Circuits and Systems
Mekelweg 4,
2628 CD Delft
The Netherlands
https://sps.ewi.tudelft.nl/

SPS-2023-00

M.Sc. Thesis

Ultrasound Imaging through Aberrating Layers using a Virtual Array

Francesca De Carlo B.Sc.

Abstract

Ultrasound images are typically generated using the Delay-And-Sum (DAS) method, which assumes a homogeneous propagation medium. When an aberrating layer is situated between the sensor array and the imaging target, this assumption does not hold, and DAS is replaced with model-based methods. These methods are computationally expensive and require to accurately model the aberrations caused by the layer. This thesis investigates novel methods for image formation and aberration estimation. The effect of the layer is described using a set of transfer functions from the sensor array to a virtual array placed after the layer. In the first part, we assume the transfer functions are known, and we propose a new method for image formation. The transfer functions allow to map the signal from the sensor array to the virtual array, and the DAS method is used on the virtual array signal. This technique is equivalent to model-based matched filtering in terms of image quality, without requiring expensive matrix computations. In the second part, the transfer functions are unknown, and a novel technique is introduced for their estimation. Using pulse-echo data, a focus-quality metric is computed to quantify the accuracy of the transfer function estimate. The transfer functions are modeled using a dictionary and the dictionary coefficients are iteratively updated to increase the defined metric. The optimization leads to improved focus quality and sharper images. In the case the layer model requires a limited dictionary, the proposed algorithm generates an accurate estimate of the transfer functions.

TU Delft

Faculty of Electrical Engineering, Mathematics and Computer Science

Delft University of Technology

Ultrasound Imaging through Aberrating Layers using a Virtual Array

Thesis

submitted in partial fulfillment of the
requirements for the degree of

Master of Science

in

Electrical Engineering

by

Francesca De Carlo B.Sc.
born in Bari, Italy

This work was performed in:

Circuits and Systems Group
Department of Microelectronics
Faculty of Electrical Engineering, Mathematics and Computer Science
Delft University of Technology

Delft University of Technology

Delft University of Technology
Department of
Microelectronics

The undersigned hereby certify that they have read and recommend to the Faculty of Electrical Engineering, Mathematics and Computer Science for acceptance a thesis entitled **"Ultrasound Imaging through Aberrating Layers using a Virtual Array"** by **Francesca De Carlo B.Sc.** in partial fulfillment of the requirements for the degree of **Master of Science**.

Dated: 24 August 2023

Chairman:

Prof. dr. ir. G.J.T. Leus

Advisors:

Prof. dr. ir. G.J.T. Leus

Dr. ir. P. Kruizinga

Committee Members:

Dr. ir. D.J. Verschuur

Abstract

Ultrasound images are typically generated using the Delay-And-Sum (DAS) method, which assumes a homogeneous propagation medium. When an aberrating layer is situated between the sensor array and the imaging target, this assumption does not hold, and DAS is replaced with model-based methods. These methods are computationally expensive and require to accurately model the aberrations caused by the layer. This thesis investigates novel methods for image formation and aberration estimation. The effect of the layer is described using a set of transfer functions from the sensor array to a virtual array placed after the layer. In the first part, we assume the transfer functions are known, and we propose a new method for image formation. The transfer functions allow to map the signal from the sensor array to the virtual array, and the DAS method is used on the virtual array signal. This technique is equivalent to model-based matched filtering in terms of image quality, without requiring expensive matrix computations. In the second part, the transfer functions are unknown, and a novel technique is introduced for their estimation. Using pulse-echo data, a focus-quality metric is computed to quantify the accuracy of the transfer function estimate. The transfer functions are modeled using a dictionary and the dictionary coefficients are iteratively updated to increase the defined metric. The optimization leads to improved focus quality and sharper images. In the case the layer model requires a limited dictionary, the proposed algorithm generates an accurate estimate of the transfer functions.

Acknowledgments

This master project has been an opportunity to learn and grow both academically and personally. I am very grateful to all the people that contributed to this journey.

First, I would like to thank my supervisors Pieter Kruizinga and Geert Leus. Pieter, thank you for sharing with me your passion about this topic and your knowledge. Since the very start, you were confident that we would have learned much from this project, and this has been extremely motivating for me. In our brainstorming sessions, you helped me find new ways to look at the problems and inspired me with countless ideas. I am truly grateful for your encouragement and your trust in my abilities. Geert, thank you for your support and guidance throughout the project. You were always available whenever I had questions, and you took the time to give insightful feedback. Knowing that I could always rely on you for a critical opinion has given me the confidence to pursue my ideas, and I learned a lot during this process.

Lastly, I would like to thank my family and my boyfriend Riccardo. Thank you for being always there for me and believing in me.

Francesca De Carlo B.Sc.
Delft, The Netherlands
24 August 2023

Contents

List of Figures

List of Tables

1 Introduction

Ultrasound imaging makes use of sound waves to generate images of a target of interest. The imaging process starts by sending an acoustic wave into the medium using a transducer array. The wave is reflected or scattered by different structures within the imaging domain and propagates back to the transducers, which record the received signal. Subsequently, this data is processed to generate an image where various structures can be distinguished based on their reflection or scattering characteristics.

To generate an image, it is necessary to reverse the wave propagation and identify the source of each received echo. To achieve this, the initial step is to model the wave propagation within the medium. The typical assumption is that the variations in speed of sound and density are large enough to generate weak scattering, but small enough to ensure that both the transmit field and received echoes propagate as if the medium was homogeneous [1]. This implies that the wave propagation delay from a source to a receiver is solely based on their distance and the average speed of sound in the medium. This forms the basis for the classical beamforming method, which is called Delay-And-Sum (DAS).

In some situations, the medium exhibits significant variations in speed of sound that challenge the validity of the homogeneous medium assumption. The variations in speed of sound can give rise to various phenomena, including refraction, multiple scattering and reverberation. As a result, the pulse-shape can be distorted and the propagation delays may differ significantly from those observed in homogeneous media. These effects are referred to as aberrations [2]. Neglecting the aberrations and using the common image formation methods leads to imaging artifacts and loss of resolution (Figure 1.1). Therefore, it becomes essential to include aberrations when modeling propagation. As the model is modified to account for aberrations, the image formation method needs to be adjusted accordingly. Instead of DAS, model-based methods are employed, which are typically more computationally expensive than DAS [3].

In some cases, the medium inhomogeneities may be confined to a specific layer situated between the transducers and the imaging target. Such a layer is referred to as an aberrating layer. In various clinical situations, the imaging target lies behind an aberrating layer. Examples of aberrating layers are the human skull and the abdominal wall [4][5]. Another interesting example of aberrating layer is the so-called coding mask. This is a random phase mask which has the purpose of distorting the transmitted wave to make the pixel response position-dependent, thus improving the image quality [6].

When dealing with aberrating layers, we can simplify the general model by introducing the concept of a virtual array [7]. The virtual array is an imaginary array positioned just after the aberrating layer. After introducing the virtual array, wave propagation through the layer can be described using a set of transfer functions from the sensor array to the virtual array. Given these transfer functions, we can model propagation in the imaging domain.

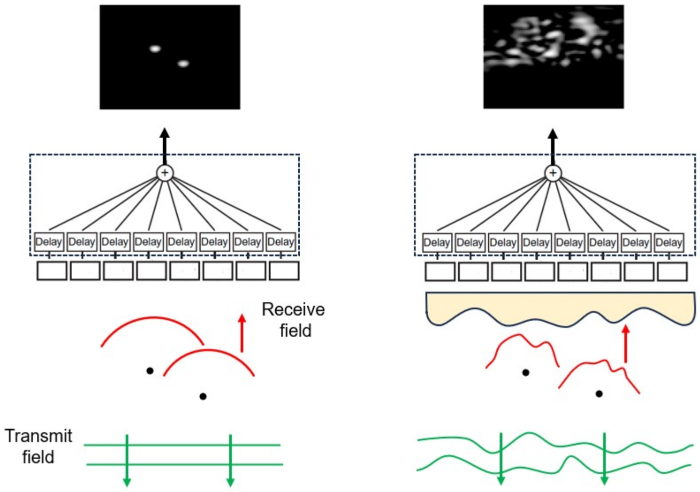


Figure 1.1: The DAS method produces accurate images when propagation occurs in homogeneous media (left). When the wave encounters an aberrating layer, severe distortion occurs, leading to corrupted images (right).

In the case of the coding mask, it is possible to place the sensors and the coding mask in a water tank and measure the transfer functions with a hydrophone [6]. However, in most in-vivo scenarios, measuring the transfer functions is not possible, since this procedure would be invasive. If we aim to correct aberrations, we first need to estimate the transfer functions.

Research Problems

This thesis addresses two distinct problems. In the first part of the thesis, we focus on cases where the transfer functions are known. In this case, the challenge lies in finding alternatives to the currently used image formation methods, which are computationally expensive. In the second part of the thesis, we address the issue of the transfer function estimation. The common thread between the two parts is the use of the virtual array framework, which proves to be a valuable tool in solving these problems.

Thesis outline

In Chapter 2, we explain the fundamental concepts on which this work is based. We derive the signal model and provide an overview of the basics of image formation. Additionally, we briefly outline the main approaches for aberration corrections in the literature. We conclude this chapter by incorporating the virtual array concept into the model.

In Chapter 3, our focus is on cases where the layer transfer functions are known. Here, we present an alternative method for image formation, which is first derived and then applied on simulated data.

In Chapter 4, we address the challenge of imaging through aberrators for which the transfer functions are unknown. We propose an iterative scheme to estimate the transfer functions, and subsequently apply it on simulated data.

Finally, in Chapter 5, we draw the conclusions for this thesis.

Signal model and image formation 2

In this chapter, we derive the signal model used in the rest of this thesis. We begin by explaining fundamental concepts related to wave propagation in Section 2.1, specifically reciprocity and time-reversal. These concepts are then linked to the matrix formalism that will be employed throughout the thesis. In Section 2.2, the signal model for pulse-echo imaging is derived and an overview of the common methods for image formation is given. In Section 2.3, the model is adapted to incorporate specific assumptions about wave propagation in the medium, taking into consideration relevant examples from the literature. Finally, in Section 2.4 we explain how the model is modified when introducing the virtual array.

2.1 Basics of ultrasound wave propagation

First, the expression for the pressure in the medium for a transducer excitation is derived. Within this thesis, we consider lossless and linear media. The pressure in the medium is a function of space and time and is denoted as $p(\mathbf{r}, t)$. One transducer element is approximated as a point source located at a position $\mathbf{r}_0$. We derive the pressure in the medium when the transducer element is excited by a Dirac pulse, and the emitted pressure is equal to its electromechanical response $s(t)$. The pressure in the medium $p(\mathbf{r}, t)$ for this input satisfies the wave equation

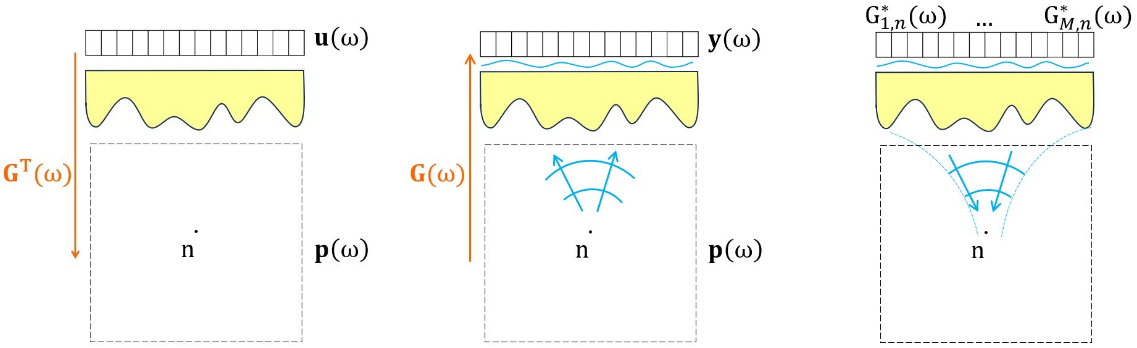
$$\nabla \cdot \left(\frac{\nabla p(\mathbf{r}, t)}{\rho(\mathbf{r})}\right) + \kappa(\mathbf{r}) \frac{\partial^2}{\partial t^2} p(\mathbf{r}, t) = -\delta(\mathbf{r} - \mathbf{r}_0) s(t), \tag{2.1}$$

where $\kappa(\mathbf{r})$ is the medium compressibility, $\rho(\mathbf{r})$ is the density and ∇ denotes spatial derivative [8].

An array of M elements is considered. Each element is approximated as a point source at position $\mathbf{r}_{0,i}$, $i = 1, .., M$. The image domain is discretized in N points at locations $\mathbf{r}_n$, $n = 1, .., N$. We define $g_{i,n}(t)$ as the pressure at $\mathbf{r}_n$ when element i sends a pulse $s(t)$. The function $g_{i,n}(t)$ will be referred to as the Green's function from sensor i to point n. It is important to note that, in this thesis, the Green's function includes the transducer response $s(t)$. The Green's function $g_{i,n}(t)$ can be derived by solving the wave equation in 2.1, after replacing $\mathbf{r}_0$ with the location of the i-th transducer $\mathbf{r}_{0,i}$. In homogeneous media, we can replace $\kappa(\mathbf{r}) = \kappa_0$ and $\rho(\mathbf{r}) = \rho_0$ and we can introduce $c_0 = (\kappa_0 \rho_0)^{-\frac{1}{2}}$. Then, $g_{i,n}(t)$ is equal to:

$$g_{i,n}(t) = \gamma_{i,n} s\left(t - \frac{|\mathbf{r}_n - \mathbf{r}_{0,i}|}{c_0}\right). \tag{2.2}$$

The wave travels in the medium with an invariant pulse-shape. The coefficient $\gamma_{i,n}$ measures the attenuation in the path from sensor i to point n. The propagation delay



(a) Green's functions (b) Reciprocity (c) Time-reversal

Figure 2.1: Schematic of basic concepts in wave propagation. The matrix $\mathbf{G}^{\mathrm{T}}(\omega)$ models propagation from the array to the medium (a), while its transpose models propagation in the opposite direction (b). If we record the wave coming from a source at n and send it back after time-reversing, the re-emitted wave will focus back at point n (c).

is equal to the distance between sensor i and point n, divided by the speed of sound. This delay is typically referred to as time-of-flight from i to n.

In inhomogeneous media, the density and the compressibility are functions of space. Therefore, solving the wave equation requires using more sophisticated methods. The resulting Green's functions may vary significantly depending on the specific medium [9]. Due to the inhomogeneity of the medium, the pulse-shape is distorted, and the delays are not solely based on geometry anymore.

If we know the Green's functions, we can derive the pressure in the medium for any transducer excitation. This relation can be written in a matrix format. We define $u_i(t)$ as the input at element i of the array, and $p_n(t)$ as the pressure at point $\mathbf{r}_n$. The frequency domain representation of $u_i(t)$ and $p_n(t)$ is indicated as $U_i(\omega)$ and $P_n(\omega)$ respectively. The relation between these variables is $\mathbf{p}(\omega) = \mathbf{G}^{\mathrm{T}}(\omega)\mathbf{u}(\omega)$, with $\mathbf{p}(\omega) = [P_1(\omega), .., P_n(\omega), ..P_N(\omega)]^{\mathrm{T}}$, $\mathbf{u}(\omega) = [U_1(\omega), ..U_i(\omega), ..U_M(\omega)]^{\mathrm{T}}$ and $\mathbf{G}(\omega)$ equal to

$$\mathbf{G}(\omega) = \begin{bmatrix} G_{1,1}(\omega) \dots G_{1,N}(\omega) \\ \vdots \qquad\qquad \vdots \\ G_{M,1}(\omega) \dots G_{M,N}(\omega) \end{bmatrix}, \tag{2.3}$$

where we have defined as $G_{i,n}(\omega)$ the Fourier-transform of $g_{i,n}(t)$.

Once the Green's functions from the array to the medium are determined, we can also model propagation in the opposite direction, i.e. from the medium back to the array. If a source at point $\mathbf{r}_n$ emits a $s(t)$ pulse, the resulting pressure at point $\mathbf{r}_{0,i}$ is $g_{i,n}(t)$. This fundamental property is known as reciprocity and is applicable in any linear, lossless medium [9]. The measured signal for an input pressure $\mathbf{p}(\omega)$ in the medium is $\mathbf{y}(\omega) = \mathbf{G}(\omega)\mathbf{p}(\omega)$, where $\mathbf{y}(\omega)$ is a M-dimensional vector in which each entry measures the received pressure at one transducer. A schematic is given in Figure 2.1b. It is important to note that by writing this relation between $\mathbf{p}(\omega)$ and $\mathbf{y}(\omega)$ we are assuming that the transducer response is reciprocal as well. At the transducers, the received signal is convolved with the transducer response. Since the response in reception is equal to one in transmission, we can conclude that $\mathbf{y}(\omega)$ is equal to $\mathbf{G}(\omega)\mathbf{p}(\omega)$.

Knowing the Green's functions allows not only to model the propagation from a source within the medium to the transducer array, but also to identify the source location based on the array measurements. To do so, we need to derive $\mathbf{p}(\omega)$ from the measurements $\mathbf{y}(\omega)$. Deriving $\mathbf{p}(\omega)$ requires solving an inverse problem, which is usually accomplished using the matched filter estimator [10]. To explain how the matching estimator performs in this case, we consider a situation where a single source located at n is transmitting a pulse $S(\omega)$. Then, $\mathbf{p}(\omega) = S(\omega)\mathbf{e}_n$, with $\mathbf{e}_n$ n-th canonical vector. The pressure at the array is $\mathbf{y}(\omega) = S(\omega)\mathbf{g}_n(\omega)$, with $\mathbf{g}_n(\omega)$ n-th column of $\mathbf{G}(\omega)$. The matching estimator is effective if $\hat{\mathbf{p}}(\omega) = \mathbf{G}^{\mathrm{H}}(\omega)\mathbf{y}(\omega) \approx \mathbf{p}(\omega)$. From now on, we neglect the multiplications by $S(\omega)$ since we set the goal on getting a peak at the n-th entry of $\hat{\mathbf{p}}(\omega)$, rather than reconstructing the original pulse shape. Then, the matching estimator is effective if $\hat{\mathbf{p}}(\omega) = \mathbf{G}^{\mathrm{H}}(\omega)\mathbf{g}_n(\omega) \approx \mathbf{e}_n$.

Since $\mathbf{G}^{\mathrm{H}}(\omega)\mathbf{g}_n(\omega) = (\mathbf{G}^{\mathrm{T}}(\omega)\mathbf{g}_n^*(\omega))^*$, $\hat{\mathbf{p}}(\omega)$ is the conjugate of the pressure over the medium when sending $\mathbf{g}_n^*(\omega)$ from the transducer array. Applying conjugation to the spectrum of a signal is equivalent to reversing it in time domain. The operation of deriving $\hat{\mathbf{p}}(\omega)$ consists of two steps. Step 1 is recording the wave coming from the source and Step 2 is re-emitting the recorded wave after time reversing. The re-emitted wave focuses back at the source location, revealing from which point the initial wave was coming from. A good estimate is obtained if this wave indeed focuses at the original source location. It can be shown that if the transducers captured all the paths originating from the source, the time-reversal focus would be perfect [11]. This happens because the wave equation is time-reversal invariant, meaning that it does not change if we replace t with $-t$, since it contains only second-order time-derivatives.

In practice, a finite aperture array is used, then it is not possible to reverse all the paths. However, it is still true that the transmitted $\mathbf{g}_n^*(\omega)$ pulses arrive at the same time at point n and sum coherently. Indeed, the pressure at n is $\sum_i G_{i,n}^*(\omega)G_{i,n}(\omega)$. Each i-th function in the sum is the convolution of $g_{i,n}(t)$ with $g_{i,n}(-t)$, thus it will have a peak at $t = 0$, meaning that the pulses will sum coherently at the focal spot n. On the other hand, the pressure at another point m will be $\sum_i G_{i,n}^*(\omega)G_{i,m}(\omega)$. Each i-th term in the sum has a different phase shift, thus the terms will add incoherently. Then, $\hat{\mathbf{p}}(\omega) = \mathbf{G}^{\mathrm{H}}(\omega)\mathbf{g}_n(\omega)$ will still have a peak at the n-the entry and lower values at the other entries.

The pulses add coherently not only at n, but also at the adjacent points. The area where they sum coherently is called focal spot. In addition, the area where they sum incoherently may exhibit other secondary peaks, which are referred to as sidelobes. In homogeneous media, the focal spot dimension and the sidelobe level can be computed based on array pitch and aperture and on the position of the focusing point. In inhomogeneous media, the focus quality depends not only on the mentioned parameters, but also on the characteristics of the medium. The focus can either degrade or improve compared to homogeneous conditions [12]. Several studies have been conducted to explore how focus quality varies in inhomogeneous media [13][8]. A remarkable result is that in disordered media the focal spot can be even smaller than in homogeneous media [8].

Overall, the matching estimator $\mathbf{G}^{\mathrm{H}}(\omega)$ is efficient in estimating the location of a source, since the off-diagonal elements $\mathbf{G}^{\mathrm{H}}(\omega)\mathbf{G}(\omega)$ can be considered to be zero.

2.2 Pulse-echo imaging

The image formation is an estimation problem, in which we estimate the scattering amplitudes for each pixel from the transducer measurements. In Section 2.2.1, the relation between the transducer measurements and the scattering amplitudes is derived. In Section 2.2.2, the most common image formation methods are briefly described. In Section 2.2.3, the link between the image and the focus quality is explained.

2.2.1 Signal model

We consider a Synthetic Aperture (SA) acquisition scheme. In the SA scheme, one transducer element is active in each transmission. The transmissions are then repeated after shifting the position of the active element until the whole array has been used [14]. As a result, a total of M scans are performed, with M equal to the number of elements in the array.

After element i transmits a pulse $s(t)$, this pulse will propagate in the medium and it will be scattered at multiple locations. The echo scatterered at location n will have a pulse-shape $g_{i,n}(t)$ and a certain intensity. We define the intensity of the scattering at n as x_n.

The wave scattered at point n propagates towards the transducer array. The propagation from point n to sensor j is modeled as the convolution of the scattered pulse $g_{i,n}(t)$ with the Green's function from point n to sensor j, $g_{j,n}(t)$. The signal measured at sensor j is $x_n(g_{j,n}(t) * g_{i,n}(t))$.

It is assumed that the scattered echoes have a low intensity, thus multiple scattering can be ignored. This approximation is known as the Born approximation [15]. As a result, the echoes that arrive at sensor j are given by the sum of the contributions from all n points.

We define $y_{i,j}(t)$ as the measured signal at sensor j when sensor i transmits. This is equal to:

$$y_{i,j}(t) = \sum_n x_n \big[g_{j,n}(t) * g_{i,n}(t) \big]. \tag{2.4}$$

In frequency domain, the relation between $Y_{i,j}(\omega)$ and x_n becomes

$$Y_{i,j}(\omega) = \sum_n x_n G_{j,n}(\omega) G_{i,n}(\omega). \tag{2.5}$$

We discretize the ω domain in L frequencies $\omega_1, .., \omega_l, ..., \omega_L$. The measurements at frequency ω_l can be arranged into a vector. Then, the relationship between this vector and the scattering amplitudes is:

$$\begin{bmatrix} Y_{1,1}(\omega_l) \\ \vdots \\ Y_{i,j}(\omega_l) \\ \vdots \\ Y_{M,M}(\omega_l) \end{bmatrix} = \begin{bmatrix} G_{1,1}(\omega_l)G_{1,1}(\omega_l) & \dots & G_{1,N}(\omega_l)G_{1,N}(\omega_l) \\ \vdots & & \vdots \\ G_{i,1}(\omega_l)G_{j,1}(\omega_l) & \dots & G_{i,N}(\omega_l)G_{j,N}(\omega_l) \\ \vdots & & \vdots \\ G_{M,1}(\omega_l)G_{M,1}(\omega_l) & \dots & G_{M,N}(\omega_l)G_{M,N}(\omega_l) \end{bmatrix} \begin{bmatrix} x_1 \\ \vdots \\ x_n \\ \vdots \\ \vdots \\ x_N \end{bmatrix} \tag{2.6}$$

This can be written as $\mathbf{y}(\omega_l) = (\mathbf{G}(\omega_l) \circ \mathbf{G}(\omega_l))\mathbf{x}$, where $\circ$ is the Khatri-rao product and $\mathbf{G}(\omega_l)$ is the matrix defined in (2.3).

The $\mathbf{y}(\omega_l)$ vecotrs can be stacked in a single vector $\mathbf{y}$. The relation between $\mathbf{y}$ and $\mathbf{x}$ is:

$$\mathbf{y} = \mathbf{A}\mathbf{x}, \tag{2.7}$$

with

$$\mathbf{A} = \begin{bmatrix} \mathbf{G}(\omega_1) \circ \mathbf{G}(\omega_1) \\ \vdots \\ \mathbf{G}(\omega_L) \circ \mathbf{G}(\omega_L) \end{bmatrix}. \tag{2.8}$$

The array measurements contain additive noise. We define $\mathbf{n}_{i,j}(\omega_l)$ as the noise on $Y_{i,j}(\omega_l)$. These can be stacked in a vector $\mathbf{n}$. The model becomes:

$$\mathbf{y} = \mathbf{A}\mathbf{x} + \mathbf{n}. \tag{2.9}$$

Throughout this thesis, we will consider independent white Gaussian noise.

The measurements can also be arranged in L matrices of dimension $M \times M$. Each l matrix is defined as follows:

$$\mathbf{Y}(\omega_l) = \begin{bmatrix} Y_{1,1}(\omega_l) \dots Y_{1,M}(\omega_l) \\ \vdots \qquad\qquad \vdots \\ Y_{M,1}(\omega_l) \dots Y_{M,M}(\omega_l) \end{bmatrix}. \tag{2.10}$$

Then, the relation between each $\mathbf{Y}(\omega_l)$ and $\mathbf{x}$ is:

$$\mathbf{Y}(\omega_l) = \mathbf{G}(\omega_l)\mathrm{diag}(\mathbf{x})\mathbf{G}^{\mathrm{T}}(\omega_l) + \mathbf{N}(\omega_l). \tag{2.11}$$

In this thesis, we will use the expression in 2.7 and the one 2.11 interchangeably.

2.2.2 Image formation methods

In this section, we briefly describe the DAS technique and the model-based methods.

The most basic beamforming method is DAS [16]. The DAS estimate for pixel n is computed as:

$$\hat{x}_n = \sum_j \sum_i y_{i,j}(t_0 + \tau_{j,n} + \tau_{i,n}) \tag{2.12}$$

with $\tau_{j,n}$ equal to the time of flight from pixel n to transducer j. Since the start of the acquisition may not correspond with the time of peak of the transmitted pulse, the offset t_0 is introduced to correct the delays [16].

In model-based imaging, a different approach is used. The image is generated by solving the system of equations $\mathbf{y} = \mathbf{A}\mathbf{x}$. To obtain an estimated image $\mathbf{x}$ from $\mathbf{y}$, the following minimization problem can be solved:

$$\hat{\mathbf{x}} = \arg\min_{x} ||\mathbf{y} - \mathbf{A}\mathbf{x}||_2^2. \tag{2.13}$$

The $\mathbf{A}$ matrix is ill-posed, then a regularization term is typically added to the cost function. Moreover, the matrix $\mathbf{A}$ is typically large, since the first dimension is proportional to the number of transmit-receive events, which in the case of SA is M^2, and the second dimension is equal to the number of pixels N. Because of the large dimension of $\mathbf{A}$, solving this problem is usually computationally intensive [3].

