# Sound Speed Estimation Using Wave-based Ultrasound Tomography: Theory and GPU Implementation

O. Roy, I. Jovanović, A. Hormati, R. Parhizkar, and M. Vetterli

Audiovisual Communications Laboratory,
Ecole Polytechnique Fédérale de Lausanne,
CH-1015 Switzerland

## ABSTRACT

We present preliminary results obtained using a time domain wave-based reconstruction algorithm for an ultrasound transmission tomography scanner with a circular geometry. While a comprehensive description of this type of algorithm has already been given elsewhere,[1,2] the focus of this work is on some practical issues arising with this approach. In fact, wave-based reconstruction methods suffer from two major drawbacks which limit their application in a practical setting: convergence is difficult to obtain and the computational cost is prohibitive. We address the first problem by appropriate initialization using a ray-based reconstruction. Then, the complexity of the method is reduced by means of an efficient parallel implementation on graphical processing units (GPU). We provide a mathematical derivation of the wave-based method under consideration, describe some details of our implementation and present simulation results obtained with a numerical phantom designed for a breast cancer detection application. The source code of our GPU implementation is freely available on the web at `www.usense.org`.

**Keywords:** acoustic wave equation, graphical processing units, inverse problems, ultrasound tomography

## 1. INTRODUCTION

When an ultrasound wave propagates through an object, it is modified as a function of the internal structure of the object. Therefore, information about the physical (acoustic) properties of the object can be obtained by carefully quantifying these modifications. This forms the basis of ultrasound transmission tomography, a powerful imaging modality which has been successfully applied in medicine, industrial non-destructive testing, seismology and oil exploration.

The imaging setup typically consists of a number of transducers surrounding the object of interest. For each transmitted signal, a set of received signals are recorded, each characterized by a different shape, amplitude and delay. These signals are generally used to produce two images: the sound speed image and the attenuation image. The focus of this work is on imaging sound speed in breast tissue in order to detect cancerous lesions. Early breast cancer detection, as an important application, relies on the ability to detect small masses (typically a few millimeters) and to accurately render the shape of the lesions as a means to distinguish benign from malign tissues.[3] Therefore, it is important to reconstruct sound speed variations with high precision. A number of reconstruction methods in ultrasound tomography rely on a simplified model for sound propagation, referred to as the ray model. It assumes that the inhomogeneities in the medium are much larger compared to the probing wavelength such that energy propagation is well described by the ray theory. In this case, the only information needed to reconstruct the sound speed image are time of flight estimates, that is, propagation times between pairs of transducers. Although the reconstruction involves solving a non-linear problem, ray-based methods have

Further author information: (Send correspondence to O. Roy)
O. Roy - email: olivier.roy@usense.org
I. Jovanović - email: ivana.jovanovic@usense.org, phone: +41 21 693 1271
A. Hormati - email: ali.hormati@usense.org, phone: +41 21 693 7663
R. Parhizkar - email: reza.parhizkar@usense.org
M. Vetterli - email: martin.vetterli@usense.org, phone: +41 21 693 5698

a reasonable complexity. However, they fail to resolve small masses and to accurately render the star-like shape of some cancerous lesions. A lot of diagnosis information may thus be lost by this lack of spatial resolution. Resolution can be improved by considering a more accurate model for sound propagation, more precisely, by solving the two-dimensional wave equation. Ultrasound tomography methods based on solutions of the wave equation have been investigated by various authors, most notably in seismology[4] and in medical imaging.[2] In these contexts, various reconstruction algorithms have been proposed. In particular, the reconstruction method described in this paper is a variation on a time domain inversion technique previously proposed in Ref. 2. The common difficulties in all these methods are twofold: convergence is difficult to obtain and their computational complexity is prohibitive.

The goal of this work is to address these two shortcomings as a means to use the method in practical settings. To solve the first problem we use an estimate of the sound speed obtained from a ray-based approach (see Ref. 5) as the starting point of the wave-based method. Then, the complexity of the method is reduced by means of an efficient time domain parallel implementation using GPUs. We run numerical simulations to assess the accuracy of the reconstruction. The results show that we successfully reconstruct the sound speed image with a resolution very close to its theoretical limit, and that a significant speedup can be achieved using GPUs.

The outline is as follows. In Section 2, we state the wave-based reconstruction problem and explain an iterative method to solve it. Section 3 details the GPU implementation. Some results obtained using a numerical breast phantom are presented in Section 4. Finally, conclusions are given in Section 5.

## 2. WAVE-BASED RECONSTRUCTION

We first start with a mathematical statement of the wave-based reconstruction problem (Section 2.1) and then describe a method to solve it (Section 2.2).

### 2.1 Problem Statement

Let us consider an array of $M$ ultrasound transducers with emit and receive capabilities, uniformly spaced on a circle $\Gamma$. This circular setup surrounds a medium whose properties need to be imaged. In our case, we estimate the speed of sound propagation inside this medium. In the domain of simulation $\Omega$, the sound is assumed to propagate following the two-dimensional wave equation

$$\nabla^2 u(x,t) - \frac{1}{c^2(x)} \frac{\partial^2}{\partial t^2} u(x,t) = s(x,t)\,, \tag{1}$$

where $u(x,t)$ denotes the sound pressure field at position $x = (x_1, x_2)^T \in \Omega$ and time $t$, $c(x)$ the sound speed, and $s(x,t)$ the source signal. The setup is illustrated in Figure 1 Typically, the field is excited at emitter $m$ with position $x_m$ by a pulse $s(t)$, such that $s(x,t)$ can be replaced by

$$s_m(x,t) = s(t)\,\delta(x - x_m)\,,$$

for $m = 0, 1, \ldots, M-1$. In this case, the induced wave field which is the solution to the wave equation (1) is denoted by $u_m(x,t)$. The pulse $s(t)$, and therefore the fields $u_m(x,t)$, are assumed to be zero for $t < 0$. In the rest of the discussion, we will find it more convenient to express results in terms of the object function $f(x)$ defined as[2]

$$f(x) = \frac{c_0^2}{c^2(x)} - 1\,,$$

where $c_0$ denotes the (constant) sound speed outside a compact subset of the domain of interest $\Omega$. The effect of propagation on the source signal $s_m$ can then be expressed as a non-linear operator $\mathcal{R}_m$, defined as

$$\mathcal{R}_m : f(x) \in L_2(\Omega) \longmapsto g_m(r,t) \in L_2(\Gamma \times (0,T))\,,$$

