Localization of small obstacles from back-scattered data at limited incident angles with full-waveform inversion

Hélène Barucq, Florian Faucher, Ha Pham

Localization of small obstacles from back-scattered data at limited incident angles with full-waveform inversion

Hélène Barucq* Florian Faucher* Ha Pham*

June 5, 2018

Abstract

We investigate numerically the inverse problem of locating small circular obstacles in a homogeneous medium from multi-frequency back-scattered data limited to four angles of incidence. The main novelty of our paper is working with the position of the obstacles as parameter space in the frame work of full-waveform inversion (FWI) procedure. The computational cost of FWI is lowered by using a method based on single-layer potential. Reconstruction results are shown up to twenty-four obstacles, from initial guesses allowed to be far from the target. In experiments with six obstacles, we supplement the reconstruction with an analysis of the performance of the nonlinear conjugate gradient and quasi-Newton methods, in used with various line search algorithms.

1 Introduction

In this work, we use full-waveform inversion (FWI) to locate small circular obstacles in a homogeneous medium using multi-frequency backscattered data generated from a limited numbers of fixed angles. Obstacle localization problem has applications in materials imaging such as non-destructive testing using acoustic waves to detect defects, buried objects location, geophysical exploration and medical imaging, *cf.* [4, 20]. For our localization problem, we work with impenetrable (hard or soft-scattering) non-overlapping obstacles. We assume that the number, size and type of the obstacles are known, and that the obstacles are located strictly inside a rectangular domain of interest in which experiments will be carried out to collect observed data. Motivated by physical experiments[1], we also impose the following constraints in data collection for all testing frequencies (see illustration in Figure 3),

- back-scattered data are obtained from the following angles: 0°, 90°, 180°, 270°;
- for each angle, the data are collected at a fixed number of 128 receivers (points).

In general, an inverse problem aims to reconstruct the unknown model/parameter which gives rise to an observed data $\mathbf{d}$. This is equivalent to solving for the inverse $\Phi^{-1}(\mathbf{d})$ of the forward map $\Phi : \mathcal{P} \to \mathcal{D}$, which assigns to a model in the parameter space $\mathcal{P}$ a corresponding data in the data space $\mathcal{D}$. In our approach, the parameter space $\mathcal{P}$ represents the set of ordered N_{par}-tuples of the coordinates of the N_{Obs} obstacles, *i.e.* $\mathcal{P} = \mathbb{R}^{N_{\text{par}}}$ with $N_{\text{par}} = 2N_{\text{Obs}}$. The inverse problem is solved by minimizing the cost/misfit function $\mathcal{J} = \frac{1}{2}\|\Phi(\mathbf{p}) - \mathbf{d}\|^2$. Minimization is carried out by gradient-based optimization and corrects iteratively an initial guess in hope of converging towards the true model. In our work, the gradient of the cost function is calculated by the adjoint-state method, which avoids the computation of the Jacobian matrix of Φ.

In seismic inversion, the above methodology (nonlinear minimization of the cost function and adjoint-state method for its gradient) is called Full Waveform Inversion (FWI), see *e.g.* [62, 57, 66, 30], and belongs to the so-called quantitative/iterative family, in contrast to qualitative/direct

*Project-Team Magique-3D, Inria Bordeaux Sud-Ouest, Université de Pau et des pays de l'Adour, UMR CNRS 5142, E2S–UPPA.

[1] In our case, backscattered data arise when one single sensory device acts as both a source (by emitting almost planewaves) and receivers. Data of 128 points is the resolution set by the devices.

one. We refer to the introduction in [15] for a list of quantitative inversion references classified by their optimization method, as well as qualitative ones; for the latter family, see also [20, 40]. Quantitative inversion has the advantage of being 'conceptually simple' and giving reconstruction with higher precision, *cf.* [45]; however, this family requires solving numerous direct and adjoint problems and can thus be computationally intensive. The general idea of using efficient methods (for the forward problem) in order to alleviate the overall computational cost in iterative inversion is not new, *cf.* [16, 15] using fast solvers to solve forward obstacle and medium scattering. In our paper, we use an integral equation method based on single-layer potential (called FSSL) which was originally studied in [64, 6], and was shown in our previous works [14, 13] to be much more efficient, compared to Finite Element methods (FEM), in simulating diffraction by a large number of small impenetrable obstacles. In addition to this advantage, FSSL works more naturally with the current choice of parametrization (obstacle positions), *cf.* Remark 5.

The first main novelty of our results is in working directly with the position of the obstacles as parameters in the framework of nonlinear optimization with restrictive back-scattered data. In literature, most localization problems employ qualitative methods such as direct sampling, factorization, MUSIC, probe methods, *cf.* [29, 3, 39, 22, 34, 27]. The remaining few which use quantitative methods are in fact (penetrable) medium reconstruction problem; they retrieve the location of obstacles from the profile of the reconstructed sound speed/contrast/conductivity function of the medium[2], *cf.* [31, 42]. For quantitative medium reconstruction without the final goal of localizing obstacles, we refer to *cf.* [7, 10, 15] and the references therein. The sensitivity and non-linearity of the scattered field and hence the forward map, with respect to the contrast function (in an inhomogeneous Helmholtz equation) are of different nature than that on the position of the obstacles (in a multiple-scattering problem with Helmholtz equation)[3]. This distinction is even clearer, when scattering regions are impenetrable (as in our cases), for which multiple-obstacle scattering framework is more natural than inhomogeneous medium. Our work serves as an initial investigation of the feasibility of using the current choice of parameter space in FWI with line-search strategy, under the aforementioned restrictive data collection and starting from arbitrary initial guesses[4].

Limited aperture data, especially back-scattered data, and limited number of radiation angles present great challenges, both theoretically and experimentally, since they increase the ill-posedness of the problem. Such restriction on data collection is a common feature in seismic inversion (that works with 'reflection data'), see [62, 63], review [66] and the references therein. In general, ill-posedness is reduced by working with multi-frequency data, which is now a common technique; in seismic, *cf.* [19, 58, 61, 59], in inverse scattering, see [25, 9, 10, 16, 15, 43] for quantitative methods, and [34] and the references therein for qualitative ones. On the other hand, in the later context most of these references assume full-aperture data, with the exception of [21, 34, 2, 40, 29] in qualitative methods, and [12, 52, 48, 68] in quantitative and the references therein. Furthermore, the number of angles of incidences and receiver points can be allowed to grow proportionally with the frequency, *cf.* [15]. In our experiments, these quantities are kept constant at all frequencies, which greatly affects the efficiency of higher frequencies, *cf.* Remark 8. Regardless of these constraints, we are able to retrieve up to twenty-four obstacles. These results are on the higher end (in terms of the number of obstacles) in both aforementioned references (for quantitative and qualitative methods).