To reduce the computation cost, the estimate can be computed using the matched filter estimator:

$$\hat{\mathbf{x}} = \mathbf{A}^{\mathrm{H}}\mathbf{y}. \tag{2.14}$$

By replacing $\mathbf{A}$ with the expression in 2.8, it can be shown that $\hat{\mathbf{x}}$ is equal to the diagonal of the matrix $\sum_{l=1}^{L} \mathbf{G}^{\mathrm{H}}(\omega_l)\mathbf{Y}(\omega_l)\mathbf{G}^*(\omega_l)$. Then, the n-th entry $\hat{x}_n$ is equal to:

$$\hat{x}_n = \sum_{l=1}^{L} \mathbf{g}_n^{\mathrm{H}}(\omega_l)\mathbf{Y}(\omega_l)\mathbf{g}_n^*(\omega_l). \tag{2.15}$$

It can be noticed that, in homogeneous media, matched filtering is closely related to DAS. In homogeneous medium, the matched filtered estimate of x_n is

$$\hat{x}_n = \sum_j \sum_i \sum_l S^*(\omega_l)S^*(\omega_l)e^{j\omega_l(\tau_{j,n}+\tau_{i,n})}Y_{i,j}(\omega_l), \tag{2.16}$$

where we have replaced $G_{i,n}(\omega)$ with the expression in 2.2. Each (i,j) entry in the sum is the correlation between $Y_{i,j}(\omega)$ and $S(\omega)S(\omega)e^{-j\omega(\tau_{j,n}+\tau_{i,n})}$. This is the same as the correlation of the time-domain signals $y_{i,j}(t)$ and $r(t-\tau_{i,n}-\tau_{j,n})$. where we have introduced $r(t) = s(t) * s(t)$. If we substitute $r(t)$ with $\delta(t-t_0)$, we obtain the expression in 2.12.

2.2.3 Relation between focus and image quality

In this Section, we highlight the link between focus and image quality, which will be important in Chapter 4.

The quality of the image depends on how closely the estimated value $\hat{x}_n$ matches the actual value of x_n. Considering the zero-noise case, we substitute $\mathbf{Y}(\omega_l) = \sum_{k=1}^{N} x_k \mathbf{g}_k(\omega_l)\mathbf{g}_k^{\mathrm{T}}(\omega_l)$ into the expression of $\hat{x}_n$ (2.15). The resulting estimate for $\hat{x}_n$ can be written as:

$$\hat{x}_n = x_n \sum_l ||\mathbf{g}_n(\omega_l)||_2^4 + \sum_{k\neq n} x_k \sum_l (\mathbf{g}_n^{\mathrm{H}}(\omega_l)\mathbf{g}_k(\omega_l))^2 \tag{2.17}$$

In this expression, we observe two terms: one proportional to x_n and another depending on the scattering sources at locations different from n. The interference from the source at point k in the estimate x_n depends on the product $\mathbf{g}_n^{\mathrm{H}}(\omega_l)\mathbf{g}_k(\omega_l)$. As explained in Section 2.1, this represents the pressure at location k when we focus on location n. To effectively suppress this interference, it is important to ensure that when we transmit $\mathbf{g}_n^{\mathrm{H}}(\omega)$ we achieve a pressure peak precisely at location n and a low pressure elsewhere. Therefore, the quality of the matched filtered image is intrinsically linked to the focus quality when applying time-reversal.

It is important to note that when noise is introduced, an additional term dependent on the noise will also be present.

2.3 Models of aberration in the literature

The image formation process requires to know the Green's function from the array to the medium. However, these are typically not known. Estimating the correct Green's functions is the first step for aberration correction. In the literature, several aberration correction methods have been proposed, which use different approaches to model wave propagation. The methods can be grouped in three main approaches:

a) The most general approach allows $G_{i,n}(\omega)$ to be any waveform. Given the speed-of-sound and density map, the Green's functions are computed by solving the wave equation.

b) The geometric approach uses $G_{i,n}(\omega) = S(\omega)e^{-j\omega\tau_{in}^{R}}$, where τ_{in}^{R} is the propagation time from sensor i to point n, computed according to refraction laws. Geometric methods do not consider diffraction and provide a good approximation when the wavelength is much smaller than the size of the inhomogeneous inclusions [9].

c) The "isoplanatic patch" approach divides the medium into multiple regions, called isoplanatic patches. One isoplanatic patch can be defined as "the area in which the Green's functions are identical after correction of the travel path differences" [17]. Then, within one isoplanatic patch, $G_{i,n}(\omega)$ can be modeled as $G_{i,n}(\omega) = S(\omega)A_i(\omega)e^{-j\omega\tau_{in}}$, where τ_{in} is the propagation time between transducer i and pixel n, computed as if the medium was homogeneous. The aberration is modeled in the $A_i(\omega)$ functions. Each $A_i(\omega)$ can be an exponential delay or any waveform. In case $A_i(\omega)$ is an exponential delay ($A_i(\omega) = e^{-j\omega\tau_{0,i}}$), the layer is modeled as a phase-screen. In this case, propagation in one isoplanatic patch is fully modeled given a set of M delays, with M number of transducers. We refer to this set of delays as an "aberration profile" [18]. In a homogeneous medium, there is only one large isoplanatic patch with $A_i(\omega) = 1$ for all i. However, as the effect of aberrations increases, the size of one isoplanatic patch decreases. Multiple isoplanatic patches need to be considered within the imaging area, each one with a different aberration profile.

The methods that use the approach in (a) or in (b) require to first estimate the speed-of-sound map. This is computed either with tomographic approaches [19] or by identifying the profile of the layer-medium interfaces based on the measured reflections [20] [21]. In

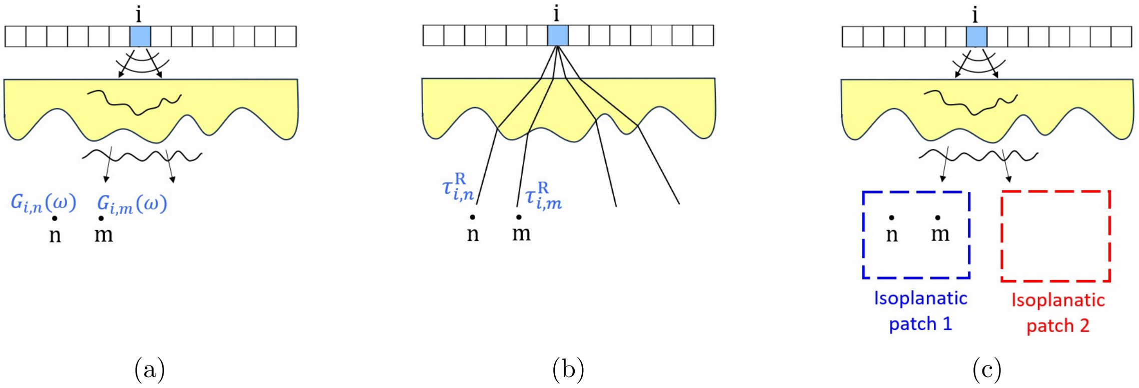


Figure 2.2: There are three main methods to model wave propagation in inhomogeneous media. In the most general approach (a), the Green's functions can take any waveform and may differ at each point. The geometric approach (b) approximates each Green's function as a delayed pulse, with delay based on refraction. The third approach (c) simplifies (a) by assuming that the Green's functions share one aberrating profile within an area defined as the isoplanatic patch.

transcranial ultrasound, a Computed Tomography (CT) scan is sometimes employed to convert Hounsfield units into a speed-of-sound map. However, this is mainly used for focusing through the skull during thermal treatment rather than for imaging purposes [22].

It can be noticed that if the geometric approach is used, matched filtering reduces to DAS with refraction-corrected delays. This variation of the DAS method is applied in [20] and [19]. Given the speed-of-sound map, the correct delays can be derived by solving a partial differential equation, known as the Eikonal Equation. After solving this equation, DAS with refraction-corrected delays can be applied. However, this requires that diffraction can be ignored.

Other methods attempt to find the Green's functions without estimating the speed-of-sound map of the aberrating layer. These methods use the "isoplanatic patch" approach, thus they require to estimate only one set of $A_i(\omega)$ functions for each isoplanatic patch in the imaging domain. In case of a phase screen ($A_i(\omega) = e^{-j\omega\tau_{0,i}}$), the set of parameters to estimate is reduced to M delays per isoplanatic patch, with M number of transducers. These delays are found by optimizing various fitting criteria [23] [18]. Other methods consider more general $A_i(\omega)$ functions, but rely on one simplifying assumption. It is assumed that a limited number of bright scatterers are present, referred to as "guiding stars". Under this assumption, $\mathbf{Y}(\omega)$ has a rank equal to the number of scatterers, as evident from (2.11). If these bright scatterers are sufficiently distant, the corresponding $\mathbf{g}_n(\omega)$ vectors become orthogonal. Consequently, the equation in (2.11) can be viewed as an eigenvalue decomposition. The work by Prada et al. on the decomposition of the time reversal operator (DORT) [24] is related to this idea. Once the $\mathbf{g}_n(\omega)$ from a guiding star location is found, the $A_i(\omega)$ components are valid within the isoplanatic patch of the guiding star.

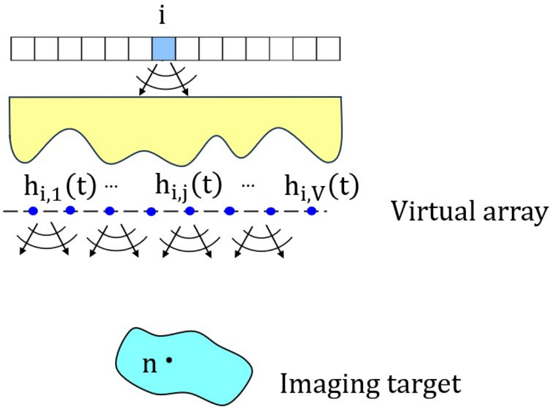


Figure 2.3: Schematic of the virtual array model. The aberrating layer distorts the transmitted field. The field at a point n in the imaging domain is derived as a function of the pressure on the virtual array, which is measured by the transfer functions $h_{i,j}(t)$.

2.4 Virtual array

Starting from the general model, we derive the model that will be used in the rest of thesis. This model has been introduced in [7] and is based on the assumption that the inhomogeneities are limited to one layer, situated between the sensor array and the imaging target.

A virtual array is placed just after the layer, as depicted in Figure 2.3. The transmit field in the imaging domain can be derived if the pressure at the virtual array is known. The field below the virtual array can be decomposed into a set of spherical waves originating from each point on the virtual array, with pulse shape equal to the pressure at that point.

We define $h_{i,j}(t)$ as the Green's function from sensor i on the real array to point j on the virtual array. We define $\tilde{g}_{j,n}(\omega)$ as the Green's function from virtual sensor j to point n. Then, the Green's function from sensor i to point n can be written as:

$$g_{i,n}(t) = \sum_{j=1}^{V} h_{i,j}(t) * \tilde{g}_{j,n}(t) \tag{2.18}$$

This equation holds if the virtual array is Nyquist sampled and the aperture is wide enough to capture all the waves emitted by the real array. The matrix $\mathbf{G}(\omega)$ can be written as:

$$\begin{aligned}\mathbf{G}(\omega) &= \begin{bmatrix} H_{1,1}(\omega) & \dots & H_{1,V}(\omega) \\ \vdots & & \vdots \\ H_{M,1}(\omega) & \dots & H_{M,V}(\omega) \end{bmatrix} \begin{bmatrix} \tilde{G}_{1,1}(\omega) & \dots & \tilde{G}_{1,N}(\omega) \\ \vdots & & \vdots \\ \tilde{G}_{V,1}(\omega) & \dots & \tilde{G}_{V,N}(\omega) \end{bmatrix} \\ &= \mathbf{H}(\omega)\tilde{\mathbf{G}}(\omega). \end{aligned} \tag{2.19}$$

Since the region below the layer is homogenous, $\tilde{\mathbf{G}}(\omega)$ can be derived based on the location of the virtual array and the imaging domain. Thus, $\tilde{\mathbf{G}}(\omega)$ does not depend on the layer. The presence of the layer only affects $\mathbf{H}(\omega)$.

The model in 2.11 can be written as:

$$\mathbf{Y}(\omega_l) = \mathbf{H}(\omega_l)\tilde{\mathbf{G}}(\omega_l)\mathrm{diag}(\mathbf{x})\tilde{\mathbf{G}}(\omega_l)^{\mathrm{T}}\mathbf{H}(\omega_l)^{\mathrm{T}} + \mathbf{N}(\omega_l), \tag{2.20}$$

for $l = 1, .., L$.

This model does not take into account that the transducers will receive reflections from the layer. In practice there may be significant reflections from the two layer-background interfaces, especially if the impedance mismatch between layer and background is high. In addition, for inhomogeneous layers scattering from inside the layer will give rise to additional echoes superimposed on the initial reflections described above. These echoes will be received before the echoes from the imaging domain. However, if the imaging area is very close to the virtual array there will be superposition between these echoes and the reflections from the layer. Thus, in this case the model in 2.20 might be inaccurate.

Imaging through known aberrators 3

In this chapter, we assume the transfer matrix $\mathbf{H}(\omega)$ is known. This assumption is true when using a coding mask, as the transfer matrix is measured during the calibration process. In other cases, the transfer matrix may be known if the speed-of-sound map is provided, and $\mathbf{H}(\omega)$ is computed based on this information. Alternatively, $\mathbf{H}(\omega)$ might be estimated from pulse-echo data, as we will explore in Chapter 4.

If the layer transfer functions are known, we can derive the Green's functions everywhere in the imaging domain and generate the model matrix $\mathbf{A}$. From here, we solve the equation $\mathbf{y} = \mathbf{A}\mathbf{x}$ using any of the model-based methods. We are forced to use the model-based methods and cannot resort to DAS. However, the model-based methods are more computationally expensive.

In this chapter, we explore the possibility of using DAS even when imaging through known aberrators. Here, we make use of the virtual array and we consider the scenario where it is feasible to transmit and receive at the virtual array. With these measurements, DAS imaging becomes possible since propagation occurs in a homogeneous medium. However, as we do not have direct access to these measurements at the virtual array, we must estimate them. Section 3.1 explains how to estimate those measurements for arbitrary transmit schemes. This method is then applied to simulated data and it is compared to model-based matched filtering. Section 3.2 provides an explanation of the simulation setup used. The estimated signals and beamformed images are displayed in Section 3.3. In Section 3.4, we discuss the expected advantage of this method compared to conventional ones and we relate this work to existing research in the field.

3.1 Transmitting and receiving at the virtual array

We start by defining the notation used in this chapter:

- $U_i(\omega)$ is the transmit field at element i of the real array, for $i = 1, .., M$
- $P_j(\omega)$ is the transmit field at element j of the virtual array, for $j = 1, .., V$
- $Z_j(\omega)$ represents the received echoes at element j of the virtual array, for $j = 1, ., V$
- $Q_i(\omega)$ represents the received echoes at element i of the real array, for $i = 1, .., M$

M is the number of real transducers and V is the number of virtual transducers. We stack $U_i(\omega)$ in a vector $\mathbf{u}(\omega) = [U_1(\omega), .., U_i(\omega), .., U_M(\omega)]^{\mathrm{T}}$. We do the same for the other variables.

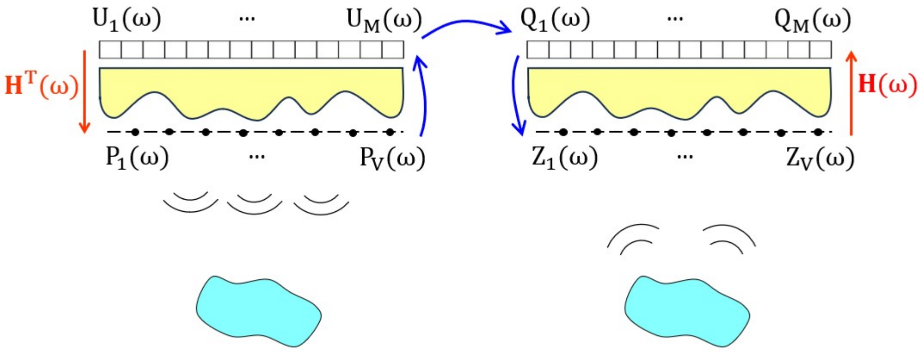


Figure 3.1: In the estimation process, we follow the order indicated by the blue arrows. First, we assign values to $\mathbf{p}(\omega)$ and estimate the corresponding input $\mathbf{u}(\omega)$. Next, we derive the real array measurements $\mathbf{q}(\omega)$, from which we estimate the received echoes at the virtual array $\mathbf{z}(\omega)$. The estimation requires solving two inverse problems.

The relations between these vectors are:

- $\mathbf{p}(\omega) = \mathbf{H}^{\mathrm{T}}(\omega)\mathbf{u}(\omega)$
- $\mathbf{q}(\omega) = \mathbf{H}(\omega)\mathbf{z}(\omega)$.

Our goal is to send a specific wave from the virtual array and capture the corresponding echoes received at the virtual array. However, since we cannot directly excite the virtual transducers, we use the real transducers to transmit a wave that, after propagating through the layer, generates the desired transmit field. We first assign values to $\mathbf{p}(\omega)$ and then proceed to estimate the input $\mathbf{u}(\omega)$ required to have that field. Once we have determined this input $\mathbf{u}(\omega)$, we obtain the measurements at the real array for this particular input, which we have defined as $\mathbf{q}(\omega)$. However, to complete our task, we require the received echoes at the virtual array $\mathbf{z}(\omega)$. Therefore, we need to estimate $\mathbf{z}(\omega)$ from $\mathbf{q}(\omega)$. These steps are depicted by the blue arrows in Figure 3.1.

The estimation of $\mathbf{z}(\omega)$ involves two inverse problems. As discussed in Section 2.1, these inverse problems can be effectively addressed using a matched filtering approach. It has been shown that it is possible to focus on each individual virtual transducer with time-reversal. Thus, the off-diagonal elements of the matrix $\mathbf{H}^{\mathrm{H}}(\omega)\mathbf{H}(\omega)$ can be assumed to be zero.

This implies that it is possible to estimate $\mathbf{u}(\omega)$ and $\mathbf{z}(\omega)$ as follows:

- $\hat{\mathbf{u}}(\omega) = \mathbf{H}^{*}(\omega)\mathbf{p}(\omega)$
- $\hat{\mathbf{z}}(\omega) = \mathbf{H}^{\mathrm{H}}(\omega)\mathbf{q}(\omega)$.

The only step that remains to be discussed is the step from $\hat{\mathbf{u}}(\omega)$ to $\hat{\mathbf{q}}(\omega)$. One option consists in transmitting the $\hat{\mathbf{u}}(\omega)$ waves from the real array and measuring $\hat{\mathbf{q}}(\omega)$. Usually, ultrasound transducers allow to send pulses or sequences of coded pulses. However, this might not be accurate enough to represent the $\hat{\mathbf{u}}(\omega)$ waveforms. If propagation and scattering in the medium are assumed to be linear and time-invariant, the received echoes $\mathbf{q}(\omega)$ for any transmission $\hat{\mathbf{u}}(\omega)$ can be derived by using a synthetic aperture scan. The received echoes for input $\hat{\mathbf{u}}(\omega)$ are equal to:

$$\hat{\mathbf{q}}(\omega) = \mathbf{Y}(\omega)\hat{\mathbf{u}}(\omega), \tag{3.1}$$

with $\mathbf{Y}(\omega)$ as defined in Section 2.2.

After combining all steps together, the estimate $\hat{\mathbf{z}}(\omega)$ can be expressed as:

$$\hat{\mathbf{z}}(\omega) = \mathbf{H}^{\mathrm{H}}(\omega)\mathbf{Y}(\omega)\mathbf{H}^{*}(\omega)\mathbf{p}(\omega). \tag{3.2}$$

If we replace $\mathbf{Y}(\omega)$ with the expression in (2.20), we get:

$$\hat{\mathbf{z}}(\omega) = \mathbf{H}^{\mathrm{H}}(\omega)\mathbf{H}(\omega)\tilde{\mathbf{G}}^{H}(\omega)\mathrm{diag}(\mathbf{x})\tilde{\mathbf{G}}^{T}(\omega)\mathbf{H}^{T}(\omega)\mathbf{H}^{*}(\omega)\mathbf{p}(\omega). \tag{3.3}$$

The pressure at virtual array when sending $\mathbf{p}(\omega)$ from the virtual array would be $\mathbf{z}(\omega) = \tilde{\mathbf{G}}^{H}(\omega)\mathrm{diag}(\mathbf{x})\tilde{\mathbf{G}}^{T}(\omega)\mathbf{p}(\omega)$. Now we clearly see that $\hat{\mathbf{z}}(\omega) \approx \mathbf{z}(\omega)$ if $\mathbf{H}^{\mathrm{H}}(\omega)\mathbf{H}(\omega) \approx \mathbf{I}$. In Section 2.1, it has been explained that the off-diagonal elements of this matrix can be considered to be zero. However, we did not elaborate on the diagonal elements. Firstly, the intensity at the focal spot can vary depending on the virtual element we focus on. Thus, the elements in the diagonal of $\mathbf{H}^{\mathrm{H}}(\omega)\mathbf{H}(\omega)$ might have different amplitudes, introducing different scaling for each entry of $\hat{\mathbf{z}}(\omega)$. Secondly, the pulse-shape at the focal spot will be $|S(\omega)|^2$. As we mentioned, $\mathbf{H}(\omega)$ models propagation and convolution with the transducer response $S(\omega)$. This is the pressure that we measure below the layer, so implicitly it contains the transducer response. Even in the scenario where the off-diagonal elements are zero and the diagonal elements have the same amplitude, $\mathbf{H}^{H}(\omega)\mathbf{H}(\omega)$ does not approximate the identity matrix, but rather $|S(\omega)|^2\mathbf{I}$. From these observations, we derive:

$$\hat{\mathbf{z}}(\omega) \approx \tilde{\mathbf{G}}^{H}(\omega)\mathrm{diag}(\mathbf{x})\tilde{\mathbf{G}}^{T}(\omega)|S(\omega)|^4\mathbf{p}(\omega). \tag{3.4}$$

This implies that the actual pulse shape in the $\hat{\mathbf{z}}(\omega)$ waveforms will be different from the pulse shape in $\mathbf{p}(\omega)$. We consider the case where we aim at estimating the pressure at the virtual array when sending a plane wave with pulse shape $S(\omega)$. Then, we set $\mathbf{p}(\omega)$ to a plane wave with pulse $S(\omega)$. However, the resulting estimate $\hat{\mathbf{z}}(\omega)$ contains the echoes we receive when transmitting $S(\omega)|S(\omega)|^4$. A longer pulse leads to a larger point spread function (PSF) in the axial dimension. If $S(\omega)$ is known, we can divide $\hat{\mathbf{z}}(\omega)$ by $|S(\omega)|^4$ and recover the desired pulse shape $S(\omega)$. However, this may increase noise at the frequencies where $S(\omega)$ is low. We will explain later how to partially solve this with regularization.

So far, we have considered a single transmission from the virtual array. Indeed, it is possible to derive the received echoes for multiple transmissions. We define $\mathbf{p}_k(\omega)$ as the input for the k-th transmission. From each one, we derive the corresponding $\hat{\mathbf{z}}_k(\omega)$. We can write everything in a single equation if we stack the vectors in two matrices $\hat{\mathbf{Z}}(\omega) = [\hat{\mathbf{z}}_1(\omega)\ldots\hat{\mathbf{z}}_k(\omega)\ldots,\hat{\mathbf{z}}_K(\omega)]$ and $\mathbf{P}(\omega) = [\mathbf{p}_1(\omega)\ldots\mathbf{p}_k(\omega)\ldots,\mathbf{p}_K(\omega)]$. We derive $\hat{\mathbf{Z}}(\omega)$ from $\mathbf{P}(\omega)$ as:

$$\hat{\mathbf{Z}}(\omega) = \mathbf{H}^{\mathrm{H}}(\omega)\mathbf{Y}(\omega)\mathbf{H}^{*}(\omega)\mathbf{P}(\omega). \tag{3.5}$$

3.1.1 Virtual transmit schemes

In this section, we define how to set $\mathbf{P}(\omega)$ for different transmit schemes from the virtual array. In each transmission k, the j-th element of the array sends a pulse that is delayed by $\tau_{j,k}$ and has an amplitude of $a_{j,k}$. We can represent this as $P_{j,k}(\omega) = a_{j,k}e^{-i\omega\tau_{j,k}}$, where $P_{j,k}(\omega)$ is the input for the j-th element during the k-th transmission. In this representation, we omit the pulse shape $S(\omega)$, for the reason explained in the previous section. The value $a_{j,k}$ is binary and is set to 1 if the j-th element is active during the k-th transmission, 0 otherwise.

As our goal is to apply DAS, we focus on transmission schemes that allow us to use DAS. Specifically, we consider three transmission schemes: Synthetic Aperture (SA), single plane wave and Plane-Wave Compounding [25]. We define how to set the coefficients in these three cases:

- For a SA scan, the number of transmissions K is equal to the number of virtual transducers V. The coefficient $a_{j,k}$ is 1 for $j = k$, 0 otherwise. The delays $\tau_{j,k}$ are set to zero. Then, $\mathbf{P}(\omega) = \mathbf{I}$ for all ω.

- For a single plane wave transmission, $K = 1$, $a_j = 1$ and $\tau_j = 0$ for all j. Then, $\mathbf{P}(\omega) = \mathbf{1}$ for all ω.

- In the case of plane-wave compounding, K waves at K angles $\theta_1, ..., \theta_K$ are sent. The amplitude coefficients are $a_{j,k} = 1$ for all j, k. The delays can be easily computed based on the array pitch, the speed of sound, and the desired angle. The formula for the delays can be found in [25].

3.1.2 Regularized inverse filter

In the previous sections, it has been explained that the inverse problems involved in the estimation can be solved by using $\mathbf{H}^{\mathrm{H}}(\omega)$. This approach is justified by the possibility of focusing on the virtual transducers through time-reversal, as explained in Section 2.1. While this method provides the desired focus, it also results in a longer pulse shape. Moreover, the intensity at the focal spot can vary depending on the virtual transducer location.