which maps the object function $f$ to the signal $g_m$ recorded on the circle $\Gamma$ over the time interval $(0,T)$. The notation $L_2(\Omega)$ denotes the set of square integrable functions over $\Omega$. It thus hold that

$$\mathcal{R}_m(f) = g_m\,, \tag{2}$$

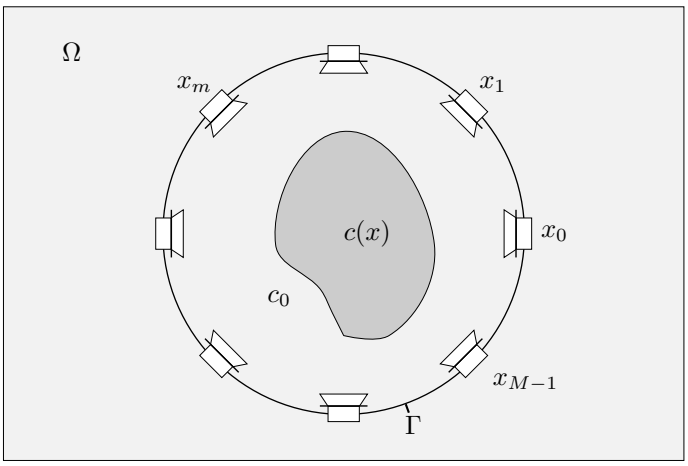


Figure 1. The ultrasound tomography setup.

for $m = 0, 1, \ldots, M-1$. The goal of the reconstruction method is to solve the above non-linear system of equations, that is, to recover the object function $f$ from the recorded signal $g_m$. Let us now describe the details of the method.

### 2.2 Method

In order to solve the non-linear system of equations (2), we follow a slightly different strategy as that adopted in Ref. 2 and minimize the following cost function

$$\mathcal{C}(f) = \sum_{m=0}^{M-1} \mathcal{C}_m(f) = \sum_{m=0}^{M-1} \|\mathcal{R}_m(f) - g_m\|^2 . \tag{3}$$

To this end, we consider the Fréchet derivative of the function $\mathcal{C}_m$ defined as the first order term of the difference $\mathcal{C}_m(f+h) - \mathcal{C}_m(f)$ (see, e.g., Ref. 1). It evaluates as

$$\begin{aligned}
&\mathcal{C}_m(f+h) - \mathcal{C}_m(f) \\
&= \|\mathcal{R}_m(f+h) - g_m\|^2 - \|\mathcal{R}_m(f) - g_m\|^2 \\
&= \langle \mathcal{R}_m(f+h), \mathcal{R}_m(f+h)\rangle - \langle \mathcal{R}_m(f+h), g_m\rangle - \langle g_m, \mathcal{R}_m(f+h)\rangle + \langle g_m, g_m\rangle \\
&\quad - \langle \mathcal{R}_m(f), \mathcal{R}_m(f)\rangle + \langle \mathcal{R}_m(f), g_m\rangle + \langle g_m, \mathcal{R}_m(f)\rangle - \langle g_m, g_m\rangle \\
&\overset{(a)}{=} \langle \mathcal{R}_m(f+h), \mathcal{R}_m(f+h)\rangle - \langle \mathcal{R}_m(f), \mathcal{R}_m(f)\rangle - \langle g_m, \mathcal{R}'_m(f)(h)\rangle - \langle \mathcal{R}'_m(f)(h), g_m\rangle + o(\|h\|^2) \\
&\overset{(b)}{=} \langle \mathcal{R}_m(f) - g_m, \mathcal{R}'_m(f)(h)\rangle + \langle \mathcal{R}'_m(f)(h), \mathcal{R}_m(f) - g_m\rangle + o(\|h\|^2) \\
&= 2\,\langle \mathcal{R}'_m(f)(h), \mathcal{R}_m(f) - g_m\rangle + o(\|h\|^2) ,
\end{aligned}$$

where in the equalities (a) and (b) we use the Taylor expansion

$$\mathcal{R}_m(f+h) = \mathcal{R}_m(f) + \mathcal{R}'_m(f)(h) + o(\|h\|^2)$$

and the fact that the involved terms are real valued. The Fréchet derivative of the function $\mathcal{C}_m$ is thus given by

$$\mathcal{C}'_m(f)(h) = 2\,\langle \mathcal{R}'_m(f)(h), \mathcal{R}_m(f) - g_m\rangle$$

which, using the definition of the adjoint operator $\langle \mathcal{R}(f), g\rangle = \langle f, \mathcal{R}^*(g)\rangle$ , reduces to

$$\mathcal{C}'_m(f)(h) = 2\,\langle h, (\mathcal{R}'_m(f))^*(\mathcal{R}_m(f) - g_m)\rangle \ .$$

**Algorithm 1** Wave-based Reconstruction

1. Start with an initial estimate $f_0$.
2. For each source $m$,
   - propagate the signal $s_m$ using the current estimate $f_k$ to obtain the field $u_m$,
   - compute the residual between the measured signal $g_m$ and the simulated signal $\mathcal{R}_m(f_k)$,
   - propagate the residual in a time-reversed manner to obtain the field $z_m$.
3. Compute the update $f_{k+1}$ using (4) and (5).
4. If $|f_{k+1} - f_k| < \epsilon$ for a prescribed threshold $\epsilon$, stop. Otherwise, go to step 2.

To decrease the cost function $\mathcal{C}_m$, the above quantity must be negative. Therefore, the update direction should be opposite to $(\mathcal{R}'_m(f))^*(\mathcal{R}_m(f) - g_m)$. By linearity, the overall cost function (3) can thus be minimized using the update equation

$$f_k = f_{k-1} + \alpha \sum_{m=0}^{M-1} (\mathcal{R}'_m(f_k))^*(g_m - \mathcal{R}_m(f_k))\,, \tag{4}$$

where $\alpha$ is a step size that can be chosen using a line search method, such as the backtracking line search algorithm.[6] Other update rules are of course possible. In particular, the updates provided by each source can be applied sequentially, as proposed in Ref. 2, instead of being summed up as we do here.