Due to the lack of reference work implementing FWI with our parametrization, in choosing line search strategy [53], we have to investigate which choice of search direction and line search algorithm are most compatible with our problem. This motivates us to compare between nonlinear conjugated gradient (NLCG) and quasi-Newton, in combination with different line search algorithms, which is the second main novelty of our work. Quasi-Newton and NLCG were considered for medium or shape reconstruction in [35, 47] and [42] respectively, however not in the framework of line search strategy

[2]Obstacles are considered as compactly supported inhomogeneities in a homogeneous background. In inhomogeneous medium scattering, the obstacles are described by the contrast function q ($n = 1 - q$ is also called refractive index), and the direct problem is modeled as $(-\Delta - \kappa^2 n)\, u = 0$ (called inhomogeneous Helmholtz equation) coupled with an outgoing radiation condition at infinity. This can also be posed on bounded domain, *e.g.* in Electrical Impedance Tomography (EIT), in which the information of the obstacles is contained in the conductivity γ, and the direct problem modeled as $\nabla \cdot (\gamma \nabla u) = 0$ coupled with a Dirichlet condition on the boundary of the domain.

[3] For inhomogeneous Helmholtz equation, see Footnote 2. For multiple-scattering with Helmholtz equation, see the discussion in Section 2.

[4]as opposed to one generated by a direct imaging method as done in [7].

and/or nonlinear optimization.

The remaining of the paper is organized as follows. Section 2 introduces FSSL and the discrete inverse problem. In Section 3, the derivative of the cost function and the frequency-hopping procedure are presented. Numerical experiments are in Section 4. Convergence comparison among different optimization methods for six obstacles is first carried out, after which the most reliable optimization method is tested with twelve and twenty-four obstacles.

2 Discrete direct and inverse problem

In nondestructive experiments to detect the location of obstacles, incident waves are sent into the domain of interest to be diffracted by the obstacles, and corresponding scattered waves u are recorded at receivers. In frequency domain, one works with time-harmonic acoustic excitation; in particular, assuming the wavespeed of the background homogeneous medium is c, and defining the wavenumber κ by the dispersion relation $\kappa = \frac{2\pi \mathrm{f}}{c}$, our sources are time-harmonic planewave $u_{\mathrm{pw}}(\mathbf{x})\, e^{\mathrm{i}2\pi \mathrm{f}\, t}$ of frequency f and angle of incidence α_{inc},

$$u_{\mathrm{pw}}(\mathbf{x}) \;=\; e^{\mathrm{i}\,\kappa\,\mathbf{x}\cdot(\cos\alpha_{\mathrm{inc}},\sin\alpha_{\mathrm{inc}})}\;,\; \mathbf{x}\in\mathbb{R}^2\,. \tag{1}$$

Testing waves are sent from N_{Acq} angles, denoted by $u_{\mathrm{pw}}^{(\mathrm{t})}$, $1 \le \mathrm{t} \le N_{\mathrm{Acq}}$. We describe below how simulated data, corresponding to one incident angle, are computed via the single-layer potential solver (conveniently) called FSSL, for more details see [14, 13].

Notations Consider a configuration of N_{Obs} non-overlapping circular obstacles. The circular obstacle I is of radius r_I and centered at $\mathbf{x}^{(I)} = (x_1^{(I)}, x_2^{(I)}) \in \mathbb{R}^2$. The configuration is described by vector $\boldsymbol{p} \in \mathbb{R}^{N_{\mathrm{par}}}$, called the parameter vector of size $N_{\mathrm{par}} = 2 \times N_{\mathrm{Obs}}$.

$$\boldsymbol{p} = (x_1^{(1)}, x_2^{(1)},\; \ldots, x_1^{(N_{\mathrm{Obs}})}\,,\, x_2^{(N_{\mathrm{Obs}})})\,. \tag{2}$$

Denote the polar coordinates relative to $\mathbf{x}^{(I)}$ by $(r_I(\cdot), \theta_I(\cdot))$,

$$\mathbf{x} \;=\; \mathbf{x}^I \;+\; r_I(\mathbf{x})\;(\cos\theta_I(\mathbf{x})\,,\,\sin\theta_I(\mathbf{x}))\,. \tag{3}$$

Denote by d_{IJ} the distance between the centers of obstacle I and J, and by θ_{IJ} and θ_{JI} their relative polar coordinates,

$$\mathbf{x}^{(I)} = \mathbf{x}^{(J)} + \mathsf{d}_{IJ}\,(\cos\theta_{JI}\,,\,\sin\theta_{JI})\,;\quad \mathbf{x}^{(J)} = \mathbf{x}^{(I)} + \mathsf{d}_{IJ}\,(\cos\theta_{IJ}\,,\,\sin\theta_{IJ})\,.$$

The non-overlapping assumption is given by $\mathsf{d}_{IJ} > \mathsf{r}_I + \mathsf{r}_J$.

The continuous forward problem in frequency domain The unknown scattered wave u is required to solve the Helmholtz equation

$$(-\Delta - \kappa^2)u = 0, \quad \text{outside of the obstacles}\;, \tag{4}$$

and to satisfy the κ-outgoing condition at infinity to ensure that it does not re-enter the domain of interest

$$\lim_{r\to\infty} \sqrt{r}\,(\partial_r u - \mathrm{i}\,\kappa\, u) = 0, \quad r = |\mathbf{x}|\,. \tag{5}$$

We assume that the obstacles are either soft or hard scattering, upon which Dirichlet boundary condition (BC) $\gamma_0^+(u + u_{\mathrm{inc}}) = 0$ or Neumann BC $\gamma_1^+(u + u_{\mathrm{inc}}) = 0$ respectively are imposed on their boundaries. See [14] for the definition of the exterior trace γ_0^+ and γ_1^+.