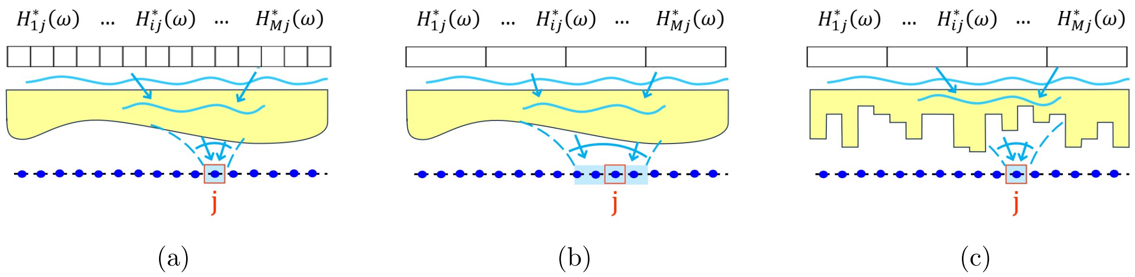
An alternative solution to the two problems is obtained through regularized least-squares. The estimate of $\mathbf{u}(\omega)$ becomes:

$$\begin{aligned}\hat{\mathbf{u}}(\omega) &= \arg\min_{\mathbf{u}(\omega)} ||\mathbf{H}^{\mathrm{T}}(\omega)\mathbf{u}(\omega) - \mathbf{p}(\omega)||_2^2 + \lambda||\mathbf{u}(\omega)||_2^2 \\ &= \mathbf{W}^{\mathrm{T}}(\omega)\mathbf{p}(\omega).\end{aligned} \tag{3.6}$$

with $\mathbf{W}(\omega) = (\mathbf{H}^{\mathrm{H}}(\omega)\mathbf{H}(\omega) + \lambda\mathbf{I})^{-1}\mathbf{H}^{\mathrm{H}}(\omega)$.

The same approach is used to estimate $\mathbf{z}(\omega)$ from $\mathbf{q}(\omega)$:

$$\begin{aligned}\hat{\mathbf{z}}(\omega) &= \arg\min_{\mathbf{z}(\omega)} ||\mathbf{H}(\omega)\mathbf{z}(\omega) - \mathbf{q}(\omega)||_2^2 + \lambda||\mathbf{z}(\omega)||_2^2 \\ &= \mathbf{W}(\omega)\mathbf{q}(\omega).\end{aligned} \tag{3.7}$$


$H^*_{1j}(\omega)$ ... $H^*_{ij}(\omega)$ ... $H^*_{Mj}(\omega)$
j
$H^*_{1j}(\omega)$ ... $H^*_{ij}(\omega)$ ... $H^*_{Mj}(\omega)$
j
$H^*_{1j}(\omega)$ ... $H^*_{ij}(\omega)$ ... $H^*_{Mj}(\omega)$
j

(a) (b) (c)

Figure 3.2: Time-reversal focusing is effective when using a fully sampled array (a). In case an undersampled array is used, focusing at a single virtual transducer is no longer feasible (b). However, it is still possible to achieve focusing using disordered aberrating layers, such as coding mask (c).

Then:

- $\hat{\mathbf{z}}(\omega) = \mathbf{W}(\omega)\mathbf{q}(\omega)$
- $\hat{\mathbf{u}}(\omega) = \mathbf{W}^{\mathrm{T}}(\omega)\mathbf{p}(\omega)$

The estimate of the full matrix $\mathbf{Z}(\omega)$ is:

$$\hat{\mathbf{Z}}(\omega) = \mathbf{W}(\omega)\mathbf{Y}(\omega)\mathbf{W}^{\mathrm{T}}(\omega)\mathbf{P}(\omega) \tag{3.8}$$

If time-reversal focusing is already effective (off-diagonal elements of $\mathbf{H}^H(\omega)\mathbf{H}(\omega)$ are zero, and the diagonal elements have the same amplitude), the primary change introduced by this approach is in the pulse shape. Specifically, $\mathbf{H}^{\mathrm{H}}(\omega)\mathbf{H}(\omega) \approx |S(\omega)|^2\mathbf{I}$, then $\mathbf{W}(\omega) \approx \frac{1}{|S(\omega)|^2+\lambda}\mathbf{H}^{\mathrm{H}}(\omega)$ and $\mathbf{W}^H(\omega)\mathbf{H}(\omega) \approx \frac{|S(\omega)|^2}{|S(\omega)|^2+\lambda}\mathbf{I}$. Otherwise, the effect is also on reducing the sidelobes and improving the focal spot size in the lateral dimension. Moreover, it can adjust the different weighting introduced on the entries of $\hat{\mathbf{u}}(\omega)$ and $\hat{\mathbf{z}}(\omega)$.

3.1.3 Undersampled array

In this derivation, no assumption has been made regarding the number of virtual transducers V and the number of real transducers M. The model in 2.20 is applicable for a Nyquist-sampled virtual array. Typically the real array is Nyquist sampled, then it is reasonable to set $V = M$.

We now shift our focus to a scenario where the real array is undersampled, leading to $M < V$. In this situation, solving the inverse problems becomes more challenging due to the mismatch between the number of unknowns (V) and the number of measurements (M). The assumption of $\mathbf{H}(\omega)^{\mathrm{H}}\mathbf{H}(\omega) \approx |S(\omega)|^2\mathbf{I}$ no longer holds. In physical terms, it is not possible to achieve focus at a single virtual transducer j by sending the time-reversed waveforms $H^*_{i,j}(\omega)$. While we discussed the possibility of focusing in the context of fully sampled arrays, the same cannot be said for an undersampled array. Generally, the focus degrades when working with a lower number of transducers (Figure 3.2b). However, it has been shown that in disordered media, a tight focal spot can still be achieved, even when using a single element [8]. We can expect a similar result for the coding mask. It has been shown that using a coding mask in front of an undersampled array significantly improves the resolution [6]. Because of the link in resolution and focus quality explained in Section 2.2.3, we can expect that a coding mask leads to a smaller focal spot when applying time-reversal on an undersampled array. Therefore, we may still be able to estimate received echoes at a fully sampled virtual array. Since $M < V$, estimating $\mathbf{Z}(\omega)$ from the real transducer measurements $\mathbf{Y}(\omega)$ can be seen as an attempt to synthesize more transducers than we actually have. The advantage of using a coding mask lies in the ability to create more virtual transducers than real transducers, thus extending imaging possibilities.

3.1.4 Delay-And-Sum at the virtual array

In this section, we present the proposed image formation method, outlining all its steps. After selecting the transmit scheme and computing the corresponding $\mathbf{P}(\omega)$, the image is computed as follows:

1. Compute $\hat{\mathbf{Z}}(\omega) = \mathbf{W}(\omega)\mathbf{Y}(\omega)\mathbf{W}^{\mathrm{T}}(\omega)\mathbf{P}(\omega)$
2. Perform an inverse Fourier-transform on each $\hat{Z}_{j,k}(\omega)$
3. Compute each x_n as: $\hat{x}_n = \sum_j \sum_k \hat{z}_{j,k}(\tau_{n,k}^{TX} + \tau_{n,j}^{RX})$

Here, we can simply set $\mathbf{W}(\omega) = \mathbf{H}^{\mathrm{H}}(\omega)$. Alternatively, we can use the regularized inverse approach and set $\mathbf{W}(\omega) = (\mathbf{H}^{\mathrm{H}}(\omega)\mathbf{H}(\omega) + \lambda\mathbf{I})^{-1}\mathbf{H}^{\mathrm{H}}(\omega)$, with λ appropriately chosen. The delay $\tau_{n,j}^{RX}$ represents the propagation time from pixel n to point j on the virtual array, while $\tau_{n,k}^{TX}$ is the time it takes for the transmit field to reach point n. It is different for each k-th transmission, as we send different types of waves.

We now derive the relation between this method and model-based matched filtering. If we consider a Synthetic Aperture at the virtual array and we use the matched filter approach in the estimation, the image estimate is computed as:

1. $\hat{\mathbf{Z}}(\omega) = \mathbf{H}^{\mathrm{H}}(\omega)\mathbf{Y}(\omega)\mathbf{H}^{*}(\omega)$
2. Inverse Fourier-transform each $\hat{Z}_{j,k}(\omega)$
3. $\hat{x}_n = \sum_j \sum_k \hat{z}_{j,k}(\tau_{n,k}^{TX} + \tau_{n,j}^{RX})$

For a Synthetic Aperture scan, $\tau_{n,k}^{TX}$ is the propagation time from n to virtual transducer k. It can be shown that the second equation can be written as:

$$\hat{x}_n = \sum_{l=1}^{L} \tilde{\mathbf{g}}_n^{\mathrm{H}}(\omega_l)\hat{\mathbf{Z}}(\omega_l)\tilde{\mathbf{g}}_n^{*}(\omega_l), \tag{3.9}$$

where $\tilde{\mathbf{g}}_n(\omega_l)$ is the vector containing the Green's functions from the virtual array to the image domain, as defined in Section 2.4. By replacing $\hat{\mathbf{Z}}(\omega)$ with the equation in Step 1, the following expression is obtained:

$$\begin{aligned}\hat{x}_n &= \sum_{l=1}^{L} \tilde{\mathbf{g}}_n(\omega_l)^{\mathrm{H}} \mathbf{H}^{\mathrm{H}}(\omega_l) \mathbf{Y}(\omega_l) \mathbf{H}^{*}(\omega_l) \tilde{\mathbf{g}}_n^{*}(\omega_l) \\ &= \sum_{l=1}^{L} \mathbf{g}_n^{\mathrm{H}}(\omega_l) \mathbf{Y}(\omega_l) \mathbf{g}_n^{*}(\omega_l).\end{aligned} \tag{3.10}$$

This equation coincides with the matched filtered estimate of $\hat{x}_n$ in Equation 2.15, therefore the method's outcome is equivalent to matched filtering with $\mathbf{A}$.

In this derivation, we have considered that $\tilde{g}_{j,n}(\omega) = e^{-i\omega\tau_{n,j}}$, thus we have ignored the attenuation $\gamma_{j,n}$ in the path from virtual transducer j to pixel n. Differences between the proposed method and matched filtering may arise if the attenuation is considered in the $\mathbf{A}$ matrix computation. However, these can be easily corrected by introducing weights equal to $\gamma_{j,n}$ in the DAS step.

3.2 Methods

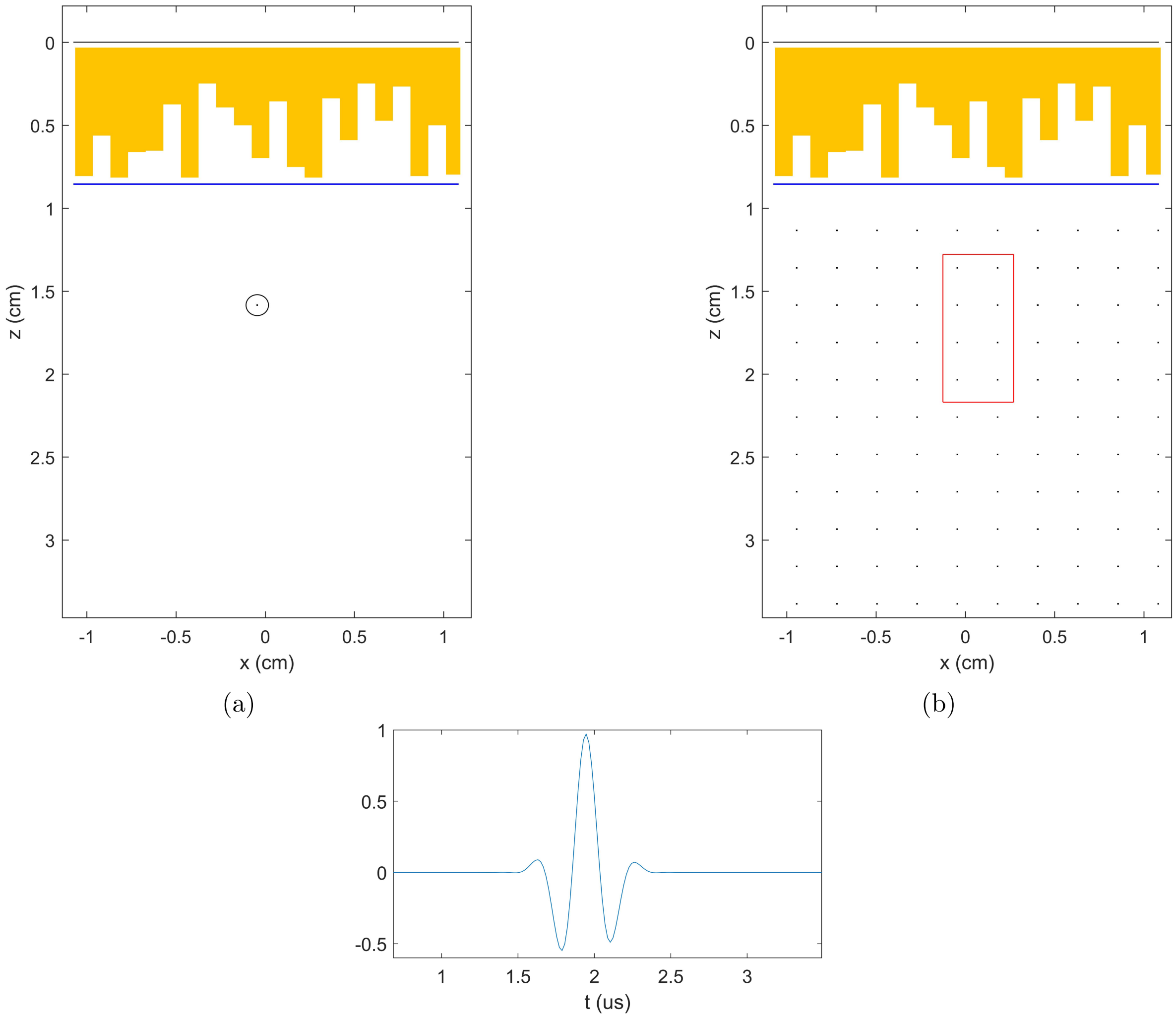
The proposed method is applied to simulation data generated using the k-Wave toolbox [26]. The simulation involves a plastic coding mask placed in front of an array, with an imaging phantom made of point scatterers. The transfer functions and the Synthetic Aperture measurements are simulated. In addition, the model matrix $\mathbf{A}$ is computed, since the proposed method will be compared to matched filtering. In this section, we provide a detailed explanation of the simulation set-up. Moreover, we explain how the model matrix is computed.

We consider a grid of 1024×512 points. The first dimension is the axial (z dimension) and the second one is the lateral (x dimension). The spacing between the points is the same in both dimensions and is equal to 90 μm. The dimension of the simulation grid is then 9.2 × 4.6 cm.

We will consider two arrays: one fully sampled and one undersampled. In the first one, the number of transducers is set to $M = 80$. Each transducer is 270 μm wide, which corresponds to 3 grid points, and the overall aperture is 2.2 cm. The aperture, pitch and number of elements is the same as the ultrasound probe GEM5ScD (General Electric, USA) used for cardiac imaging and transcranial Doppler. The second array has the same aperture size as the GEM5ScD. However, each transducer covers 24 grid points, providing a pitch 8 times larger than the Nyquist limit.

The background speed of sound c_0 is set to the average speed of sound in tissues, which is 1540 m/s. The background density ρ_0 is set to 997 kg/m^3. The transmitted signal is a pulse with central frequency $f_0 = 2.8$ MHz and bandwidth $B = 6$ MHz. Then, the wavelength is equal to $\lambda_0 = 550\mu$m and the array spacing is smaller than the Nyquist limit. The sampling frequency is set to $f_s = 57$ MHz.

The shape of the layer is displayed in Figure 3.3a. The mask covers the entire transducer aperture. There is a small gap of 360 μm between the sources and the layer interface. The width of the pillars is equal to 1 mm, while the thickness varies in the interval [2,8] mm, which corresponds to [2,8] wavelengths in plastic. The thickness of each pillar is selected randomly in this interval.



(a)

(b)

(c)

Figure 3.3: Set-up for k-Wave simulations. A plastic coding mask (yellow) is placed in front of the array (black), and the virtual array (blue) is located just after the mask. For the first simulation (a), a single scattering point is placed at the center of the array. This point is highlighted in a circle. In the second simulation (b), a grid of scattering points is used. The red rectangle in (b) highlights the area at which the pressure is recorded and the $\mathbf{A}$ matrix is computed. Both simulations use the pulse shown in (c).

In the layer region, the speed of sound and density are set to 2750 m/s and 940 kg/m^3 respectively. These values are selected to simulate a plastic layer.

The virtual array is at 1 cm depth. It has the same aperture and pitch of the real array. Then, the number of virtual transducers is $V = 80$.

The imaging phantom consists of scattering points. At the scattering points, the speed of sound is set to 2050 m/s and the density is set to 500 kg/m^3. We will consider two phantoms: a single point and a grid of points. The single point is located at position (1.6, 0) cm. The grid consists of points spaced 2 mm apart (approximately 4 wavelengths) and covers the entire simulation grid, starting from a depth of 1.1 cm.

The position of the two phantoms with respect to the two arrays and the coding mask is displayed in Figures 3.3a and 3.3b. The figure displays only a section of the simulation grid, not the entire grid.

The comparison with matched filtering is performed only on the area highlighted in Figure 3.3b. Thus, the model matrix $\mathbf{A}$ will contain only the $\mathbf{a}_n$ columns corresponding to this area.

We begin by computing the transfer functions through a simulation without the scattering points. A synthetic aperture scan is acquired, and for each transmission, the pressure at the virtual array is recorded. This collection of functions is then Fourier transformed to create one transfer matrix per frequency bin $\mathbf{H}(\omega_l)$. During this simulation, the pressure at the real array is also recorded. This pressure originates at the medium-mask interfaces and will be used in the next step. In addition, the pressure in the region highlighted in Figure 3.3b is recorded. This pressure is used to make the matrices $\mathbf{G}(\omega_1), .., \mathbf{G}(\omega_L)$ and to generate the matrix $\mathbf{A}$ as in (2.8). We choose to measure the Green's functions in k-wave, instead of deriving them from the transfer functions using (2.20). In this way, we can not only validate the relation presented in Section 3.1.4, but we can also indirectly assess whether the model in (2.20) produces the same pressure in the medium as the one obtained through the k-Wave simulation.

Subsequently, a second Synthetic Aperture scan is repeated after introducing the scattering pixels. Two simulations are performed, one for each phantom. At this step, only the pressure at the real array is stored. The reflections from the mask, measured at the previous step, are subtracted from the data. Moreover, zero-mean white Gaussian noise is added to this data. The noise power is adjusted to achieve a signal-to-noise ratio of 20 dB. Finally, the data is Fourier transformed to form one matrix $\mathbf{Y}(\omega_l)$ per frequency bin.

Instead of repeating the simulations for the undersampled array, we just group the transducers in 10 sets of 8 and we sum the entries of $\mathbf{H}(\omega_l)$ and $\mathbf{Y}(\omega_l)$ over these sets. The summation is performed on the noiseless measurement. Next, the signal power is computed and noise is added. The noise power is adjusted to obtain an SNR of 20 dB. Finally, the summation is repeated on the $\mathbf{G}(\omega_l)$ matrices and the model matrix $\mathbf{A}$ for the undersampled array is computed.

In addition, a further simulation is performed after removing the coding mask from the speed-of-sound and density maps. This data will be used for comparison.

3.3 Results

In this section, we apply the proposed method to image point scatterers through a plastic coding mask. We begin with the single point phantom in Section 3.3.1, and use a single plane wave transmission as the virtual transmit scheme. We analyze the estimates of the wavefields $\mathbf{u}(\omega)$, $\mathbf{q}(\omega)$ and $\mathbf{z}(\omega)$, as defined in Section 3.1. Next, we perform beamforming on $\hat{\mathbf{z}}(\omega)$ and examine the Point Spread Function (PSF). In Section 3.3.2, we consider the grid phantom. In this case, we focus solely on the beamformed images and do not analyze the estimated waves. The main goal is to verify that the PSF is centered at the correct location, exhibits relatively tight dimensions, and has low sidelobes. We assess the method's performance with various transmission schemes and using both fully sampled and undersampled arrays. Additionally, we evaluate the effectiveness of the two inverse problem solution strategies described in this chapter. Finally, we compare the proposed method to matched filtering with the matrix $\mathbf{A}$.

3.3.1 Single point phantom

We consider the data for the single point simulation and we derive the echoes at the virtual array for a plane wave transmission. For this transmission, the received echoes form a parabola in the space-time domain. Our goal is to retrieve this parabolic shape with our method. Instead of directly computing the waveforms in $\hat{\mathbf{z}}(\omega)$ as in (3.2), we will derive them following the steps mentioned in Section 3.1. In this way, we also gain insights into the characteristics of the transmit field.

First, we use the matched filter approach and we set $\mathbf{W}(\omega)$ to $\mathbf{H}^{\mathrm{H}}(\omega)$. Then, the input needed to have a plane wave transmission on the virtual array is $\hat{\mathbf{u}}(\omega) = \mathbf{H}^{*}(\omega)\mathbf{1}$. The time domain representation is shown in Figure 3.4b. The transmit field at the virtual array is $\hat{\mathbf{p}}(\omega) = \mathbf{H}^{\mathrm{T}}(\omega)\mathbf{H}^{*}(\omega)\mathbf{1}$. The time domain is shown in Figure 3.4c. It can be seen that it is an approximation of a plane wave. However, the pressure amplitude varies depending on the array element. As mentioned before, the amplitude at i depends on the pressure peak when focusing at element i with time-reversal. Thus, it exhibits a lower amplitude at elements where it is more challenging to focus.

Next, we derive the echoes at the real array for the $\hat{\mathbf{u}}(\omega)$ input. In k-Wave it is possible to set the input signals to these waveforms. This is not the case in a real situation, as explained in Section 3.1. Then, we use the Synthetic Aperture scan as in (3.1). The echoes at the real array when sending $\hat{\mathbf{u}}(\omega)$ are computed as $\hat{\mathbf{q}}(\omega) = \mathbf{Y}(\omega)\hat{\mathbf{u}}(\omega)$. The time domain representation is shown in Figure 3.4d. In this Figure, we can observe the effect of aberrations on a wavefront. In a homogeneous medium, the echo from a point source forms a parabola. In this case, it deviates significantly from the parabolic shape. We can recover the parabolic shape after computing $\hat{\mathbf{z}}(\omega) = \mathbf{H}^{\mathrm{H}}(\omega)\mathbf{q}(\omega)$. This is displayed in Figure 3.4e. The amplitude is not constant over the parabola. We observe a decrease as we move towards the borders of the array. This decrease is partially due to the attenuation that occurs in the medium. We can also notice that the pulse shape has a broader support with respect to $s(t)$, for the reasons explained in Section 3.1.

Finally, the data $\hat{\mathbf{z}}(\omega)$ are beamformed to create an image. An image domain of dimension 100×80 is defined, centered at the scattering point location. Delay-and-sum is applied to generate an image. A hilbert transform is applied to obtain the envelope signal which is dispalyed in the image. Furthermore we normalize each pixel in the image to the overal maximum and apply a log transformation to the image to enhance visibilty. These steps will be applied to all the images displayed in this thesis. The resulting image is displayed in Figure 3.4f.

In Section 3.1.2, we have discussed the option of using an inverse filter instead of a matched filter. This is now applied on this data. $\hat{\mathbf{z}}(\omega)$ is computed as $\hat{\mathbf{z}}(\omega) = \mathbf{W}(\omega)\mathbf{Y}(\omega)\mathbf{W}^{\mathrm{T}}(\omega)\mathbf{1}$. The time domain of $\hat{\mathbf{z}}$ is shown in Figure 3.4g. Here we just plot $\hat{\mathbf{z}}(\omega)$ to show the effect on the final estimate. The parameter λ is tuned based on the results. Once again, we can observe the parabola shape appearing in the plot, and this

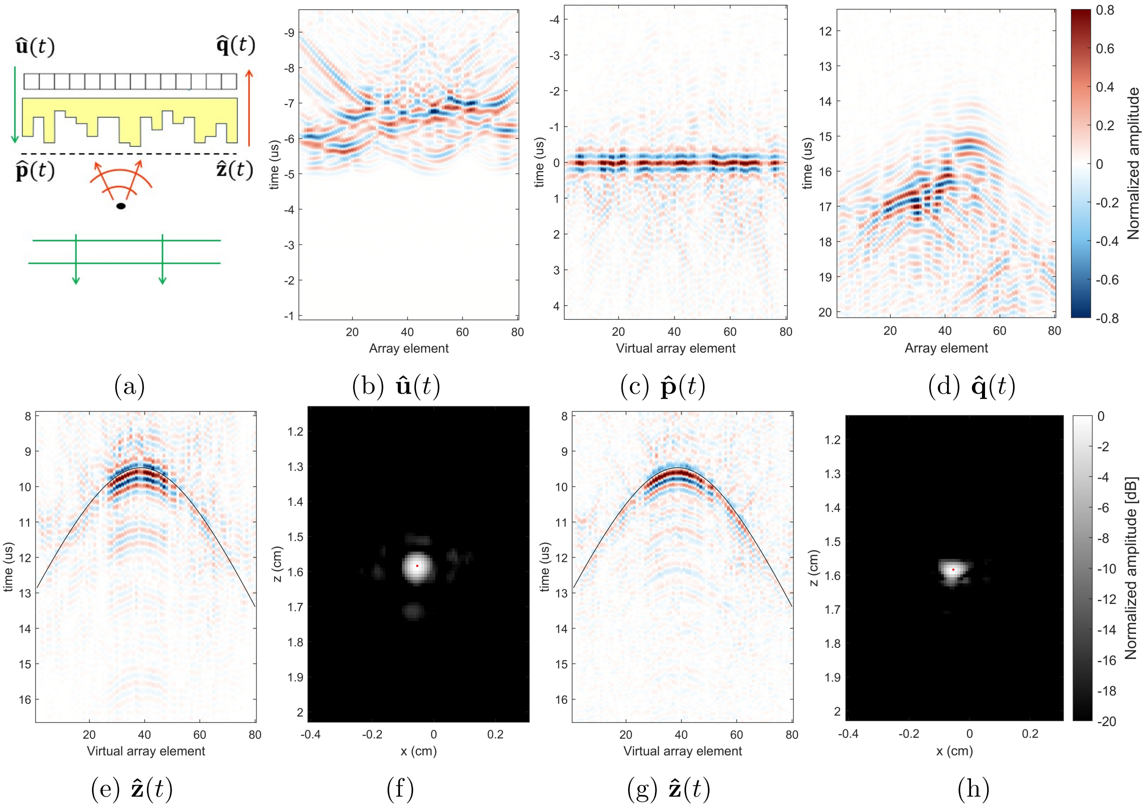
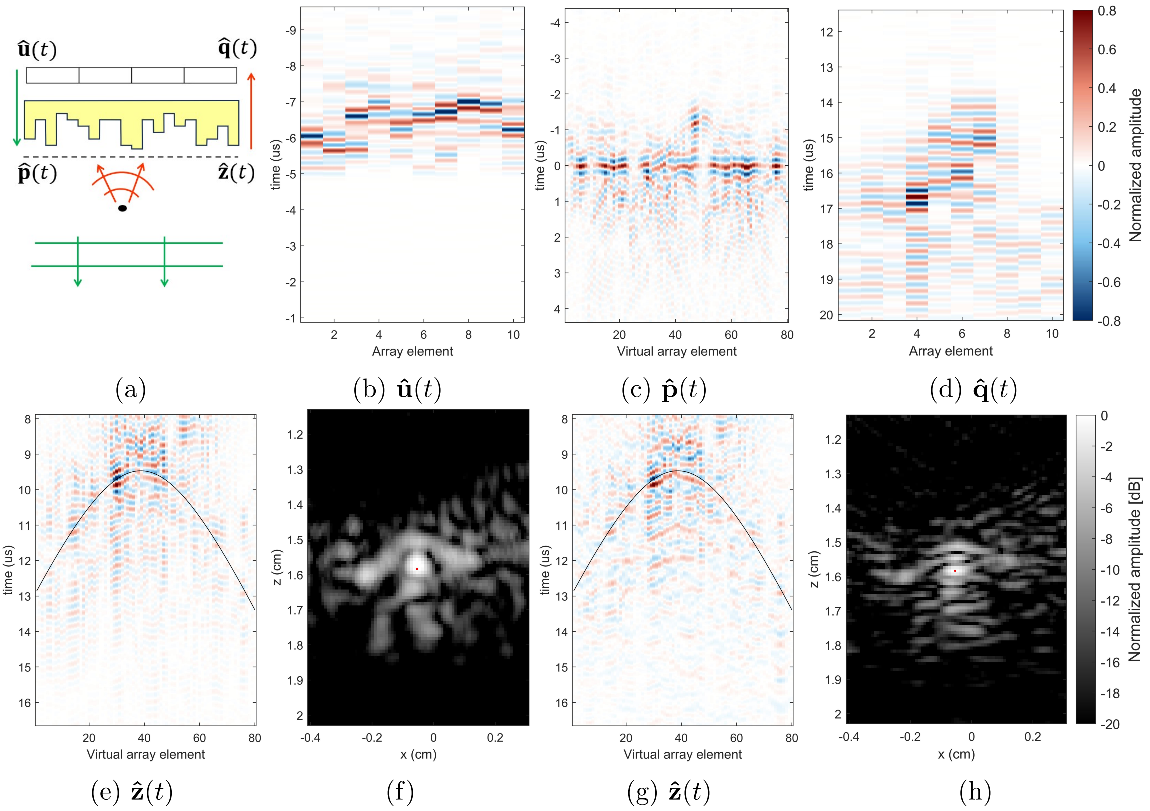


Figure 3.4: Transmitting a plane wave through the coding mask (a) is possible by transmitting the waveforms in (b), which will form a plane wave at the virtual array (c). From the received echoes at the real (d), we derive the echoes at the virtual array (e). The data in (e) is DAS beamformed, resulting in image (f). The estimation is repeated using the regularized inverse filter, resulting in a better estimate of the echoes (g) and a smaller PSF (h).

time it is even more pronounced than before. The pulse shape exhibits a shorter time support, and we also notice a more uniform amplitude distribution over the parabola. After beamforming, we achieve a smaller PSF in the axial dimension, while the lateral dimension remains almost unchanged. The sidelobes are lower than -20 dB in both cases.