The above update relies on the computation of two main quantities. The term $\mathcal{R}_m(f)$ simply corresponds to the signals recorded on the circle $\Gamma$ when the source $s_m$ propagates with sound speeds given by $f$. It can thus be obtained by simulation using a finite difference method. The computation of the adjoint operator $(\mathcal{R}'_m(f))^*$ is more involved. It can be shown[2] (see details in Appendix A) that for any $g_\Gamma \in L_2(\Gamma, (0,T))$, the following relation holds

$$(\mathcal{R}'_m(f))^*(g_\Gamma) = \frac{1}{c_0^2} \int_0^T z_m \, \frac{\partial^2 u_m}{\partial t^2} \, dt\,, \tag{5}$$

where $z_m(x,t)$ is the solution to the wave equation

$$\nabla^2 z_m(x,t) - \frac{1}{c^2(x)} \frac{\partial^2}{\partial t^2} z_m(x,t) = g(x, T-t)\,, \tag{6}$$

with $z_m(x,t) = 0$ for $t < 0$. The signal $g(x,t)$ is the distribution defined by

$$\int_0^T \int_\Omega g\, \varphi \, dx \, dt = \int_0^T \int_\Gamma g_\Gamma \, \varphi \, dx \, dt\,, \tag{7}$$

for all $\varphi \in L_2(\Omega, (0,T))$. The above equations mean that the quantity $(\mathcal{R}'_m(f))^*(g_\Gamma)$ can be computed from two different wave fields: (i) the field $u_m$ induced by the source $s_m$, and (ii) the field $z_m$ induced by the source $g$ emitted in a time-reversed manner. As apparent in the update (4), the source $g$ is equal, on the circle $\Gamma$, to the residual between the measured signal $g_m$ and the signal $\mathcal{R}_m(f_k)$ simulated using the current sound speed values. Of course, in practice we can only measure $g_m$ at a finite number of transducer locations. Hence, some approximation is involved. For example, the unknown values can be interpolated from the existing ones or simply set to zero. The method is summarized in Algorithm 1. Note that this method requires to compute two wave fields using finite difference methods. This operation must be performed many times and can quickly become prohibitive when using a fine simulation grid. In the next section, we propose to use the parallel computing power of GPUs to reduce simulation time.

## 3. GPU PROCESSING

We now describe the GPU implementation of our wave-based reconstruction algorithm. We first provide some generalities on GPU processing (Section 3.1). Details on the time domain finite difference method are then given (Section 3.2). Finally, we describe the chosen absorbing boundary conditions (Section 3.3) and comment on some further implementation details that can be used to speed up computations (Section 3.4).

### 3.1 Overview

Graphical processing units have been designed for the rendering of complex graphic scenes that involve a very large amount of highly parallel computations. More precisely, the GPU computing architecture is particularly well suited to solve problems with high arithmetic intensity, defined as the ratio of arithmetic operations to memory operations.[7] Quite recently, NVIDIA released a programming architecture, referred to as CUDA, that enables general purpose computations on GPUs using a high-level programming language such as C. While a comprehensive description of this framework is beyond the scope of this paper, we review a few important concepts that need to be understood to efficiently use the computing power available on GPUs. For a more detailed exposition, we refer to the available literature on this subject.[7–9]

In CUDA, a GPU program is simply a C program running on the CPU that makes a number of calls to the GPU for parallel execution. For this reason, the CPU is referred to as the *host*, and the GPU as the *device*. A call to the device takes the form of a function, referred to as a *kernel*, which is executed in parallel by all the threads available on the GPU. To this end, each thread has a unique identifier used inside the kernel to index operations. For example, to add two vectors using parallel computations, each thread can add up the component indexed by its own identifier. The host and device each have their own memory. Therefore, the data read by kernels must be transferred from the host to the device. Conversely, the result of a kernel execution should be copied from the device to the host. GPU threads are grouped into blocks whose size is limited by the hardware (e.g., $16 \times 16$ blocks). Each thread has access to its private *local memory*. The treads of the same block have access to a common *shared memory*. Finally, all the threads have access to the same *global memory*. The global memory is large (e.g., 1GB) and its state is persistent across kernel launches. Shared memory is much smaller (e.g., 16KB), its state is reset every kernel call and its content cannot be directly copied from/to the host memory. However, read and write operations in shared memory are typically two order of magnitude faster than in global memory.[9] This memory size, persistency and access speed trade-off is an important aspect of GPU programming and should be optimized carefully.

The computational burden of our wave-based reconstruction algorithm lies in the simulation of propagating wave fields using a finite difference method. More precisely, two fields $u_m$ and $z_m$ must be simulated for each source. In practice, however, this is not exactly true. The computation of $z_m$ requires the values of $u_m$ on the circle $\Gamma$. In other words, the two fields must be evaluated sequentially. In order to compute the integral (5), we thus need to store the values of $u_m$ at every position and every time instant. This is unfortunately not practical. Instead, once the propagation of $u_m$ has been computed, one can simultaneously simulate the propagation of $z_m$ as well as propagate the field $u_m$ back in time such that, at every time instant, the term inside the integral (5) can be evaluated. This approach thus requires the simulation of three different wave fields. This is achieved using an efficient finite difference method implemented on GPUs, as proposed in Ref. 9, which we now describe in more details.

### 3.2 Finite Difference Computation

The fields are discretized in time with an interval $\Delta t$, and in space using a grid with cells of dimension $\Delta x \times \Delta y$. The field value at time $n$ and position $(i, j)$ is thus given by

$$u_{i,j}^n = u(x_{i,j}, n\Delta t)\,,$$

where $x_{i,j} = (i\Delta x, j\Delta y)$. Approximating the differential operators using finite differences leads to the following discrete version of the wave equation (1)

$$u_{i,j}^{n+1} = 2u_{i,j}^n - u_{i,j}^{n-1} + \Delta t^2 c_{i,j}^2 \left( \frac{u_{i+1,j}^n - 2u_{i,j}^n + u_{i-1,j}^n}{\Delta x^2} + \frac{u_{i,j+1}^n - 2u_{i,j}^n + u_{i,j-1}^n}{\Delta y^2} \right) .$$

Note that the above discretization is often referred to as the leapfrog scheme.[2] Assuming that $\Delta x = \Delta y$ and defining the matrices $C^2$ and $U^n$ such that $(C^2)_{i,j} = c_{i,j}^2$ and $(U^n)_{i,j} = u_{i,j}^n$, respectively, the above update can be conveniently written in matrix form as

$$U^{n+1} = 2\,U^n - U^{n-1} + \frac{\Delta t^2}{\Delta x^2}\,C^2 \odot (U^n * H)\;, \tag{8}$$

where $\odot$ denotes the element-wise product, $*$ the two-dimensional convolution operator and $H$ the filter matrix defined as

$$H = \begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix}\;.$$

The formulation (8) reveals that the computationally intensive part of the update is the filtering operation: for every grid point, a weighted average must be computed. At each time instant, all these operations can fortunately be performed in parallel using GPU. An important observation is that a grid point is involved in the update of four of its neighbors. The same value must thus be accessed five times. The approach proposed in[9] is to update the whole field on a block by block basis (stencil computation), where each block is processed by a different thread block. The field values are first copied from global memory to shared memory such that subsequent memory accesses can be significantly faster. The field update is computed in the shared memory and the global memory is then updated with the new field values. This strategy reduces redundant access to global memory and thus increases the data processing throughput.[9] Using this approach, we observed a speedup of about an order of magnitude compared to an optimized CPU implementation.