Discretization of the forward problem by FSSL Denote by Γ_I the boundary of obstacle I, $\mathrm{H}_0^{(1)}$ the Hankel function of the first kind, and by $\mathbf{w}_{J,l}$ the l-th order Fourier nodes on the boundary of obstacle J, *i.e.*

$$\mathbf{w}_{J,l}(\theta_J(\mathbf{x})\,,\, r_J(\mathbf{x})) \;:=\; e^{\mathrm{i}\,l\,\theta_J(\mathbf{x})}. \tag{6}$$

Define the single-layer potential along Γ_J with continuous density $v \in \mathcal{C}(\Gamma_J)$ by

$$(\mathcal{S}_J\, v)(\mathbf{x}) \;:=\; \int_{\Gamma_I} \tfrac{\mathrm{i}}{4}\,\mathrm{H}_0^{(1)}(\kappa\,|\mathbf{x}-\mathbf{y}|)\,v(\mathbf{y})\,d\sigma(\mathbf{y})\,. \tag{7}$$

Such a quantity satisfies the Helmholtz equation and the radiation condition. At approximation order $\mathbf{m}$, the approximate diffractive field is written as a linear combination of single-layer potentials with approximate single-layer densities $\{v_{h,J}\}$, the latter expressed in terms of $2\mathbf{m}+1$ Fourier nodes between $-\mathbf{m}$ and $\mathbf{m}$,

$$u_h \;=\; \sum_{J=1}^{N_{\mathrm{Obs}}} \mathcal{S}_J\; v_{h,J}\;;\; \text{with } v_{h,J} \;=\; \sum_{l=-\mathbf{m}}^{\mathbf{m}} V_{J,l}\,\mathbf{w}_{J,l}\,. \tag{8}$$

In this way, with the Helmholtz equation and the radiation condition already fulfilled, it remains to satisfy the boundary conditions imposed on the boundary of obstacles. When the obstacles are circular, the scattered wave can be written as multipole expansion defined in terms of Hankel functions of the first kind $\mathrm{H}_k^{(1)}$,

$$u_h(\mathbf{x}) \;=\; \frac{\mathrm{i}\,\pi}{2} \sum_{J=1}^{N_{\mathrm{Obs}}} \mathsf{r}_J \sum_{l=-\mathbf{m}}^{\mathbf{m}} V_{J,l}\;\mathrm{H}_l^{(1)}(\kappa\,r_J(\mathbf{x}))\;e^{\mathrm{i}\,l\,\theta_J(\mathbf{x})}\,. \tag{9}$$

To obtain the linear system solved by the unknowns V, exterior traces corresponding the boundary conditions along the obstacles are applied to the Ansatz (8) to obtain,

$$\mathbf{A}(\boldsymbol{p})\,V \;=\; F(\boldsymbol{p}, u_{\mathrm{pw}}). \tag{10}$$

The size of the linear system is given by

$$N = (2\mathbf{m}+1)N_{\mathrm{Obs}}. \tag{11}$$

That V is a solution of (10) is equivalent to $V \;=\; \mathsf{S}(\boldsymbol{p}, u_{\mathrm{pw}})$, where solution operator S is defined as

$$\mathsf{S} \;:\; \mathbb{R}^{N_{\mathrm{par}}} \quad\longrightarrow\quad \mathbb{C}^N \quad,\quad \boldsymbol{p} \quad\mapsto\quad \mathbf{A}(\boldsymbol{p})^{-1}\,F(\boldsymbol{p}, u_{\mathrm{pw}}). \tag{12}$$

When there are several angles of incidence, we denote by S_{t} the solution operator corresponding to planewave $u_{\mathrm{pw}}^{(\mathrm{t})}$. The multi-scattering matrix $\mathbf{A}$ composes of $N_{\mathrm{Obs}} \times N_{\mathrm{Obs}}$ block matrices, each of which is a matrix of size $(2\mathbf{m}+1)\times(2\mathbf{m}+1)$.

$$\mathbf{A} \;=\; \begin{pmatrix} \mathbf{A}_1 & \mathbf{A}_{12} & \dots & \mathbf{A}_{1\,(N-1)} & \mathbf{A}_{1\,N} \\ \mathbf{A}_{21} & \mathbf{A}_2 & \dots & \mathbf{A}_{2\,(N-1))} & \mathbf{A}_{2\,N} \\ \vdots & & & \ddots & \vdots \\ \mathbf{A}_{(N-1)\,1} & \mathbf{A}_{(N-1)\,2} & \dots & \mathbf{A}_{N-1} & \mathbf{A}_{(N-1)N} \\ \mathbf{A}_{N\,1} & \mathbf{A}_{N\,2} & \dots & \mathbf{A}_{N\,(N-1)} & \mathbf{A}_N \end{pmatrix}. \tag{13}$$

The diagonal operator $\mathbf{A}_I$, describes self-reflection by obstacle I. The off-diagonal $\mathbf{A}_{IJ}$ with $I \neq J$ describes the diffraction by obstacle I of the wave emitted by J.

For circular obstacles, the multiple-scattering linear system can be described explicitly in the form of multipole expansion, using the Hankel functions of the first kind $\mathrm{H}_k^{(1)}$ and the Bessel function J_k and their derivatives. The diagonal blocks $\mathbf{A}_I$ are diagonal matrices, with diagonal components given by

$$(\mathbf{A}_I)_{ll} = \mathrm{i}\,\pi\,\mathsf{r}_I\,\mathrm{J}_l(\kappa\,\mathsf{r}_I) \;\times \begin{cases} \mathrm{H}_l^{(1)}(\kappa\,\mathsf{r}_I) & \text{, soft-scattering,} \\ \kappa\,\mathrm{H}_l^{(1)\prime}(\kappa\,\mathsf{r}_I) & \text{, hard-scattering.} \end{cases} \tag{14}$$

For $I \neq J$, the components of the off-diagonal block $\mathbf{A}_{IJ}$ are given by,

$$(\mathbf{A}_{IJ})_{lm} = \mathrm{i}\,\pi\,\mathsf{r}_J \mathrm{J}_m(\kappa\,\mathsf{r}_J)\,\mathrm{H}^{(1)}_{m-l}\,(\kappa\,\mathsf{d}_{IJ})\;e^{\mathrm{i}\,(m-l)\boldsymbol{\theta}_{JI}} \times \begin{cases} \mathrm{J}_l(\kappa\,\mathsf{r}_I) & \text{, soft-scattering,} \\ \kappa\,\mathrm{J}'_l(\kappa\,\mathsf{r}_I) & \text{, hard-scattering.} \end{cases} \tag{15}$$

The components of the right-hand-side in (10) corresponding to planewave u_{pw} (1) are given by

$$F_{I,l} \;= -\,2\,u_{\mathrm{pw}}(\mathbf{x}^{(I)})\,\mathrm{i}^l\;\; e^{-\mathrm{i}\,l\,\alpha_{\mathrm{inc}}} \times \begin{cases} \mathrm{J}_l(\kappa\,\mathsf{r}_I) & \text{, soft-scattering,} \\ \kappa\,\mathrm{J}'_l(\kappa\,\mathsf{r}_I) & \text{, hard-scattering.} \end{cases} \tag{16}$$