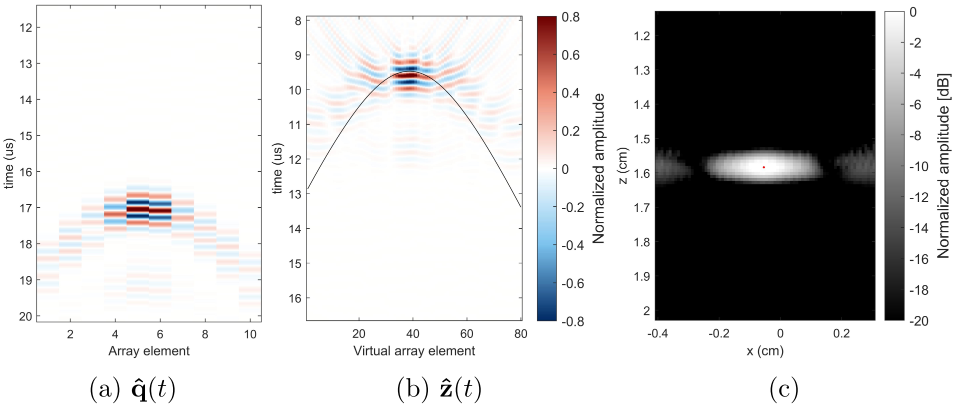
In Section 3.1.3, we mentioned that this method can, in principle, be applied when the transducers' spacing is smaller than the Nyquist limit. Now we consider the data for the undersampled array and we repeat the steps explained before. The results are shown in Figure 3.5. In this case, the number of real transducers is $M = 10$, while the number of virtual transducers is $V = 80$. Transmitting a plane wave through the medium with just 10 elements is more challenging, as we can see in Figure 3.5c. For one part of the elements, the pressure is a pulse with peak at $t = 0$. However, there are also elements where the support is longer. Moreover, the amplitude is less uniform than in Figure 3.4c. Similarly, the estimate of $\hat{\mathbf{z}}(\omega)$ is more distant from a parabola, resulting in a poorer PSF. As we can see in Figure 3.5f, an intensity peak at the correct location is obtained. However, 4 more spots at approximately -2 dB can be observed. Improvements can be obtained when using the regularized inverse filter, as the intensity



(a) (b) $\hat{\mathbf{u}}(t)$ (c) $\hat{\mathbf{p}}(t)$ (d) $\hat{\mathbf{q}}(t)$

(e) $\hat{\mathbf{z}}(t)$ (f) (g) $\hat{\mathbf{z}}(t)$ (h)

Figure 3.5: Transmitting a plane wave with an undersampled array (a) is still possible, but more challenging. By transmitting the waveforms in (b), an approximation of a plane wave is obtained (c). From the received echoes at real (d), we derive the echoes at the virtual array (e). The data in (e) is DAS beamformed, resulting in image (f). The PSF is worse than the one for fully sampled array. Improvements can be obtained by using a regularized inverse filter (g),(h).



(a) $\hat{\mathbf{q}}(t)$ (b) $\hat{\mathbf{z}}(t)$ (c)

Figure 3.6: Results obtained from a simulation without coding mask. $\hat{\mathbf{z}}(t)$ (b) is a more densely sampled version of $\hat{\mathbf{q}}(t)$ (a), thus it is not possible to reproduce the desired curvature. This results in a large PSF (c). The advantage of using a mask is evident when comparing these plots to the estimates in Figure 3.5.

of the sidelobes becomes lower and the size of the PSF becomes smaller.

Even though the $\hat{\mathbf{z}}(\omega)$ waveforms appear quite different from the expected parabola, the estimation process successfully captures the correct curvature in the central part of the virtual array. It is important to consider that $\hat{\mathbf{z}}(\omega)$ is obtained from $\hat{\mathbf{q}}(\omega)$. The spatial variations of $\hat{\mathbf{q}}(\omega)$ in the lateral dimensions would not be sufficient to achieve the desired curvature, since we have only 10 transducers. Through a multiplication with $\mathbf{H}^{\mathrm{H}}(\omega)$, we are able to synthesize a more spatially varying field. This would not hold if we didn't use a coding mask. In Figure 3.6 the results for a simulation with no layer are shown. Although $\hat{\mathbf{z}}(\omega)$ is more densely sampled than $\mathbf{q}(\omega)$, it does not contain higher spatial frequencies, resulting in a larger focal spot in the lateral dimension.

3.3.2 Grid phantom

In this section, we consider the data for the simulation with a grid of scatterers. So far, we have only considered a single plane wave transmission. However, the PSF can be significantly improved by combining multiple transmissions. This compounding technique implies no additional cost in terms of acquisition time, as a Synthetic Aperture scan is required for both approaches. In this section, we will investigate two virtual transmit schemes: Synthetic Aperture (SA) and Plane-Wave Compounding (PWC).

First, we estimate the received echoes for a SA scan at the virtual array. These are computed as in (3.8). Both the fully sampled and the undersampled array are used. For each array, we compare the matched filter and the regularized inverse. Again, the λ parameter has been tuned based on the results. The resulting images for the fully sampled array are shown in Figures 3.7b and 3.7c. The images for the undersampled array are in Figures 3.8b and 3.8c.

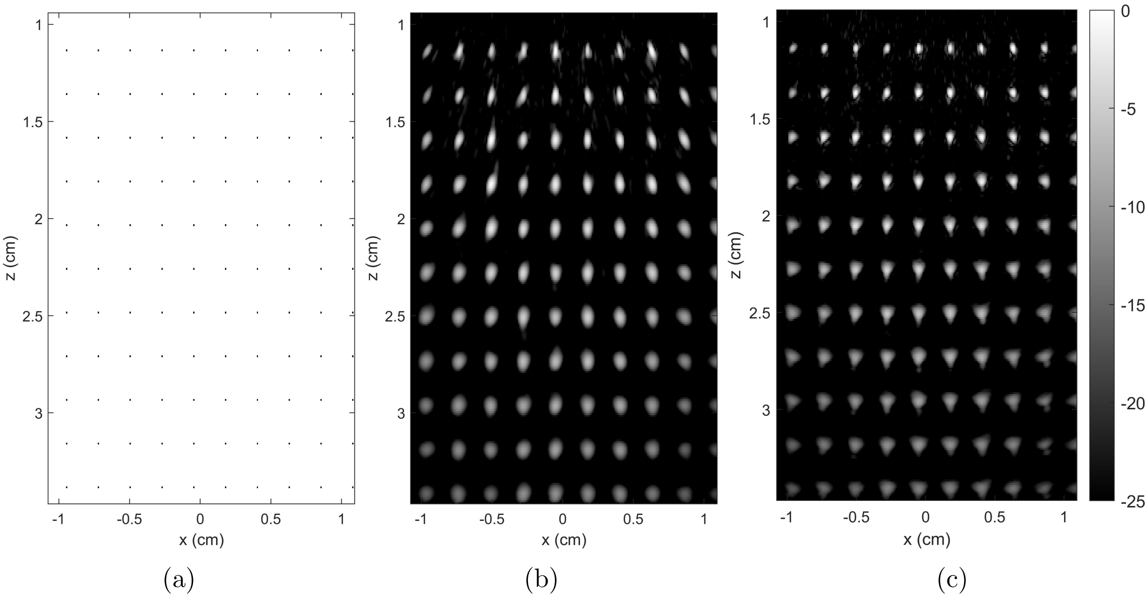
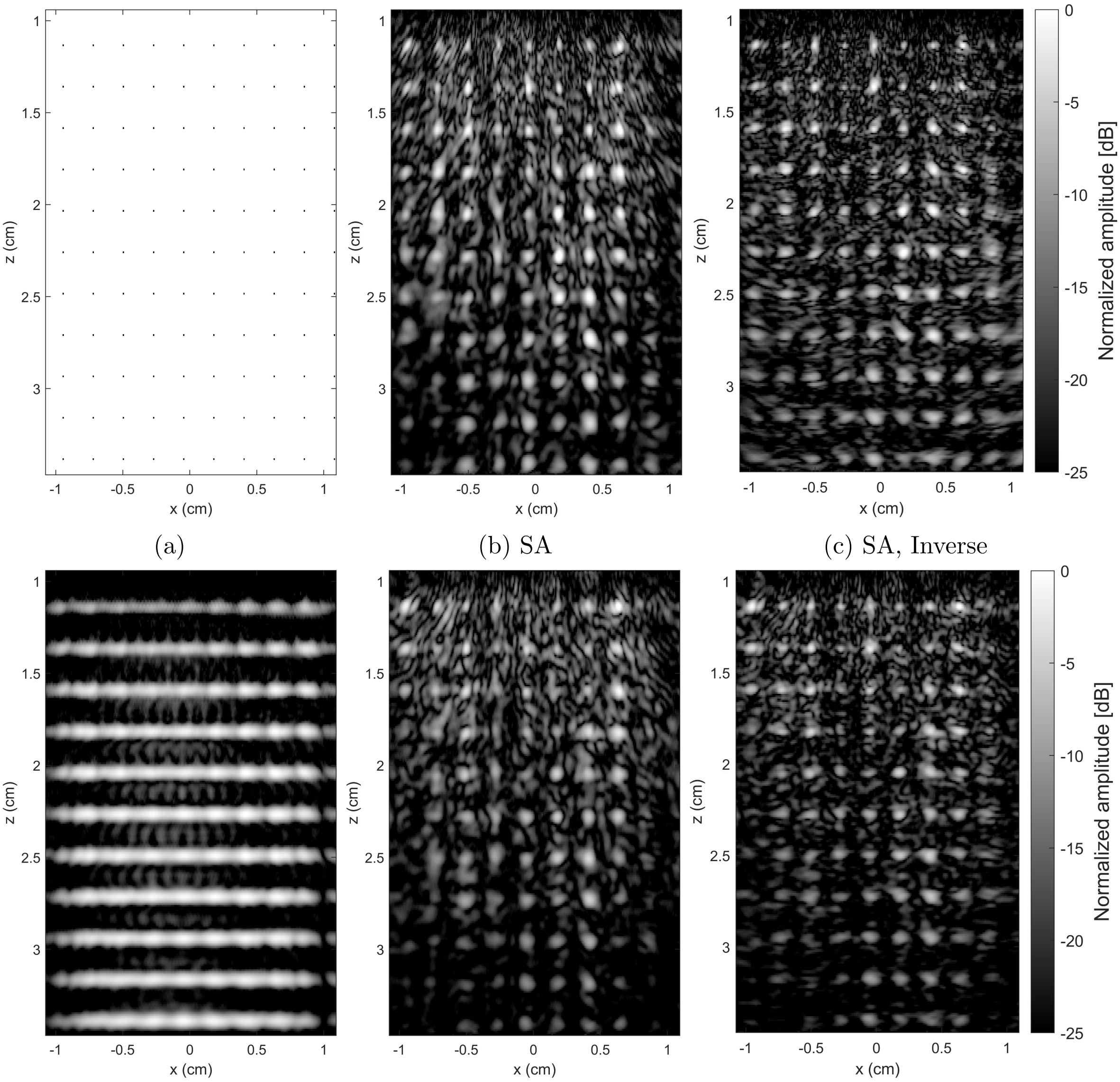


Figure 3.7: Images of a grid of scatterers (a) obtained with a fully sampled array ($M = 80$). A virtual SA scan is considered and the matched filter approach is applied (b). Next, the regularized inverse approach (c) is applied, leading to a slightly smaller PSF.



(a) (b) SA (c) SA, Inverse

(d) SA, no coding mask (e) PWC (f) PWC, Inverse

Figure 3.8: Images of a grid of scatterers (a) obtained with an undersampled array ($M = 10$). A virtual SA scan is considered, and both the matched filter (b) and regularized inverse (c) approaches are applied. To reduce the computations, the virtual transmit scheme can be set to Plane-Wave Compounding with 9 plane waves. In all cases, the results show significant improvement compared to a simulation without a coding mask (c).

The PSF varies across the image. The quality of the PSF at a particular point depends on the focus quality that can be achieved at that specific location. Since the focus quality beyond a layer is not uniform over the imaging domain, the PSF differs accordingly.

When using the regularized filter, the PSF becomes smaller. This effect is noticeable in both the fully sampled and undersampled array cases. In the undersampled array scenario, we can also observe better sidelobe suppression. If there is no coding mask

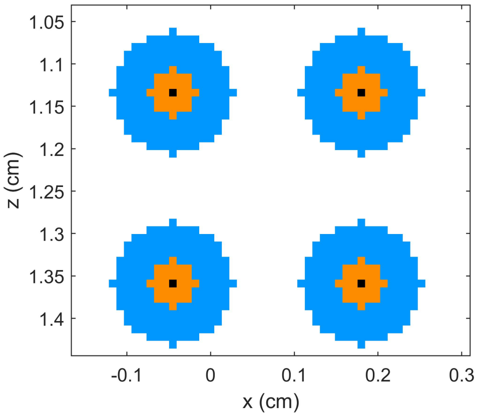


Figure 3.9: The two masks used for the image quality metric computation are displayed. For each scattering point, the ratio between the mean intensity in the inner circle (orange) and in the outer circle (blue) is computed. The image quality is the average over the whole grid.

between the real and virtual array, it is not possible to estimate the data for a fully sampled virtual array. In Figure 3.8d, the resulting image for the no-layer simulation is shown. The PSF is so large that the single points cannot be distinguished.

In Section 3.3.1, we applied only one virtual transmission. In this section, the number of virtual transmissions is increased to 80, which corresponds to the number of virtual transducers. Although the acquisition time remains the same, as we need Synthetic Aperture from the real array in any case, a higher number of virtual transmissions leads to longer image computation times. Considering this, we look at what happens when the number of virtual transmissions is reduced. Specifically, we will consider 9 plane waves at evenly spaced angles in the interval $[-15°, 15°]$. The resulting images are displayed in Figures 3.8e and 3.8f.

To provide a quantitative comparison, we introduce an image quality metric. For each scattering point, we define two concentric circles centered at the point's location. The metric computation involves calculating the mean image intensity within the inner circle and the mean in the area between the two circles, followed by taking their ratio. This process is repeated for all scattering points, and the mean is then computed over the entire image grid.

The inner circle has a diameter equal to 7 grid points, which is equivalent to 1 wavelength, while the outer circle has a diameter equal to 17 pixels. Figure 3.9 displays the area considered for the metric computation for 4 points. Only 4 points are shown in the figure to illustrate the circle's dimension compared to the distance between the points. The image metric is calculated for all the images presented in Figures 3.7 and 3.8, and the image quality values are reported in Table 3.1.

N. transducers	Inverse problem solution	Virtual TX scheme	Figure label	Image quality
80	Matched	SA	3.7b	1.24
80	Regularized inverse	SA	3.7c	1.44
10	Matched	SA	3.8b	0.83
10	Regularized inverse	SA	3.8c	0.87
10	Matched	PWC	3.8e	0.68
10	Regularized inverse	PWC	3.8f	0.77

Table 3.1: Image quality metric for all the images in Figure 3.7 and Figure 3.8

Finally, we compare the proposed method with matched filtering. We apply matched filtering on both the fully sampled and undersampled array data. The resulting images are displayed in Figures 3.10a and 3.10c. In section 3.1.4 it has been shown that matched filtering with $\mathbf{A}$ is equivalent to this method if we consider a SA scan at the virtual array and we set $\mathbf{W}(\omega) = \mathbf{H}^{\mathrm{H}}(\omega)$. These images have already been computed for both the fully sampled and undersampled array and are displayed in Figures 3.7b and 3.8b. The area selected for the comparison is extracted and is displayed again in Figures 3.10b and 3.10d. In the fully sampled case, the dynamic range has been increased to 40 dB to better display the similarities in the sidelobe levels.

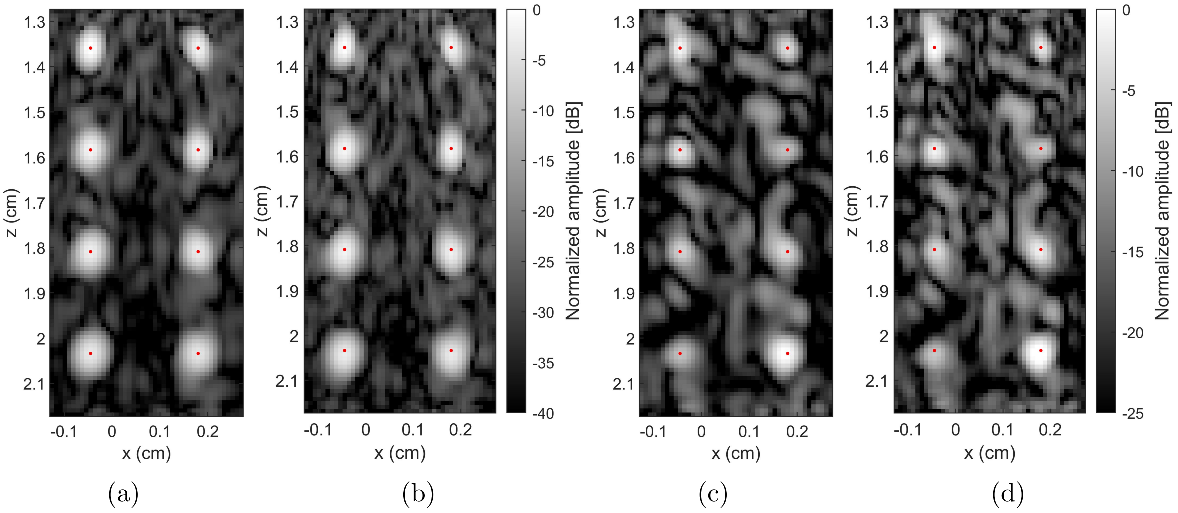


Figure 3.10: Comparison between the matched filtering with the model matrix $\mathbf{A}$ and the proposed method. In (a) and (b), the images for the fully sampled array are shown, while (c) and (d) are computed using the undersampled array measurements. (a) and (c) display the matched filtered images, while (b) and (d) display the images obtained with the proposed method. The red dots indicate the location of the scatterers.

3.4 Discussion

When imaging through aberrating layers, using the traditional DAS method becomes ineffective, and model-based methods are required. In this chapter, we have introduced an alternative approach that enables the application of DAS in the presence of aberrating layers. In this section, the advantages and disadvantages of the proposed method are discussed. Additionally, we relate this work to existing research in the field.

Comparison with model-based methods

In previous sections, it has been mentioned that the model-based methods are computationally expensive. The number of computations needed to generate an image varies significantly depending on the array and on the dimension of the imaging domain. In this discussion, we focus on a scenario where model-based imaging is particularly inconvenient. We consider a situation where a 2D array is used to perform 3D imaging. The model linking the SA measurements $\mathbf{y}$ to the scattering amplitudes $\mathbf{x}$ is $\mathbf{y} = \mathbf{A}\mathbf{x}$, where $\mathbf{A}$ has dimensions $M^2L \times N$. The computation of $\mathbf{A}$ involves propagating the field through the entire imaging domain using the Angular Spectrum Approach (ASA) [27]. The procedure to generate the $\mathbf{A}$ matrix is described in [6]. This process is illustrated in Figure 3.11. We assume the number of frequencies is $L = 256$, the array dimension is 10×10, then $M = 10^2$, while the virtual array is 80×80, then $V = 80^2$. We assume the number of pixels in the image domain is $N = 120^3$ and each entry of the $\mathbf{A}$ matrix occupies 4 bytes. These have been selected arbitrarily and represent realistic values. In this case, the memory required to store $\mathbf{A}$ is $M^2LN = 10^4 \times 256 \times 120^3 \times 4$ bytes ≈ 17 TeraBytes. Storing this matrix would be impractical. Therefore, computing a matched filtered estimate of the image would not be possible in a single operation. Instead, it would require multiple steps. First, the field must be propagated to the first plane, and then the $\hat{x}_n$ values for the pixels at that plane can be computed. The process is then repeated for the next plane, and so on until the entire image is formed. The major

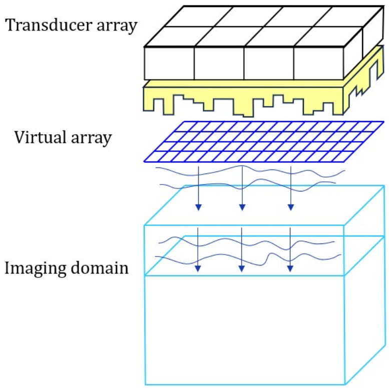


Figure 3.11: In this scenario, a 2D array is employed for 3D imaging. To construct the $\mathbf{A}$ matrix, field propagation using the Angular Spectrum Approach (ASA) is required. However, due to its large size, storing the matrix is not feasible. Field propagation needs to be repeated when generating each frame $\hat{\mathbf{x}}$, leading to a significant computational burden.

challenge lies in the fact that field propagation needs to be performed for each frame $\hat{\mathbf{x}}$, resulting in a substantial increase in computational cost.

If the method proposed in this thesis is used, there is no need to propagate the field through ASA. Instead, we can directly apply DAS. Before applying DAS, the $\hat{\mathbf{Z}}(\omega_l)$ matrices need to be computed, and each $[\hat{Z}_{i,j}(\omega_l), .., \hat{Z}_{i,j}(\omega_L)]$ has to be inverse Fourier transformed. The cost of this operation needs to be added to the cost of DAS. However, this operation is done just once for the whole image, then its cost does not scale with the number of pixels. Overall, the cost is expected to be lower than matched filtering with $\mathbf{A}$.

Since the virtual array contains a large number of elements, performing DAS at the virtual array on SA data can be quite computationally demanding. For this reason, the option of using Plane-Wave Compounding has been proposed. When using PWC as virtual transmit scheme, the image quality may be degraded. Therefore, one needs to evaluate whether the image quality is sufficient depending on the specific scenario.

Inverse filter as an alternative to time-reversal:

In this chapter, it has been shown that a regularized inverse filter can be a valid alternative to a matched filter/time-reversal approach. Several papers in the literature have analyzed the potential of reducing the focus spot size and sidelobes by using an inverse filter instead of time-reversal [12] [28]. In [12], Tanter et al. demonstrated that using an inverse filter leads to improvements when dealing with lossless media. This is in line with our results, where we observed an improvement using the regularized inverse filter. However, it is essential to note that this observation is valid for the current level of noise and may change as the noise increases. It is well-known that the performance of the regularized inverse filter heavily depends on the noise level. Moreover, it changes depending on the rank of the transfer matrix $\mathbf{H}(\omega_l)$ and the regularization parameter λ. Prior research has explored methods to determine the rank of the transfer matrix, as seen in Tanter et al. [12] and Aubry et al. [29]. In this thesis, the rank of $\mathbf{H}(\omega_l)$ has not been analyzed and the parameter λ has been tuned based on the results.

Imaging through unknown aberrators 4

In the previous chapters, it has been shown that aberrating layers do not pose an obstacle for successful imaging, if the layer transfer functions between the real and virtual array are known. In the case of a coding mask, these transfer functions can be measured. However, when the layer cannot be removed, measuring $\mathbf{H}(\omega)$ becomes invasive and impractical. This chapter addresses the challenge of imaging through layers for which the transfer matrix cannot be directly measured.

Typically, there is no prior knowledge about the imaging target, and the only available data are the pulse-echo measurements. The relation between the measurements, the image and the layer transfer functions is shown in (2.20). Imaging through unknown aberrators involves estimating both $\mathbf{H}(\omega)$ and $\mathbf{x}$ from the given $\mathbf{y}$. This is challenging as there is a large number of unknowns compared to the number of knowns.

In this chapter, we investigate the possibility of estimating $\mathbf{H}(\omega)$ by optimizing a cost function. The cost function is designed to measure the focus quality beyond the layer. Instead of directly optimizing for $\mathbf{H}(\omega)$, we adopt a different approach. We model $\mathbf{H}(\omega)$ and treat the model's parameters as optimization variables. By doing so, we reduce the dimensionality of the solution space. In Section 4.1, the cost function is introduced, while Section 4.2 is dedicated to the parametrization. The main objectives are:

1. To investigate whether the transfer functions can be approximated using the proposed parametrization
2. To explore whether increasing the defined cost function results in an improved estimate of the transfer matrix, and subsequently improves the quality of the generated image.

To achieve these objectives, we apply the proposed model and optimization approach to simulated data. The details of the data generation and algorithm implementation are explained in Section 4.3. The results are presented in Section 4.4. In Section 4.5, we discuss the obtained results and the limitations of this work.

4.1 Focus quality measure

In this section, we derive the cost function that will be employed in the optimization. The goal is to find a function of $\mathbf{H}(\omega)$ that yields high values for good estimates of $\mathbf{H}(\omega)$ and low values for poor estimates.

