partial^2}{\partial x \partial z} \\ \frac{1}{k^2} \frac{\partial^2}{\partial x \partial y} & 1 + \frac{1}{k^2} \frac{\partial^2}{\partial y^2} & \frac{1}{k^2} \frac{\partial^2}{\partial y \partial z} \\ \frac{1}{k^2} \frac{\partial^2}{\partial x \partial z} & \frac{1}{k^2} \frac{\partial^2}{\partial y \partial z} & 1 + \frac{1}{k^2} \frac{\partial^2}{\partial z^2} \end{pmatrix} \frac{e^{-jk|\mathbf{r}-\mathbf{r}'|}}{4\pi|\mathbf{r} - \mathbf{r}'|}. \tag{3.10}$$

Note that taking the derivatives of the already singular Green function makes the components of this dyadic even more singular in the source point, even up to a point where its singularities cease to be integrable. Extreme care needs to be applied in the interpretation of integrals with this Green dyadic in the integrand.

3.3 Integral representation of the wave equation

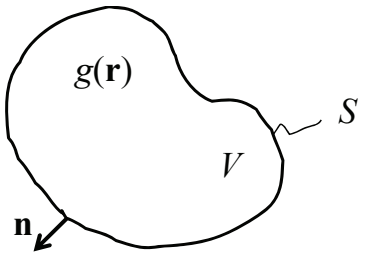

Figure 3.1: Pertinent to the scalar wave equation: Volume V with boundary S and external surface normal $\mathbf{n}$, in which a source $g(\mathbf{r})$ is located.

As in previous section, let us start from the scalar wave equation, which now holds in a volume V bounded by an interface S, as shown in Fig. 3.1, but now with a general source term $g(\mathbf{r})$ in its right-hand side

$$\nabla^2 f(\mathbf{r}) + k^2 f(\mathbf{r}) = g(\mathbf{r}) \tag{3.11}$$

and combine this with (3.1), repeated here for convenience:

$$\nabla^2 G(\mathbf{r}, \mathbf{r}') + k^2 G(\mathbf{r}, \mathbf{r}') = -\delta(\mathbf{r} - \mathbf{r}'). \tag{3.12}$$

Multiplying (3.11) by $G(\mathbf{r},\mathbf{r}')$, (3.12) by $f(\mathbf{r})$, and then subtracting the results yields

$$G(\mathbf{r},\mathbf{r}')\nabla^2 f(\mathbf{r}) - f(\mathbf{r})\nabla^2 G(\mathbf{r},\mathbf{r}') = G(\mathbf{r},\mathbf{r}')g(\mathbf{r}) + f(\mathbf{r})\delta(\mathbf{r}-\mathbf{r}'). \tag{3.13}$$

Integrating over the volume V

$$\int_V \left[G(\mathbf{r},\mathbf{r}')\nabla^2 f(\mathbf{r}) - f(\mathbf{r})\nabla^2 G(\mathbf{r},\mathbf{r}')\right]\, dV = \int_V G(\mathbf{r},\mathbf{r}')g(\mathbf{r})\, dV + f(\mathbf{r}'), \tag{3.14}$$

allows us to apply Gauss theorem to express the solution $f(\mathbf{r}')$ of the wave equation (3.11) in all $\mathbf{r}'$ of V as a function of the solution on the boundary S:

$$\oint_S \left[G(\mathbf{r},\mathbf{r}')\frac{\partial}{\partial n} f(\mathbf{r}) - f(\mathbf{r})\frac{\partial}{\partial n} G(\mathbf{r},\mathbf{r}')\right]\, dS = \int_V G(\mathbf{r},\mathbf{r}')g(\mathbf{r})\, dV + f(\mathbf{r}'), \tag{3.15}$$

with $\mathbf{n}$ the external normal to V and $\frac{\partial}{\partial n} = \mathbf{n}\cdot\nabla$. This formula is called a *representation formula* for the field $f(\mathbf{r}')$ within V. It can be computed from the source term $g(\mathbf{r})$ and the so-called *Cauchy data* on the boundary S of V, being $f(\mathbf{r}')$ on the boundary, and its normal derivative on S. The kernel function appearing in the integrals is the Green function for an infinite homogeneous space. Let us now simplify notation by interchanging the roles of $\mathbf{r}$ and $\mathbf{r}'$, making the dependence on the distance $|\mathbf{r}-\mathbf{r}'|$ explicit in the Green function, and writing $\partial_{n'} = \frac{\partial}{\partial n'} = \mathbf{n}'\cdot\nabla'$ for the derivative with respect to the external normal direction and

$$f^{\mathrm{inc}}(\mathbf{r}) = -\int_V G(|\mathbf{r}-\mathbf{r}'|)g(\mathbf{r}')\, dV' \tag{3.16}$$

for the source term. We obtain:

$$f(\mathbf{r}) = -\oint_S \partial_{n'} G(|\mathbf{r}-\mathbf{r}'|)f(\mathbf{r}')\, dS' + \oint_S G(|\mathbf{r}-\mathbf{r}'|)\partial_{n'} f(\mathbf{r}')\, dS' + f^{\mathrm{inc}}(\mathbf{r}). \tag{3.17}$$

Conceptually, it is also relatively easy to also express the gradient of the field $f(\mathbf{r})$ by an integral representation formula

$$\begin{aligned}\nabla f(\mathbf{r}) = &-\oint_S \nabla\left(\partial_{n'} G(|\mathbf{r}-\mathbf{r}'|)\right) f(\mathbf{r}')\, dS' \\ &+ \oint_S \nabla G(|\mathbf{r}-\mathbf{r}'|)\partial_{n'} f(\mathbf{r}')\, dS' + \nabla f^{\mathrm{inc}}(\mathbf{r}).\end{aligned} \tag{3.18}$$

However, extreme care has to be taken when evaluating the integrals, as they may contain strong singularities when $\mathbf{r}'$ approaches the boundary S. This will be the topic of next section.

3.4 Boundary Integral Equation

Two more steps are necessary to transform an integral representation of the fields in a volume V, as derived in Section 3.3, into an integral equation. First, we must take the limit of observation point $\mathbf{r}$ in V approaching the boundary S. This is not a trivial step, since, given the singular behavior of the Green kernel function, the pertinent integrals contain singularities. Second, we express the boundary conditions holding on the boundary S to come to an integral equation in which the boundary fields and their normal derivatives act as unknowns.

3.4.1 Treatment of singularities in the integrals

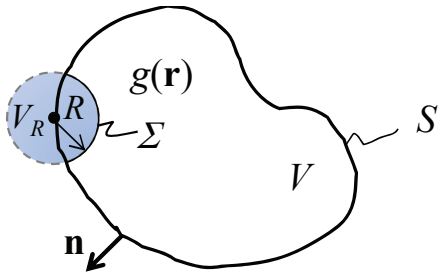

Figure 3.2: Splitting the integral into a regular and a selfpatch part: Isolating the singularity by defining a spherical exclusion area V_R with boundary Σ around the observation point.

Let us now first discuss the different integrals and the singularities involved in the evaluation process when we take the limit for an observation point approaching the boundary S in the field representation formula (3.17), evaluating

$$\begin{aligned}\lim_{\mathbf{r}\to S} f(\mathbf{r}) = &- \lim_{\mathbf{r}\to S} \oint_S \partial_{n'} G(|\mathbf{r}-\mathbf{r}'|) f(\mathbf{r}')\, dS' \\ &+ \lim_{\mathbf{r}\to S} \oint_S G(|\mathbf{r}-\mathbf{r}'|) \partial_{n'} f(\mathbf{r}')\, dS' + \lim_{\mathbf{r}\to S} f^{\mathrm{inc}}(\mathbf{r}),\end{aligned} \tag{3.19}$$

and for (3.18), calculating

$$\begin{aligned}\lim_{\mathbf{r}\to S} \mathbf{n}\cdot\nabla f(\mathbf{r}) &= \lim_{\mathbf{r}\to S} \partial_n f(\mathbf{r}) \\ &= - \lim_{\mathbf{r}\to S} \oint_S \partial_n \partial_{n'} G(|\mathbf{r}-\mathbf{r}'|) f(\mathbf{r}')\, dS' \\ &\quad + \lim_{\mathbf{r}\to S} \oint_S \partial_n G(|\mathbf{r}-\mathbf{r}'|) \partial_{n'} f(\mathbf{r}')\, dS' + \partial_n f^{\mathrm{inc}}(\mathbf{r}).\end{aligned} \tag{3.20}$$

As for the integral

$$\lim_{\mathbf{r}\to S} \int_S G(|\mathbf{r}-\mathbf{r}'|) \partial_{n'} f(\mathbf{r}') dS' = \int_S G(|\mathbf{r}-\mathbf{r}'|) \partial_{n'} f(\mathbf{r}') dS', \tag{3.21}$$

the Green function's singularity is integrable, no matter where the observation point $\mathbf{r}$ is located. Therefore, the limit can simply be removed. However, the integral

$$\lim_{\mathbf{r}\to S} \int_S \partial_{n'} G(|\mathbf{r}-\mathbf{r}'|) f(\mathbf{r}') dS' \tag{3.22}$$

requires a much more careful analysis. To isolate the singularity, we introduce a sphere with radius R, volume V_R and surface $\Sigma = \delta V_R = \Sigma_R \bigcup S_R$, centered around the observation point $\mathbf{r}$ on the interface S, as shown in Fig. 3.2. Assume that the volume of the sphere V_R inside V is bounded by the surface Σ_R. We then split up the integral over S into a regular part over $S-\Sigma$, by making the detour around the sphere's surface,

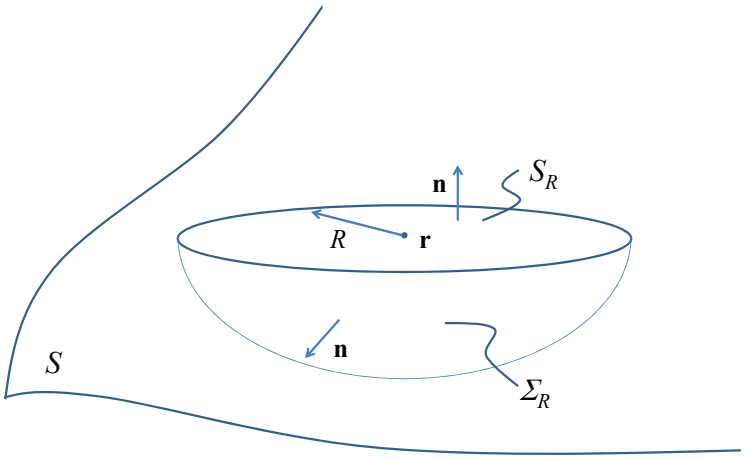

Figure 3.3: Evaluation of the selfpatch part: Integrating over the flat disc S_R and the hemisphere Σ_R.