Remark 1 (Invertibility and condition number of (10))**.** *FSSL is efficient in low-frequency scattering, i.e. up to medium-sized obstacles. For circular obstacles, to guarantee the invertibility of* (10)*, the testing wavenumber and the radius of the obstacles have to satisfy:* $\kappa\,\mathsf{r}$ *are not in the set of zeros of Bessel functions. A 'cheap' requirement however sufficient for our tests is that* $0 < \kappa\,\mathsf{r} \leq 2$ *(also holds for arbitrarily-shaped obstacles), for more details see [14, Remark 2]. With* $\mathrm{r} = 0.5$ *(size of the obstacles taken in the experiments), this means* $0 < \kappa \leq 4$*. In our inversion experiments, we use the wavenumber range* $0.08 \leq \kappa \leq 3.0$*, in which the condition number of the scattering matrix* $\mathbf{A}$ *is bounded, see Figure 1.*

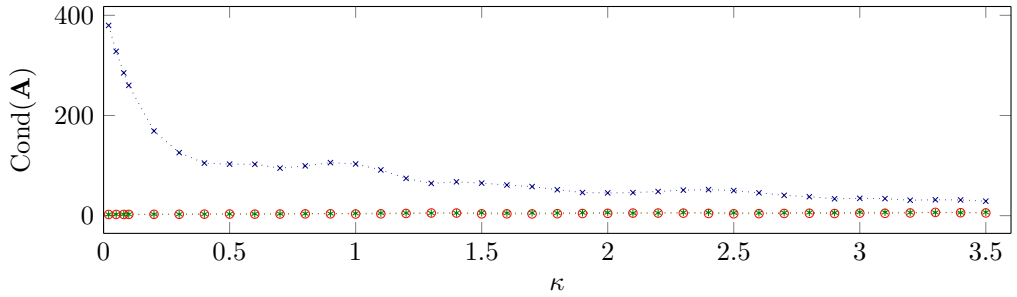


Figure 1: Condition number of the scattering matrix $\mathbf{A}$ (13), shown for $\mathbf{m} = 12$ and in the frequency range used for inversion: for six hard obstacles with the red circles (○ , see Subsection 4.1); for twelve soft obstacles in periodic configuration with the blue crosses (× , see Subsection 4.2) and for twelve hard obstacles in irregular configuration with the green asterisks (* , see Subsection 4.3).

Remark 2. *Comparable to our approach for the direct problem is the generalized Foldy-Lax model used in [8] and the references therein. However a combination of single-layer and double-layer potentials, called 'Combined Field Integral Equation' (CFIE) [8, Eqn 2.13] was used. This has the advantage of guaranteeing invertibility at all wavenumbers. As noted in Remark 1, we only work in the interval* $0.08 \leq \kappa \leq 3$*, and do not incur problem with instability. Since we can get away with a simpler operator (less effort with matrix construction and storage, simpler integral equations) with the same behavior in invertibility, we decide to go with this option. Note that inversion method in [8] is qualitative.*

Remark 3. *In dealing with small scatterers or homogeneities, a reductive model is usually employed by asymptotic expansion (see e.g. [4]), or the interaction between the obstacles is approximated by single-scattering (also called Born approximation), or the scatterers are considered point-like (e.g. in Foldy-Lax model [39]). In contrast, FSSL imposes no reductive assumption; it takes into consideration all levels of interaction between the scatterers, and treats both small and extended scatterers, cf. Remark 1.*

Remark 4. *It was shown in [14, 13] that compared with highly-optimized software using Finite Element methods (FEM), a 'naive' implementation of FSSL is still drastically faster*[5]*, e.g. see [14, Experiment 3] in which it is 2000 times faster. However, it is noted that we are in a very special case*

[5]The current implementation of FSSL does not employ any technique of fast solver, in the sense of FMM [32], FMPS [33] and PVFMM [50]. However, they will be necessary for the case of general-shaped obstacles.

for FSSL: we have analytic expressions for the linear system, which is constructed and solved, using intrinsic operations in `BLAS3` *with* `Lapack` *([5]), or* `Mumps` *([1]). A crude upper bound for a basic LU-factorization (without further optimization or speed up) is* $\mathsf{O}(N_{Obs}^3)$ *(the size of the linear system is given in* (11)*, we focus here on the complexity with respect to the number of obstacles).*

The discrete forward map Φ Consider a set of receivers located at $\mathsf{R}_1, \ldots, \mathsf{R}_{N_{\rm rec}} \in \mathbb{R}^2$. Denote by $\mathcal{R}$ the linear evaluation (restriction) of a function to this set. For a given incident angle and configuration $\boldsymbol{p}$, we define the forward map Φ which gives the simulated data, which is the field scattered by the configuration, recorded at receivers positioned at $\mathsf{R}_1, \ldots, \mathsf{R}_{N_{\rm rec}} \in \mathbb{R}^2$,

$$
\begin{array}{rccl}
\Phi & : & \boldsymbol{\mathcal{P}} = \mathbb{R}^{N_{\rm par}} & \longrightarrow \quad \boldsymbol{\mathcal{D}} = \mathbb{C}^{N_{\rm rec}} \\
 & & \boldsymbol{p} & \mapsto \quad \Phi(\boldsymbol{p}) := \mathcal{R}\, u_h = \left(\displaystyle\sum_{J=1}^{N_{\rm Obs}} \sum_{l \in \mathbb{Z}} v_{J,l} \left(\mathcal{S}_J \mathbf{w}_{J,l} \right)(\mathsf{R}_k) \right)_{1 \le k \le N_{\rm rec}} .
\end{array}
\tag{17}
$$

It is convenient to write

$$
\Phi(\boldsymbol{p}) \;=\; \mathfrak{R}(\boldsymbol{p})\, \mathsf{S}(\boldsymbol{p}) \,, \tag{18}
$$

where $\mathfrak{R}(\cdot)$, called the evaluation matrix, is of size $N_{\rm rec} \times N$ with row indexed by k with $1 \le k \le N_{\rm rec}$ and the columns indexed by (J,l) with $1 \le J \le N_{\rm Obs}$, $-\mathbf{m} \le l \le \mathbf{m}$. Its component at row k and column (J,l) is given by

$$
\Big(\mathfrak{R}(\boldsymbol{p}) \Big)_{k,(J,l)} \;=\; \left(\mathcal{S}_J\, \mathbf{w}_{J,l} \right)(\mathsf{R}_k). \tag{19}
$$

When there are several angles of incidence, we write $\Phi_{\rm t}$ the forward map corresponding to planewave $u_{\rm pw}^{(\rm t)}$.