Our implementation follows the aforementioned strategy but considers instead a discretization of the wave equation using the Optimal Nearly Analytic Discrete Method (ONADM).[10,11] In this case, the update equation can be derived as

$$\begin{aligned} U^{n+1} = {} & 2\,U^n - U^{n-1} \\ & + \frac{\Delta t^2}{\Delta x^2}\,C^2 \odot \left[(U^n * H_u) + \frac{\Delta t^2}{\Delta x^2}\,C^2 \odot (U^n * H_u')\right] \\ & + \frac{\Delta t^2}{\Delta x^2}\,C^2 \odot \left[(V^n * F_u) + \frac{\Delta t^2}{\Delta x^2}\,C^2 \odot (V^n * F_u')\right] \\ & + \frac{\Delta t^2}{\Delta x^2}\,C^2 \odot \left[(W^n * G_u) + \frac{\Delta t^2}{\Delta x^2}\,C^2 \odot (W^n * G_u')\right]\;, \end{aligned} \tag{9}$$

where $U^n$, $V^n$ and $W^n$ are matrices whose components $(i,j)$ are given by $u_{i,j}^n$, $(\partial u/\partial x_1)_{i,j}^n$ and $(\partial u/\partial x_2)_{i,j}^n$, respectively. The filters can be computed as

$$H_u = \begin{bmatrix} 0 & 2 & 0 \\ 2 & -8 & 2 \\ 0 & 2 & 0 \end{bmatrix}, \quad H_u' = \begin{bmatrix} 1/6 & -4/3 & 1/6 \\ -4/3 & 14/3 & -4/3 \\ 1/6 & -4/3 & 1/6 \end{bmatrix}, \quad F_u = \begin{bmatrix} 0 & 0 & 0 \\ 1/2 & 0 & -1/2 \\ 0 & 0 & 0 \end{bmatrix}, \quad F_u' = \begin{bmatrix} 0 & 0 & 0 \\ -1/2 & 0 & 1/2 \\ 0 & 0 & 0 \end{bmatrix},$$

$G_u = F_u^T$ and $G_u' = F_u'^T$. The update of the matrix $V^n$ follows the same rule as above with the first ratio $\Delta t^2/\Delta x^2$ in (9) replaced by $\Delta t^2/\Delta x^3$. The filters are given by

$$H_v = \begin{bmatrix} -5/4 & 3/2 & -1/4 \\ -13/2 & 0 & 13/2 \\ 1/4 & -3/2 & 5/4 \end{bmatrix}, \quad F_v = \begin{bmatrix} 0 & 1 & 0 \\ -3/2 & -14 & -3/2 \\ 0 & 1 & 0 \end{bmatrix}, \quad G_v = \begin{bmatrix} -1/2 & 1/2 & 0 \\ -1 & 2 & -1 \\ 0 & 1/2 & -1/2 \end{bmatrix},$$

$$H_v' = \begin{bmatrix} 3/4 & -3/2 & 3/4 \\ 15/2 & 0 & -15/2 \\ -3/4 & 3/2 & -3/4 \end{bmatrix}, \quad F_v' = \begin{bmatrix} 0 & -1 & 0 \\ 5/2 & 12 & 5/2 \\ 0 & -1 & 0 \end{bmatrix} \quad \text{and} \quad G_v' = \begin{bmatrix} 1/2 & -1/2 & 0 \\ 1 & -2 & 1 \\ 0 & -1/2 & 1/2 \end{bmatrix}.$$

Similarly, the update of the matrix $W^n$ can be computed using the filters

$$H_w = H_v^T, \quad H_w' = H_v'^T, \quad F_w = G_v^T, \quad F_w' = G_v'^T, \quad G_w = F_v^T \quad \text{and} \quad G_w' = F_v'^T.$$

Note that the ONADM approach additionally requires the computation and storage of the first order derivatives of the field in space. However, it allows to significantly reduce the numerical dispersion that appears when the grid is too coarse. In other words, the cell size $\Delta x$ can be increased while still keeping the same amount of numerical dispersion as that of the leapfrog scheme.

### 3.3 Absorbing Boundary Conditions

In many practical scenarios, the ultrasound waves are assumed to propagate in an unbounded medium, that is, reflections on the parts of the ultrasound tomography scanner are neglected. In other words, waves should travel freely outside of the domain of simulation. This can be modeled by imposing absorbing conditions at the boundary of the domain. To this end, let us consider the propagation of a plane wave in a rectangular domain $\Omega$ and let us derive the corresponding absorbing condition on the left border (i.e., for positions $x = (x_1, x_2)^T$ with $x_1 = 0$, or grid points $(i, j)$ with $i = 0$).[12] The other cases follow similarly. A plane wave can be expressed as

$$u(x,t) = e^{j(kx-\omega t)}\,,$$

where $\omega$ is the temporal frequency and $k = (k_1, k_2)^T$ is the wave vector satisfying $\|k\| = \omega/c$. The derivative in the horizontal direction $x_1$ satisfies

$$\left(\frac{\partial}{\partial x_1} - j\,k_1\right) u = \left(\frac{\partial}{\partial x_1} - j\,\frac{w}{c}\sqrt{1 - \frac{c^2k_2^2}{\omega^2}}\right) u = 0\,.$$

Using the approximation $\sqrt{1 - c^2k_2^2/\omega^2} \approx 1 + o(c^2k_2^2/\omega^2)$ yields the following boundary condition in the time domain

$$\left(\frac{\partial}{\partial x_1} - \frac{1}{c}\frac{\partial}{\partial t}\right) u\Big|_{x_1=0} = 0\,.$$