and the selfpatch part contained within the integration over the surface Σ, yielding

$$\lim_{\mathbf{r}\to S}\int_S \partial_{n'}G(|\mathbf{r}-\mathbf{r}'|)f(\mathbf{r}')dS' = \lim_{R\to 0}\int_{S-\Sigma}\partial_{n'}G(|\mathbf{r}-\mathbf{r}'|)f(\mathbf{r}')dS' + \lim_{R\to 0}\int_{\Sigma}\partial_{n'}G(|\mathbf{r}-\mathbf{r}'|)f(\mathbf{r}')dS'. \tag{3.23}$$

The limit of the first, regular, term can be taken without any problem. To evaluate the second term, we carefully take the limit on the infinitesimally small surface $\Sigma = S_R + \Sigma_R$. When the surface S is sufficiently regular around point $\mathbf{r}$, it may be considered flat in an infinitesimally small area around $\mathbf{r}$, and S_R looks like a disc, as shown in Fig. 3.3. The limit reduces to

$$\begin{aligned}\lim_{R\to 0}\int_{\Sigma}\partial_{n'}G(|\mathbf{r}-\mathbf{r}'|)f(\mathbf{r}')dS' &= \lim_{R\to 0}\int_{S_R+\Sigma_R}\mathbf{n}'\cdot\nabla' G(|\mathbf{r}-\mathbf{r}'|)f(\mathbf{r}')dS' \\ &= f(\mathbf{r})\lim_{R\to 0}\int_{S_R+\Sigma_R}\mathbf{n}'\cdot\mathbf{u}_R\left(-jk\frac{e^{-jkR}}{4\pi R}-\frac{e^{-jkR}}{4\pi R^2}\right)R^2 d\Omega \\ &= -\frac{f(\mathbf{r})}{4\pi}\lim_{R\to 0}\int_{\Sigma_R}d\Omega = -\frac{f(\mathbf{r})}{2}\end{aligned} \tag{3.24}$$

In the last step, we made use of the fact that $\mathbf{n}'\cdot\mathbf{u}_R = 0$ on S_R and $\mathbf{n}'\cdot\mathbf{u}_R = 1$ on Σ_R, resulting in a final integral that equals the solid angle of half a sphere. This leads to the final result

$$\lim_{\mathbf{r}\to S}\int_S \partial_{n'}G(|\mathbf{r}-\mathbf{r}'|)f(\mathbf{r}')dS' = \int_S \partial_{n'}G(|\mathbf{r}-\mathbf{r}'|)f(\mathbf{r}')dS' - \frac{1}{2}f(\mathbf{r}), \tag{3.25}$$

where we should not forget that the first integral must be interpreted in the sense outlined in (3.23). Applying the same process, we obtain for the second integral appearing in (3.20):

$$\lim_{\mathbf{r}\to S}\int_S \partial_n G(|\mathbf{r}-\mathbf{r}'|)\partial_{n'}f(\mathbf{r}')dS' = \int_S \partial_n G(|\mathbf{r}-\mathbf{r}'|)\partial_{n'}f(\mathbf{r}')dS' + \frac{1}{2}\partial_n f(\mathbf{r}). \tag{3.26}$$

The minus sign in the correction term turns into a plus sign, as the derivative is taken with respect to $\mathbf{r}$ instead of $\mathbf{r}'$. The last remaining integral in (3.20) requires an even more careful treatment as the singularity of the double normal derivative of the Green's function is much too strong to be integrable in the usual sense. We must first carefully evaluate the integral while keeping the observation point away from the singularity point, after which the limit of the resulting integration remain finite only if $f(\mathbf{r})$ is sufficiently smooth. In that case, the value of this integral does not depend on the direction from which we approach the surface. We do not go into detail here, but simply write

$$\lim_{\mathbf{r}\to S}\int_S \partial_n\partial_{n'}G(|\mathbf{r}-\mathbf{r}'|)f(\mathbf{r}')dS' = \int_S \partial_n\partial_{n'}G(|\mathbf{r}-\mathbf{r}'|)f(\mathbf{r}')dS'. \tag{3.27}$$

Owing to the symmetry of the Green function, we do not obtain any additional contributions as for the previous integrals. Making use of all these results, we may now cast the representation formulas in operator form, yielding

$$\begin{pmatrix} f \\ \partial_n f \end{pmatrix} = \begin{pmatrix} \frac{1}{2}-K' & S \\ -D & \frac{1}{2}+K \end{pmatrix}\begin{pmatrix} f \\ \partial_n f \end{pmatrix} + \begin{pmatrix} f^{inc} \\ \partial_n f^{inc} \end{pmatrix}, \tag{3.28}$$

with

$$Sf(\mathbf{r}) = \int_S G(|\mathbf{r}-\mathbf{r}'|)\partial_{n'}f(\mathbf{r}')dS' \tag{3.29}$$

$$K'f(\mathbf{r}') = \int_S \partial_{n'}G(|\mathbf{r}-\mathbf{r}'|)f(\mathbf{r}')dS' \tag{3.30}$$

$$K\partial_n f(\mathbf{r}) = \int_S \partial_n G(|\mathbf{r}-\mathbf{r}'|)\partial_{n'}f(\mathbf{r}')dS' \tag{3.31}$$

$$Df(\mathbf{r}) = \int_S \partial_n\partial_{n'}G(|\mathbf{r}-\mathbf{r}'|)f(\mathbf{r}')dS'. \tag{3.32}$$

On the one hand, this representation is valid for all solutions to the Helmholtz equation (3.11) in V. On the other hand, if a pair $(f, \partial_n f)$ can be found for which these formulas hold on S, their extension into the volume V will solve the wave equation. Similar formulas can be constructed for the region outside of V. Do not forget that some of the integral operators are extremely singular. For example, the D operator contains contributions of $\frac{1}{R^3}$ in 3D and $\frac{1}{R^2}$ in 2D. These integrands are not integrable over a surface and a line, respectively. Each time we have to deal with such an integral representation, we must interpret them in a correct manner, that is, positioning the observation point at a small distance from the boundary (on the inside of V), perform the integration, and take the limit of this result afterwards. Note that the representation formulas are a mathematical version of Huygens principle and of the introduction of equivalent currents on the boundary of a solution domain, which are well-known principles in Acoustics and Electromagnetics.

3.4.2 Applying the boundary conditions

To transform the representation formulas for a domain V bounded by S into integral equations that have unique solutions, we introduce the boundary conditions on the surface S. Since representations exist for both the boundary value $f(\mathbf{r})$ on S and its

normal derivative $\partial_n f(\mathbf{r})$ on S, we obtain two equations for the Dirichlet case, by setting $f(\mathbf{r}) = 0$ on S, yielding

$$S\partial_n f = -f^{inc} \tag{3.33}$$

$$\left(\frac{1}{2} - K\right)\partial_n f = \partial_n f^{inc}, \tag{3.34}$$

and two equations for the Neumann case, by imposing $\partial_n f(\mathbf{r}) = 0$ on S, to obtain

$$\left(\frac{1}{2} + K'\right) f = f^{inc} \tag{3.35}$$

$$Df = \partial_n f^{inc}. \tag{3.36}$$

Since the integral equations involved lack regularity, their spectrum is not correctly predicted by the theory of compact integral operators, outlined in Section 1.5. Moreover, these operators are not self-adjoint, because the Green function contains the imaginary unit. Yet, although not mathematically correct, (3.33) and (3.36) are often called equations of the first kind, whereas (3.34) and (3.35) are denoted equations of the second kind.

The above equations were constructed for a bounded domain, such that we did not have to worry about contributions at infinity in the application of the Green theorem. Although out of the scope of the course, it can be proven that for unbounded regions, the contribution of the boundary surface at infinity goes to zero thanks to the radiation condition embedded in the Green function kernel. Therefore, the representation formulas hold for unbounded domains, provided the external normal is chosen correctly.

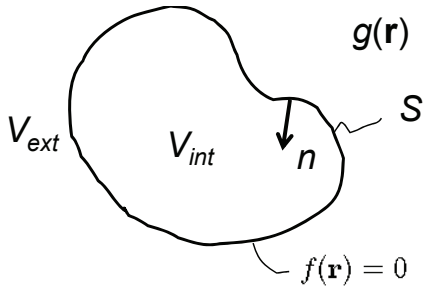

Figure 3.4: The solution of the Dirichlet equation of the first kind in $V_{\rm ext}$ is non-unique at Dirichlet eigenfrequencies of $V_{\rm t}$, bounded by S, since Dirchlet eigenfunctions of the inner problem may be added to the external region's solution.

Note that integral equations (3.33), (3.36),(3.34) and (3.35), which only involve one unknown on the boundary S, cannot be solved uniquely at certain discrete frequencies that correspond to the eigenfrequencies of the cavity S. To demonstrate the problem, we focus on the Dirichlet equation of the first kind (3.33), for the volume *external* to V, which may be written as

$$\lim_{\mathbf{r}\to S} \int_S G(|\mathbf{r} - \mathbf{r}'|) \frac{\partial}{\partial n'} f(\mathbf{r}')\, dS' = -f^{inc} \tag{3.37}$$

Now consider the sourceless wave equation in V, hence for the interior problem, at an eigenfrequency with corresponding wave number k_0:

$$\nabla^2 f_0(\mathbf{r}) + k_0^2 f_0(\mathbf{r}) = 0. \tag{3.38}$$

At k_0, this equation has a nontrivial solution corresponding to the Dirichlet eigenfunction $f_0(\mathbf{r})$, which fulfills the inner representation formulas, hence:

$$\lim_{\mathbf{r}\to S} \int_S G(|\mathbf{r}-\mathbf{r}'|) \frac{\partial}{\partial n'} f_0(\mathbf{r}')\, dS' = 0. \tag{3.39}$$

Adding (3.37) and (3.39) proves that $f(\mathbf{r}) + f_0(\mathbf{r})$ also satisfies (3.33), making its solution non-unique. Since $f_0(\mathbf{r})$ is an eigenfunction of the internal problem, the non-uniqueness issue is denoted the problem of *internal resonances*. One way of avoiding internal resonances consists of combining the first and second kind integral equations. The resulting combined integral equation does not support any internal resonances and has therefore a unique solution at all frequencies.

3.5 Application to Maxwell's equations

Let us now apply the above theory to the Maxwell equations. We first discuss the two-dimensional case, distinguishing between TE and TM polarized sources. Then, we turn our attention to the three-dimensional configuration.