Discrete inverse problem Denote by $\mathbf{d}_{\rm t}$ the observed data corresponding to planewave $u_{\rm pw}^{(\rm t)}$, $1 \le {\rm t} \le N_{\rm Acq}$. In FWI approach, the discrete inverse problem is written as an unconstrained optimization problem

$$
\min_{\boldsymbol{p} \,\in\, \boldsymbol{\mathcal{P}}} \; \mathcal{J}(\boldsymbol{p}) \tag{20}
$$

of the reduced cost function $\mathcal{J}$, which represents the difference between observed data and simulated data over all angles of incidence,

$$
\mathcal{J}(\boldsymbol{p}) \;:=\; \frac{1}{2} \sum_{\rm t=1}^{N_{\rm Acq}} \; \Big\| \, \Phi_{\rm t}(\boldsymbol{p}) \;-\; \mathbf{d}_{\rm t} \, \Big\|_{\boldsymbol{\mathcal{D}}}^2 . \tag{21}
$$

3 Optimization method

3.1 Calculation of the gradient by adjoint-state method

We will use a gradient-based optimization to correct the initial guess to the true models. The main ingredient of these first-order methods is the Fréchet derivatives of the cost function $\mathcal{J}$ with respect to the parameters we wish to retrieve, which in this case is the position of the obstacles $\boldsymbol{p}$. Variable $\boldsymbol{p}$ is real, hence we are only requiring real-Fréchet differentiability of the forward map $\Phi_{\rm t}$ and $\mathcal{J}$. If they exist (under sufficient regularity assumption), $\partial_{\boldsymbol{p}} \Phi_{\rm t}$ is identified with a matrix of size $N_{\rm rec} \times N_{\rm par}$ and $\partial_{\boldsymbol{p}}\, \mathcal{J}$, also written $\mathcal{J}'(\boldsymbol{p})$, a row vector of size $N_{\rm par}$. The differentiability of $\mathcal{J}$ is related to that of $\Phi_{\rm t}$ via, see *e.g.* [55, Prop. 1],

$$
\partial_{\boldsymbol{p}}\, \mathcal{J}(\boldsymbol{p}) \;=\; \sum_{\rm t=1}^{N_{\rm Acq}} \, {\rm Re} \Big((\Phi_{\rm t} - \mathbf{d}_{\rm t})^{\star} \; \partial_{\boldsymbol{p}} \Phi_{\rm t} \Big) . \tag{22}
$$

The differentiability of Φ_t depends on that of the incident wave (*i.e.* the right-hand side of (10)), the scattered field and the multiple-scattering matrix, *cf.* (39) and (40), the latter two quantities depend on the differentiability of the single-layer potential. For arbitrarily-shaped obstacles, techniques in [56] can be adapted to show the Fréchet differentiability of the single-layer potential with respect to the position of obstacles[6].

Remark 5. *For circular obstacles, the explicit multipole expansions can be used to show differentiability and give explicit expressions for the derivatives*[7]*, cf. [55, App. B]. In FSSL, the parameters to be reconstructed, $\boldsymbol{p}$, appear explicitly in the quantities that need to be differentiated, in particular, the coefficient matrix $\mathbf{A}$, the evaluation matrix $\mathfrak{R}$, and the right-hand side F. This is another advantage (in addition to reducing the computational cost) of FSSL compared to discretization methods like finite elements or finite differences.*

Identity (22) shows that $\mathcal{J}'$ is the action of the Jacobian $\partial_{\boldsymbol{p}}\Phi_t$ on the vector $\Phi_t(\boldsymbol{p}) - \mathbf{d}_t$. Since this is the only needed information, we calculate this action using the adjoint-state method, and avoid computing the whole Jacobian (a method called 'the sensitivity approach')[8]. The adjoint-state method was originally developed in the context of control theory by [49], and was used for the computation of a gradient with respect to a parameter in [23]. For general discussion of the adjoint-state method, see [41, 24]. In literature, all formulations of the adjoint-state method in FWI work with volume discretization, *e.g.* [30, 18]. In addition to the advantage discussed in Remark 5, FSSL, typical of integral equation methods, allows a simple implementation in FWI without the need of domain truncation. In our case, the main idea of adjoint-state method is the derivation of (25) which is then used, instead of (22), to calculate $\mathcal{J}'$. The details of this derivation are presented from two perspectives, the 'traditional' one in A.1 and another one using a Lagrangian in A.2. To finish the discussion, we summarize the procedure and formulas used for the calculation.

For a fixed set of parameter p, to calculate $\mathcal{J}'(\boldsymbol{p})$, we use the following steps (the complexity of which is addressed afterwards).

Step 1: We solve for $V_t = \mathsf{S}_t(\boldsymbol{p})$, solution to the direct problem defined in (10), now with multiple right-hand-sides,

$$\mathbf{A}(\boldsymbol{p})\big[V_1 \quad \dots \quad V_{N_{\rm Acq}}\big] = \big[F(\boldsymbol{p}, u_{\rm pw}^{(1)}) \quad \dots \quad F(\boldsymbol{p}, u_{\rm pw}^{(N_{\rm Acq})})\big]. \tag{23}$$

Recall that matrix $\mathbf{A}$ is given by (14), and the right-hand side F corresponding to one angle of incidence in (16). This step also provides the value of $\mathcal{J}$.

Step 2: We then solve the adjoint equation, also with multiple right-hand-sides,

$$\mathbf{A}(\boldsymbol{p})^\star \Big[\boldsymbol{\gamma}_1(\boldsymbol{p}) \quad \dots \quad \boldsymbol{\gamma}_{N_{\rm Acq}}(\boldsymbol{p})\Big] \;=\; -\mathfrak{R}^\star(\boldsymbol{p}) \,\big[\big(\Phi_1(\boldsymbol{p}) - \mathbf{d}_1\big) \quad \dots \quad \big(\Phi_{N_{\rm Acq}}(\boldsymbol{p}) - \mathbf{d}_{N_{\rm Acq}}\big)\big]\,, \tag{24}$$

for the adjoint-states $\boldsymbol{\gamma}_t(\boldsymbol{p})$, which is a vector of size $N = (2\mathbf{m}+1)\times N_{\rm Obs}$. The matrix $\mathfrak{R}(\boldsymbol{p})$ is defined in (19).