The partial derivatives $\partial/\partial x_1$ and $\partial/\partial t$ are approximated by the operators $D_{x_1}$ and $D_t$ defined by $(u^n_{i+1,j} - u^n_{i,j})/\Delta x$ and $(u^{n+1}_{i,j} - u^n_{i,j})/\Delta t$, respectively. The above boundary condition is then discretized as[12]

$$D_{x_1}(u^n_{0,j} + u^{n+1}_{0,j}) - \frac{1}{c}D_t(u^n_{0,j} + u^n_{1,j}) = 0\,.$$

The update equation follows as

$$u^{n+1}_{0,j} = u^n_{1,j} + \frac{\Delta x - c\Delta t}{\Delta x + c\Delta t}\,(u^n_{0,j} - u^{n+1}_{1,j})\,.$$

### 3.4 Further Implementation Details

In order to further increase the speed of simulation, we propose two simple strategies. The first one is based on the observation that wave fields travel at a finite speed. Thus, a region at a distance $d_0$ from the emitter will not observe any fluctuations before time instant $t_0 = d_0/c_{\max}$, where $c_{\max}$ is the maximum assumed speed of propagation. Therefore, the strategy amounts to update the field only after time $t_0$. This can be easily implemented on GPU by storing the time instants $t_0$ for every thread block and every emitter. Since an additional parameter needs to be read from global memory, this approach will provide large gains only if a significant portion of the updates could be avoided. The second strategy amounts to copy the field values from global memory to shared memory, but to only apply the update equations if at least one value of the block exceeds a prescribed threshold. Below this threshold, we avoid a number of write operations in the global memory. Varying this threshold hence provides a trade-off between simulation accuracy and computational complexity.

## 4. RESULTS

As a means to assess the reconstruction accuracy of the proposed scheme, we present a number of synthetic results obtained using our GPU implementation and the numerical breast phantom shown in Figure 2(a). In this

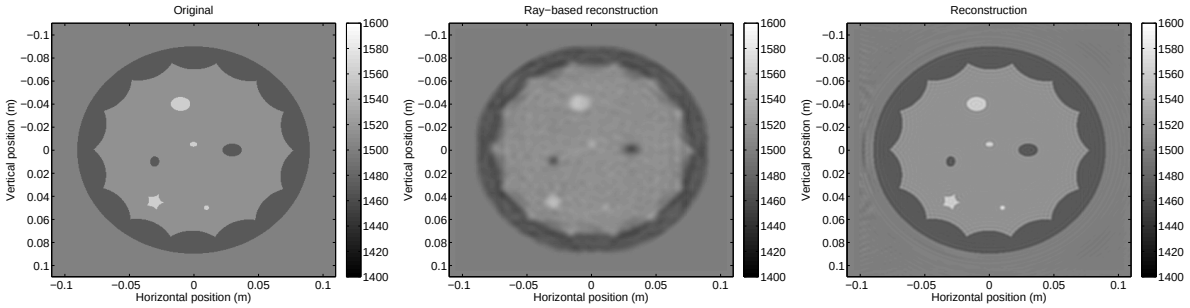


Figure 2. Ray-based vs. wave-based reconstruction. (a) Original phantom. (b) Ray-based reconstruction used as a starting point for the wave-based method. (c) Wave-based reconstruction. We observe that the higher spatial resolution provided by the wave-based method allows to recover the shape of the masses with a much higher accuracy.

phantom, the inclusions have diameters ranging from 4 to 8 mm. We consider an array of $M = 256$ ultrasound transducers, uniformly spaced on a circle $\Gamma$ with radius 10 cm. The source signal $s(t)$ is of the form

$$s(t) = e^{-t^2/(2\tau^2)} \cos(2\pi f_c\, t)\,,$$

where $\tau = 1/(2f_c)$. The center frequency is set to $f_c = 150$ kHz which corresponds to a bandwidth of $b = 134$ kHz. Here the bandwidth is defined as the decay of the spectrum of $s(t)$ to 1/e of its central value. It can be computed as $b = 2\sqrt{2}f_c/\pi$. Simulations were performed on a $0.22 \times 0.22$ square meter area using a $402 \times 402$ grid. The pixel size $\Delta x$ is thus approximatively 0.55 mm. In order to ensure stability of the ONADM method, we chose the sampling interval $\Delta t$ such that[11]

$$c_{\max} \frac{\Delta t}{\Delta x} \leq 0.527,$$

where in our case $c_{\max} \approx 1600$ m/s. Regarding the GPU architecture, we use an NVIDIA GeForce 9600 GT graphics card with 8 multiprocessors, about 1GB of global memory, 16KB of shared memory per thread block, each thread block being of size $16 \times 16$.

The original numerical breast phantom is depicted in Figure 2(a). In Figure 2(c), we show the reconstruction obtained by the proposed wave-based implementation. The initial sound speed estimate is obtained using a ray-based algorithm[5] and is depicted in Figure 2(b). For illustration purpose, we also plot in Figure 3(a) the difference, in absolute value, between the original and its reconstruction. The reconstruction quality is good but it is apparent that the scheme is not able to recover the high spatial frequencies that arise at the edges of the inclusions. In other words, the reconstruction is a lowpass version of the original one. This is further illustrated in Figure 3(b), where the squared magnitude of the ratio between the reconstructed and original bi-dimensional spectrums is depicted. We clearly observe the lowpass effect of the reconstruction scheme. The bandwidth of this filter is proportional to the maximum frequency of the input signal. Note that the strong frequency components at the zero horizontal and vertical frequencies are due to the rectangular tilling used in our experiments.

In Figure 4, we show the squared magnitude of the bi-dimensional spectrum of the original phantom, together with those obtained at different stages of the iterative reconstruction algorithm. We clearly observe that, as the number of iterations increases, the method is able to capture higher and higher spatial frequencies. We thus expect the corresponding reconstruction to exhibit sharper and sharper edges. Accordingly, the root mean squared error of the reconstruction decreases with the number of iterations, as depicted in Figure 5. This shows that the method converges properly. This convergence however requires a fairly good initial estimate. For example, choosing a constant sound speed as the starting point does not lead to convergence in this case.

To further illustrate the lowpass effect of the method, we plot in Figure 6(a) a one-dimensional cut of the phantom together with its reconstruction. We observe a Gibb's phenomenon around the points of discontinuity. A zoomed version of this cut is depicted in Figure 6(b), where the approximation obtained at different iterations is also plotted. This clearly illustrates the ability of the wave-based method to better capture singularities and thus better reconstruct sharp edges.