3.5.1 The 2D TM problem

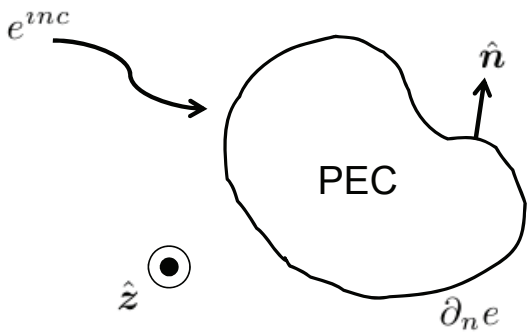

Figure 3.5: Scattering of a plane wave by a perfect electric conducting cylinder S: TM case with the plane wave's electric field polarized along the z-axis.

Consider a two-dimensional electromagnetic TM-problem, where a plane wave is incident on a perfect electric conducting cylinder S. The electric field and the induced current on the scatterer are directed along the z-axis, being the axis of the cylinder. In that case, the magnetic field is completely transversal.

The problem is governed by the homogeneous two-dimensional Helmholtz equation

$$(\Delta + k^2) e(\mathbf{r}) = 0 \tag{3.40}$$

$$e|_S = -\, e^{inc}\big|_S \tag{3.41}$$

where the second condition expresses that the total electric field (being the sum of the scattered and incident electric field) must be zero to satisfy the PEC boundary condition on S. Hence, we are dealing with a Dirichlet problem. The z-component (and only

component) of the electric field $\mathbf{e}(\mathbf{r}) = e(\mathbf{r})\hat{\boldsymbol{z}}$ takes up the role of $f(\mathbf{r})$. Given the analysis of previous section, we find the two following integral equations to model this type of scattering

$$\left(\frac{1}{2} - K\right)\partial_n e = \partial_n e^{inc} \tag{3.42}$$

$$S\partial_n e = -e^{inc} \tag{3.43}$$

The first equation has the normal derivative of the incident electric field in its right hand side, the second one the incident electric field itself. We now make use of Maxwell's equations to transform the normal derivatives of the z-directed fields to some more conventional field quantities. The normal derivative of the electric field is transformed into the magnetic field, as follows

$$\begin{aligned}
\nabla \times \boldsymbol{e} &= -\jmath\omega\mu\boldsymbol{h} \\
\nabla e \times \hat{\boldsymbol{z}} &= -\jmath\omega\mu\boldsymbol{h} \\
\hat{\boldsymbol{z}} \times (\nabla e \times \hat{\boldsymbol{z}}) &= -\jmath\omega\mu\hat{\boldsymbol{z}} \times \boldsymbol{h} \\
\nabla e - (\hat{\boldsymbol{z}} \cdot \nabla e)\,\hat{\boldsymbol{z}} = \nabla_{xy} e &= -\jmath\omega\mu\hat{\boldsymbol{z}} \times \boldsymbol{h} \\
\partial_n e &= -\jmath\omega\mu\hat{\boldsymbol{n}} \cdot (\hat{\boldsymbol{z}} \times \boldsymbol{h}) \\
&= \jmath\omega\mu\hat{\boldsymbol{z}} \cdot (\hat{\boldsymbol{n}} \times \boldsymbol{h}) \\
&= \jmath\omega\mu j
\end{aligned}$$

with $\jmath$ the imaginary unit and $\hat{\boldsymbol{z}}$ the unit vector along the axis of the cylinder. We find that on S the normal derivatives of the z-directed field is parallel to the current j induced on the surface of the scatterer. This result enables us to reformulate the integral equations as

$$\left(\frac{1}{2} - K\right) j = \hat{\boldsymbol{z}} \cdot (\hat{\boldsymbol{n}} \times \boldsymbol{h}^{inc}) \tag{3.44}$$

$$Sj = -\frac{1}{\jmath\omega\mu} e^{inc} \tag{3.45}$$

We obtain two equations: one having the incident magnetic field in its right hand side, hence called the magnetic field integral equation or MFIE, and one having the incident electric field in its right hand side, therefore the electric field integral equation or EFIE.

3.5.2 The 2D TE problem

The dual TE case can be treated in exactly the same manner. The pertinent equation and boundary condition are

$$(\Delta + k^2)h = 0 \tag{3.46}$$

$$\partial_n h|_S = -\,\partial_n h^{inc}\big|_S \tag{3.47}$$

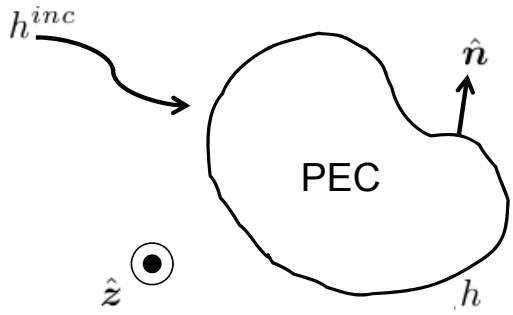

Figure 3.6: Scattering of a plane wave by a perfect electric conducting cylinder: TE case with the plane wave's magnetic field oriented along the z-axis.

As for the TM case, we relate the normal derivative of the magnetic field $\mathbf{h}(\mathbf{r}) = h(\mathbf{r})\hat{\boldsymbol{z}}$ to tangential components of the electric field, as follows

$$\begin{aligned}
\nabla \times \boldsymbol{h} &= \jmath\omega\epsilon\boldsymbol{e} \\
\nabla h \times \hat{\boldsymbol{z}} &= \jmath\omega\epsilon\boldsymbol{e} \\
\nabla_{xy} h &= \jmath\omega\epsilon\hat{\boldsymbol{z}} \times \boldsymbol{e} \\
\partial_n h &= \jmath\omega\epsilon\hat{\boldsymbol{n}} \cdot (\hat{\boldsymbol{z}} \times \boldsymbol{e}) \\
&= -\jmath\omega\epsilon\hat{\boldsymbol{z}} \cdot (\hat{\boldsymbol{n}} \times \boldsymbol{e})
\end{aligned}$$

This relation transforms the PEC boundary condition into a Neumann boundary condition on the magnetic field. Hence, we obtain the following integral equations

$$\left(\frac{1}{2} + K'\right) h = h^{inc} \tag{3.48}$$

$$Dh = \partial_n h^{inc} \tag{3.49}$$

making use of $\mathbf{j} = \hat{\boldsymbol{n}} \times \hat{\boldsymbol{h}} = h\hat{\boldsymbol{n}} \times \hat{\boldsymbol{z}}$, we can rewrite these equations in terms of the induced current $\mathbf{j} = j\hat{\boldsymbol{t}} = -h\hat{\boldsymbol{t}}$, with $\hat{\boldsymbol{t}} = \hat{\boldsymbol{z}} \times \hat{\boldsymbol{n}}$, which is tangential to S in the xy-plane:

$$\left(\frac{1}{2} + K'\right) j = -h^{inc} \tag{3.50}$$

$$Dj = \jmath\omega\epsilon\hat{\boldsymbol{z}} \cdot (\hat{\boldsymbol{n}} \times \boldsymbol{e}^{inc}) \tag{3.51}$$

Again, one may solve any of the two Neumann-condition boundary integral equations. In the TE case, the "second kind" equation is the MFIE (3.50), the "first kind" equation is the EFIE (3.51).

3.5.3 The 3D problem

In the 3D EM case, we need to solve the vectorial Helmholtz equation

$$\nabla \times \nabla \times \boldsymbol{e} - k^2\boldsymbol{e} = -\jmath\omega\mu\boldsymbol{j} \tag{3.52}$$

Hence, we need to develop representation formulas that take into account the vectorial nature of the fields. The details are out of the scope of this course, but we can proceed

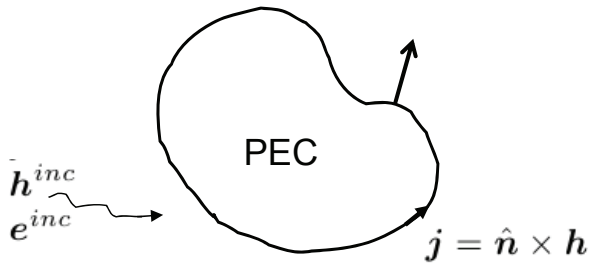

Figure 3.7: Scattering of a plane wave by a perfect electric conducting volume V due to induced current $\hat{\boldsymbol{n}} \times \boldsymbol{h}$ on the boundary S.

just as in the scalar case. This means that by cross multiplying the wave equations with the defining equation for the Green dyadic and by applying a suitable Green-like theorem to reduce integrations to the boundary, we arrive at the representation formula

$$\begin{pmatrix} \boldsymbol{e} \times \hat{\boldsymbol{n}} \\ \hat{\boldsymbol{n}} \times \boldsymbol{h} \end{pmatrix} = \begin{pmatrix} \frac{1}{2} + K & -\eta T \\ T/\eta & \frac{1}{2} + K \end{pmatrix} \cdot \begin{pmatrix} \boldsymbol{e} \times \hat{\boldsymbol{n}} \\ \hat{\boldsymbol{n}} \times \boldsymbol{h} \end{pmatrix} + \begin{pmatrix} \boldsymbol{e}^{inc} \times \hat{\boldsymbol{n}} \\ \hat{\boldsymbol{n}} \times \boldsymbol{h}^{inc} \end{pmatrix} \tag{3.53}$$

with $\eta = \sqrt{\frac{\mu}{\epsilon}}$ the wave impedance. Thanks to duality, only two operators appear

$$T\boldsymbol{f}(\mathbf{r}) = \frac{1}{jk}\hat{\boldsymbol{n}} \times \int_S \nabla \times \nabla \times G(|\mathbf{r} - \mathbf{r}'|)\boldsymbol{f}(\mathbf{r}')dS'$$
$$K\boldsymbol{f}(\mathbf{r}) = \hat{\boldsymbol{n}} \times \int_S \nabla \times G(|\mathbf{r} - \mathbf{r}'|)\boldsymbol{f}(\mathbf{r}')dS'.$$

Now, the Cauchy data are the tangential electric and magnetic fields. If and only if a pair of boundary values belong to a solution to the vectorial Helmholtz equation, they fulfill these representation formulas. Note that the kernel of the operator T is, up to a factor, the Green dyadic we already encountered. For a 2D problem, the operators T en K fall apart in two decoupled components, giving rise to the operators we already encountered in the previous subsections.