Step 3: The derivative of the reduced cost function $\mathcal{J}$ at p is given by , *cf.* Proposition 1

$$\mathcal{J}'(\boldsymbol{p}) = \sum_{t=1}^{N_{\rm Acq}} \mathrm{Re}\,\Big[\big(\Phi_t(\boldsymbol{p}) - \mathbf{d}_t\big)^\star \partial_{\boldsymbol{p}}\mathfrak{R} \underset{(J,l)}{\cdot} \mathsf{S}_t(\boldsymbol{p}) + (\boldsymbol{\gamma}_t)^\star\big(\partial_{\boldsymbol{p}}\mathbf{A} \underset{(J,l)}{\cdot} \mathsf{S}_t(\boldsymbol{p}) - \partial_{\boldsymbol{p}} F(\boldsymbol{p}, u_{\rm pw}^{(t)})\big)\Big]. \tag{25}$$

In expression (25), the derivative $\partial_{\boldsymbol{p}}F$ is a matrix (a tensor of order 2) of size $N \times N_{\rm par}$. Matrix $\mathfrak{R}$ is of size $N_{\rm rec} \times N$. The Fréchet derivatives $\partial_{\boldsymbol{p}}\mathfrak{R}$ and $\partial_{\boldsymbol{p}}\mathbf{A}$ are tensors of order 3, their sizes

[6] In [56], the differentiability of Φ_t and the potentials are shown with respect to the boundary of the obstacles. Other references for the differentiability of Φ_t with respect to the boundary of the obstacles with the range of Φ_t being far-field patterns, are given in [37, 38, 44, 45].

[7] This task is further simplified in the discretized problem, since the multi-pole expansions involved comprise of a finite number of terms.

[8] The Jacobian $\partial_p\Phi$ is also needed, if one uses Newton's method to solve the inverse problem which is viewed as solving the nonlinear equation $\Phi(p) = \mathbf{d}$, *cf.* [16, 15, 9, 44]. In this approach, the linear problem is first linearized as $\Phi(\boldsymbol{p}_{\text{initial guess}}) + \partial_p\Phi'(\boldsymbol{p}_{\text{initial guess}})\delta_{\boldsymbol{p}} = \mathbf{d}$ and one solves for $\delta_{\boldsymbol{p}}$.

are $N_{\rm rec} \times N \times N_{\rm par}$ and $N \times N \times N_{\rm par}$ respectively. The notation $\underset{(J,l)}{\cdot}$, borrowed from tensor calculus, denotes the contraction over the (second) index labeled by the tuples (J,l). The quantity $\partial_{\boldsymbol{p}}\mathfrak{R} \underset{(J,l)}{\cdot} \mathsf{S}_{\rm t}(\boldsymbol{p})$ is then a matrix of size $N_{\rm rec} \times N_{\rm par}$, and $\partial_{\boldsymbol{p}}\mathbf{A} \underset{(J,l)}{\cdot} \mathsf{S}_{\rm t}(\boldsymbol{p})$ of size $N \times N_{\rm par}$.

We next describe how each summand in (25) is calculated for each incidence angle.

- To calculate the first term in (25), we need the difference in data given in Step 1 and the action of matrix $\partial_{\boldsymbol{p}}\mathfrak{R} \underset{(J,l)}{\cdot} \mathsf{S}_t(\boldsymbol{p})$, whose k-th row is given by the following expression evaluated at $V_t = \mathsf{S}_t(\boldsymbol{p})$.

$$\Big(\sum_{-\mathbf{m}\le m\le \mathbf{m}} \frac{\mathrm{i}\pi\,\mathsf{r}_I}{2}\,\mathrm{J}_m(\kappa\,\mathsf{r}_I) e^{\mathrm{i}\,m\,\theta_I(\mathsf{R}_k)}\,V_{Im} \Big[-\kappa_e\,\mathrm{H}^{(1)\prime}_m\big(\kappa\,r_I(\mathsf{R}_k)\big) \begin{pmatrix}\cos\theta_I(\mathsf{R}_k)\\ \sin\theta_I(\mathsf{R}_k)\end{pmatrix}^t + \frac{\mathrm{i}\,m\,\mathrm{H}^{(1)}_m\big(\kappa\,r_I(\mathsf{R}_k)\big)}{\|\mathsf{R}_k - \mathbf{x}_I\|} \begin{pmatrix}\sin\theta_I(\mathsf{R}_k)\\ -\cos\theta_I(\mathsf{R}_k)\end{pmatrix}^t \Big] \Big)_{1\le I\le N_{\rm Obs}}, \tag{26}$$

  The above formula comes from the Fréchet derivatives of the single layer, an explicit calculation of which is given in [55, Eqn (66) App. B1].

- The calculation of the second term in (25) uses the explicit calculations of $\partial_{\boldsymbol{p}}\mathbf{A}$ and $\partial_{\boldsymbol{p}}F$, *cf.* [55, App. B1 and B3]. Below, we list the final formula for this term; its I-th block, denoted by $(2I-1:2I)$, is given by the following expression evaluated at $W = \overline{\gamma_{\rm t}}$ and $V_t = \mathsf{S}_t(\boldsymbol{p})$.

$$-(W^t\,\partial_{\boldsymbol{p}}F)(2I{-}1:2I) \;=\; 2\,\kappa\,u^{(\rm t)}_{\rm pw}(\mathbf{x}^{(I)}) \begin{pmatrix}\cos\alpha_{\mathbf{inc}}\\ \sin\alpha_{\mathbf{inc}}\end{pmatrix} \sum_{l=-\mathbf{m}}^{\mathbf{m}} W_{Il}\,\mathrm{i}^{l+1}\,e^{-\mathrm{i}\,l\,\alpha_{\mathbf{inc}}} \times \begin{cases}\mathrm{J}_l(\kappa\,\mathsf{r}_I) & ,\text{soft-scattering}\\ \kappa\,\mathrm{J}'_l(\kappa\,\mathsf{r}_I) & ,\text{hard-scattering}\end{cases}.$$

$$\begin{aligned}
&(W^t\,\partial_{\boldsymbol{p}}\,\mathbf{A} \underset{(J,l)}{\cdot} V)(2I-1:2I) \;=\; \big(\partial_{\boldsymbol{p}}\,W^t\mathbf{A}(\boldsymbol{p})\,V\big)(2I-1:2I)\\
&= \sum_{J=1}^{N_{\rm Obs}} \Big[\sum_{l,m=-\mathbf{m}}^{\mathbf{m}} W_{Il}\,\boldsymbol{\alpha}_{Il}\,\mathcal{M}^{(1)}_{IJ,lm}\,\boldsymbol{\beta}_{Jm}\,V_{Jm} \;+\; W_{Jm}\,\boldsymbol{\alpha}_{Jm}\,\mathcal{M}^{(2)}_{IJ;lm}\,\boldsymbol{\beta}_{Il}\,V_{Il} \Big] \begin{pmatrix}\cos\theta_{JI}\\ \sin\theta_{JI}\end{pmatrix}^t\\
&+ \sum_{J=1}^{N_{\rm Obs}} \Big[\sum_{l,m=-\mathbf{m}}^{\mathbf{m}} W_{Il}\,\boldsymbol{\alpha}_{Il}\,\mathcal{M}^{(3)}_{IJ,lm}\,\boldsymbol{\beta}_{Jm}\,V_{Jm} \;+\; W_{Jm}\,\boldsymbol{\alpha}_{Jm}\,\mathcal{M}^{(4)}_{IJ,lm}\,\boldsymbol{\beta}_{Il}\,V_{Il} \Big] \begin{pmatrix}-\sin\theta_{JI}\\ \cos\theta_{JI}\end{pmatrix}^t.
\end{aligned}$$