Given an estimate of $\mathbf{H}(\omega)$, we can derive an estimate of the Green's functions at any point n using the equation $\hat{\mathbf{g}}_n(\omega) = \hat{\mathbf{H}}(\omega)\tilde{\mathbf{g}}_n(\omega)$, with $\tilde{\mathbf{g}}_n(\omega)$ as defined in Section 2.4. A good estimate of $\mathbf{H}(\omega)$ is one for which $\hat{x}_n = \sum_\omega \hat{\mathbf{g}}_n^H(\omega)\mathbf{Y}(\omega)\hat{\mathbf{g}}_n^*(\omega)$ accurately reflects

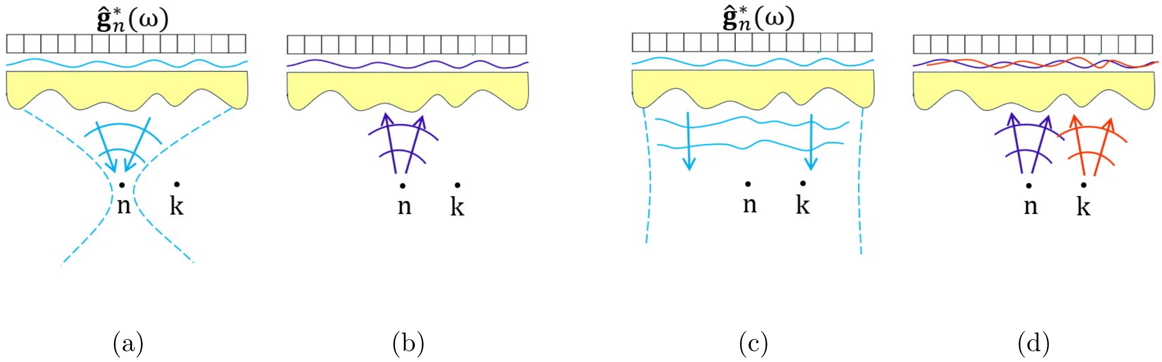

$\hat{\mathbf{g}}_n^*(\omega)$
n
k
n
k
$\hat{\mathbf{g}}_n^*(\omega)$
n
k
n
k
(a)
(b)
(c)
(d)

Figure 4.1: A good estimate $\hat{\mathbf{g}}_n(\omega)$ is one that produces a field focused at point n after propagating through the layer (a). As a consequence, when transmitting $\hat{\mathbf{g}}_n(\omega)$, we receive only echoes from n (b). In contrast, a poor estimate results in an unfocused transmit field and echoes scattered from locations different from n, as depicted in (c) and (d) respectively.

x_n. This is the case if the pressure beyond the layer when transmitting $\hat{\mathbf{g}}_n^*(\omega)$ forms a beam focused at n. To identify a good $\hat{\mathbf{H}}(\omega)$, we need a metric that measures how well we can focus at n using $\hat{\mathbf{g}}_n(\omega)$. To simplify the notation, throughout this section we will refer to these $\hat{\mathbf{g}}_n(\omega)$ estimates without explicitly writing the dependency on $\hat{\mathbf{H}}(\omega)$. However, it is essential to remember that what we are actually estimating is $\mathbf{H}(\omega)$, and $\hat{\mathbf{g}}_n(\omega)$ is a function of $\hat{\mathbf{H}}(\omega)$.

As we aim to use a non-invasive approach, measuring the pressure beyond the layer is not possible. Our assessment of focus quality must rely solely on the pulse-echo data. To assess focus quality based on pulse-echo data, we rely on the idea that when focusing at a location n, we should receive echoes only from that location. This concept is illustrated in Figure 4.1.

As mentioned in Section 3.1, it is possible to derive the received echoes for any input from Synthetic Aperture measurements. Therefore, we can obtain the received echoes when trasmitting $\hat{\mathbf{g}}_n^*(\omega)$ as $\mathbf{q}(\omega) = \mathbf{Y}(\omega)\hat{\mathbf{g}}_n^*(\omega)$. These should only contain echoes from point n. Then, if we cross-correlate these echoes with the Green's functions at another location $\hat{\mathbf{g}}_k(\omega)$ ($k \neq n$), the result should be zero. This is the main idea behind our cost function for the transfer matrix optimization.

We introduce a new variable $x_{k,n}$ which represents the intensity of the echo scattered from point k when focusing at point n. This is equal to:

$$x_{k,n} = \sum_l \hat{\mathbf{g}}_k^{\mathrm{H}}(\omega_l)\mathbf{Y}(\omega_l)\hat{\mathbf{g}}_n^*(\omega_l). \tag{4.1}$$

It can be noticed that for $k = n$, this corresponds to the matched filtered estimate of x_n.

The indices k and n vary from 1 to N, with N equal to the number of pixels. We define a matrix $\mathbf{X}$ of dimension $N \times N$, in which the (k, n) entry is $x_{k,n}$. This matrix is equal to:

$$\begin{aligned}\mathbf{X} &= \sum_l \hat{\mathbf{G}}^{\mathrm{H}}(\omega_l)\mathbf{Y}(\omega_l)\hat{\mathbf{G}}^*(\omega_l)\\ &= \sum_l \hat{\mathbf{G}}^{\mathrm{H}}(\omega_l)\mathbf{G}(\omega_l)\mathrm{diag}(\mathbf{x})\mathbf{G}^T(\omega_l)\hat{\mathbf{G}}^*(\omega_l),\end{aligned} \tag{4.2}$$

where we have replaced $\mathbf{Y}(\omega_l)$ with the expression in 2.11. A simplified model with zero noise is considered.

We aim to exploit that a good $\hat{\mathbf{G}}(\omega)$ is one for which $x_{k,n}$ is zero for $k \neq n$. As we have previously explained, if $\hat{\mathbf{g}}_n^*(\omega)$ allows focusing at point n, and this holds for all points, the matrix $\hat{\mathbf{G}}^{\mathrm{H}}(\omega)\mathbf{G}(\omega)$ becomes diagonal. Consequently, to assess the quality of the estimate, we should examine the diagonality of the matrix $\mathbf{X}$. However, it has been found that considering all $x_{k,n}$ entries may not be the best approach, and it is preferable to focus on a subset of these entries. From this point on, we consider only the $x_{k,n}$ for k and n at the same depth.

To explain why, we replace $\mathbf{Y}(\omega_l) = \sum_m x_m \mathbf{g}_m(\omega_l)\mathbf{g}_m(\omega_l)^{\mathrm{T}}$ in (4.1) and we write the variables $x_{k,n}$ as:

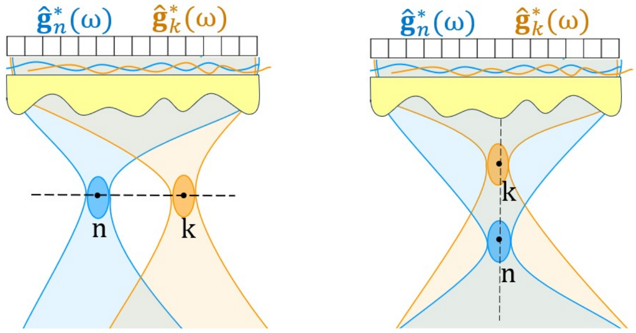
$$\begin{aligned}x_{k,n} &= \sum_l \hat{\mathbf{g}}_k^{\mathrm{H}}(\omega_l)\Big(\sum_m x_m \mathbf{g}_m(\omega_l)\mathbf{g}_m^T(\omega_l)\Big)\hat{\mathbf{g}}_n^*(\omega_l)\\ &= \sum_m x_m \sum_l \mathbf{g}_m^T(\omega_l)\hat{\mathbf{g}}_k^*(\omega_l)\mathbf{g}_m^T(\omega_l)\hat{\mathbf{g}}_n^*(\omega_l)\\ &= \sum_m x_m w_m^{kn},\end{aligned} \tag{4.3}$$

where we have defined $w_m^{kn} = \sum_l \mathbf{g}_m^T(\omega_l)\hat{\mathbf{g}}_k^*(\omega_l)\mathbf{g}_m^T(\omega_l)\hat{\mathbf{g}}_n^*(\omega_l)$.

For fixed n and k, $x_{k,n}$ can be interpreted as a weighted sum over m of the underlying pixel echogenicity values x_m. We now derive an upper bound for the weights w_m^{kn}. As explained in the Section 2.1, $\mathbf{g}_m^T(\omega)\hat{\mathbf{g}}_k^*(\omega)$ and $\mathbf{g}_m^T(\omega)\hat{\mathbf{g}}_n^*(\omega)$ represent the pressure at m when we transmit $\hat{\mathbf{g}}_k^*(\omega)$ and $\hat{\mathbf{g}}_n^*(\omega)$ respectively. We define $P_k(\mathbf{r},\omega)$ as the pressure in the medium when sending $\hat{\mathbf{g}}_k^*(\omega)$, and as $P_n(\mathbf{r},\omega)$ the pressure when sending $\hat{\mathbf{g}}_n^*(\omega)$. By using basic algebra relations, we can derive:

$$\begin{aligned}w_m^{kn} &\leq \sum_l |P_n(\mathbf{r}_m,\omega_l)|^2 |P_k(\mathbf{r}_m,\omega_l)|^2\\ &\leq \Big(\sum_l |P_n(\mathbf{r}_m,\omega_l)|^2\Big)\Big(\sum_l |P_k(\mathbf{r}_m,\omega_l)|^2\Big).\end{aligned} \tag{4.4}$$

$\sum_l |P_n(\mathbf{r}_m,\omega_l)|^2$ for varying $\mathbf{r}_m$ represents the beam obtained when focusing on point n. Similarly, $\sum_l |P_k(\mathbf{r}m,\omega_l)|^2$ corresponds to the beam obtained when focusing on point k. Figure 4.2 shows a schematic depicting the relation between these two beams. When points n and k are at the same depth (Figure 4.2a), the two beams do not overlap significantly. As a result, the upper bound for w_m^{kn} is low at all points m, leading to an overall low value for $x_{k,n}$. However, for two points aligned in the axial dimension (Figure 4.2b), the overlap between the two beams is more significant, resulting in higher upper bounds for the weights w_m^{kn}.



(a) (b)

Figure 4.2: Schematic of the beams when transmitting $\hat{\mathbf{g}}_n(\omega)$ (blue) and $\hat{\mathbf{g}}_k(\omega)$(orange). When points n and k are at the same depth (a), the two beams do not overlap significantly. As as result, the corresponding $x_{k,n}$ is guaranteed to be low. This is not true for two points aligned in the axial dimension (b). Considering n and k at the same depth is expected to be more effective in identifying a good $\hat{\mathbf{G}}(\omega)$.

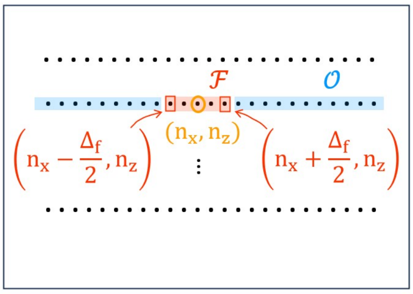

$\mathcal{F}$
$\mathcal{O}$
(n_x, n_z)
$\left(n_x - \frac{\Delta_f}{2}, n_z\right)$
$\left(n_x + \frac{\Delta_f}{2}, n_z\right)$
⋮

Figure 4.3: For each point n, we compute the $x_{k,n}$ for the k points at the same depth as n. The cost contribution of this point is given by $\sum_{k\in\mathcal{F}} |x_{k,n}|^2$ divided by $\sum_{k\in\mathcal{O}} |x_{k,n}|^2$.

Since the goal is to identify a good estimate as the one for which the $x_{k,n}$ terms have low values, it is expected that considering only n, k at the same depth is more effective.

To compute the cost function, we select a fixed n and calculate $x_{k,n}$ for all k at the same depth as n. Next, we divide the points into two sets: those inside the focal spot $\mathcal{F}$ and those outside the focal spot $\mathcal{O}$. The two sets for a specific point n are displayed in Figure 4.3. A good estimate $\hat{\mathbf{G}}(\omega)$ is one for which the sum of squared magnitudes $\sum_{k\in\mathcal{F}} |x_{k,n}|^2$ is high, while the sum for $\sum_{k\in\mathcal{O}} |x_{k,n}|^2$ is low. Thus, we will use the ratio of these sums as the cost function:

$$f_n = \frac{\sum_{k\in\mathcal{F}} |x_{k,n}|^2}{\sum_{k\in\mathcal{O}} |x_{k,n}|^2} \tag{4.5}$$

This operation is repeated for all n in a pre-defined grid, and the overall cost function is calculated as the sum of individual point contributions, denoted as $f = \sum_n f_n$.

To explicate the dependency of f on $\mathbf{H}(\omega)$, we reformulate the cost function. For this purpose, we introduce the matrices $\mathbf{X}_{n_z}$, for all the n_z in the defined grid. Each matrix has dimension $K \times K$, where K represents the number of points in the x-dimension, and it contains only the $x_{k,n}$ for k, n at a specific depth n_z. $\mathbf{X}_{n_z}$ is equal to:

$$\mathbf{X}_{n_z} = \sum_l \hat{\mathbf{G}}_{n_z}^{\mathrm{H}}(\omega_l)\mathbf{Y}(\omega_l)\hat{\mathbf{G}}_{n_z}^{*}(\omega_l). \tag{4.6}$$

$\mathbf{G}_{n_z}(\omega_l)$ is $M \times K$ and it contains only the $\mathbf{g}_n(\omega_l)$ columns that correspond to points at the depth n_z.

Using $\mathbf{X}_{n_z}$, we can write f_n as:

$$f_n = \frac{||\mathbf{W}_{n_x}^{F} \odot \mathbf{X}_{n_z}||_F^2}{||\mathbf{W}_{n_x}^{O} \odot \mathbf{X}_{n_z}||_F^2}, \tag{4.7}$$

where $\odot$ represents element-wise multiplication. We define the matrices $\mathbf{W}_{n_x}^{F}$ for n_x in the pre-defined grid. Each matrix is of dimension $K \times K$. The matrix $\mathbf{W}_{n_x}^{F}$ contains all zeros, except for the column n_x. In this column, the elements $(n_x - \Delta_f/2, n_x)$, ..., (n_x, n_x), ..., $(n_x + \Delta_f/2, n_x)$ are set to 1, while the rest are 0. Here, Δ_f represents the focal spot width in grid points.

Similarly, we define the matrices $\mathbf{W}_{n_x}^{O}$, also with dimension $K \times K$ and n_x varying in the pre-defined grid. The matrix $\mathbf{W}_{n_x}^{O}$ contains all zeros, expect for the column n_x. In this column, the elements in the focal spot are 0, while the other elements are 1.

The cost function f can be finally written as:

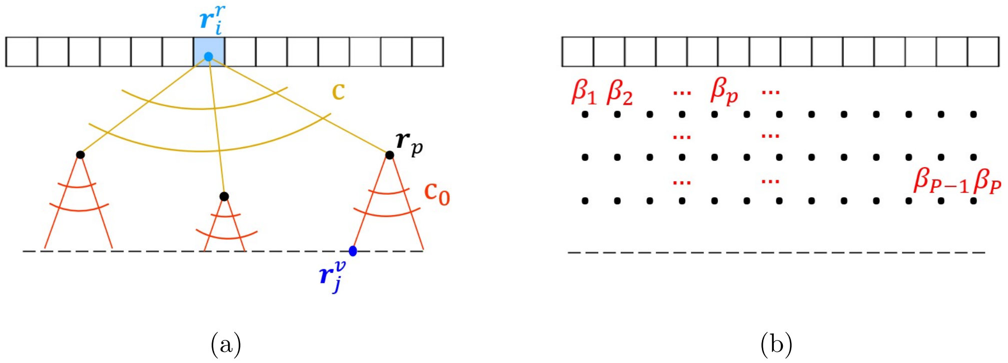
$$\begin{aligned} f &= \sum_{n_z}\sum_{n_x} \frac{||\mathbf{W}_{n_x}^{F} \odot \mathbf{X}_{n_z}||_F^2}{||\mathbf{W}_{n_x}^{O} \odot \mathbf{X}_{n_z}||_F^2} \\ \mathbf{X}_{n_z} &= \sum_l \tilde{\mathbf{G}}_{n_z}^{\mathrm{H}}(\omega_l)\mathbf{H}^{\mathrm{H}}(\omega_l)\mathbf{Y}(\omega_l)\mathbf{H}^{*}(\omega_l)\tilde{\mathbf{G}}_{n_z}^{*}(\omega_l) \end{aligned} \tag{4.8}$$

Even though we have provided an intuition for why a good estimate results in a high cost and a bad estimate yields a low cost, we cannot definitively demonstrate that this cost is maximized by the true $\mathbf{H}(\omega)$. It is essential to acknowledge that other matrices $\mathbf{H}(\omega)$ may also lead to higher costs, particularly if we do not impose any specific structure and allow $\mathbf{H}(\omega)$ to be arbitrary.

To mitigate this issue, we introduce a model for $\mathbf{H}(\omega)$ and optimize for its parameters. By constraining the model, we can better guide the optimization process and increase the likelihood of obtaining improved focus and better images by maximizing this cost function.

4.2 Modeling the transfer matrix

We model wave propagation through the layer as a set of scattering events. The (i, j) entry of the matrix $\mathbf{H}(\omega)$ measures the field at the virtual element j when the real element i transmits a pulse $S(\omega)$. We assume $H_{i,j}(\omega)$ can be approximated as a superposition of waves scattered from multiple locations $\mathbf{r}_p$, as illustrated in Figure 4.4a. We consider a speed of sound c in the path from the transducer to the scatterer and a different speed of sound c_0 in the path from the scatterer to the virtual transducer. We assume a constant attenuation α for both paths.



(a)

(b)

Figure 4.4: Each $H_{i,j}(\omega)$ is approximated as a weighted sum of waves scattered from multiple locations $\mathbf{r}_p$ (a). The scattering from each location is weighted by a coefficient β_p. To estimate $\mathbf{H}(\omega)$, a grid of possible scattering locations is considered (b), and the task is reduced to estimating the coefficients $[\beta_1, \beta_2, ..., \beta_P]$.

First, we derive the contribution of a single scatterer at $\mathbf{r}_p$ to the pressure $H_{i,j}(\omega)$. We define $D_{i,j,p}(\omega)$ as the field measured at the virtual transducer j when a pulse $S(\omega)$ is sent by the real element i and then is scattered at location $\mathbf{r}_p$. We define $\mathbf{r}_i^{\mathrm{r}}$ as the location of the real element i and $\mathbf{r}_j^{\mathrm{v}}$ as the location of virtual element j. Then, $D_{i,j,p}(\omega)$ is equal to:

$$D_{i,j,p}(\omega) = \exp\Big(- \frac{\Delta(\mathbf{r}_p, \mathbf{r}_i^{\mathrm{r}}) + \Delta(\mathbf{r}_p, \mathbf{r}_j^{\mathrm{v}})}{\alpha} - j\omega\Big(\frac{\Delta(\mathbf{r}_p, \mathbf{r}_i^{\mathrm{r}})}{c} + \frac{\Delta(\mathbf{r}_p, \mathbf{r}_j^{\mathrm{v}})}{c_0}\Big)\Big) S(\omega) \qquad (4.9)$$

where we have defined $\Delta(\mathbf{r}_1, \mathbf{r}_2) = ||\mathbf{r}_1 - \mathbf{r}_2||_2$.

$D_{i,j,p}(\omega)$ is the field for a single scatterer at $\mathbf{r}_p$. We assume that scattering occurs at multiple locations simultaneously. However, we do not have prior knowledge of which specific locations to consider as active scattering points. To address this, a grid of P points is defined. An example of such a grid is shown in Figure 4.4b. After assigning a value for c, c_0 and α, the $D_{i,j,p}(\omega)$ functions can be computed for each grid location. Then, the transfer functions $H_{i,j}(\omega)$ can be decomposed as a weighted sum of the waves scattered at all points:

$$H_{i,j}(\omega) = \sum_p \beta_p D_{i,j,p}(\omega). \qquad (4.10)$$

The coefficient β_p is zero if the corresponding location $\mathbf{r}_p$ does not scatter. At the location where it is non-zero, we can have different amplitudes, as the intensity of the scattered waves can vary depending on the scatterer's location. In addition, the coefficient β_p is considered complex, which allows for variations in the phase of the waves.

We will define a grid a priori. Then, the estimation problem is reduced to estimating the $[\beta_1, .., \beta_P]$ coefficients.

We define $\mathbf{D}_p(\omega)$ as a matrix whose (i, j) entry is $D_{i,j,p}(\omega)$. We also discretize the ω domain in L frequencies ω_1,..., ω_L. Then, we have a set of matrices $\mathbf{D}_p(\omega_1)$,..,$\mathbf{D}_p(\omega_L)$ for each location in the grid. These can be arranged in a single vector:

$$\mathbf{d}_p = \begin{bmatrix} \mathrm{vec}(\mathbf{D}_p(\omega_1)) \\ \vdots \\ \mathrm{vec}(\mathbf{D}_p(\omega_L)) \end{bmatrix}. \tag{4.11}$$

Then, we reconstruct the transfer functions as:

$$\mathbf{h} = \mathbf{D}\mathbf{b}, \tag{4.12}$$

where we have defined $\mathbf{b} = [\beta_1, ..., \beta_P]^T$, $\mathbf{D} = [\mathbf{d}_1 ... \mathbf{d}_P]$ and

$$\mathbf{h} = \begin{bmatrix} \mathrm{vec}(\mathbf{H}(\omega_1)) \\ \vdots \\ \mathrm{vec}(\mathbf{H}(\omega_L)) \end{bmatrix}. \tag{4.13}$$

The assumption is that the vector $\mathbf{h}$ can be reconstructed by using P vectors $\{\mathbf{d}_1, .., \mathbf{d}_P\}$. We use the term "dictionary" to refer to this set.

Before proceeding with the optimization, the dictionary needs to be computed. This requires defining the grid dimension and spacing, the speed of sound from transducer to scatterer c, speed of sound from scatterer to virtual transducer c_0 and the attenuation α. The next section will provide further details on how these parameters are defined in the results' derivation.

In general, the grid should cover the transducer aperture in the x-dimension. For the z-dimension, the minimum and maximum depths should be chosen based on the expected layer thickness. It is important to keep a larger range to compensate for errors in the speed of sound. If the c used in the $\mathbf{D}_p(\omega)$ computation is higher than the actual speed of sound, we may need to place the scatterers closer to the transducers to compensate for the speed of sound error and achieve an accurate time of arrival.

Regarding the grid spacing, it is necessary to ensure that it is small enough to sufficiently sample the layer area. However, excessive sampling with respect to the wavelength does not provide additional information, as the $\mathbf{d}_p$ vectors become more linearly dependent.

4.3 Methods

In Section 4.1, a cost function for the transfer function estimation has been derived. In Section 4.2, a decomposition for the transfer matrix $\mathbf{H}(\omega)$ has been introduced. It is possible to replace the dictionary-based representation of $\mathbf{H}(\omega)$ in the cost function, which then becomes a function of the dictionary coefficients $[\beta_1, .., \beta_P]$. The overall cost function is:

$$\begin{aligned} f &= \sum_{n_z}\sum_{n_x} \frac{||\mathbf{W}^F_{n_x} \odot \mathbf{X}_{n_z}||_F^2}{||\mathbf{W}^O_{n_x} \odot \mathbf{X}_{n_z}||_F^2} \\ \mathbf{X}_{n_z} &= \sum_l \tilde{\mathbf{G}}^{\mathrm{H}}_{n_z}(\omega_l)\mathbf{H}^{\mathrm{H}}(\omega_l)\mathbf{Y}(\omega_l)\mathbf{H}^*(\omega_l)\tilde{\mathbf{G}}^*_{n_z}(\omega_l) \\ \mathbf{H}(\omega_l) &= \sum_p \beta_p \mathbf{D}_p(\omega_l) \;\; l = 1, .., L. \end{aligned} \tag{4.14}$$

The optimization will be performed on simulated data. In Section 4.3.1, the set-up for the simulations is explained.

After this, we explain the methodology used for the cost function optimization. It is important to consider that the evaluation of this function involves several parameters that need to be defined before the optimization process. Even though these parameters have been mentioned in previous sections, we provide an overview of all the matrices involved in the cost computation and we explicitly state which parameters are required:

- The matrices $\mathbf{D}_p(\omega_l)$ for varying l and varying p collectively form the dictionary. The dictionary computation requires to define a pulse shape, a speed of sound for the layer c, a speed of sound for the background c_0, an attenuation α and a grid for the scatterer's positions. The choices for these parameters are explained in Section 4.3.2.

- The matrices $\mathbf{Y}(\omega_l)$ for varying l are computed from a SA scan at the real array. This data is given.

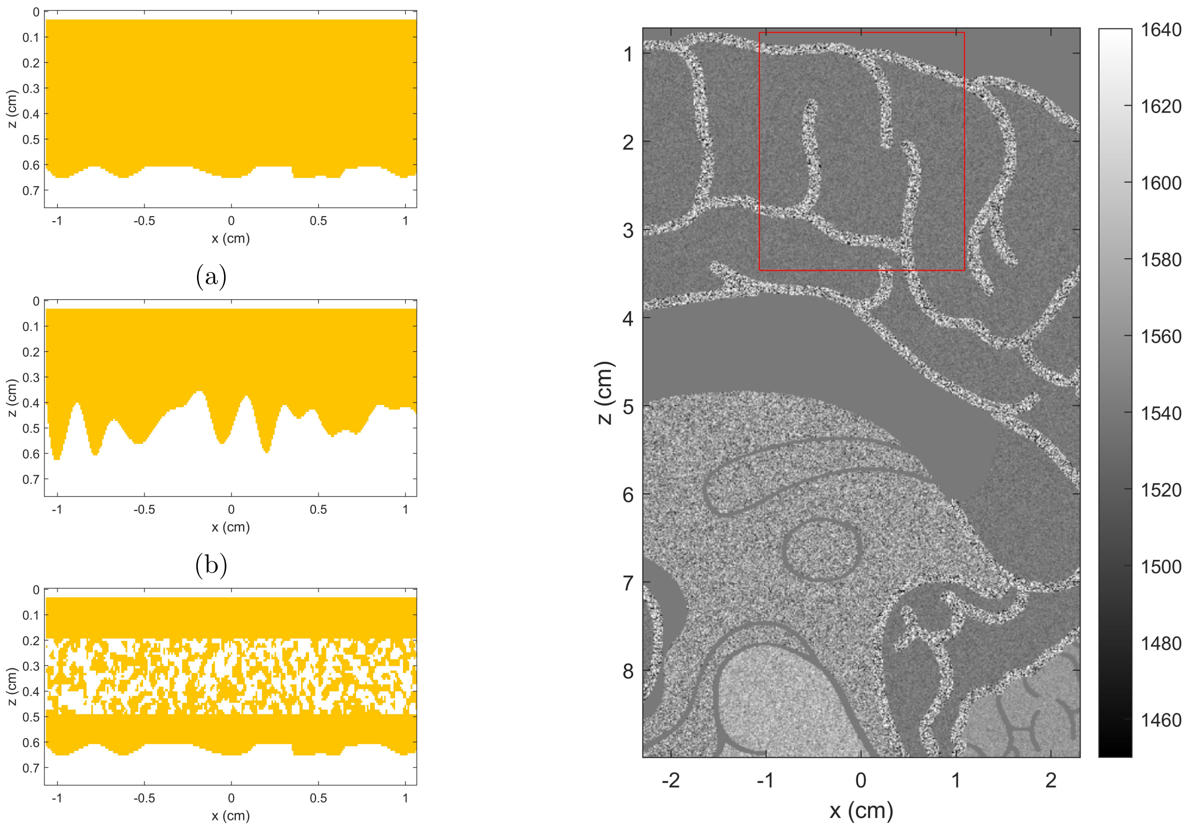
- Each $\tilde{\mathbf{G}}_{n_z}(\omega_l)$ is a $V \times K$ matrix with V number of virtual transducers, K number of pixels in the x dimension. It contains the Green's functions from the virtual array to the line at depth n_z. To compute $\tilde{\mathbf{G}}_{n_z}(\omega_l)$ at all frequencies, we only need to compute the geometry-based delays from the virtual array to the points on this line. We compute one set of L matrices for each depth n_z included in the summation. To accelerate the cost computation, we limit the summation to a selected subset of depths within the imaging domain. In Section 4.3.3 we explain how these depths are selected.