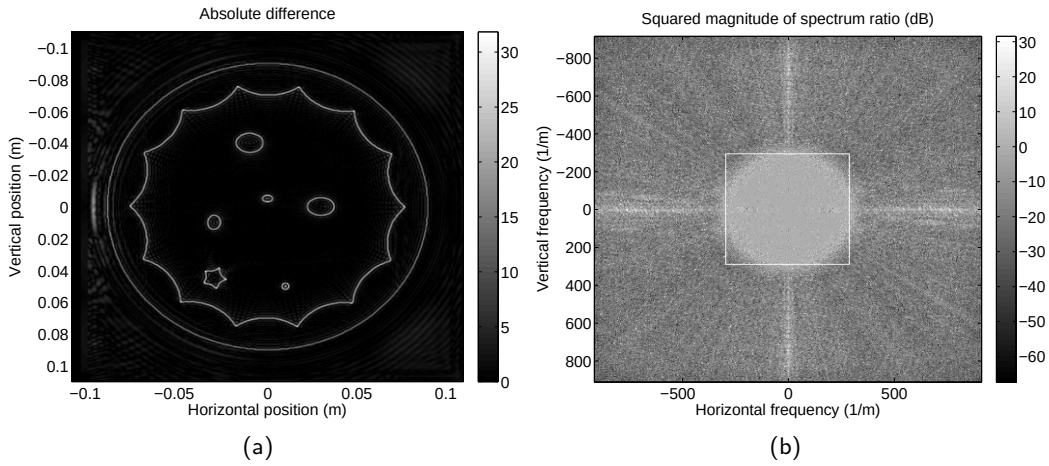


Figure 3. Accuracy of the reconstruction. (a) Absolute difference between the reconstructed and the original phantom. (b) Squared magnitude of the ratio between the spectrum of the reconstructed and the original phantom. We observe that the reconstruction is a lowpass version of the original. The spatial bandwidth, indicated by the square in Figure (b), is equal to $4/\lambda_0 = 4f_0/c$, where $c$ is the average speed of sound, $f_0$ the maximum temporal frequency in the input signal and $\lambda_0$ the corresponding wavelength. Here, $c \simeq 1500$ m/s, $f_0 \simeq 217$ kHz such that the spatial bandwidth is approximatively equal to 580 1/m.

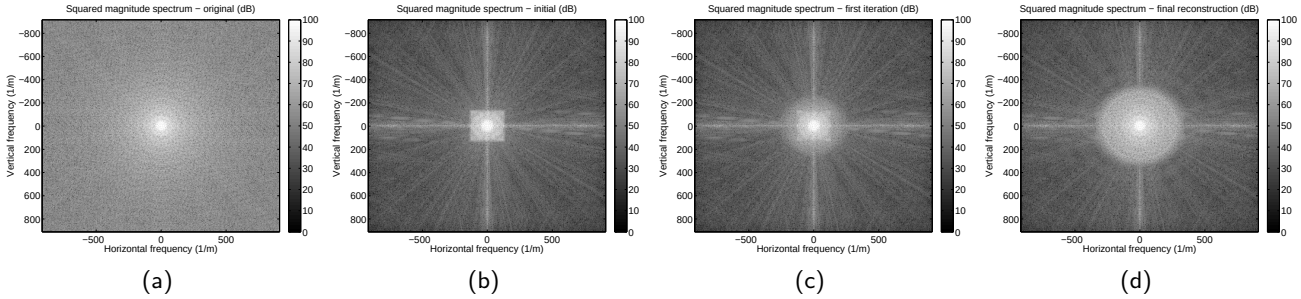


Figure 4. Squared magnitude of the original and estimated spectrums. (a) Original phantom. (b) Spectrum of the ray-based reconstruction. (c) Spectrum after the first iteration of the wave-based method. (d) Final reconstruction. We observe that the wave-based method allows to obtain a lowpass version of the original phantom. Its spatial bandwidth increases with the number of iterations. The ray-based reconstruction involves a regularization step using an ideal separable lowpass filter, as observed in Figure (b).

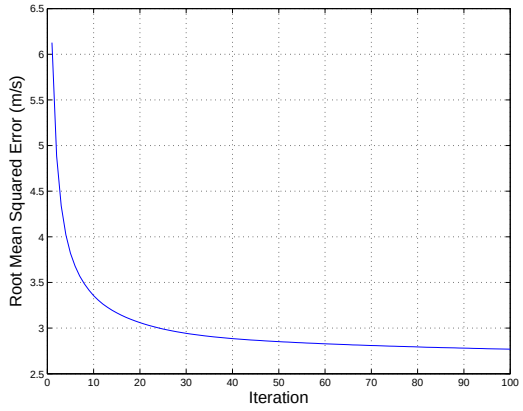


Figure 5. Root mean squared error between the original phantom and its reconstruction as a function of the number of iterations. The initial estimate obtained using a ray-based reconstruction scheme allows for the wave-based method to converge.

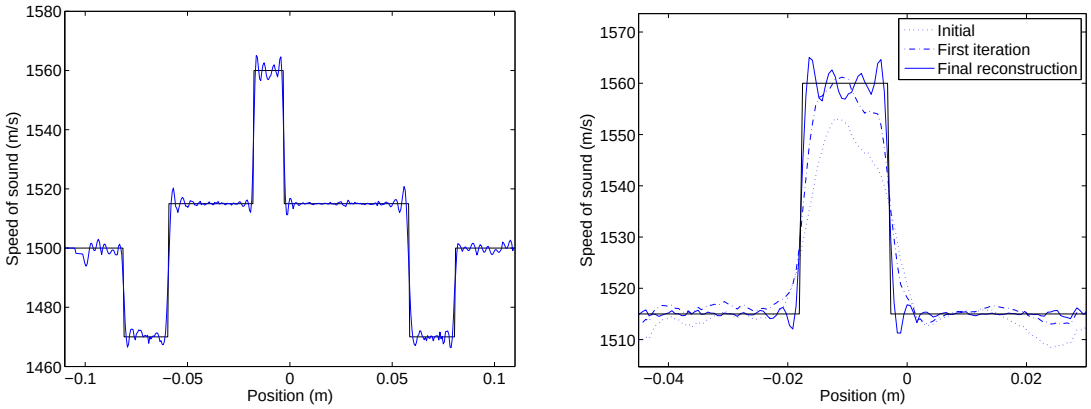


Figure 6. One-dimensional cross section of the synthetic phantom corresponding to the horizontal line with origin $-0.036$ m. (a) Original and reconstructed sound speed values. (b) Zoomed version with, in addition, the sound speeds used as the starting point of the iterative method and those estimated after the first iteration. We observe that the reconstruction is a lowpass approximation of the original piecewise constant sound speed profile. Furthermore, the wave-based method allows for a significant sharpening of the edges compared to the initial estimate obtained using a ray-based method.