  The block matrices $\mathcal{M}^{(1)}_{IJ}$, $\mathcal{M}^{(2)}_{IJ}$, $\mathcal{M}^{(3)}_{IJ}$, $\mathcal{M}^{(4)}_{IJ}$, are defined as follows: for $I \neq J$,

$$\begin{aligned}
&\mathcal{M}^{(1)}_{IJ,lm} \;=\; e^{\mathrm{i}\,(m-l)\,\theta_{JI}}\,\kappa\,\mathrm{H}^{(1)\prime}_{m-l}(\kappa\,\mathsf{d}_{IJ})\;; \qquad \mathcal{M}^{(2)}_{IJ,lm} \;=\; e^{\mathrm{i}\,(l-m)\,\theta_{JI}}\,\kappa\,\mathrm{H}^{(1)\prime}_{m-l}(\kappa\,\mathsf{d}_{IJ})\;;\\
&\mathcal{M}^{(3)}_{IJ,lm} \;=\; \mathrm{i}\,e^{\mathrm{i}\,(m-l)\,\theta_{JI}}\,\frac{(m-l)\,\mathrm{H}^{(1)}_{m-l}(\kappa\,\mathsf{d}_{IJ})}{\mathsf{d}_{IJ}}\;; \quad \mathcal{M}^{(4)}_{IJ,lm} \;=\; \mathrm{i}\,e^{\mathrm{i}\,(l-m)\,\theta_{JI}}\,\frac{(l-m)\,\mathrm{H}^{(1)}_{m-l}(\kappa\,\mathsf{d}_{IJ})}{\mathsf{d}_{IJ}};
\end{aligned} \tag{27}$$

  for $I = J$, $\mathcal{M}^{(1)}_{IJ} = \mathcal{M}^{(2)}_{IJ} = \mathcal{M}^{(3)}_{IJ} = \mathcal{M}^{(4)}_{IJ} = \mathbf{0}_{(2\mathbf{m}+1)\times(2\mathbf{m}+1)}$. Matrix $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ are given by

$$\boldsymbol{\alpha}_{Il} \;=\; \begin{cases}\mathrm{J}_l(\kappa\,\mathsf{r}_I) & \text{soft-scattering}\\ \kappa\,\mathrm{J}'_l(\kappa\,\mathsf{r}_I) & \text{hard-scattering}\end{cases} \quad;\quad \boldsymbol{\beta}_{Jm} \;=\; \mathrm{i}\,\pi\,\mathsf{r}_J\,\mathrm{J}_m(\kappa\,\mathsf{r}_J)\,. \tag{28}$$

Complexities for the cost function and its gradient The complexities are stated in terms of $N_{\rm Obs}$ the number of obstacles, with other variables considered bounded or constant, $N_{\rm rec} = 128$, $N_{\rm Acq} \le 4$, and $\mathbf{m} \le 6$. We assume only a brute-force implementation of FSSL, see Remark 4.

- The complexities of $\mathcal{J}$, denoted by $\mathrm{work}_{\mathcal{J}}$, comes from the resolution of the forward linear system (23) whose main cost resides in the factorization of the scattering matrix $\mathbf{A}$. This linear system has to be solved for each angle of incidence, however, the use of direct solver (*e.g.* `Mumps` or `Lapack`) allows multiple right-hand sides resolution, at low cost, *i.e.* a cost that is of smaller order compared to that of the factorization[9]. From Remark 4, an upper bound of $\mathrm{work}_{\mathcal{J}}$ is $\mathsf{O}(N^3_{\rm Obs})$.

[9] For a specific example of CPU time, see [14, Table 1– 2] which shows that the construction time and resolution time for the matrix is less than that needed for its factorization.

- Denote by $\text{work}_{\nabla\mathcal{J}}$ the complexities for the gradient. Using direct solvers, the adjoint problem (24) can reuse the LU factorization of the forward problem. In this way, the extra cost for $\nabla\mathcal{J}$ only includes back substitution for the adjoint state and matrix-vector products in (25), and $\text{work}_{\nabla\mathcal{J}} = \mathsf{O}(N_{\text{Obs}}^2)$. This highlights the benefits of direct solvers, compared to iterative solvers, in solving the forward and adjoint problems (23)–(24).

These advantages of direct solvers are critical in frequency domain seismic inversion (due to the high computational cost, *e.g.* [54, 28]) and are also exploited in [16, 15].

3.2 Frequency-hopping inversion procedure

In this section, we describe the algorithm used to correct a prescribed initial guess to the true model. The overall process, called frequency-hopping, uses multi-frequency data and carries out gradient-based optimization procedure described in Figure 2 at each prescribed frequency called a 'run'. We first proceed with the lowest frequency and then proceed consecutively to higher ones, in a process called 'low-to-high frequency progression'. At the lowest frequency, the algorithm uses (user-)prescribed initial guess; at successive frequencies, initial guesses are the reconstructed parameters given by the algorithm at the previous frequency. For the twenty-four obstacles, after some frequency progression, we employ a recycling of frequency, because the fixed number of receivers prevents us from using arbitrarily high frequencies, *cf.* Remark 8.

Reconstruction at lower frequencies allows the initial guesses, although far from the true models both in terms of position and nature, to converge to the vicinity of the true position; higher frequencies are essential for taking into account the multiple-scattering effect between the obstacles, and are necessary for convergence to positions hidden (in the middle of) in the configurations. As mentioned in the introduction, low-to-high frequency marching is a common technique is seismic inversion, *e.g.* [19, 58, 61]. It is also the principle in Recursive linearization Algorithm [25], see discussions in [43, 60, 16]. Multifrequencies data are also used in [34, 11, 17] for location of homogeneities, [7, 10, 15] for medium reconstruction, [16] for shape reconstruction.