- The matrices $\mathbf{W}^F_{n_x}$, $\mathbf{W}^O_{n_x}$ for varying n_x have been defined in Section 4.1. They are determined once a focal spot width Δ_f is fixed. This value is defined in Section 4.3.3.

Once all the parameters are defined, we can proceed with the optimization. We start with an initial guess $[\beta_1^0, .., \beta_P^0]$. Then, we iteratively update each β_p using Gradient Ascent:

$$\beta_p^{(i+1)} = \beta_p^{(i)} + \mu \frac{\partial f}{\partial \beta_p^*}(\beta_p^{(i)}) \tag{4.15}$$

In Section 4.3.4, we elaborate on how to determine the initial guess, how to compute the gradients, and how to assess if the updates lead to improvements with respect to the initial guess.



(a)

(b)

(c)

(d)

Figure 4.5: Three different layers are considered: two homogeneous layers with a range of variations in thickness of 0.5λ (a) and 3λ (b) respectively, and a third layer with the same profile as the first one but with a porous structure inside (c). The speed-of-sound map of the imaging phantom resembles the brain structure (d). The red rectangle highlights the imaging domain.

4.3.1 K-Wave simulations

The simulation grid and the array characteristics are the same as those described in Section 3.2. However, the layer shape and the imaging phantom are changed.

Three distinct layers are considered. The speed-of-sound map of each layer is displayed in Figures 4.5b, 4.5b and 4.5c respectively. We assign a speed of sound $c_0 = 1540$ m/s and a density$\rho_0 = 997$ kg/m^3 to the white pixels. At the yellow pixels the speed of sound is $c = 2750$ m/s and the density is $\rho =$940 kg/m^3 to the yellow pixels. As in the previous chapter, we simulate plastic layers.

The profile of the first two layers is generated by randomly selecting a set of x locations and assigning a random thickness to each location. The thickness values are drawn from a specified range. The thickness values at all x locations are then determined through interpolation. For the first layer, the difference between the minimum and maximum thickness is set to 0.5λ, with λ equal to the wavelength in plastic. In the second layer, this interval is increased to 3λ. The third layer has the same profile as the first one. However, it is heterogeneous inside. The heterogeneous part resembles the porous structure of the skull.

First, the transfer functions are simulated, as they will serve as ground truth for the estimation. The speed of sound and density are set to c_0 and ρ_0 in the rest of the k-Wave grid. Then, a Synthetic Aperture scan is simulated. The matrices $\mathbf{H}(\omega_1), .., \mathbf{H}(\omega_L)$ are computed as explained in Section 3.2. This is repeated for each of the three layers.

Secondly, the speed-of-sound and density maps are modified to simulate the presence of an imaging target. The speed-of-sound map used for k-Wave simulations is displayed in Figure 4.5d. The density for each pixel is calculated by dividing the speed of sound by a factor of 1.5. The speed-of-sound map is generated based on a brain illustration. It should be noted that the phantom does not accurately model brain structures as we know them from in-vivo measurements. However, the use of sub-resolution scatterers with relatively low contrast does provide a realistic and difficult imaging scenario unlike single bright scatterers often used in this aberration correction work.

The red rectangle in Figure 4.5d highlights the area right below the layer, which is the part that we will image. This area has dimension (2.70 × 2.16) cm, which corresponds to 300 × 240 pixels.

The simulation is then repeated after inserting the phantom and the matrices $\mathbf{Y}(\omega_1), .., \mathbf{Y}(\omega_L)$ are computed. As in the previous chapter, the reflections from the layer are removed. Moreover, in the skull simulation, the brain phantom is shifted 1 mm below. The reasons for shifting the phantom and the implications of removing the reflections are discussed in Section 4.5.

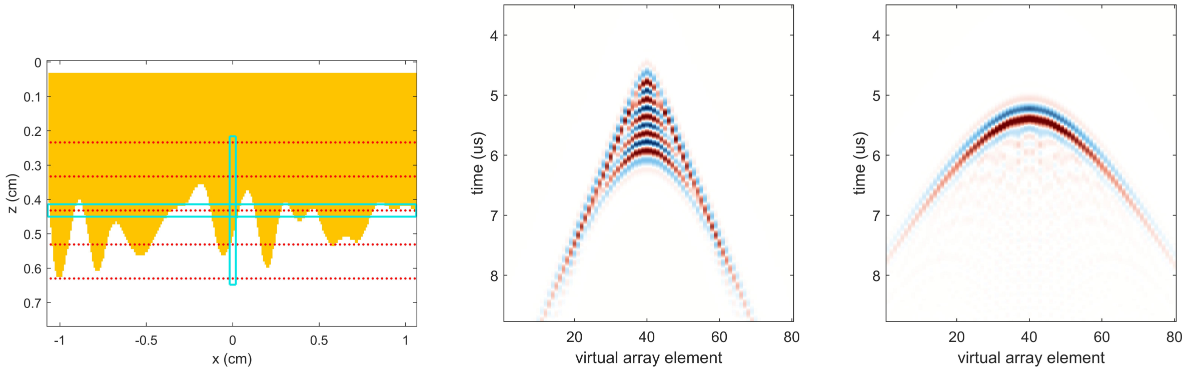
4.3.2 Dictionary definition

We assume the speed of sound of the layer is known, then we set c to 2750 m/s. The speed of sound of the background is $c_0 = 1540$ m/s. The attenuation value has been arbitrarily set to $\alpha = 500m^{-1}$.

In the x dimension, the grid covers the entire transducer aperture. The spacing is set to $\lambda_0/2$, with λ_0 equal to the wavelength in the background. This results in 80 points in the x-dimension, which is the same as the number of transducer elements. In the z dimension, we select 5 depths for the dictionary, ranging from 2.3 to 6.3 mm with 1 mm spacing. Figure 4.6a displays the 5×80 grid overlaid on the shape of the second layer.

Each transfer matrix $\mathbf{H}(\omega)$ and each dictionary entry $\mathbf{D}_p(\omega)$ consist of $M \times V$ functions of ω. In the remainder of this thesis, we will visualize these functions as follows. We will only display a subset of these functions, specifically the ones corresponding to the $i = 40$ row of the matrix. These functions represent the measurements at the virtual array when element $i = 40$ of the real array sends a pulse. Moreover, we will always display the time domain of these functions.

Figure 4.6b displays the dictionary entries corresponding to the vertical line highlighted in Figure 4.6a. This plot shows the sum of the five dictionary entries. It can be observed that the time supports of the pulses do not overlap due to the chosen spacing between the lines. Additionally, Figure 4.6c presents another significant plot, which shows the sum of the dictionary entries corresponding to the points on the horizontal line highlighted in Figure 4.6a. The plot reveals that the sum forms a parabola.



(a) (b) (c)

Figure 4.6: In (a), the 5 × 80 grid for the dictionary computation is displayed. (b) shows the sum of the dictionary entries for $x = 40$ and varying z. (c) displays the sum of the dictionary entries for $z = 3$ and varying x. Only the waves corresponding to a transmission from element 40 of the real array are shown.

This observation implies that if we consider $\beta_p = 1$ over a single horizontal line and $\beta_p = 0$ elsewhere, we are modeling $\mathbf{H}(\omega)$ as the transfer matrix for propagation in a homogeneous medium. However, the speed of sound in this medium is different from c_0. Instead, it is equal to an average speed of sound between the background and the layer. This average speed depends on the location of the line.

For each layer, the ground truth $\mathbf{H}(\omega)$ is available. Therefore, before proceeding with the optimization, we will verify if this dictionary can provide a good approximation of the true transfer matrix. To achieve this, we compute the least-squares fit:

$$\mathbf{b}_{\mathrm{LS}} = \arg\min_{\mathbf{b}} ||\mathbf{h} - \mathbf{D}\mathbf{b}||_2^2 = \mathbf{D}^{\dagger}\mathbf{h}, \tag{4.16}$$

with † indicating pseudo-inversion. Then, we compute $\mathbf{h}_{\mathrm{LS}} = \mathbf{D}\mathbf{b}_{\mathrm{LS}}$ and we compare it to $\mathbf{h}$.

4.3.3 Optimization grid and focal spot width

Two more aspects need to be discussed before proceeding with the optimization. Firstly, we need to determine the specific values of n_x and n_z that will be included in the summation in (4.14). Secondly, we need to define an appropriate focal spot width Δ_f for the weighting matrices.

The focal spot width is set to 3 pixels, which corresponds to half wavelength. The n_x and n_z in the summation cover the entire image domain, with steps of 3 pixels in the x-dimension, and 10 pixels in the z-dimension. This configuration results in a 29 × 80 optimization grid, as illustrated in Figure 4.7.

4.3.4 Optimization procedure

We use Gradient Ascent (GA) to maximize the cost function. First, we make an initial guess $\mathbf{b}^{(0)}$. Then, we iteratively update each entry as in (4.15). In principle, an analytic

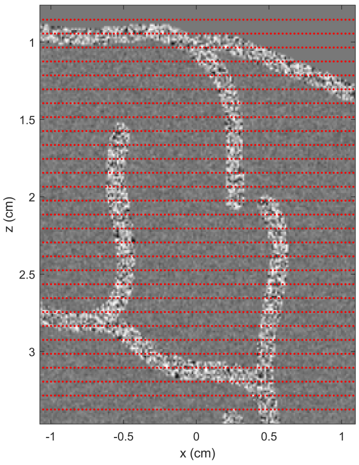


Figure 4.7: Grid considered for the cost computation. The image domain is sampled with steps of 3 pixels ($0.5\lambda_0$) in the x dimension and 10 pixels ($1.7\lambda_0$) in the z dimension, resulting in a 29×80 grid.

expression for the gradients of f with respect to each β_p can be derived using the chain rule. However, in this thesis, we take advantage of the PyTorch framework and we use Automatic Differentation to compute the gradients at each iteration [30].

Since the function f is not convex in β_p, there is no guarantee that the gradient ascent will reach the global maximum. However, our goal is not to find a global or local maximum, but rather to determine whether the transfer matrix estimate can be improved at each update.

In this scenario, the choice of $\mathbf{b}^{(0)}$ is crucial. This vector is selected among 5 possible ones. These 5 are defined in the following way. For each of the 5 depths included in the dictionary, we define a vector $\mathbf{b}^{(0)}$ in which $\beta_p = 1$ for the β_p corresponding to points at that depth, while all other β_p are set to zero. This process yields 5 possible $\mathbf{b}^{(0)}$, each corresponding to one depth. Then, we evaluate the cost at the 5 vectors and we select the one with highest cost. In Section 4.3.2, it has been explained that if we set $\beta_p = 1$ over one line only, we are essentially modeling propagation as if the medium was homogeneous, with a different speed of sound depending on the depth of the line. Therefore, this search for the optimal depth can be seen as analogous to a search for the average speed of sound in the area between the real and the virtual transducer.

Starting from this guess, we apply Gradient Ascent iterations with an appropriately selected learning rate μ. At each iteration, we compute $\hat{\mathbf{h}} = \mathbf{D}\mathbf{b}^{(i)}$. Then, we compute the normalized correlation coefficient between $\mathbf{h}$ and $\hat{\mathbf{h}}$, defined as:

$$\rho = \mathbf{h}^{\mathrm{H}}\hat{\mathbf{h}}/||\mathbf{h}||_2||\hat{\mathbf{h}}||_2 \tag{4.17}$$

Another approach to evaluate the estimate $\hat{\mathbf{h}}$ consists in analyzing the focus quality it provides. We reshape $\hat{\mathbf{h}}$ into L matrices $\hat{\mathbf{H}}(\omega_1)$,..,$\hat{\mathbf{H}}(\omega_L)$. We decide a focusing point n and we compute $\hat{\mathbf{g}}_n^*(\omega_l) = \hat{\mathbf{H}}(\omega_l)\tilde{\mathbf{g}}_n^*(\omega_l)$ for $l = 1, .., L$. Then, we transform each $\hat{g}_{n,i}(\omega_1), .., \hat{g}_{n,i}(\omega_L)$ into the time-domain to obtain M waveforms. Next, we run a k-Wave simulation in which transducer i sends the i-th waveform, and we measure the pressure in the medium. By visualizing the maximum pressure per point, we can check whether a beam focused at n is obtained.

As explained in Section 2.1, the pressure resulting from this excitation could be computed as $\mathbf{G}^T(\omega)\hat{\mathbf{g}}_n^*(\omega)$. However, for a large number of pixels, running a k-Wave simulation is faster than computing this product.

4.4 Results

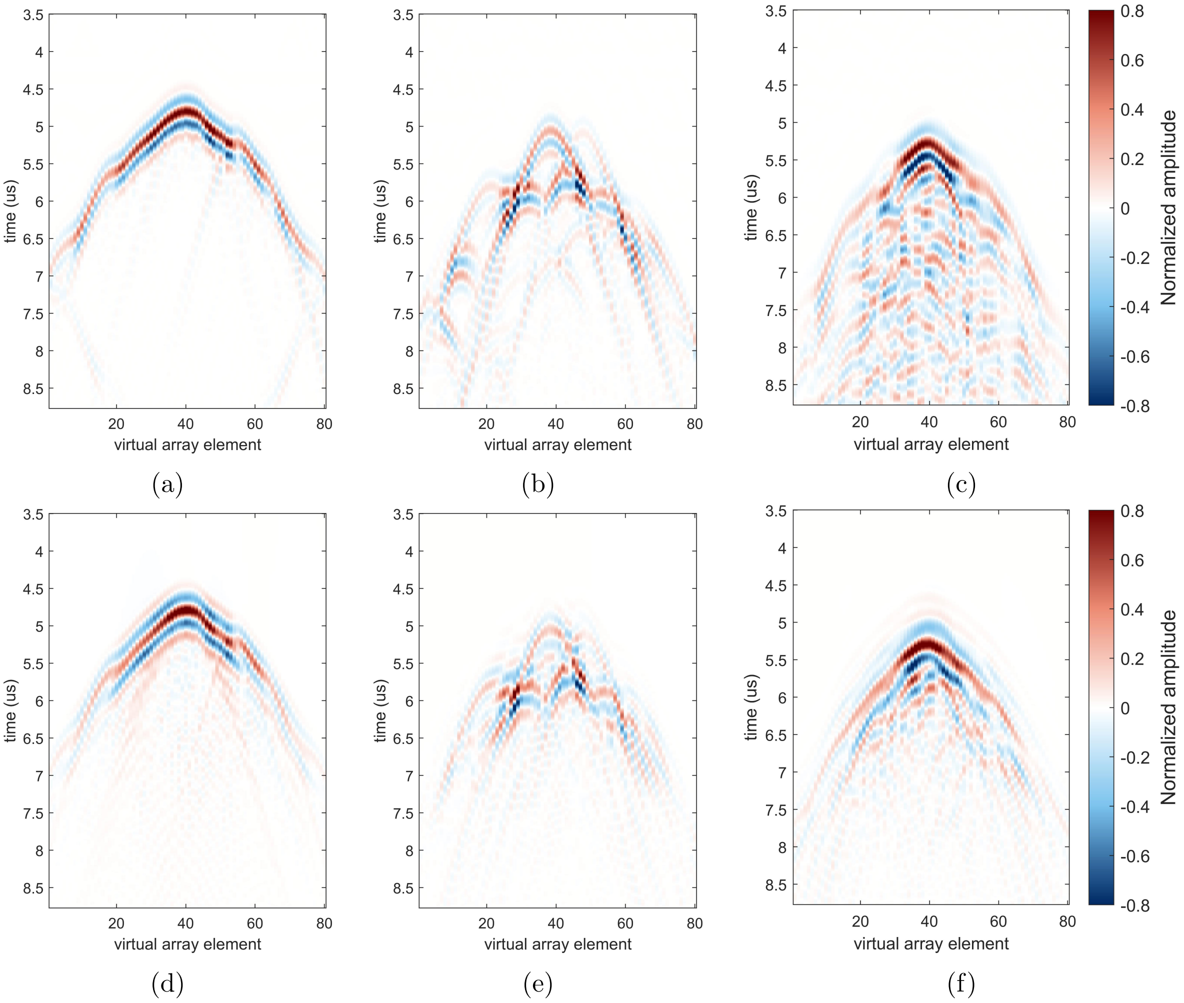
In this Section, the results for the transfer matrix estimation are presented. First, we consider the ground truth transfer matrix and we compute the least-squares dictionary fit to assess whether a good approximation can be achieved with the defined dictionary. Next, a starting point for the optimization is defined and Gradient Ascent is applied. This is repeated for all the three layers described.

4.4.1 Dictionary Least-Squares fit

First, we visualize the transfer functions for the three different layers. We only display the transfer functions related to the transmission from the central element of the array ($i = 40$). The transfer functions for the three layers are shown in the top row of Figure 4.8. The transfer functions for Layer 1 exhibit slight deviations from a parabolic shape. The small variations in the delays can be attributed to the relatively small changes in layer thickness. As the layer profile becomes more varied, the transfer functions experience a wider range of delays, as evident in Figure 4.8b, which displays the transfer functions for Layer 2. In the third plot, the time support of the transfer functions becomes more extended, due to the presence of the internal porous structure which generates multiple late reflections.

Next, the coefficients that best approximate the ground truth in the least-squares sense are computed as $\mathbf{b}_{\text{LS}} = \mathbf{D}^\dagger \mathbf{h}$, with $\mathbf{D}^\dagger$ pseudo-inverse of $\mathbf{D}$. The $\mathbf{h}$ approximation is derived as $\mathbf{h}_{\text{LS}} = \mathbf{D}\mathbf{b}_{\text{LS}}$. We reshape $\mathbf{h}_{LS}$ in L matrices $\mathbf{H}_{\text{LS}}(\omega_1), .., \mathbf{H}_{\text{LS}}(\omega_L)$. Then, each (i, j) function is transformed in time-domain. The functions for $i = 40$ are shown in the bottom row of Figure 4.8. For Layer 1 and 2, a good reconstruction can be obtained. The transfer functions at the borders of the virtual array have lower amplitude, indicating that we are using a too high attenuation value. A worse reconstruction is obtained for Layer 3, because modeling the late reflections is not possible with the current dictionary, but would require considering more lines of scatterers.

The $\mathbf{b}_{\text{LS}}$ coefficients are shown in Figure 4.9. The $\mathbf{b}_{\text{LS}}$ vector is reshaped in a 5×80 matrix to simplify the mapping of each coefficient to its corresponding grid location. The variations in the $\mathbf{b}_{\text{LS}}$ plots resemble the layer profiles. This can be attributed to the fact that the highest scattering occurs at the boundary between the layer and the background medium. In the third plot, the high-intensity β's are at a smaller depth compared to Layer 1, even though the boundary is the same. This occurs because the average speed of sound in Layer 3 is smaller, causing the wave to arrive slightly later than in Layer 1. This time delay is also evident in the plots of $\mathbf{h}$ in Figure 4.8a and 4.8c.

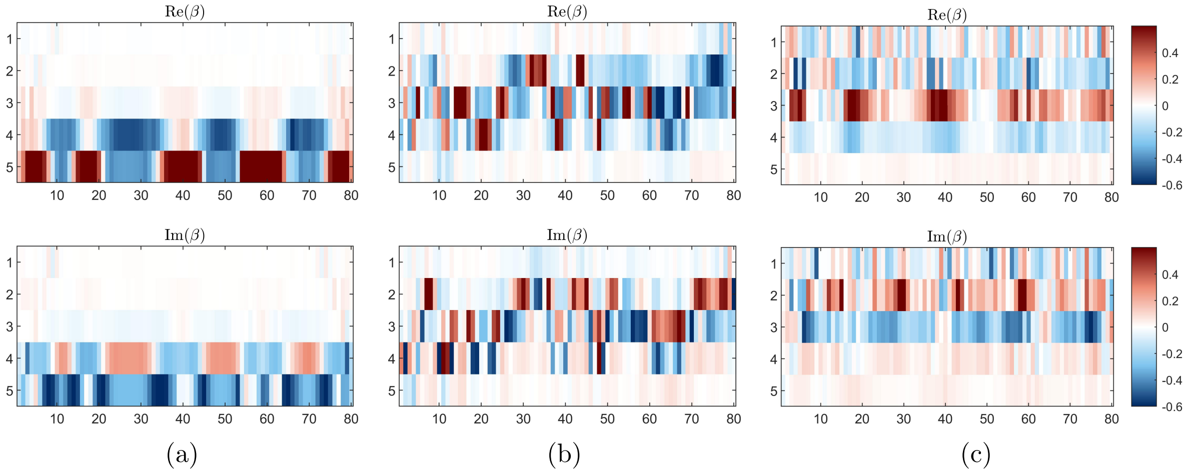


(a) (b) (c)

(d) (e) (f)

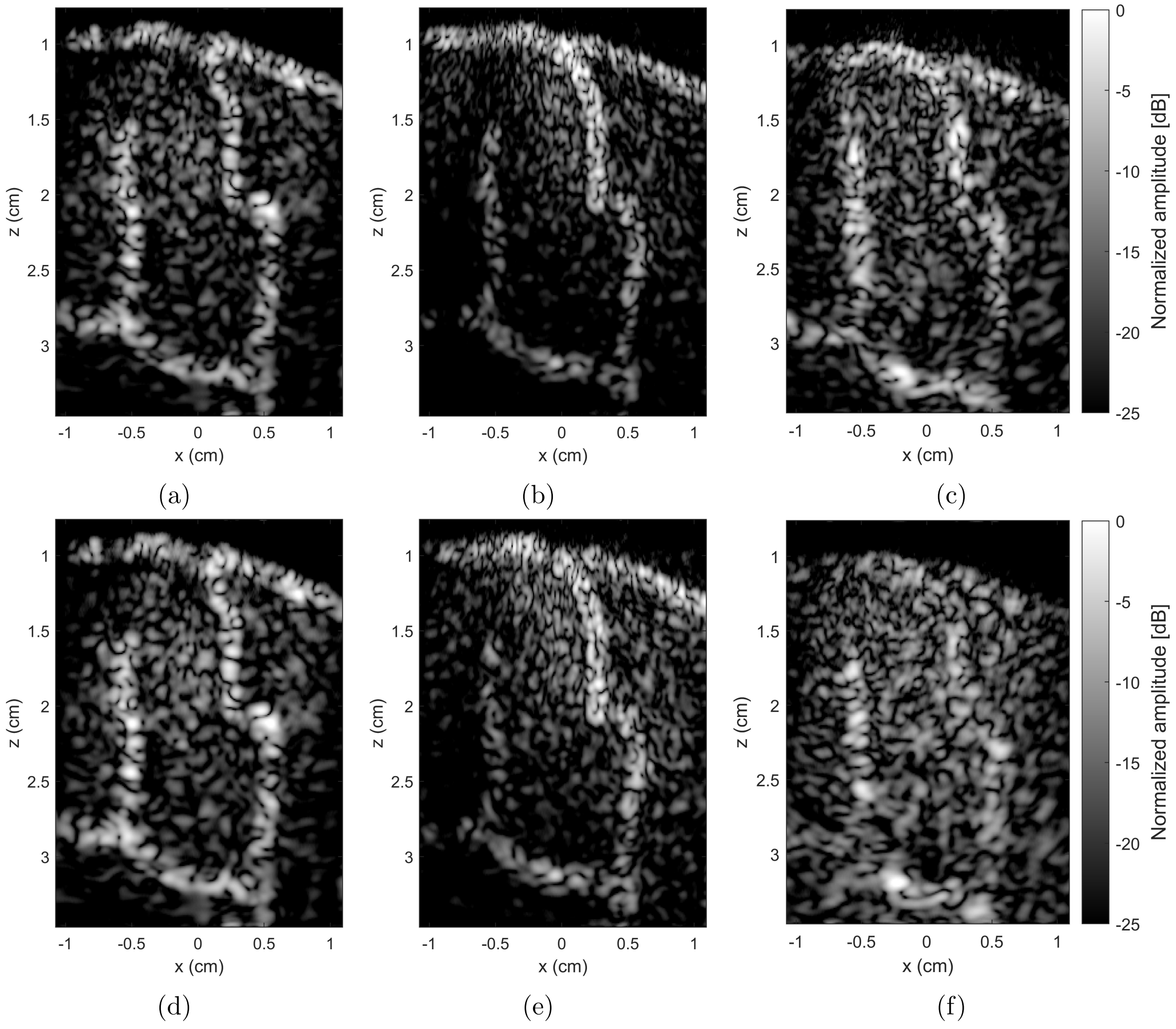
Figure 4.8: Tranfer functions $h_{i,j}(t)$ for transmission from real element $i = 40$. The top row shows the ground truth transfer functions for Layer 1 (a), Layer 2 (b) and Layer 3 (c). The bottom row displays the transfer functions obtained through a least-squares dictionary fit for Layer 1,2,3 respectively.

Finally, we generate images by using both the ground truth $\mathbf{h}$ and the least-squares fit $\mathbf{h}_{\text{LS}}$. The imaging domain corresponds to the area highlighted in red in Figure 4.5d. The image is computed using the method described in Chapter 3. First, we estimate the data for a SA scan at the virtual array as in (3.5). Next, we apply DAS. The resulting images are shown in Figure 4.10. For Layer 1 and 2, the same image is obtained whether we use $\mathbf{h}$ or $\mathbf{h}_{\text{LS}}$. However, for Layer 3, the image quality is degraded when using the least-squares fit. This indicates that including the late reflections is important when attempting to correct for aberrations.



(a) (b) (c)

Figure 4.9: β coefficients obtained from the least-squares dictionary fit. The results for Layer 1 (a), Layer 2 (b), and Layer 3 (c) are shown. Both the real and imaginary parts of the coefficients are displayed.



(a) (b) (c)

(d) (e) (f)

Figure 4.10: Images generated using the ground truth $\mathbf{h}$ (top row) and the least-squares dictionary fit $\mathbf{h}_{\text{LS}}$ (bottom row). The results for Layer 1 (left), Layer 2 (center) and Layer 3 (right) are displayed.

4.4.2 Optimization

In this Section, we apply the proposed method for transfer function estimation to the simulated data, considering each of the three layers separately.