As for scattering at a 3D PEC object with boundary S, we impose as boundary conditions $\hat{\boldsymbol{n}} \times \boldsymbol{e} = \boldsymbol{0}$ and $\hat{\boldsymbol{n}} \times \boldsymbol{h} = \boldsymbol{j}$ on S. Substituting these conditions into the representation formulas yields

$$-\eta T\boldsymbol{j} = -\boldsymbol{e}^{inc} \times \hat{\boldsymbol{n}} \tag{3.54}$$
$$\left(\frac{1}{2} + K\right)\boldsymbol{j} = -\hat{\boldsymbol{n}} \times \boldsymbol{h}^{inc} \tag{3.55}$$

being the EFIE, often denoted as first kind, and the MFIE, often named second kind, respectively. Also these equations suffer from internal resonances. Forming a combined field integral equation again solves the problem.

3.6 Method of Moments

Discretizing an integral equation closely follows the steps we already encountered in Chapter 2. This entails first subdividing the solution space, being the boundaries of

the solution domain, thanks to the representation formulas with Green function kernel. Next, we propose the form of a candidate solution in each element by selecting a set of basis functions from the correct function space, plus any global continuity constraints. Finally, we select the degrees of freedom to expand the solution into a linear combination of these basis functions. Then, this candidate solution is substituted in the integral equation, after which, following the Galerkin scheme, it is tested by multiplying with each of the shape functions and integrating. This process yields a square system of linear equations that we can solve using direct or iterative solvers. These are exactly the steps already discussed in the course Electromagnetism II. Let us briefly repeat them for a two-dimensional scalar Fredholm equation of the first kind.

3.6.1 Example: Two-dimensional problem

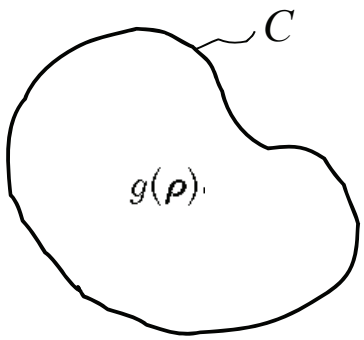

Figure 3.8: Pertinent to the two-dimensional scalar Fredholm equation of the first kind: cylinder S with boundary C containing a source $g(\boldsymbol{\rho})$.

Consider the two-dimensional scalar Fredholm equation of the first kind

$$\lim_{\mathbf{r}\to C} \oint_C G(\boldsymbol{\rho}|\boldsymbol{\rho}') f(\boldsymbol{\rho}')\, dc' = \lim_{\mathbf{r}\to C} g(\boldsymbol{\rho}) \tag{3.56}$$

on a boundary C, for some kind of excitation $g(\boldsymbol{\rho})$ within the surface S. In a first step, we subdivide the contour C into N segments, as shown in Fig. 3.9. For ease of implementation, we choose them straight. Second, we select a suitable subspace in which to construct an expansion for the candidate solution, based on the physical properties of the unknown field $f(\boldsymbol{\rho})$ on S. Assuming that no additional continuity is required, a candidate solution may be piecewise constant, hence the proposed representation is a constant field value in each segment. Therefore, the degrees of freedom may be chosen the function values in the middle of each segment. In this case, trivial shape functions are obtained, being pulses spanning one segment, taking the value one on that segment, and zero on all others. We thus obtain a basis of N shape functions $p_i(\boldsymbol{\rho}), i = 1, \ldots, N$, spanning the approximate solution subspace. Now expanding the unknown function $f(\boldsymbol{\rho}')$ into a linear combination of these N basis or shape functions yields

$$f(\boldsymbol{\rho}) = \sum_{j=1}^{N} f_j p_j(\boldsymbol{\rho}). \tag{3.57}$$

Introducing this candidate solution into (3.56) and imposing the integral equation in a weak sense by weighting with a set of test functions yields

$$\sum_{j=1}^{N} f_j \int_{\Delta_i} w_i(\boldsymbol{\rho}) \int_{\Delta_j} G(\boldsymbol{\rho}|\boldsymbol{\rho}') p_j(\boldsymbol{\rho}')\, dc'\, dc = \int_{\Delta_i} w_i(\boldsymbol{\rho}) g(\boldsymbol{\rho})\, dc. \tag{3.58}$$

The end result is a matrix equation

$$\sum_{j=1}^{N} K_{i,j} f_j = g_i \tag{3.59}$$

with

$$K_{i,j} = \int_{\Delta_i} w_i(\boldsymbol{\rho}) \int_{\Delta_j} G(\boldsymbol{\rho}|\boldsymbol{\rho}') p_j(\boldsymbol{\rho}')\, dc'\, dc \tag{3.60}$$

and

$$g_i = \int_{\Delta_i} w_i(\boldsymbol{\rho}) g(\boldsymbol{\rho})\, dc, \tag{3.61}$$

which must be solved by the unknown set of expansion coefficients f_j. Note that the system matrix is dense due to the nature of the integral operator. Given the presence of the Green function kernel, a source at some point will radiate a field in every point of the solution space. This will definitely affect the CPU time and memory requirements of our algorithms. Besides the method of moments, this process is also called the boundary element method, being the finite element method on boundaries. Yet, because the applied integral operators are not self-adjoint, we cannot reformulate this problem into a search for a stationary point of some functional.

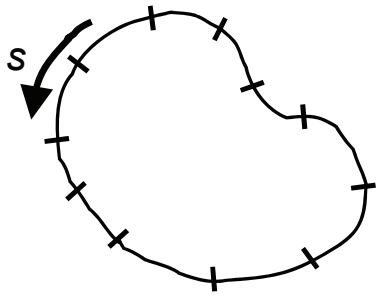

Figure 3.9: Cylinder S with boundary C subdivided into N segments.

A first step of the solution process entails the evaluation of the interaction integrals (3.60). The integrations of the basis and test functions with the Green function kernel may be evaluated numerically using Gauss quadrature rules. However, since the Green function $G(\boldsymbol{\rho}|\boldsymbol{\rho}')$ becomes singular in its source point, i.e. when $\boldsymbol{\rho} = \boldsymbol{\rho}'$, care should be taken when the integration interval of the basis function overlaps with the integration interval of the test function. Such a situation is called a *self-patch contribution*.

To deal with the selfpatch contribution, the following two-step procedure is one of the options available to accurately evaluate the interaction integrals containing a singularity. In a first step, one extracts this singularity by splitting the integral into a singular

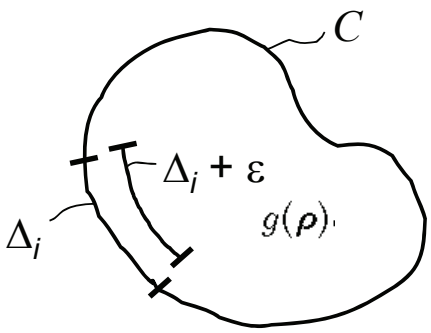

Figure 3.10: Evaluating the selfpatch contribution by first leaving an infinitesimal distance ϵ between the test function segment and the base function segment, analytically calculating the singular part of the integral and then taking the limit $\epsilon \to 0$.

part originating from the Green function and a remaining part of the Green function that is non-singular. The latter part can be directly integrated over both test and basis functions using quadrature formulas. Next, we revert our attention to a correct interpretation of the singular part. Therefore, we first integrate over the test function and afterwards take the limit to the boundary C, as is dictated by the integral equation. The integrations over basis and test functions usually can be performed in closed form, while still leaving a separation ϵ between the test function segment and the base function segment, as in Fig. 3.10. Only after all integration has been evaluated analytically, we perform the limit operation $\epsilon \to 0$.

After all matrix elements $K_{i,j}$ have been computed, we will solve the matrix system. When applying the Galerkin Method of Moments, the test and basis functions are equal, hence $w_i(\boldsymbol{\rho}') = p_i(\boldsymbol{\rho}')$. Given that the Green function is symmetric in its arguments, we find that the coefficient matrix of the linear system of equations is symmetric. This property is important in terms of reciprocity and energy conservation, but, unfortunately, it is very difficult to exploit complex symmetry of the system matrix in the solution process.

3.6.2 Basis and test functions

Scalar case

To make a judicious choice for the basis and test functions, we follow the guidelines provided in (2.6.2). Specifically for boundary element methods, where, in general, no sparse matrix is obtained anyhow, we must first decide between subdomain functions and entire domain functions. Subdomain functions are defined over a finite interval only, which is exactly the property that yielded the sparse system matrix in the finite element method. Entire domain functions span the whole contour of the object.

To construct a set of subdomain functions on a contour C, the boundary is divided into a number of segments, as in the example in the previous subsection. Again, pulse functions may be applied on each segment, as shown in Fig. 3.11, providing a piecewise constant approximation, but also overlapping triangular functions, as plotted in Fig. 3.12, may be used, allowing for a piecewise linear approximation. The latter expansion generates a continuous approximation of the solution which is important when derivatives must be taken along the contour. Hence, as explained in (2.6.2), the oorrect

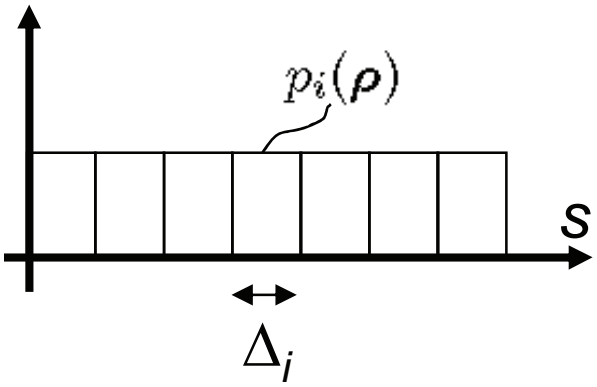

Figure 3.11: Piecewise constant expansion in terms of pulse functions

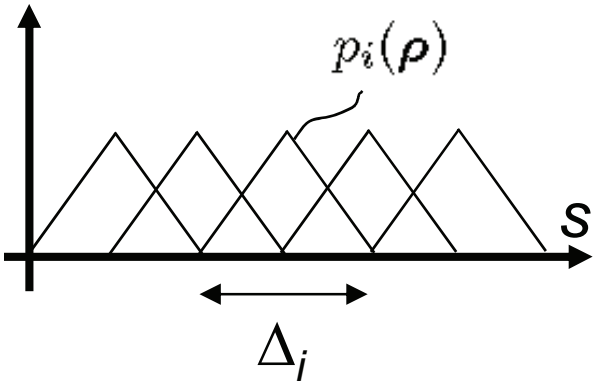

Figure 3.12: Piecewise linear expansion in terms of triangle functions

continuity conditions must be applied by ensuring that the set of basis functions and the set of test functions spans the correct subspace. As an example, a surface current $\mathbf{j} = j\hat{\boldsymbol{t}}$ flowing on the contour C due to TE excitation, as in Section 3.5.2, requires a continuous representation, in order to respect the continuity condition for the current, imposed by the law of charge conservation. Therefore, currents should have a regular divergence, and since the current in a TE problem in tangentially oriented, this implies continuity along the contour. While pulse functions may be used for the TM problem of Section 3.5.1, they are not an option for the TE configuration.