## 5. CONCLUSIONS

We have shown some preliminary results obtained using a wave-based reconstruction method and a numerical phantom designed for a breast cancer detection application. A detailed derivation of the algorithm has been provided and an efficient GPU implementation has been described. Future work will focus on the application of this method to the reconstruction of physical phantoms as well as patient data obtained from an ultrasound scanner prototype.[13] Then, the computational complexity of the method will be further decreased to make it usable in a clinical environment.

## APPENDIX A. ADJOINT OPERATOR

We provide a detailed derivation of the adjoint operator $(\mathcal{R}'(f))^*(h)$. Let us first compute the Fréchet derivative operator $\mathcal{R}'(f)$. It is defined as the first order term of the difference $\mathcal{R}(f+h)-\mathcal{R}(f)$. We have that $\mathcal{R}(f)$ is equal to the solution $u(x,t)$ of the wave equation

$$\nabla^2 u(x,t) - \frac{1+f(x)}{c_0^2}\frac{\partial^2}{\partial t^2}u(x,t) = s(x,t) \tag{10}$$

evaluated on the circle $\Gamma$, with $u(x,t)=0$ for $t<0$. Similarly, $\mathcal{R}(f+h)$ is equal to the solution $v(x,t)$ of the wave equation

$$\nabla^2 v(x,t) - \frac{1+f(x)+h(x)}{c_0^2}\frac{\partial^2}{\partial t^2}v(x,t) = s(x,t) \tag{11}$$

evaluated on the circle $\Gamma$, with $v(x,t)=0$ for $t<0$. By linearity, $\mathcal{R}(f+h)-\mathcal{R}(f)$ thus follows from the solution $w(x,t)$ of the equation obtained by subtracting (10) from (11), that is,

$$\nabla^2 w(x,t) - \frac{1+f(x)}{c_0^2}\frac{\partial^2}{\partial t^2}w(x,t) = \frac{h(x)}{c_0^2}\frac{\partial^2}{\partial t^2}v(x,t)\,. \tag{12}$$

Furthermore, we have that

$$h(x)\frac{\partial^2}{\partial t^2}v(x,t) = h(x)\frac{\partial^2}{\partial t^2}u(x,t) + o(|h|^2)\,,$$

such that the Fréchet derivative operator $\mathcal{R}'(f)$ is equal to the solution of the wave equation

$$\nabla^2 w(x,t) - \frac{1+f(x)}{c_0^2}\frac{\partial^2}{\partial t^2}w(x,t) = \frac{h(x)}{c_0^2}\frac{\partial^2}{\partial t^2}u(x,t) \tag{13}$$

evaluated on the circle $\Gamma$, with $w(x,t) = 0$ for $t < 0$. Note that, as expected from the definition of the Fréchet derivative, $\mathcal{R}'(f)$ is linear with respect to $h$. Let us now compute its adjoint. We have that $\mathcal{R}'(f) : L_2(\Omega) \to L_2(\Gamma \times (0,T))$ such that there exists a unique adjoint operator $(\mathcal{R}'(f))^* : L_2(\Gamma \times (0,T)) \to L_2(\Omega)$ which satisfies

$$\langle R'(f)(h), s\rangle = \langle h, (R'(f))^*(s)\rangle \ , \quad \forall h \in L_2(\Omega) \text{ and } s \in L_2(\Gamma \times (0,T)) \, .$$

In integral form, this translates to

$$\int_0^T \int_\Gamma \mathcal{R}'(f)(h) s \, dxdt = \int_\Omega h (\mathcal{R}'(f))^*(s) \, dx \, . \tag{14}$$

Using (13) and integration by parts (see details in Appendix B), we can write

$$\begin{aligned} \int_0^T \int_\Omega \frac{h}{c_0^2} \frac{\partial^2 u}{\partial t^2} \, z \, dxdt &= \int_0^T \int_\Omega \left( \nabla^2 w - \frac{1}{c^2} \frac{\partial^2 w}{\partial t^2} \right) z \, dxdt \\ &= \int_0^T \int_\Omega w \left( \nabla^2 z - \frac{1}{c^2} \frac{\partial^2 z}{\partial t^2} \right) dxdt + \left[ \int_\Omega \frac{1}{c^2} \left( \frac{\partial z}{\partial t} w - \frac{\partial w}{\partial t} z \right) dx \right]_0^T . \end{aligned} \tag{15}$$

Let us choose $z(x,t)$ as the solution of the wave equation

$$\nabla^2 z(x,t) - \frac{1}{c^2(x)} \frac{\partial^2}{\partial t^2} z(x,t) = g(x,t) \, ,$$

with $z(x,t) = 0$ for $t > T$. The distribution $g(x,t)$ is defined in (7). Using the change of variable $t' = T - t$, it is readily seen that the field $z(x,t)$ is equivalent to the field $z'(x,t)$ generated by the time-reversed source $g(x, T-t)$ with initial condition $z'(x,t) = 0$ for $t > 0$. Using the above definitions, it holds from (15) that

$$\int_0^T \int_\Omega \frac{h}{c_0^2} \frac{\partial^2 u}{\partial t^2} \, z \, dxdt = \int_0^T \int_\Omega w \, g \, dxdt = \int_0^T \int_\Gamma w \, g_\Gamma \, dxdt = \int_0^T \int_\Gamma (\mathcal{R}'(f)(h)) \, g_\Gamma \, dxdt. \tag{16}$$

Note that the second term in (15) is zero since $w(x,t)$ (hence $\partial w/\partial t$) is zero for $t < 0$, and $z(x,t)$ (hence $\partial z/\partial t$) is zero for $t > T$. Comparing (14) and (16) reveals that

$$(\mathcal{R}'(f))^*(g_\Gamma) = \frac{1}{c_0^2} \int_0^T z \frac{\partial^2 u}{\partial t^2} \, dt$$

for any $g_\Gamma \in L_2(\Gamma \times (0,T))$.