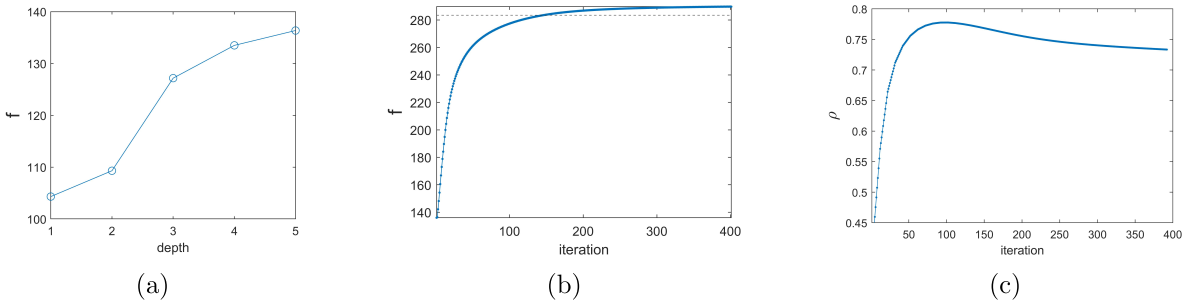
Layer 1

Following the steps mentioned in Section 4.3.4, we define five possible initial vector $\mathbf{b}^0$ vectors (one per dictionary depth). The cost function is evaluated at these five vectors and the resulting values are shown in Figure 4.11a. The best approximation is obtained by setting $\beta_p = 1$ along the 5-th line. This result aligns with the least-squares fit displayed in Figure 4.9a.

Based on this result, we assume that only one more line is needed to achieve a good approximation of $\mathbf{h}$. Consequently, we set the β_p values for depths 1, 2, and 3 to zero and only optimize for the β_p values at depths 4 and 5. This choice simplifies the problem, since it reduces the feasible set dimension.

The step size is set to $\mu = 1 \times 10^{-4}$ and a total of 400 iterations are executed. Figure 4.11b shows how the cost f evolves over the iterations. The dashed line in the plot indicates the cost evaluated at the ground truth $\mathbf{h}$. To evaluate the cost at the ground truth, we just replace the ground truth $\mathbf{h}$ in (4.14), bypassing the use of the dictionary model.

The cost increases over the iterations until it reaches the cost for the ground truth. Interestingly, it surpasses the cost of the ground truth as the iterations continue. Figure 4.11c shows the normalized correlation coefficient ρ between the estimated $\mathbf{h}$ and the ground truth. Initially, the correlation increases, indicating that the solution is getting closer to the true $\mathbf{h}$. However, after 100 iterations, the correlation starts to decrease.



(a) (b) (c)

Figure 4.11: Results for Layer 1. (a) shows the 5 values of the cost f in the search for the best $\mathbf{b}^0$ vector. (b) and (c) show the evolution of the cost function f and the normalized correlation between $\hat{\mathbf{h}}$ and the ground truth $\mathbf{h}$.

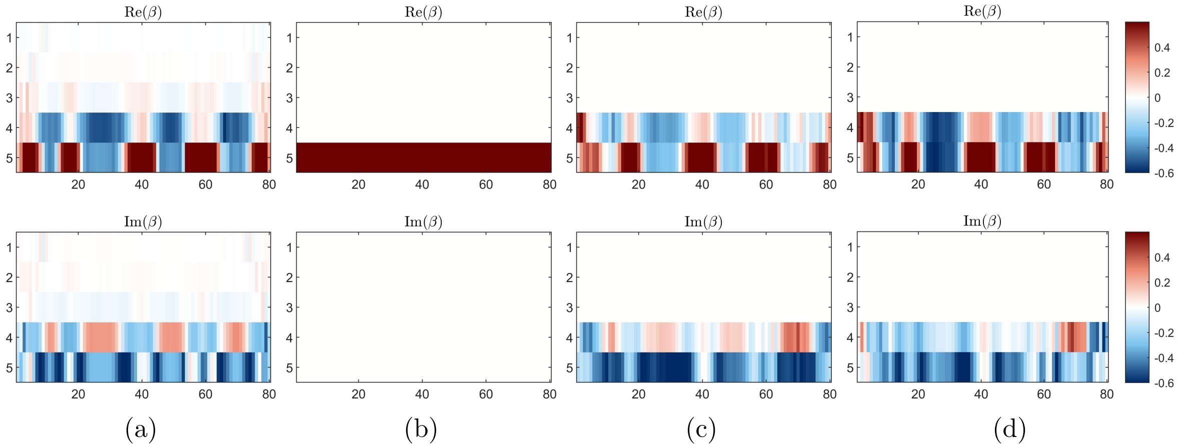


Figure 4.12: **b** coefficients during the optimization for Layer 1. Each vector is reshaped in a 5 × 80 matrix. The least-squares fit of the ground truth is shown in (a). Then, the **b** coefficients for iteration 1, 100 and 400 are shown in (b), (c) and (d) respectively.

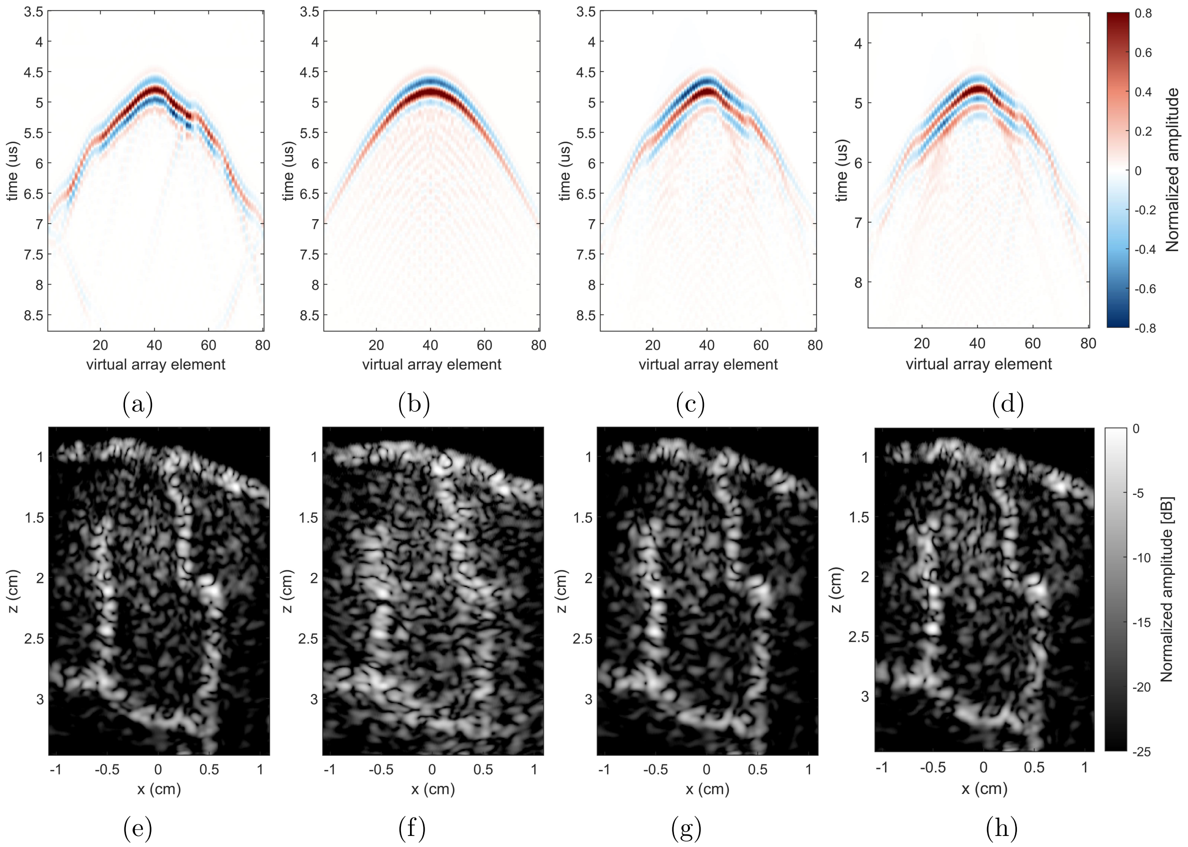
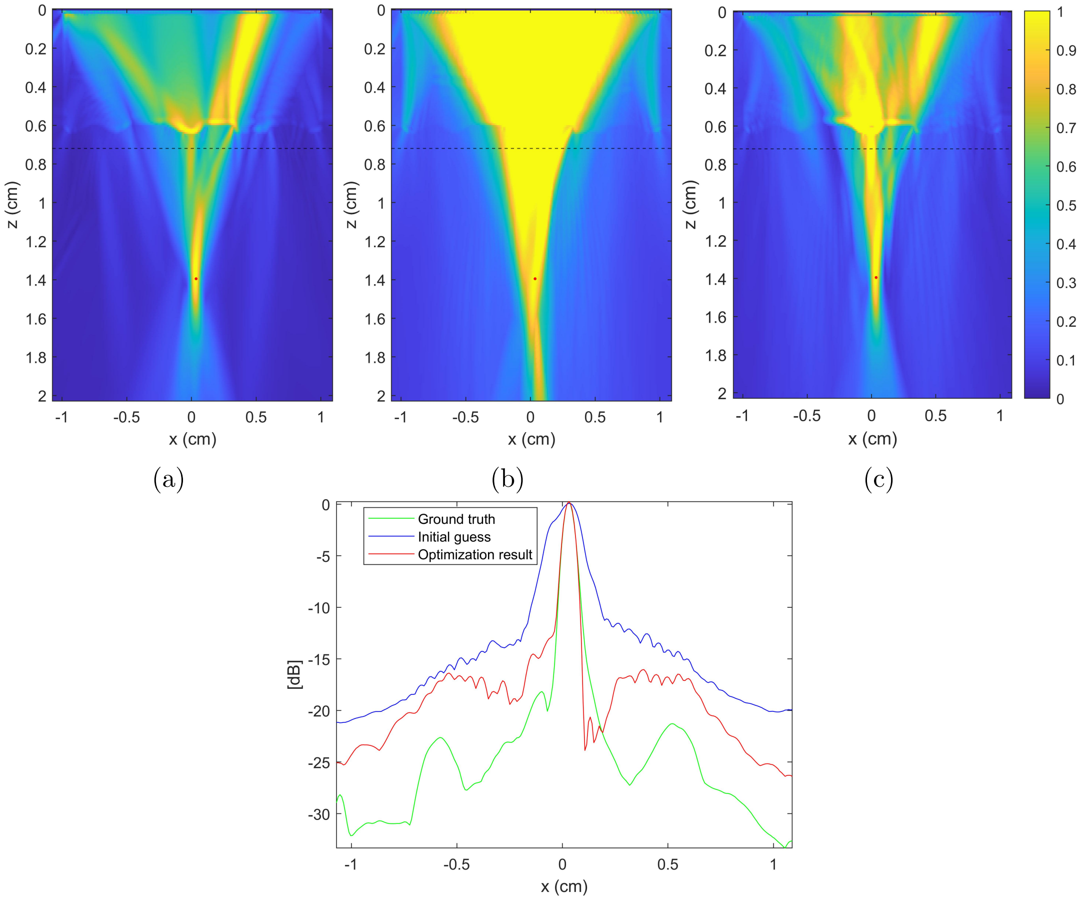


Figure 4.13: Transfer functions evolution during the optimization for Layer 1 and corresponding images: ground truth (a,e), initial guess (b,f), iteration 100 (c,g) and iteration 400 (d,h).

Figure 4.12 illustrates the evolution of the **b** coefficients. The least-squares fit is displayed again in this Figure to facilitate the comparison with the estimates. Figure 4.13 displays the evolution of the transfer functions (top row) and the corresponding beamformed images (bottom row). The true **h** and its corresponding image are included a second time for easier comparison. When using the initial **h**, a decent image is obtained, where the important features can be recognized. However, the PSF appears larger, and the sidelobes are higher when compared to Figure 4.13e. As the optimization progresses, the spatial variations in the β_p coefficients are correctly estimated. We notice improvements in both **h** and the corresponding image. Although the estimated **h** is not exactly equal to the ground truth, and the normalized correlation reaches a maximum of 0.78, the resulting images show no noticeable difference.



(a)

(b)

(c)

(d)

Figure 4.14: Pressure distribution during the transmission of $\hat{\mathbf{g}}_n(\omega)$ waveforms through Layer 1. The point n is highlighted in red on the figures. The maximum pressure per point is extracted and the beam is normalized with respect to the value at the focusing point. The beams for the ground truth (a), the initial guess (b) and the estimate at the last iteration (c) are displayed. In (d), the pressure distribution along a line at the focal point depth is shown.

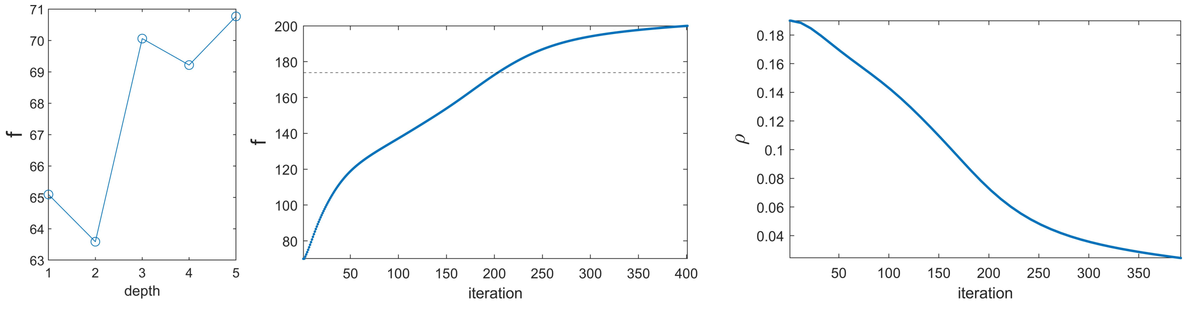
We conclude by analyzing the beams obtained for a focused transmission. A single focusing point is considered, located at position (1.4, 0) cm. The beams are computed as explained in Section 4.3.4. We consider the ground truth $\mathbf{h}$, the initial $\mathbf{h}^0$ and the estimate computed at the iteration number 100. The three beams are shown in Figure 4.14. The intensity has been normalized to the value at the focusing point, which is indicated with a red dot. In Figure 4.14d, the pressure over the line $z = 1.4$ cm is shown. The pressure for the three $\mathbf{h}$ is overlaid for comparison. When using $\mathbf{h}^0$, we obtain a peak at the correct location, but the focal spot is larger, and the sidelobes are higher compared to the beam in Figure 4.14a. This is the reason why we get a bigger PSF in the image. After the optimization, the focus becomes as small as for the true $\mathbf{h}$, but the sidelobes are 10 dB higher.

Layer 2

We begin with the search for the optimal initial vector. In this case, the search for the average depth does not yield the expected result. The cost is maximized at 5. However, we know that 3 is a better choice. To guide the optimization process, we decide to select 3 as the depth for the initial vector.

The step size is again $\mu = 1 \times 10^{-4}$, and the number of iterations is set to 400. The evolution of the cost and the normalized correlation coefficients is shown in Figure 4.15b and 4.15c, respectively. As the cost increases, the normalized correlation coefficient decreases, indicating that the $\mathbf{h}$ estimate is getting further from the ground truth $\mathbf{h}$. This is also evident in the $\mathbf{h}$ plots displayed in the top row of Figure 4.17

Even though the $\mathbf{h}$ estimate is moving further from the true, we can observe that the image becomes sharper after the optimization. The image obtained using the initial $\mathbf{h}$ is completely blurred, but after the optimization, more details become visible. For instance, a vertical bright line, which is present in the true image, becomes visible in the optimized image as well, although there has been a shift.



(a) (b) (c)

Figure 4.15: Results for Layer 2. (a) shows the 5 values of the cost f in the search for the best $\mathbf{b}^0$ vector. (b) and (c) show the evolution of the cost function f and the normalized correlation between $\hat{\mathbf{h}}$ and the ground truth $\mathbf{h}$.

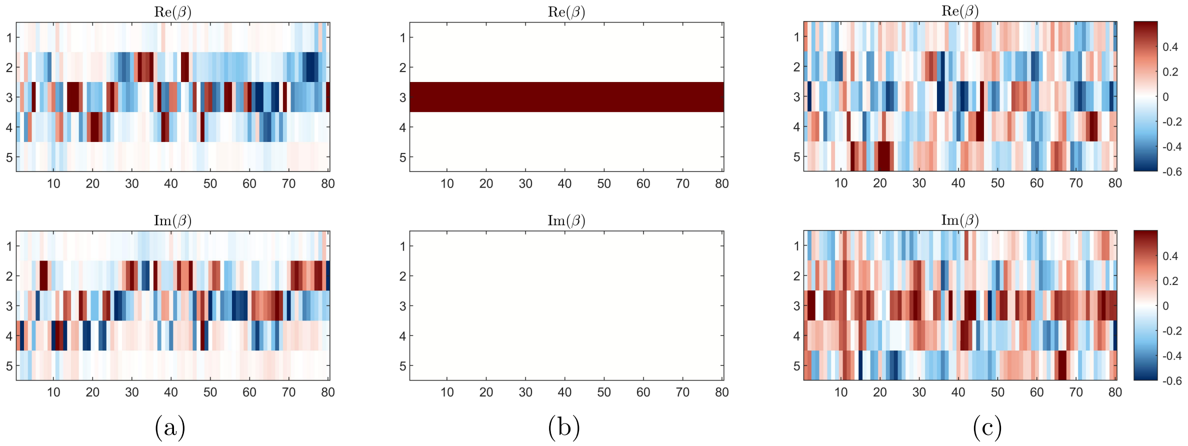


Figure 4.16: **b** coefficients during the optimization for Layer 2. Each vector is reshaped in a 5 × 80 matrix. The least-squares fit of the ground truth is shown in (a). Then, the **b** coefficients for iteration 1 and 400 are shown in (b) and (c) respectively.

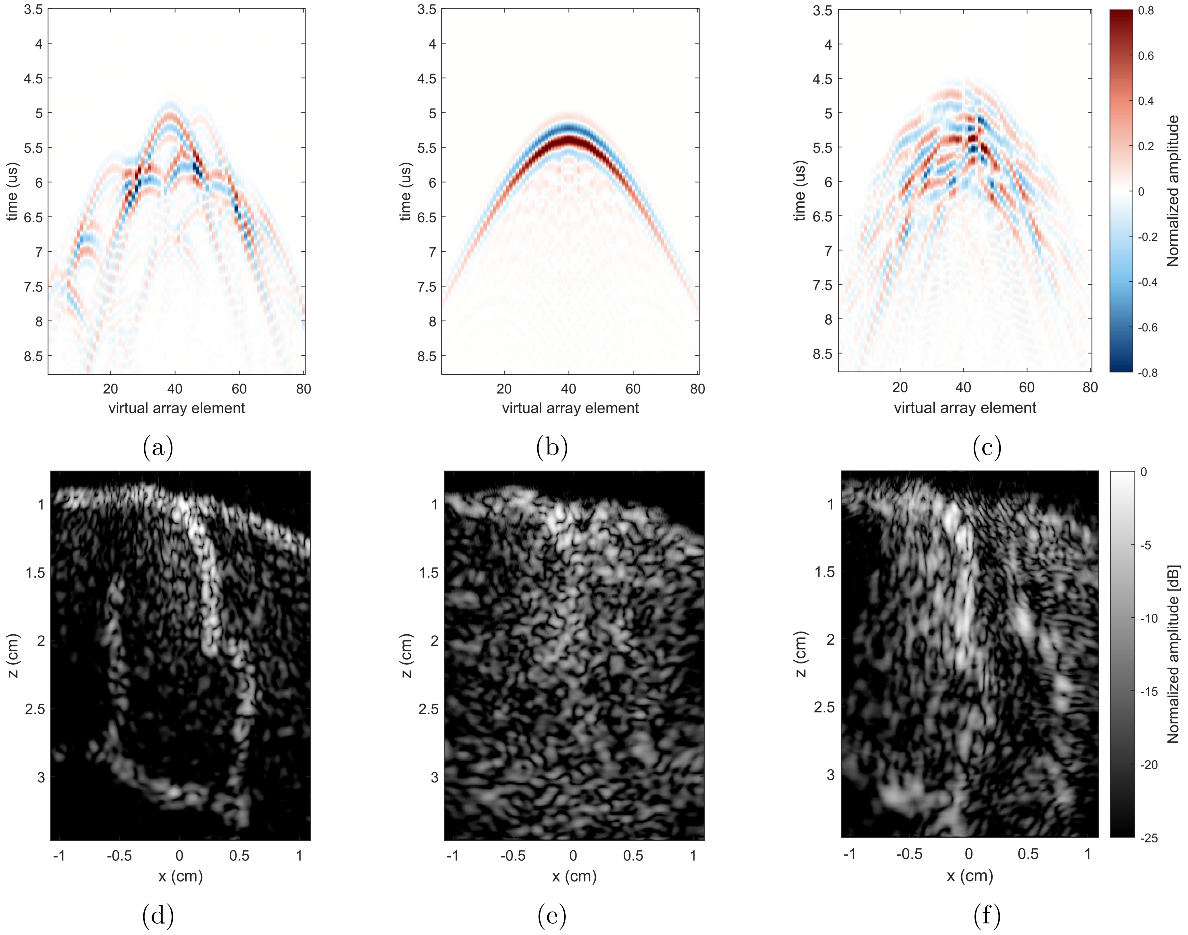


Figure 4.17: Transfer functions evolution during the optimization for Layer 2 and corresponding images: ground truth (a,d), initial guess (b,e) and iteration 400 (c,f).

Looking at the beams in Figure 4.18, we can see that using the initial **h** results in a completely unfocused beam. However, after the optimization, the energy beam appears to be more focused. If we observe the pressure at the line $z = 1.4$ cm (Figure 4.18d), we can see that, after the optimization, the energy is higher at the center and lower at the sides. However, the peak is not at $x = 0$. Instead, there is one peak on the left and two peaks on the right.

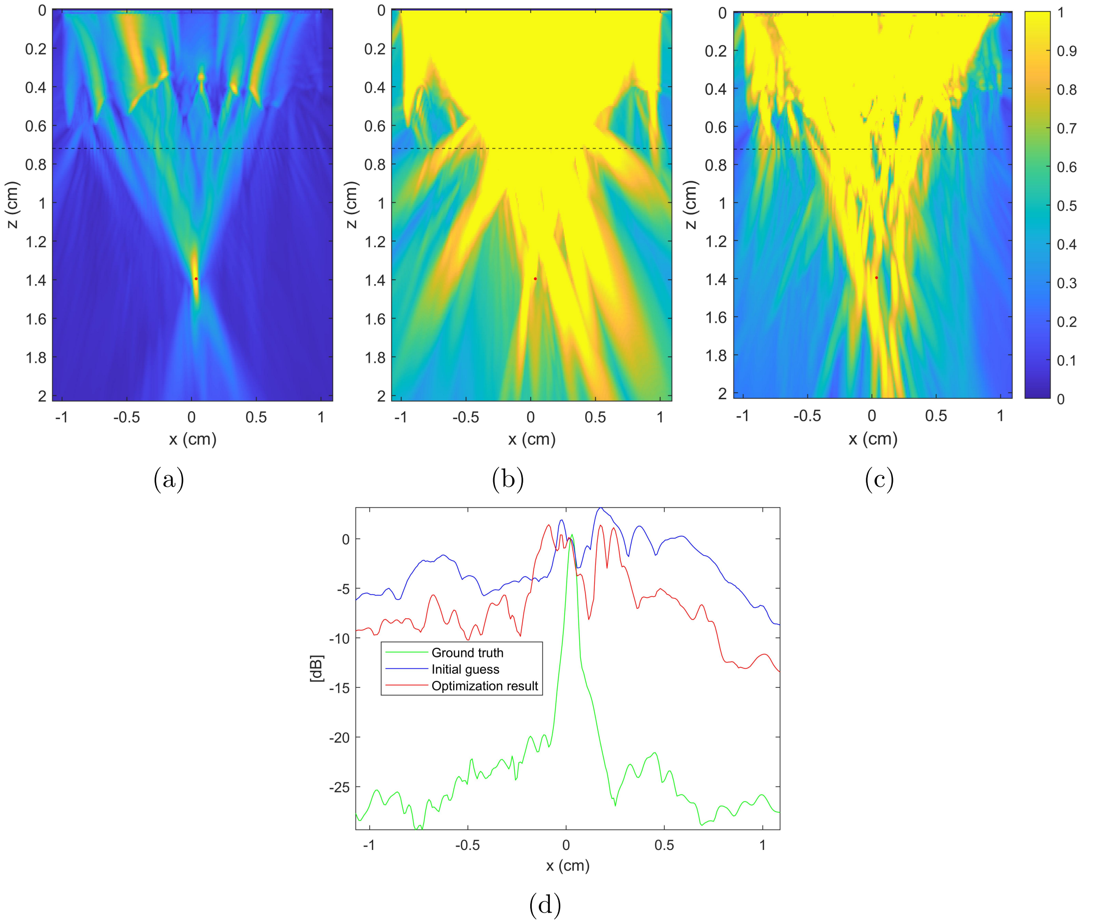
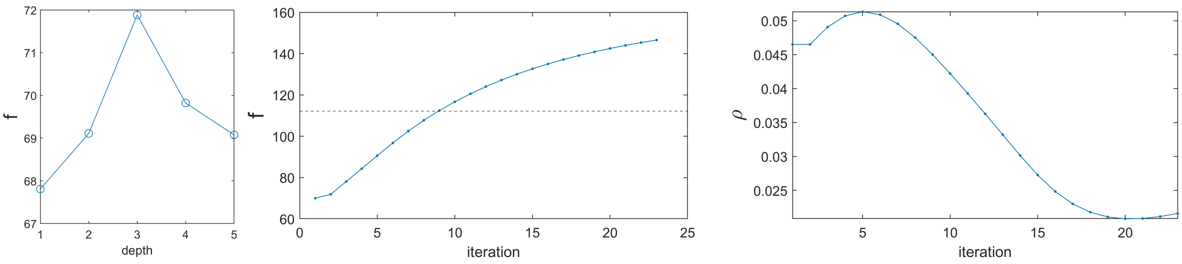


Figure 4.18: Pressure distribution during the transmission of $\hat{\mathbf{g}}_n(\omega)$ waveforms through Layer 2. The point n is highlighted in red on the figures. The maximum pressure per point is extracted and the beam is normalized with respect to the value at the focusing point. The beams for the ground truth (a), the initial guess (b) and the estimate at the last iteration (c) are displayed. In (d), the pressure distribution along a line at the focal point depth is shown.

Layer 3

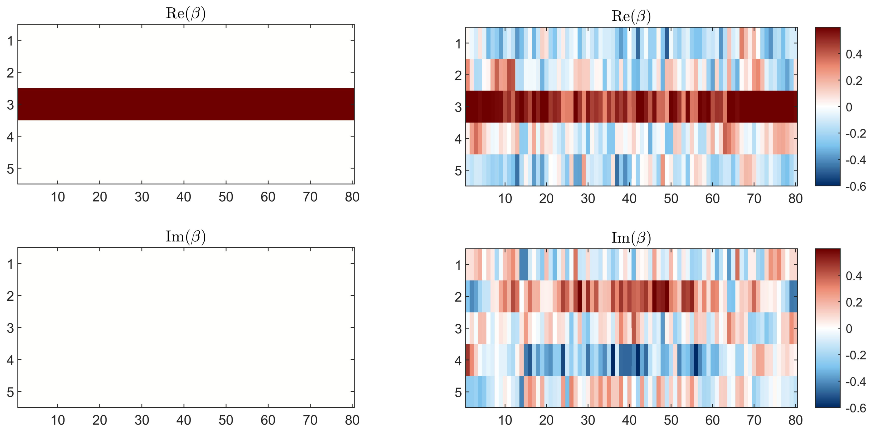
We begin by selecting the optimal initial point. The cost values for the 5 depths are shown in Figure 4.19a. The highest cost is obtained at the 3rd depth. This result is consistent with the $\mathbf{b}_{\mathrm{LS}}$ plot in Figure 4.9c, where the highest β coefficients correspond to depth 3.