As for entire domain basis functions, extending over the complete contour C, an expansion in a Fourier basis is the obvious choice when an object has a cylindrical cross-section. This corresponds to a Fourier series expansion $p_n(\boldsymbol{\rho}) = e^{jn\phi}, n = -N, \ldots, -1, 0, 1, \ldots, N$ of the unknown function along the azimuth angle ϕ. Another example of entire domain basis functions are orthogonal polynomials. Chebychev polynomials are often used on open lines.

Vectorial case— Rao-Wilton-Glisson basis functions

When solving the Maxwell equations in a 3D configuration, we are confronted with vectorial unknowns. More in particular, we have to expand the electric current $\boldsymbol{j}(\boldsymbol{r})$

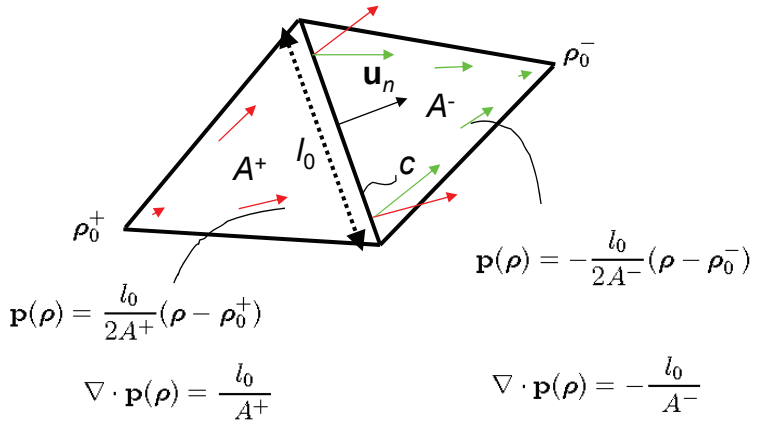

Figure 3.13: Rao-Wilton-Glisson basis function on a triangular element for the expansion of the current distribution.

induced on the surface S of a 3D scatterer. Choosing an expansion and testing function for this current can be done in a similar way as for the finite element method.

In terms of subspace, we need to take into account that currents should be divergence conforming, as their divergence has a physical meaning, being the charge density, owing to conservation of charge. Hence, basis functions should be chosen in the Sobolev subspace $\mathcal{H}(\text{div}; S)$. Hence, in the absence of singular charge distributions, which radiate infinite energy fields and should be avoided, the normal components of the current distribution must be continuous. We never actually measure the current density, only its flux through a segment of the 2D surface mesh.

The approach to discretize the surface S of the scatterer V and to approximate the induced current S on its boundary then proceeds by the following steps:

1. We partition the boundary S by applying a mesh of triangles. Again, for simplicity, flat triangular cells may be used. The Delaunay algorithm can be used to improve the quality of the mesh, avoiding long sharp triangles.

2. Next, in each triangle T, we define local shape functions belonging to $\mathcal{H}(\text{div}; T)$. We propose a Rao-Wilton-Glisson basis function of the form

$$\boldsymbol{p}(\boldsymbol{\rho}) = \boldsymbol{P_0} + P_1\boldsymbol{\rho}, \tag{3.62}$$

   generating a zero current in the triangle node $-\frac{\boldsymbol{P_0}}{P_1}$, increasing linearly up to a maximum value at the opposite edge, on which the normal component of the current is constant. Moreover, the divergence, which is proportional to the charge, remains constant within the triangle. One such basis function is shown in Fig. 3.13.

3. We then connect the different shape functions, which are at the moment local to each triangle, by imposing continuity of the (constant) normal component of the current at each edge of the triangle. This ensures that the candidate solution respects the continuity conditions of the current on the complete surface S. As unknowns, we then select the constant amplitudes of the normal components of the current on each edge of the mesh. Equivalently, we may choose as degrees

of freedom the current fluxes through each edge, being the normal component integrated over that edge.

3.7 Fast techniques: The Fast Multipole Method

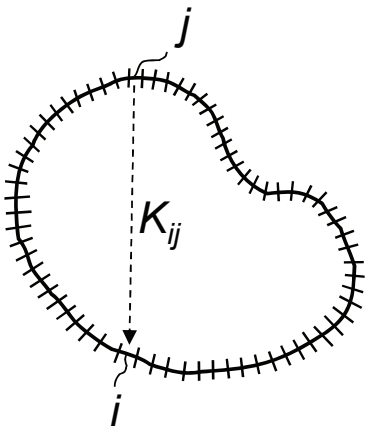

Figure 3.14: Discretization of a 2D scatterer: $K_{i,j}s_j$ corresponds to tested field at segment i by source s_j at segment j.

In recent years, many advances have been made in solving problems using integral equations by means of fast techniques. These are algorithms that solve the matrix system faster than the conventional $\mathcal{O}\left(N^3\right)$ order of complexity, with N the number of unknowns. To be really efficient, such fast formalisms should also be able to store the matrix in a form that requires a memory size much less than order $\mathcal{O}\left(N^2\right)$. In the same process, the time required for the calculation of the $\mathcal{O}\left(N^2\right)$ interaction integrals (3.60) should also be drastically reduced. Recent developments in methods that satisfy all these requirements resulted in a dramatic increase of the size and complexity of problems that can be handled.

For a large number of unknowns N, it becomes advantageous to solve the resulting system of linear equations iteratively. This reduces the complexity from $\mathcal{O}\left(N^3\right)$ to

$$\text{Constant} \times N^{\text{iter}} \times N^{\text{matrix}-\text{vector}} \times \text{cost}^{\text{matrix}-\text{vector}}, \tag{3.63}$$

with N^{iter} the number of iterations required to solve the matrix system with a prescribed accuracy, $N^{\text{matrix}-\text{vector}}$ the number of matrix-vector products in each iteration, and $\text{cost}^{\text{matrix}-\text{vector}}$ the cost to compute one matrix-vector product. Hence, in an iterative solution process, the computational complexity is mainly due to the evaluation matrix-vector products in each iteration. The Fast Multipole Method optimizes the computation of the matrix-vector products in each iteration of the iterative solution process.

Let us apply the method to the two-dimensional scattering problem governed by the first kind integral equation for Dirichlet problems, as discussed in Section 3.6.1, Remember that we discretized the integral equation (3.58) using pulse basis and test functions. During the iterative solution of the matrix system (3.59), in each iteration

we have to evaluate the product

$$\overline{\overline{K}} \cdot \boldsymbol{s} = \sum_{j=1}^{N} K_{i,j} s_j, \tag{3.64}$$

involving the system matrix $\overline{\overline{K}}$ and a search vector $\boldsymbol{s}$. As we can interpret $K_{i,j}s_j$ in terms of the weighted field

$$K_{i,j}s_j = \int_{\Delta_i} w_i(\boldsymbol{\rho}) \int_{\Delta_j} G(|\boldsymbol{\rho} - \boldsymbol{\rho}'|) p_j(\boldsymbol{\rho}') s_j \, dc' \, dc \tag{3.65}$$

due to source s_j at segment j, as shown in Fig. 3.14, we find that the matrix vector product yields the tested fields in all segments generated by the combined effect of the sources s_j active in all segments on the boundary. To accelerate this matrix vector product, different kinds of fast multipole methods exist, each having their own strengths and weaknesses. However, they all start from the same basic idea.

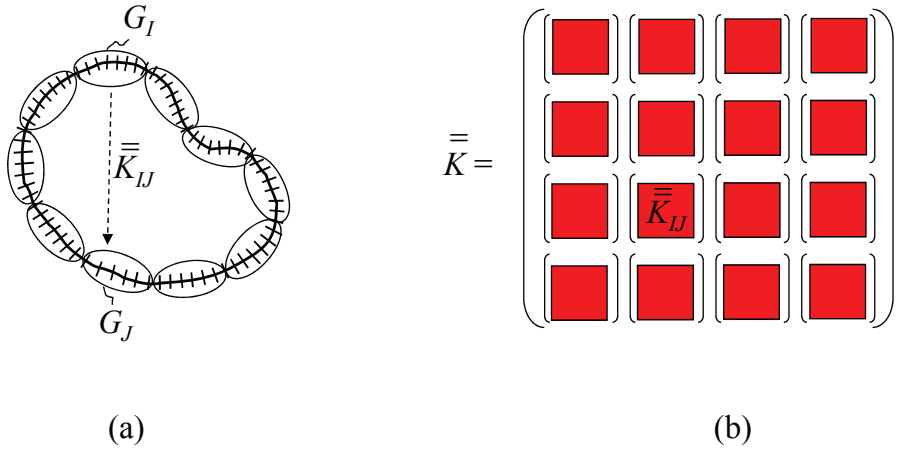

Figure 3.15: Partitioning the geometry and the system matrix: (a) Geometric partition of the scatterer into groups with (b) corresponding matrix partition, showing the interaction block $\overline{\overline{K}}_{I,J}$ between groups G_I and G_J.

First, we group the segments, acting as sources, into groups or boxes. Then, the system matrix $\overline{\overline{K}}$ can be partitioned into a number of submatrices, each describing the interaction between two groups G_I and G_J. In the process, we reorder the unknowns such that geometric blocks correspond to matrix blocks, as shown in Fig. 3.15.

We have to distinguish between two situations:

1. If the structure contains many small geometrical details, the segments will be much smaller compared to wavelength. This fine discretisation is then necessary to capture the fine geometrical structure. Furthermore, we assume that the object as a whole is much smaller than the wavelength. This is called the *low frequency regime*.

2. If the structure is large compared to wavelength, the discretisation is chosen such that it accurately captures the wavelike behavior. Segment sizes are now comparable to the wavelength (e.g. one tenth of a wavelength). This is called the *high frequency regime*.