## APPENDIX B. INTEGRATION BY PARTS

We give a detailed derivation of equality (15). We have that

$$\int_0^T \int_\Omega \left( \nabla^2 w - \frac{1}{c^2} \frac{\partial^2 w}{\partial t^2} \right) z \, dxdt = \underbrace{\int_0^T \int_\Omega z \nabla^2 w \, dxdt}_{=I_1} - \underbrace{\int_0^T \int_\Omega \frac{1}{c^2} \frac{\partial^2 w}{\partial t^2} z \, dxdt}_{=I_2} \, .$$

The integral $I_1$ can be expressed as

$$I_1 = \int_0^T \left( \int_\Omega z \frac{\partial^2 w}{\partial x_1^2} \, dx + \int_\Omega z \frac{\partial^2 w}{\partial x_2^2} \, dx \right) dt \, . \tag{17}$$

Let $n = (n_1, n_2)^T$ be the outer-pointing normal to the surface $\partial\Omega$. Using integration by parts, the first integral in (17) evaluates as

$$\begin{aligned} \int_\Omega z \frac{\partial^2 w}{\partial x_1^2} \, dx &= \int_{\partial\Omega} \frac{\partial w}{\partial x_1} z n_1 \, dr + \int_\Omega \frac{\partial w}{\partial x_1} \frac{\partial z}{\partial x_1} \, dx \\ &= \int_{\partial\Omega} \frac{\partial w}{\partial x_1} z n_1 \, dr - \int_{\partial\Omega} \frac{\partial z}{\partial x_1} w n_1 \, dr + \int_\Omega \frac{\partial^2 z}{\partial^2 x_1} w \, dx \, . \end{aligned}$$

Assuming that the fields $w$ and $z$ decay sufficiently fast as $|x|$ tends to infinity, their value on the boundary of the simulation domain $\Omega$ is negligible, thus the first two terms in the above equation vanish, such that

$$\int_\Omega z \frac{\partial^2 w}{\partial x_1^2}\, dx \approx \int_\Omega \frac{\partial^2 z}{\partial^2 x_1} w\, dx\,.$$

Under the same conditions, the second integral in (17) satisfies

$$\int_\Omega z \frac{\partial^2 w}{\partial x_2^2}\, dx \approx \int_\Omega \frac{\partial^2 z}{\partial^2 x_2} w\, dx\,,$$

such that

$$I_1 = \int_0^T \left( \int_\Omega \left( \frac{\partial^2 z}{\partial^2 x_1} + \frac{\partial^2 z}{\partial^2 x_2} \right) w\, dx \right) dt = \int_0^T \int_\Omega w \nabla^2 z\, dx\, dt\,.$$

Similarly, the integral $I_2$ is given by

$$\begin{aligned} I_2 &= \left[ \int_\Omega \frac{1}{c^2} \frac{\partial w}{\partial t} z\, dx \right]_0^T - \int_0^T \int_\Omega \frac{1}{c^2} \frac{\partial w}{\partial t} \frac{\partial z}{\partial t}\, dx dt \\ &= \left[ \int_\Omega \frac{1}{c^2} \frac{\partial w}{\partial t} z\, dx \right]_0^T - \left( \left[ \int_\Omega \frac{1}{c^2} \frac{\partial z}{\partial t} w\, dx \right]_0^T - \int_0^T \int_\Omega \frac{1}{c^2} \frac{\partial^2 z}{\partial^2 t} w\, dx dt \right) \\ &= \int_0^T \int_\Omega \frac{1}{c^2} \frac{\partial^2 z}{\partial^2 t} w\, dx dt - \left[ \int_\Omega \frac{1}{c^2} \left( \frac{\partial z}{\partial t} w - \frac{\partial w}{\partial t} z \right) dx \right]_0^T . \end{aligned}$$

Subtracting $I_2$ from $I_1$ yields the claimed equality.

## REFERENCES


[1] Tarantola, A., [*Inverse Problem Theory and Methods for Model Parameter Estimation*], SIAM (2005).

[2] Natterer, F. and Wubbeling, F., [*Mathematical Methods in Image Reconstruction*], Society for Industrial and Applied Mathematics, Philadelphia, PA, USA (2001).

[3] Stavros, A. T., Thickman, D., Rapp, C. L., Dennis, M. A., Parker, S. H., and Sisney, G. A., "Solid breast nodules: use of sonography to distinguish between benign and malignant lesions," *Radiology* **196**(1), 123–134 (1995).

[4] Pratt, R. G., "Seismic waveform inversion in the frequency domain, part 1: Theory and verification in a physical scale model," *Geophysics* **64**(3), 888–901 (1999).

[5] Jovanovic, I., *Inverse Problems in Acoustic Tomography: Theory and Applications*, PhD thesis, EPFL, Lausanne, Switzerland (2008).

[6] Boyd, S. and Vandenberghe, L., [*Convex Optimization*], Cambridge University Press (2004).

[7] NVIDIA Corporation, *NVIDIA CUDA Programming Guide 2.1* (2008).

[8] Lindholm, E., Nickolls, J., Oberman, S., and Montrym, J., "NVIDIA Tesla: A unified graphics and computing architecture," *IEEE Micro* **28**(2), 39–55 (2008).

[9] Micikevicius, P., "3D finite difference computation on GPUs using CUDA," in [*2nd Workshop on General Purpose Processing on Graphics Processing Units*], 79–84 (2009).

[10] Yang, D. H., Lu, M., Wu, R. S., and Peng, J. M., "An optimal nearly analytic discrete method for 2D acoustic and elastic wave equations," *Bulletin of the Seismological Society of America* **94**(5), 1982–1992 (2004).

[11] Yang, D., Peng, J., Lu, M., and Terlaky, T., "Optimal nearly analytic discrete approximation of the scalar wave equation," *Bulletin of the Seismological Society of America* **96**(3), 1114–1130 (2006).

[12] Engquist, B. and Majda, A., "Absorbing boundary conditions for the numerical simulation of waves," *Mathematics of Computations* **31**(139), 629–651 (1977).

[13] Duric, N., Littrup, P., Poulo, L., Babkin, A., Holsapple, E., Rama, O., and Glide, C., "Detection of breast cancer with ultrasound tomography: First results with the computed ultrasound risk evaluation (CURE) prototype," *Medical Physics* **2**(34), 773–785 (2007).