The learning rate is now set to $\mu = 0.001$. Since the learning rate is higher than before, we run only 22 iterations. The evolution of the cost and the normalized correlation is shown in Figure 4.19. As before, we observe an increase in the cost and a decrease in the normalized correlation. The top row of Figure 4.21 displays the true $\mathbf{h}$, the initial $\mathbf{h}$, and the estimated $\mathbf{h}$ after 22 iterations. It is evident that the optimization result looks different from the true $\mathbf{h}$. However, when examining the images in the bottom row of Figure 4.21, we notice an improvement. The image is significantly blurred when using $\mathbf{h}^0$ and becomes sharper when using the optimized $\mathbf{h}$. A better focusing can also be observed in the beam plots, shown in Figure 4.22.



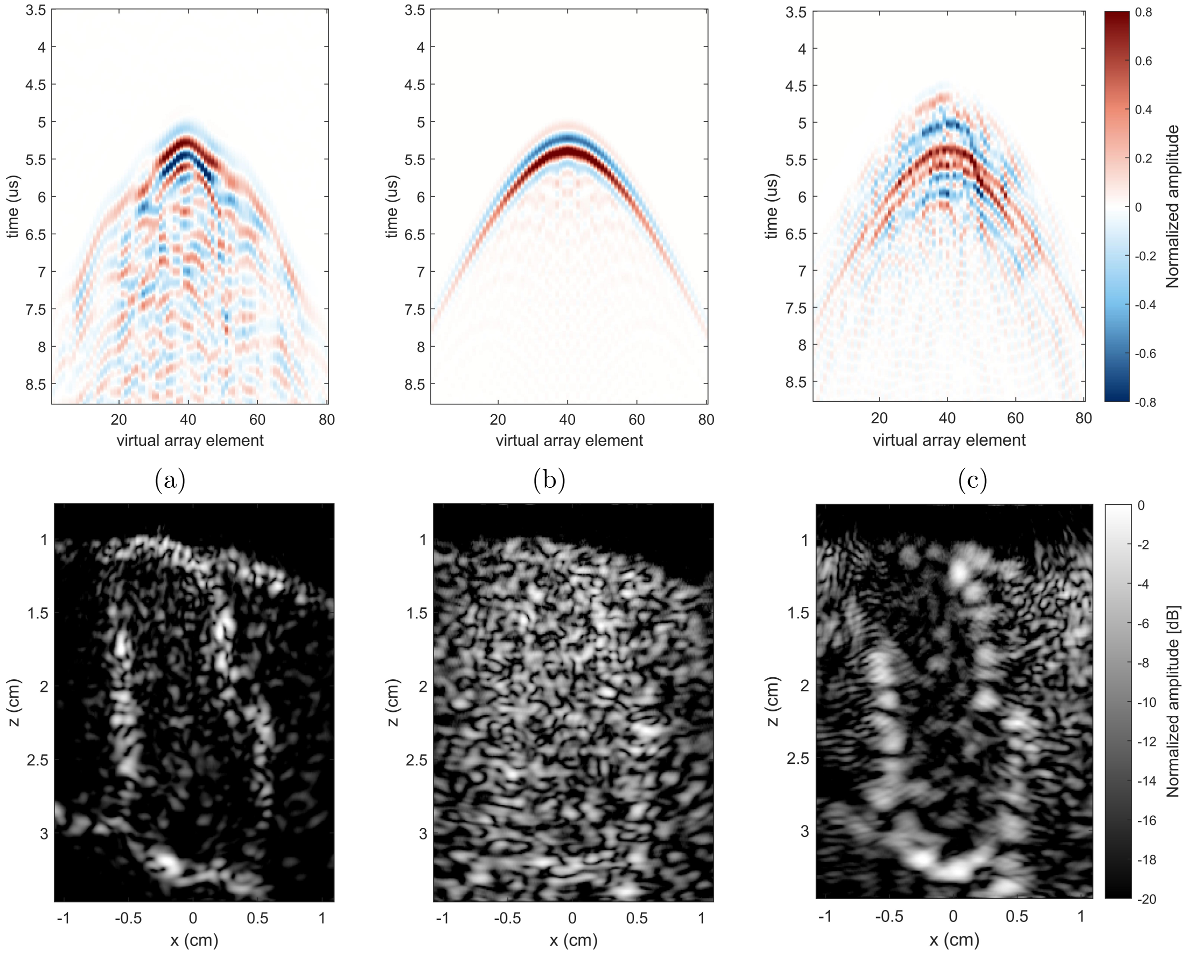
(a) (b) (c)

Figure 4.19: Results for Layer 3. (a) shows the 5 values of the cost f in the search for the best $\mathbf{b}^0$ vector. (b) and (c) show the evolution of the cost function f and the normalized correlation between $\hat{\mathbf{h}}$ and the ground truth $\mathbf{h}$.



(a) (b)

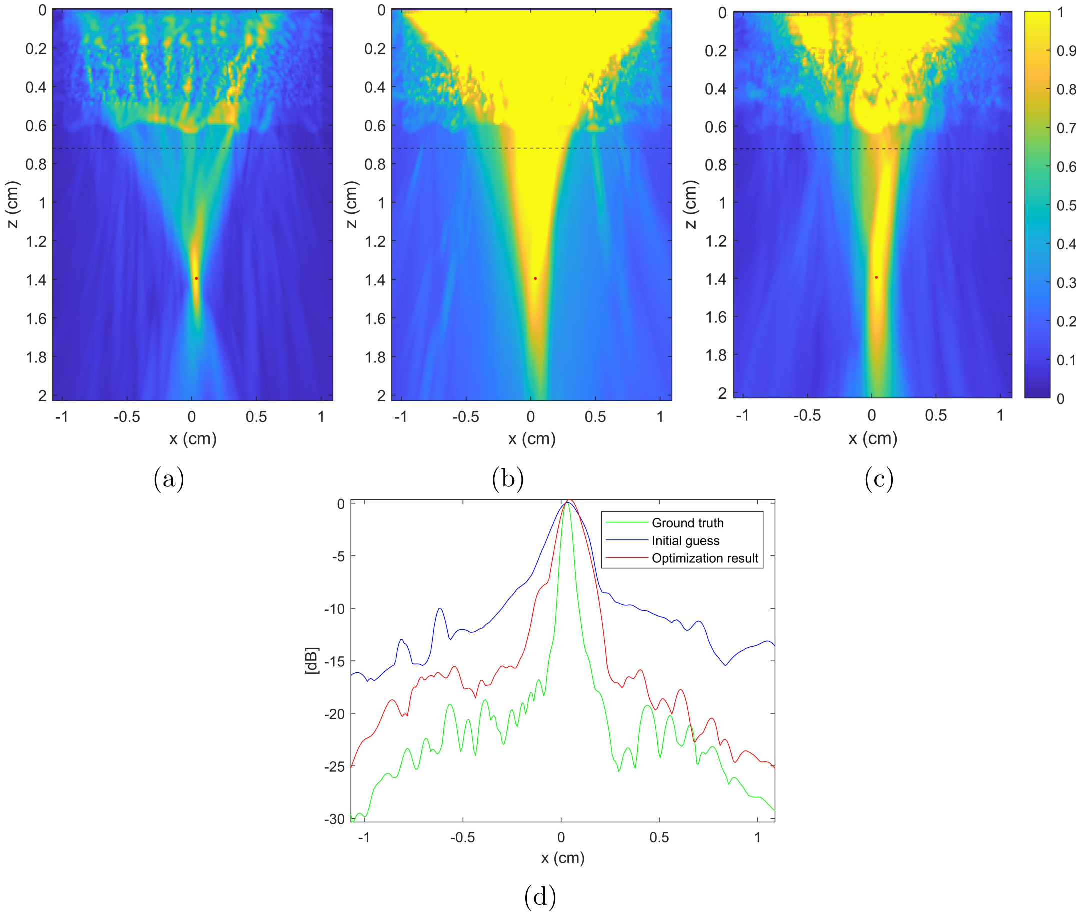
Figure 4.20: $\mathbf{b}$ coefficients during the optimization for Layer 3. Each vector is reshaped in a 5×80 matrix. The $\mathbf{b}$ coefficients for iteration 1 and 22 are shown in (a), (b) respectively.



(a) (b) (c)

(d) (e) (f)

Figure 4.21: Transfer functions evolution during the optimization for Layer 3 and corresponding images: ground truth (a,d), initial guess (b,e) and iteration 22 (c,f).



(a)

(b)

(c)

(d)

Figure 4.22: Pressure distribution during the transmission of $\hat{\mathbf{g}}_n(\omega)$ waveforms through Layer 3. The point n is highlighted in red on the figures. The maximum pressure per point is extracted and the beam is normalized with respect to the value at the focusing point. The beams for the ground truth (a), the initial guess (b) and the estimate at the last iteration (c) are displayed. In (d), the pressure distribution along a line at the focal point depth is shown.

4.5 Discussion

This chapter addresses the challenge of imaging through aberrating layers for which the transfer functions cannot be directly measured and need to be estimated. To address this issue, we have introduced a cost function designed to yield higher values when good focus can be achieved. Additionally, we have proposed a parametrization for the transfer functions, with the purpose of reducing the dimension of the solution space.

To evaluate the effectiveness of the parametrization, we compute the least-squares dictionary fit for each layer. Three different layers are considered in this study: two homogeneous layers and one heterogeneous layer. The results show that for Layer 1 and Layer 2, the dictionary fit is good, indicating that the model accurately represents the transfer functions for these layers. Moreover, errors in the attenuation coefficient do not significantly affect the image quality in these cases. However, for Layer 3, the dictionary fit is poor, leading to a lower-quality image.

Despite the poor least-squares fit for Layer 3, running the optimization is still valuable. It is possible that within the feasible set there are other solutions that yield better images even if they are farther from the ground truth $\mathbf{h}$ in the least-squares sense.

The optimization process is applied to all three layers, yielding different results:

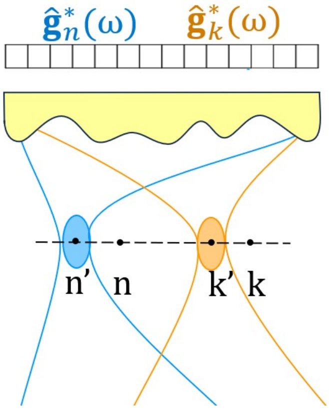
1. The first layer represents the simplest scenario. The feasible set is small, since we optimize only for the β corresponding to the lines 4 and 5. With the initial $\mathbf{h}$ we already obtain a focused beam. However, the focal spot obtained with this initial estimate is larger compared to the true focal spot. As a result, the image exhibits a larger PSF. The optimization leads to improved sidelobe suppression and a reduction in the focal spot size. The image obtained using the estimated transfer functions closely resembles the one generated with the true $\mathbf{h}$, indicating the success of the optimization approach in this simple layer.

2. Layer 2 presents more challenges due to the higher variations in thickness. The initial guess results in a completely unfocused beam. The optimized beam appears more focused, but not centered at the correct point. Consequently, there is a shift in the resulting image. Moreover, in the final image we observe structures that are not present in the correct image, indicating that the optimization process is not progressing in the right direction.

3. Layer 3 presents additional challenges in estimation compared to Layer 1, primarily due to its porous structure. Interestingly, when comparing with Layer 2, we observe that the initial beam for Layer 3 is better, indicating that high variations in thickness are more problematic than the porous structure itself. However, the initial image is still completely blurred. After the optimization, the focus is improved, resulting in a sharper image.

Comparing the results with the initial guess, the optimization using the proposed cost function yields more focused beams and sharper images for all three cases. However, the fact that the image becomes sharper does not mean that it becomes closer to the true image. For layer 2, we know that in the feasible set there is a solution that gives a good image, which is the least-squares solution. However, the optimization leads to other solutions, which have higher cost but generate a worse image.

In this thesis, we have only provided an intuitive explanation of why a good estimate is expected to have a high cost. When the beams $\mathbf{G}^T(\omega_l)\hat{\mathbf{g}}_n^*(\omega_l)$ and $\mathbf{G}^T(\omega_l)\hat{\mathbf{g}}_k^*(\omega_l)$ have clearly separated focal spots, the corresponding $x_{k,n}$ tends to be low. On the other hand, if the beams are unfocused, the $x_{k,n}$ tends to be high. It is important to note that for $x_{k,n}$ to be low, it is sufficient that the focal spots are clearly separate, without requiring the first focal spot to be centered at n and the second one at k. Other wrong solutions can lead to low $x_{k,n}$ and then high cost. For instance, we consider a situation where $\mathbf{G}^T(\omega_l)\hat{\mathbf{g}}_n^*(\omega_l)$ generates a beam centered at a point $n' \neq n$ and $\mathbf{G}^T(\omega_l)\hat{\mathbf{g}}_k^*(\omega_l)$ generates a beam centered at a point $k' \neq k$. If the focal spots are clearly separated, the coefficient $x_{k,n}$ will be low. However, the estimates are poor since the focal spots are not centered at n and k. This situation is depicted in Figure 4.23. A scenario like this would result in mapping the intensity of point n' to n and the intensity of point k' to k, leading to a shift on the left in the beamformed image.



$\hat{\mathbf{g}}_n^*(\omega)$
$\hat{\mathbf{g}}_k^*(\omega)$
n'
n
k'
k

Figure 4.23: To achieve a low $x_{k,n}$ value, it is necessary that $\mathbf{G}^T(\omega_l)\hat{\mathbf{g}}_n^*(\omega_l)$ and $\mathbf{G}^T(\omega_l)\hat{\mathbf{g}}_k^*(\omega_l)$ represent two beams with clearly separated focal spots. However, this does not guarantee that the focal spot locations will be precisely aligned with the correct ones, which could result in a shift in the resulting image.

The idea of computing the $x_{n,k}$ coefficients has been used also in [31]. In this paper, Lambert et al. derive a matrix that can be considered as the equivalent of $\mathbf{X}_{n_z}$ in (4.6). They compute $\mathbf{X}_{n_z}$ using the Green's functions in homogeneous medium $\mathbf{G}_0(\omega)$. Then, they define one metric to measure aberrations, which they call “F-factor”. This metric could be examined to improve the cost function. However, it's important to remember that the final cost must remain suitable for gradient descent.

We introduce the parametrization of $\mathbf{h}$ with the purpose of excluding wrong solutions that result in a high cost. While this approach works for Layer 1, it is not sufficient for Layer 2. Therefore, it becomes evident that the dimension of the feasible space should be further reduced.

The parametrization we have used has the advantage of being linear in the parameters. However, other parametrization would be possible. For instance, for Layer 1 and 2 we could approximate each transfer function as $H_{i,j}(\omega) = S(\omega)e^{j\omega\tau_{i,j}}$. This approximation works well when diffraction can be neglected, and the geometric approach described in Section 2.3 can be applied. In such a case, the optimization could be performed on the $\tau_{i,j}$ parameters. However, it is important to highlight that $\mathbf{h}$ is no longer linear in the optimization variables. While gradient ascent could still be employed, introducing non-linearity may not be beneficial in finding a stable solution. Additionally, this approach would require constraints on the minimum and maximum delays. Moreover, we would need to ensure there are no sharp discontinuities in the delays of adjacent elements.

The model used in this thesis is linear in the parameters. The β coefficients can be discontinuous in the x dimension. However, the overall $\mathbf{H}(\omega)$ will not be discontinuous, as the dictionary entries overlap. Furthermore, we automatically limit the time support without the need for additional constraints, which simplifies the optimization problem.

Despite these advantages, we have observed that the current approach may not be sufficient to exclude solutions that result in a higher cost compared to the optimal one.

Therefore, further investigation is required to address this limitation.

To identify the initial point, prior information about the layer speed of sound has been employed. The procedure involves keeping the speed of sound constant while varying the thickness, with the goal of finding the thickness that maximizes the cost. We believe that knowing the exact speed of sound is not crucial. The key factor is to determine a pair consisting of average thickness and average speed of sound which provides an accurate estimate of the pulse travel time from the sensor array to the virtual array. To find this pair, one can fix the speed of sound and vary the thickness, as we did in this thesis. Alternatively, one could fix the thickness and vary the speed of sound. In this situation, this method could be integrated with more sophisticated techniques for identifying the average speed of sound. In any case, it is expected that errors in one variable will not pose significant issues, as long as the other one compensates for the errors, leading to an accurate travel time estimate.

Limitations of this thesis

It is important to acknowledge that the general problem is more complex than the one presented in this thesis. Several simplifications have been adopted in this work:

- no noise is added to the k-Wave simulation data;
- the speed of sound and density are set to the ones of plastic, resulting in a relatively low impedance mismatch and low losses compared to for example skull bone;
- the reflections from the masks are removed from the $\mathbf{Y}(\omega)$ data.

The focus of this chapter has been restricted to distortion correction. Even with low losses and zero noise, this is already a challenging problem. Indeed, distortion correction becomes more challenging as we introduce noise and increase the signal losses.

Moreover, the optimization for Layer 3 did not work when the phantom was placed at the same depth as for the other two layers. To achieve an improvement in the optimized image, it was necessary to move it 1 mm below. We believe that this issue is related to the presence of consistent reflections, which have a long duration because of the internal porous structure. Even though the reflections are subtracted from the measurements, numerical errors may still lead to residual effects, impacting the optimization process.

In addition, another aspect that requires further investigation is the convergence analysis. While the optimization for Layer 1 appeared to work well, it remains unclear whether starting from different initial points may lead to convergence issues and getting stuck in local minima.

Lastly, the search for the average depth in Layer 2 did not lead to the optimal result, and it requires further investigation.

Conclusion 5

In this thesis, the concept of a virtual array is exploited to develop methods for image formation and aberration estimation.

In Chapter 3, an alternative method for image formation is presented. The initial step involves estimating the measurements that would result from transmission and reception at the virtual array. This estimation can be done using either a matched filter or a regularized inverse filter, with the latter resulting in a shorter pulse-shape. After estimating this data, DAS is applied to form an image. The PSF is examined for different virtual transmit schemes. Improvements are observed when incorporating more virtual transmissions. However, this improvement comes with a corresponding increase in the number of computations. In addition, the method remains applicable even when the real array is undersampled. In this case, a coding mask is placed between the real and the virtual array. We demonstrate that the advantage of using a coding mask lies in the ability to virtually synthesize more transducers than those physically available. Finally, the method is compared to model-based matched filtering and it is shown that the resulting images are equivalent. However, unlike matched filtering, the proposed method does not require to compute the large model matrix. Consequently, it can potentially lead to a faster implementation.

In Chapter 4, an iterative scheme to estimate the layer transfer functions is proposed. This method uses pulse-echo data and does not rely on specific assumptions about the image structure. However, it requires to define a speed of sound and a thickness range for the layer. Based on this, a dictionary is generated and the dictionary coefficients are updated using Gradient Ascent. The proposed algorithm effectively estimates the transfer functions, when the layer can be modeled using a small dictionary. When dealing with more complex layers and larger dictionaries, the optimization process remains effective in achieving enhanced focus quality and sharper images compared to the initial guess. However, it does not converge to the optimal estimate. Instead, it leads to other transfer functions which result in higher values for the cost function, but correspond to worse estimates.

Future research

Future research could investigate how to prevent the optimization from yielding inaccurate estimates. One possibility is to further narrow down the feasible set, for instance by enforcing sparsity on the dictionary coefficients. Moreover, a deeper analysis is necessary to understand why the cost function assumes high values at incorrect solutions. To this end, it might be useful to examine how the different points contribute to the overall cost. A reliable estimate should result in a simultaneous cost increase across the entire image region. On the other hand, other solutions might lead to cost increases limited to one specific pixel cluster. Improvements might be obtained by penalizing these solutions.

Bibliography

[1] T. L. Szabo, *Diagnostic Ultrasound Imaging: Inside Out.* Elsevier, 2014.

[2] R. Ali, T. Brevett, L. Zhuang, H. Bendjador, A. S. Podkowa, S. S. Hsieh, W. Simson, S. J. Sanabria, C. D. Herickhoff, and J. J. Dahl, "Aberration correction in diagnostic ultrasound: A review of the prior field and current directions," *Zeitschrift für Medizinische Physik*, 2 2023.

[3] P. Q. van der Meulen, *Ultrasound imaging through aberrating layers.* PhD thesis, 2023.

[4] G. Pinton, J.-F. Aubry, E. Bossy, M. Muller, M. Pernot, and M. Tanter, "Attenuation, scattering, and absorption of ultrasound in the skull bone," *Medical Physics*, vol. 39, pp. 299–307, 12 2011.

[5] L. M. Hinkelman, T. D. Mast, L. A. Metlay, and R. C. Waag, "The effect of abdominal wall morphology on ultrasonic pulse distortion. Part I. Measurements," *The Journal of the Acoustical Society of America*, vol. 104, pp. 3635–3649, 12 1998.

[6] P. Kruizinga, P. van der Meulen, A. Fedjajevs, F. Mastik, G. Springeling, N. de Jong, J. G. Bosch, and G. Leus, "Compressive 3D ultrasound imaging using a single sensor," *Science Advances*, vol. 3, 12 2017.

[7] P. van der Meulen, M. Coutiño, J. G. Bosch, P. Kruizinga, and G. Leus, "Ultrasonic imaging through aberrating layers using covariance matching," *IEEE Transactions on Computational Imaging*, pp. 1–15, 2023.

[8] M. Fink, "Time-reversal acoustics in complex environments," *Geophysics*, vol. 71, no. 4, 2006.

[9] P. M. Morse and K. U. Ingard, *Theoretical acoustics.* New York: McGraw-Hil, 1968.

[10] M. Fink, "Time reversal of ultrasonic fields. I. Basic principles," *IEEE Transactions on Ultrasonics, Ferroelectrics and Frequency Control*, vol. 39, pp. 555–566, 9 1992.

[11] D. Cassereau and M. Fink, "Time-Reversal of Ultrasonic Fields—Part III: Theory of the Closed Time-Reversal Cavity," *IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control*, vol. 39, no. 5, pp. 579–592, 1992.

[12] M. Tanter, J.-L. Thomas, and M. Fink, "Time reversal and the inverse filter," *The Journal of the Acoustical Society of America*, vol. 108, pp. 223–234, 7 2000.

[13] M. Tanter, J.-L. Thomas, and M. Fink, "Focusing and steering through absorbing and aberrating layers: Application to ultrasonic propagation through the skull," *The Journal of the Acoustical Society of America*, vol. 103, pp. 2403–2410, 5 1998.

[14] J. A. Jensen, S. I. Nikolov, K. L. Gammelmark, and M. H. Pedersen, “Synthetic aperture ultrasound imaging,” *Ultrasonics*, vol. 44, pp. e5–e15, 12 2006.

[15] M. Verweij, B. Treeby, K. van Dongen, and L. Demi, “Simulation of Ultrasound Fields,” in *Comprehensive Biomedical Physics*, pp. 465–500, Elsevier, 2014.

[16] V. Perrot, M. Polichetti, F. Varray, and D. Garcia, “So you think you can DAS? A viewpoint on delay-and-sum beamforming,” *Ultrasonics*, vol. 111, 3 2021.

[17] G. Montaldo, M. Tanter, and M. Fink, “Time reversal of speckle noise,” *Physical Review Letters*, vol. 106, no. 5, 2011.

[18] L. Nock, G. E. Trahey, and S. W. Smith, “Phase aberration correction in medical ultrasound using speckle brightness as a quality factor,” *The Journal of the Acoustical Society of America*, vol. 85, pp. 1819–1833, 5 1989.

[19] R. Ali, T. Brevett, D. Hyun, L. L. Brickson, and J. J. Dahl, “Distributed Aberration Correction Techniques Based on Tomographic Sound Speed Estimates,” *IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control*, vol. 69, pp. 1714–1726, 5 2022.

[20] M. Mozaffarzadeh, E. Verschuur, M. D. Verweij, V. Daeichin, N. De Jong, and G. Renaud, “Refraction-Corrected Transcranial Ultrasound Imaging Through the Human Temporal Window Using a Single Probe,” *IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control*, vol. 69, no. 4, 2022.

[21] K. Shapoori, A. N. Sinclair, and R. G. Maev, “An Ultrasonic Adaptive Beamforming Method and Its Application for Trans-Skull Imaging of Certain Types of Head Injuries; Part II: Reception Mode and Adaptive Imaging,” *IEEE Transactions on Biomedical Engineering*, vol. 70, pp. 544–552, 2 2023.

[22] C. Angla, B. Larrat, J. Gennisson, and S. Chatillon, “Transcranial ultrasound simulations: A review,” *Medical Physics*, vol. 50, pp. 1051–1072, 2 2023.

[23] M. Karaman, A. Atalar, H. Koymen, and M. O’Donnell, “A phase aberration correction method for ultrasound imaging,” *IEEE Transactions on Ultrasonics, Ferroelectrics and Frequency Control*, vol. 40, pp. 275–282, 7 1993.

[24] C. Prada, S. Manneville, D. Spoliansky, and M. Fink, “Decomposition of the time reversal operator: Detection and selective focusing on two scatterers,” *The Journal of the Acoustical Society of America*, vol. 99, pp. 2067–2076, 4 1996.

[25] G. Montaldo, M. Tanter, J. Bercoff, N. Benech, and M. Fink, “Coherent plane-wave compounding for very high frame rate ultrasonography and transient elastography,” *IEEE Transactions on Ultrasonics, Ferroelectrics and Frequency Control*, vol. 56, pp. 489–506, 3 2009.

[26] B. E. Treeby and B. T. Cox, “k-Wave: MATLAB toolbox for the simulation and reconstruction of photoacoustic wave fields,” *Journal of Biomedical Optics*, vol. 15, no. 2, p. 021314, 2010.

[27] Y. Du, H. Jensen, and J. A. Jensen, “Investigation of an angular spectrum approach for pulsed ultrasound fields,” *Ultrasonics*, vol. 53, pp. 1185–1191, 8 2013.

[28] B. E. Anderson, J. Douma, T. Ulrich, and R. Snieder, “Improving spatio-temporal focusing and source reconstruction through deconvolution,” *Wave Motion*, vol. 52, pp. 151–159, 1 2015.

[29] A. Aubry and A. Derode, “Singular value distribution of the propagation matrix in random scattering media,” *Waves in Random and Complex Media*, vol. 20, pp. 333–363, 8 2010.

[30] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Köpf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, and S. Chintala, “PyTorch: An Imperative Style, High-Performance Deep Learning Library,” 12 2019.

[31] W. Lambert, J. Robin, L. A. Cobus, M. Fink, and A. Aubry, “Ultrasound Matrix Imaging - Part I: The Focused Reflection Matrix, the F-Factor and the Role of Multiple Scattering,” *IEEE Transactions on Medical Imaging*, vol. 41, pp. 3907–3920, 12 2022.