3.7.1 Low-Frequency Fast Multipole Method

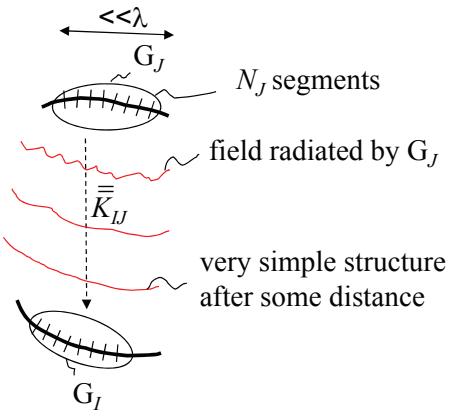

Figure 3.16: Interaction $\overline{\overline{K}}_{I,J}$ between groups G_I and G_J. In the LF regime, the radiated field has a very simple structure for well-separated groups.

Let us first consider the LF situation. We assume that not only the segments but also the group containing such segments remains small compared to the wavelength. Therefore, all segments are in the near field of all other segments. The field can hence be approximated very accurately by replacing the Green function in the interaction integrals by its approximation (3.5), repeated here for convenience

$$\lim_{\boldsymbol{\rho}\to\boldsymbol{\rho}'} G(\boldsymbol{\rho},\boldsymbol{\rho}') \sim -\frac{j}{4} - \frac{1}{2\pi}\left[\gamma + \ln\left(\frac{k\,|\boldsymbol{\rho}-\boldsymbol{\rho}'|}{2}\right)\right], \tag{3.66}$$

valid near the source point. In 2D, we observe the logarithmic singularity. This term can now be expanded into multipoles as follows

$$\ln\left(\frac{k\,|\boldsymbol{\rho}-\boldsymbol{\rho}'|}{2}\right) = \ln\left(\frac{kr}{2}\right) - \sum_{q=1}^{Q}\frac{1}{q}\left(\frac{r}{r'}\right)^q \cos\left[q\,(\theta-\theta')\right] \tag{3.67}$$

where we introduced a cylindrical coordinate system with its origin in the center of the observation group G_I, as shown in Fig 3.17, (r,θ) the coordinates of a point $\boldsymbol{r}=\boldsymbol{\rho}$ in the observation group G_I and (r',θ') the coordinates of a point $\boldsymbol{r}'=\boldsymbol{\rho}'$ in the excitation group G_J. We observe that the convergence of the series depends on the fraction $\frac{r}{r'}$, hence on the size of the observation group compared to the source-observation distance. Assuming that all boxes enclosing the different groups are of the same size, we say in general that the method is applicable when the separation between groups is much larger than the box size. Note also that the number of terms required for an accurate approximation does not depend on electric size, as no wavelength is involved in the convergence of the series.

We now make the evaluation of the matrix-vector product (3.64) more efficient. We

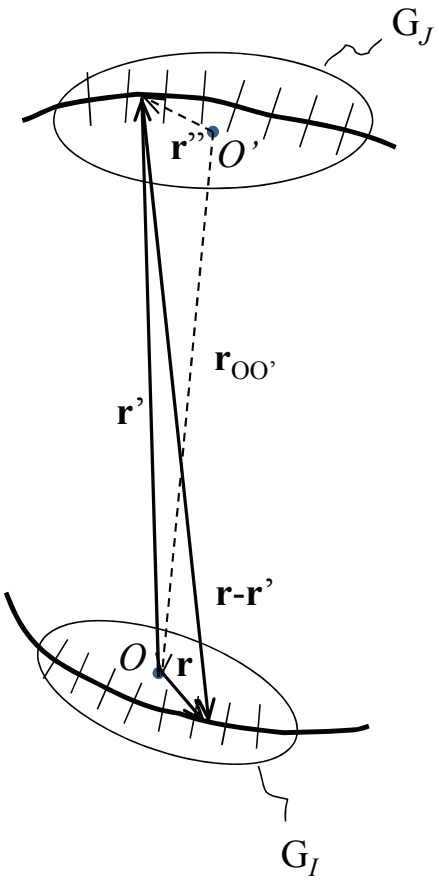

Figure 3.17: Local coordinate systems belonging to the G_I and G_J groups.

start by approximating (3.66) as a series of the form

$$\lim_{\boldsymbol{\rho}\to\boldsymbol{\rho}} G(\boldsymbol{\rho},\boldsymbol{\rho}') \approx A_0 + B_0 \ln\left(\frac{kr}{2}\right) + \sum_{q=1}^{Q} A_q r^q \cos q\theta \, \frac{\cos q\theta'}{r'^q} + \sum_{q=1}^{Q} B_q r^q \sin q\theta \, \frac{\sin q\theta'}{r'^q}, \tag{3.68}$$

in which we note that we can expand the kernel function in a series of terms that are products of a function depending on the source point and a function of the observation point. Inserting the multipole expansion (3.67) into that part of the matrix-vector

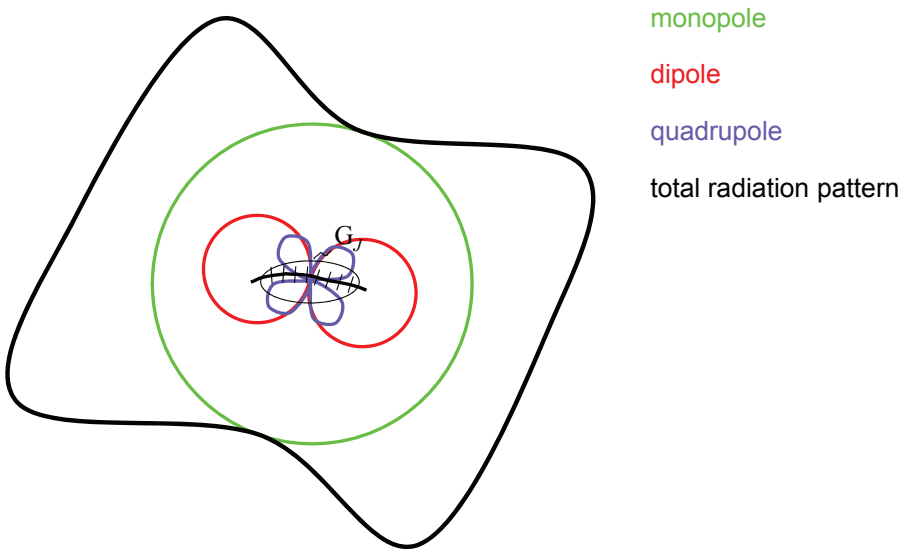

Figure 3.18: Multipole expansion of the radiation pattern of the source group.

product (3.68) that is relevant to the interaction between G_I and G_J yields

$$\begin{aligned}\sum_{j=1}^{N_J} K_{I,J,i,j} s_j = &\int_{\Delta_i} \left[A_0 + B_0 \ln\left(\frac{kr}{2}\right)\right] w_i(\boldsymbol{\rho})\, dc \sum_{j=1}^{N_J} s_j \int_{\Delta_j} p_j(\boldsymbol{\rho}')\, dc' \\ &+ \sum_{q=1}^{Q} A_q \int_{\Delta_i} w_i(\boldsymbol{\rho}) r^q \cos q\theta\, dc \sum_{j=1}^{N_J} s_j \int_{\Delta_j} p_j(\boldsymbol{\rho}') \frac{\cos q\theta'}{r'^q}\, dc' \\ &+ \sum_{q=1}^{Q} B_q \int_{\Delta_i} w_i(\boldsymbol{\rho}) r^q \sin q\theta\, dc \sum_{j=1}^{N_J} s_j \int_{\Delta_j} p_j(\boldsymbol{\rho}') \frac{\sin q\theta'}{r'^q}\, dc'.\end{aligned} \tag{3.69}$$

The first thing we observe is that all integrals in this expression can be precomputed before the iterative solution process starts. The number of integrals to evaluate has now reduced from order $\mathcal{O}(N_I N_J)$ to order $\mathcal{O}(QN_J)$ for the excitation segments and order $\mathcal{O}(QN_I)$ for the observation segments. We then apply a two-step approach to compute the matrix-vector product. We demonstrate these two steps by considering, for example, the second term:

1. *Aggregation* to Q radiation patterns of the excitation group by computing

$$R_q = \sum_{j=1}^{N_J} s_j \int_{\Delta_j} p_j(\boldsymbol{\rho}') \frac{\cos q\theta'}{r'^q}\, dc'. \tag{3.70}$$

   This process is independent of the observation group and takes $\mathcal{O}(QN_J)$ operations. It can be interpreted as an expansion of a radiation pattern in a monopole, dipole and quadrupole, ..., term, as graphically shown in Fig. 3.18.

2. *Disaggregation* of the Q radiation patterns at the observation group by comput-

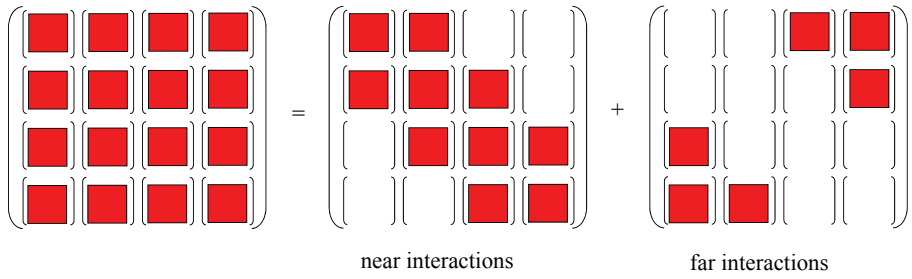

Figure 3.19: Splitting the system matrix in near and far interactions.

ing

$$\sum_{j=1}^{N_J} K_{I,J,i,j} s_j = \sum_{q=1}^{Q} A_q R_q \int_{\Delta_i} w_i(\boldsymbol{\rho}) r^q \cos q\theta \, dc. \tag{3.71}$$

This process is independent of the source group and takes $\mathcal{O}(QN_I)$ operations.

In each step of the iterative process, we hence reduce the number of operations to $\mathcal{O}(QN_J)$ + $\mathcal{O}(QN_I)$. This results in an efficient algorithm, provided that Q is small, hence that both groups G_I and G_J are well separated. As the above expansion can therefore only be applied efficiently for groups that are well separated, we first subdivide the coefficient matrix in a part that contains the near interactions and a part that contains the far interactions

$$\overline{\overline{K}} \cdot \boldsymbol{s} = \overline{\overline{K}}_{\text{near}} \cdot \boldsymbol{s} + \overline{\overline{K}}_{\text{far}} \cdot \boldsymbol{s}, \tag{3.72}$$

as schematically shown in Fig. 3.19

We then apply the fast multipole decomposition to the far interactions only. The end result is schematically shown in Fig. 3.20. There, one additional step has been applied. Remember that the expansion (3.69) was carried at around the origin O, located at the center of each observation group G_I. This means that the aggregated radiation patterns of a source group G_J, computed w.r.t. this origin, will differ for each observation group G_I, as they should be referred w.r.t. a different origin. This problem can be avoided by introducing a new expansion, taken now w.r.t. center of the observation group G_J. The details of this expansion are beyond the scope of this course. We only mention that the far interactions are efficiently calculated by aggregating the radiation pattern based on expansion such as (3.69) around the source region's origin O', translating the radiation pattern from the source region's origin O' to the observation region's origin O, and applying the disaggregation using (3.71) to resolve the different interactions in the observation group. The complexity of the resulting algorithm is $\mathcal{O}\left(N^{4/3}\right)$ instead of $\mathcal{O}\left(N^2\right)$.

A similar multipole expansion may be applied in three dimensions but then spherical harmonics are involved. For acoustic problems, these are scalar spherical harmonics, such as, e.g., used in the study of the hydrogen atom. For an electromagnetic problem, vectorial spherical harmonics are needed. These are a bit more complicated and are also encountered in quantum electro dynamics.

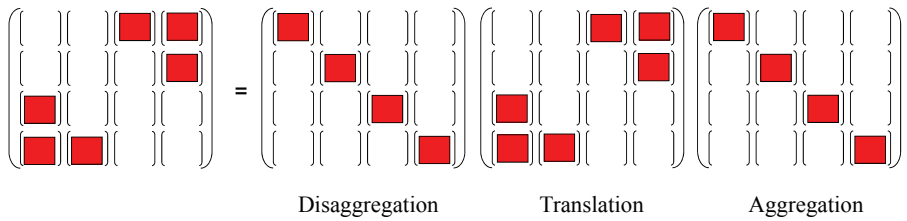

Figure 3.20: LF-FMM: Decomposing the far interactions in the system matrix into a aggregation, translation and disaggregation step.

3.7.2 High-Frequency Fast Multipole Method

We now revert our attention to the HF case, where a group of segments will be comparable to wavelength. This means that its radiation pattern, even after a certain distance, will be of high complexity. Although a full-blown multipole expansion of the form (Fig. 3.17)

$$G(\boldsymbol{\rho},\boldsymbol{\rho}') = -\frac{j}{4}H_0^{(2)}\left(k\left|\boldsymbol{\rho}-\boldsymbol{\rho}'\right|\right) \approx -\frac{j}{4}\sum_{q=-Q}^{Q} j^q J_q(kr')e^{jq\theta'}H_q^{(2)}(kr)e^{jq\theta}, \quad (3.73)$$

based on the addition theorem of the Hankel function, gives again rise to a series of terms that are products of a function in the observation group's coordinates and the source group's coordinates, many degrees of freedom Q are needed in the dynamic far field for the series to "somewhat" converge. We shall therefore take a different approach, based on a different expansion. We start by subdividing the scatterer's surface into groups of bounded *electric* size, being bounded w.r.t. wavelength. In that case, the group's radiation pattern as a function of the observer's angle is bounded by the electric size of the source box, i.e. the proportion of the box diameter and the wavelength. This means that, instead of expansion (3.73), we can represent the box's radiation pattern by a limited number of plane wave terms of the form

$$G(\boldsymbol{\rho},\boldsymbol{\rho}') = -\frac{j}{4}H_0^{(2)}\left(k\left|\boldsymbol{\rho}-\boldsymbol{\rho}'\right|\right) \approx -\frac{j}{4}\sum_{q=-Q}^{Q} T_q(k,r',\theta')e^{jkr\cos(\theta_q-\theta)}. \quad (3.74)$$

Replacing the multipole expansion by this plane wave expansion will significantly increase the efficiency of the resulting algorithm. Yet extreme care must be taken when applying the series, as it does not exhibit guaranteed convergence for increasing Q due to the behavior of $T_q(k,r',\theta')$. The expansion is also very unstable at low frequencies, which is the reason why the multipole expansion is used there. It is very hard (though not impossible) to eliminate this instability. The instability originates from the fact that the terms in the summation (in the dynamic expansion) are very much larger than the final summation result. Therefore, a lot of cancellation occurs, which can completely wipe out all precision of a floating-point representation. Again, the function $T_q(k,r',\theta')$ can be split up into a translation operator, which translates plane waves (instead of multipoles) w.r.t. origin of the the source group to plane waves w.r.t. the origin of the observation, and an aggregation operator. This decomposition is a key ingredient in the FMM algorithm. Hence, for groups that are well separated, the matrix

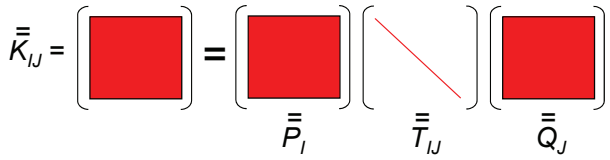

Figure 3.21: HF-FMM: Decomposing the far group interactions into a aggregation, translation and disaggregation step.

$\overline{\overline{K}}_{I,J}$ can be written as (Fig. 3.21)

$$\overline{\overline{K}}_{I,J} = \overline{\overline{P_I}} \cdot \overline{\overline{T}}_{I,J} \cdot \overline{\overline{Q}}_J. \tag{3.75}$$

This expansion has the following physical meaning:

1. the aggregation matrix $\overline{\overline{Q}}_J$ describes the expansion of the field generated by *all segments* in group G_J into plane waves, propagating in different directions away from the box center. This matrix will only depend on the source group G_J and, hence, needs to be determined only once for each group.

2. $\overline{\overline{T}}_{I,J}$ describes a translation of the plane waves from the centre of group G_J to centre of group G_I. A plane wave originating from the center of box G_J will still be only one plane wave arriving at the center of box G_I. This means that the matrix $\overline{\overline{T}}_{I,J}$ is a diagonal matrix, only changing the phase of each plane wave. If we would use multipoles in our expansion, this block would be dense. This is the main reason why the plane waves are used in the high-frequency case. Because the size of this matrix scales with the size of the groups, it is necessary to minimize the computational complexity of multiplying by $\overline{\overline{T}}_{I,J}$.

3. the disaggregation matrix $\overline{\overline{P}}_I$ performs the inverse operation of $\overline{\overline{Q}}_J$. It translates incoming plane waves coming from all directions towards the box center into fields in all segments of group G_I. Hence, it only depends on the observation group G_I and, again, needs to be calculated only once for each group.

Although the field of a source can always be expanded into plane waves (cfr. electromagnetics I – fourier decomposition), the method will only work if the groups are well separated. The reason is that, for groups that are near to each other, the number of terms $2Q + 1$ we need to take into account will increase up to the point where the mathematically fragile expansion we used will start to behave numerically unstable. Moreover, it is clear that, using plane waves, we cannot approximate the near singular behavior of the field near the source. Therefore, as in the LF-FMM case, we will split up the system matrix again into near and far interactions, according to Fig. 3.19. The far interactions can then be decomposed as explained, yielding an expansion of disaggregation, translation and aggregation block matrices of the form shown in Fig. (3.22). Note that this is only a very small example. For a real example with hundreds of groups,it becomes clear that memory requirements are drastically reduced with this decomposition and that the computation time to perform the matrix vector products is also accelerated considerably. The resulting algorithm's computational complexity is $\mathcal{O}\left(N^{3/2}\right)$ instead of $\mathcal{O}\left(N^2\right)$, or $\mathcal{O}\left(N^{4/3}\right)$ for LF-FMM.

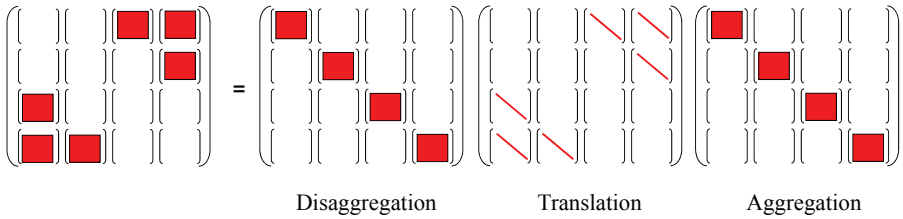

Figure 3.22: HF-FMM: Decomposing the far interactions in the system matrix into a aggregation, translation and disaggregation step.

3.7.3 Multilevel Fast Multipole Method

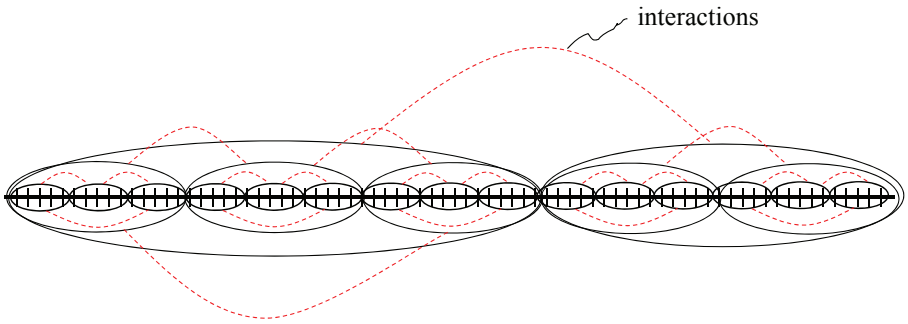

Figure 3.23: Computation of the interactions in a multilevel fast multipole method scheme.

Even though we already improved on the classic $\mathcal{O}\left(N^2\right)$ complexity in both the low frequency and high frequency regimes, we can do even better. The key ingredient for further improvement is the extension of the subdivision into groups in the fast multipole method to a multi-level scheme. In such a scheme, the boxes themselves are grouped into parent boxes, on so on, until we arrive at a level on which only one box remains. The number of levels is proportional to the logarithm of the number of lowest level boxes. This extension is called the Multi-Level Fast Multipole Algorithm (MLFMA). This approach, which is similar to the way links in a wired switched telephony systems are made, optimizes both the LF-FMM and HF-FMM, leading to complexities of $\mathcal{O}\left(N\log(N)\right)$, both for CPU-time for one iteration of the iterative solver and in terms of memory requirements. Figures 3.24 and 3.25 demonstrate the effect on CPU-time and memory requirements, respectively, proving the drastic improvement provided by the ML-FMM scheme over the conventional boundary element method when modelling scattering at a metamaterial consisting of an array of dielectric cylinders.

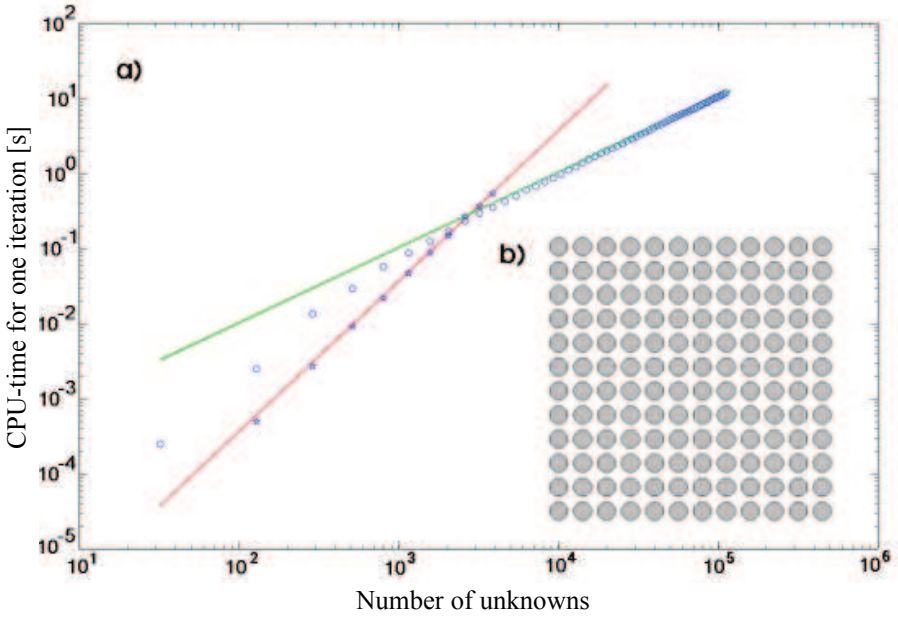

Figure 3.24: CPU-time: Conventional boundary element method versus Multilevel Fast Multipole Method.

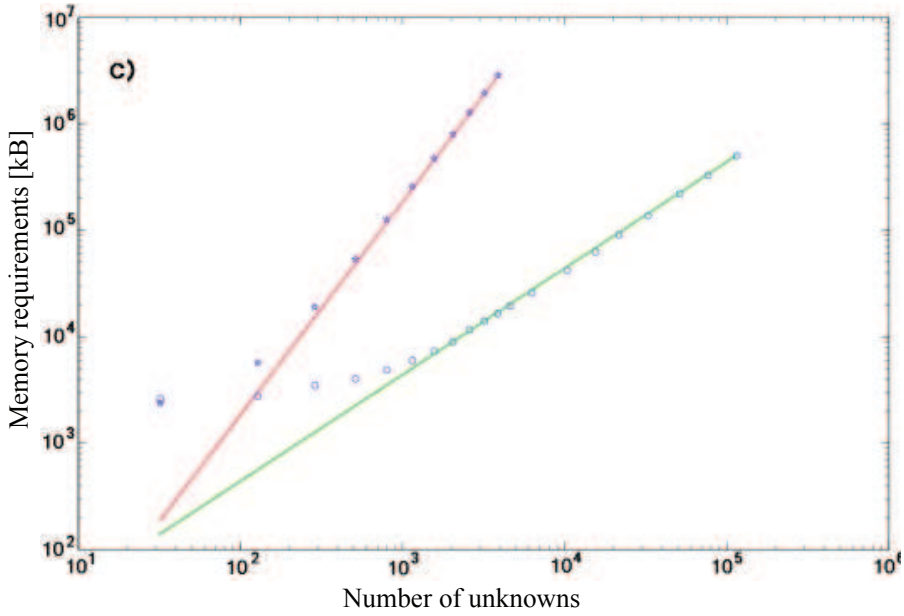

Figure 3.25: Memory requirements: Conventional boundary element method versus Multilevel Fast Multipole Method.

Appendix A

Problems

In this appendix, we provide some problems that the reader may want to solve in order to better understand the different chapters of this course.

A.1 Chapter 1: Mathematical techniques

A.1.1 Positive-Definite Matrices.

A Hermitian matrix $\mathsf{A} \in \mathbb{C}^{N\times N}$ is positive-definite iff

$$\boldsymbol{x}^H \cdot \mathsf{A} \cdot \boldsymbol{x} > 0, \quad \forall \boldsymbol{x} \in \mathbb{C}^N, \tag{A.1}$$

which is equivalent to requiring that all leading principal minors[1] are strictly positive.

A real symmetric matrix $\mathsf{A} \in \mathbb{R}^{N\times N}$ is positive-definite iff

$$\boldsymbol{x}^T \cdot \mathsf{A} \cdot \boldsymbol{x} > 0, \quad \forall \boldsymbol{x} \in \mathbb{R}^N, \tag{A.2}$$

which is equivalent to requiring that all leading principal minors are strictly positive.

1. Now consider a *general* (i.e. not necessarily symmetric) real matrix $\mathsf{A} \in \mathbb{R}^{N\times N}$. Demonstrate that, to determine whether the matrix A is positive-definite, it is sufficient to concentrate on the symmetric part of A, being $\frac{\mathsf{A}+\mathsf{A}^T}{2}$, only. Therefore, show that, for the skew-symmetric part of A, we have that

$$\boldsymbol{x}^T \cdot \left(\frac{\mathsf{A}-\mathsf{A}^T}{2}\right) \cdot \boldsymbol{x} \equiv 0, \quad \forall \boldsymbol{x} \in \mathbb{R}^N. \tag{A.3}$$

2. Now verify that the following matrices are positive definite:

$$A_1 = \begin{pmatrix} 1 & -1 \\ 1 & 1 \end{pmatrix}, \tag{A.4}$$

$$A_2 = \begin{pmatrix} 1 & -4 & 5 \\ 4 & 2 & 6 \\ -5 & -6 & 3 \end{pmatrix}. \tag{A.5}$$

[1] To calculate a k^{th} leading principal minor, eliminate the $N-k$ last column and rows and calculate the determinant of the remaining matrix.

A.1.2 Iterative Solution Methods.

We consider the solution of the matrix system

$$\mathsf{A} \cdot \boldsymbol{v} = \boldsymbol{b}. \tag{A.6}$$

with A a complex $N \times N$ matrix.

1. Demonstrate that, for the matrix A Hermitian and positive-definite, the solution of (A.6) i.e. $\mathsf{A}^{-1} \cdot \boldsymbol{b}$, is the minimum of the functional

$$\phi(\boldsymbol{x}) = \frac{1}{2}\left(\boldsymbol{v} - \mathsf{A}^{-1}\boldsymbol{b}, \boldsymbol{v} - \mathsf{A}^{-1}\boldsymbol{b}\right)_{\mathsf{A}}. \tag{A.7}$$

   with $(\boldsymbol{x}, \boldsymbol{y})_{\mathsf{A}} = \boldsymbol{y}^H \mathsf{A}\boldsymbol{x}$ being the inproduct with the matrix A as a kernel matrix. Explain why the matrix A must be Hermitian and positive-definite for this property to hold.

2. Explain why, for a general complex $N \times N$ matrix A, iterative methods such as the conjugate gradient method may still be applied to solve (A.6) after rewriting the matrix system as

$$\mathsf{A}^H \mathsf{A} \cdot \boldsymbol{v} = \mathsf{A}^H \boldsymbol{b}. \tag{A.8}$$

   Prove that this procedure squares the condition number.

A.1.3 Fredholm Integral Equations.

Consider the Fredholm integral equation of the second kind

$$g(t) = \int_a^b K(t,s) f(s) \mathrm{d}s - \sigma f(t), \forall t \in [a,b]. \tag{A.9}$$

1. Under the assumption of a bounded kernel $K(t,s)$ and self-adjoint integral operator, prove that the solution of (A.9) is given by

$$f(s) = \sum_p \frac{\phi_p(s)}{\lambda_p - \sigma} \int_a^b \overline{\phi_p(t)} g(t) \mathrm{d}t, \tag{A.10}$$

   with λ_p the eigenvalues and $\phi_p(t)$ the eigenfunctions of the integral operator

2. Explain the Fredholm alternative based on the solution (A.10) of the second-kind Fredholm integral equation.

A.2 Chapter 2: Finite elements

A.2.1 Magnetic Field Formulation (MFF)

Construct the magnetic field formulation (MFF):

1. Derive a weak-form Galerkin formulation for the Maxwell equations, based on the magnetic field as unknown.

2. What are the natural and essential boundary conditions in this magnetic field formulation (MFF)?

3. Write down a functional in the magnetic field, being the dual form of (2.26) in the syllabus. Demonstrate that the former weak-form equation is then the Euler equation yielding the stationary point of this functional.

A.2.2 Vector Finite Elements.

Write a function `plottrianglebasisfunction(x1,y1,x2,y2,x3,y3)` in MatLAB that plots the first-order curl-conforming edge elements, connected to the three edges of a triangle formed by the points $(x1, y1)$, $(x2, y2)$ and $(x3, y3)$.

1. Construct a triangular grid in the barycentric coordinates $(\lambda_1, \lambda_2, \lambda_3)$, with $\lambda_1 + \lambda_2 + \lambda_3 = 1$. Select as values for $\lambda_1 = 0, 0.1, \ldots, 1$.

2. Now construct the vectors X and Y, representing the points $\mathbf{r} = (x, y)$ in the triangle based on

$$\mathbf{r} = \lambda_1 \mathbf{r}_1 + \lambda_2 \mathbf{r}_2 + \lambda_3 \mathbf{r}_3 \tag{A.11}$$

   with $\mathbf{r}_1 = (x_1, y_1)$, $\mathbf{r}_2 = (x_2, y_2)$ and $\mathbf{r}_3 = (x_3, y_3)$.

3. Now construct vectors U and V, corresponding to the basis function vectors $\mathbf{w}_{\text{tr,edge}_i} = (u, v)$ in the grid points, with edge_i the edge opposite to point i and with

$$\mathbf{w}_{\text{tr,edge}_i} = \lambda_{i+1} \nabla_{\text{tr}} \lambda_{i+2} - \lambda_{i+2} \nabla_{\text{tr}} \lambda_{i+1}. \tag{A.12}$$

   To compute $\nabla_{\text{tr}} \lambda_i$, make use of the fact that $\lambda_i = \dfrac{1}{2A} (a_i + b_i x + c_i y)$, with the auxiliary variables $a_i = x_{i+1} y_{i+2} - x_{i+2} y_{i+1}$, $b_i = y_{i+1} - y_{i-1}$, and $c_i = x_{i-1} - x_{i+1}$, where all indices i should be interpreted modulo 3.
   $A = \frac{1}{2} (b_{i+1} c_{i+2} - b_{i+2} c_{i+1})$ is the area of triangle.

4. Apply the MatLAB function `quiver(X,Y,U,V)` to generate the plot of the basisfunction within the triangle. Generate three figures, one for each edge of the triangle. Interpret the result.