Within a gradient-based procedure at a fixed frequency, *cf.* Figure 2, each iteration to update or correct a guess uses line search-based strategy. To go from a current iterate $\boldsymbol{p}_k$ to the next value $\boldsymbol{p}_{k+1}$ at which the (cost) function $\mathcal{J}$ would have a lower value, one needs a search direction $\mathbf{s}_k$ (usually chosen to be descent *i.e.* $\mathbf{s}_k^t \nabla\mathcal{J}(\boldsymbol{p}_k) < 0$), and a step length α_k in this direction, to update $\boldsymbol{p}_{k+1} = \boldsymbol{p}_k + \alpha_k\,\mathbf{s}_k$. To find α_k, one can solve exactly the one-dimensional minimization problem $\min_{\alpha>0} \mathcal{J}(\boldsymbol{p}_k + \alpha\mathbf{s}_k)$, but this can be expensive. A line search algorithm is employed instead, to generate a limited number of trial step lengths until a candidate, which approximates the minimum by satisfying decrease criteria, is found. We investigate three line search algorithms, differing from one another by decrease criteria and how search intervals and trial step lengths are updated. The line search algorithms are based on the optimization theory in [53], however their implementation can vary with different choices of initial step length, interval update and stop criteria. We briefly describe their main features and refer to [55, Subsection 4.1] for more details on their actual implementation in our codes. To describe the decrease criteria, we introduce the following notations,

$$\psi_k(\alpha) := \mathcal{J}(\boldsymbol{p}_k + \alpha\,\mathbf{s}_k)\ ;\ \psi_k(0) := \mathcal{J}(\boldsymbol{p}_k)\ ;\ \psi_k'(0) = \mathcal{J}'(\boldsymbol{p}_k)\,\mathbf{s}_k. \tag{29}$$

- The most expensive of the three algorithms presented here, is the line search algorithm LS3, *cf.* [55, Algo 3–4], which requires trial step lengths to satisfy the strong Wolfe conditions comprising of

$$\begin{aligned} \psi(\alpha) \;&\le\; \psi(0) \;+\; \mu\;\alpha\;\psi'(0)\;, \quad \mu \in (0,1) && \text{Armijo condition;} \\ |\psi'(\alpha)| \;&\le\; \eta\;|\psi'(0)|\;, \quad 0 < \mu \le \eta && \text{Curvature condition.} \end{aligned} \tag{30}$$

  The first condition, also called the sufficient decrease condition, controls the decrease in $\mathcal{J}$, while the second one prevents trial step lengths α from being too small. For the interval updates in LS3, see detailed description in [55, Algo 3,4].

- As an intermediate, line search algorithm LS2, *cf.* [55, Algo 2], imposes sufficient decrease (Armijo condition) and second-order quadrature (minimizer of a quadratic) for update.

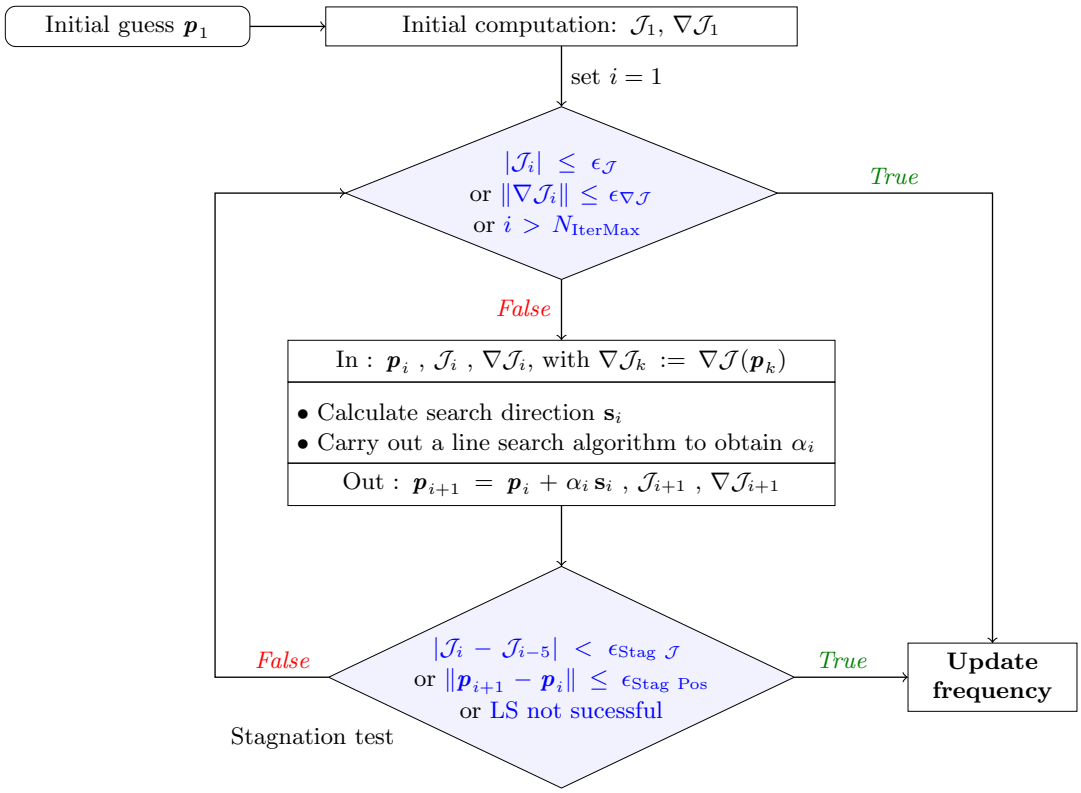


Figure 2: Optimization algorithm at each fixed frequency. At the lowest frequency, the algorithm uses prescribed initial guess; at later frequencies, initial guesses are reconstructed parameters given by the algorithm at previous one. The values of $\mathcal{J}$ and its gradient $\nabla\mathcal{J}$ are obtained by the adjoint-state method, using Step 1–3 in Subsection 3.1. To skip to the next frequency or to stop the overall procedure, the following criteria are employed. Parameters $\epsilon_{\text{critical point}}$ and $\epsilon_{\text{err tol}}$ control the smallness of $\nabla\mathcal{J}$ and $\mathcal{J}$. Stagnation of the algorithm is controlled by $\epsilon_{\text{Stag Steplength}}$ (which signals that a new step length is too close to the previous one), $\epsilon_{\text{Stag Pos}}$ (new configuration is too close to the current one), and $\epsilon_{\text{Stag }\mathcal{J}}$ (the algorithm does not reduce the value of the cost function). The number of outer iterations is controlled by N_{IterMax}, while those in a line search are by $N_{\text{LS IterMax 1}}$ and $N_{\text{LS IterMax 2}}$ (the latter is only used for LS3). Other criteria are whether a line search is successful or not, and the number of frequencies prescribed by user.

- The 'cheapest' line search algorithm is LS1, *cf.* [55, Algo 1], which imposes no control over how much $\mathcal{J}$ decreases and requires only $\psi(\alpha) < \psi(0)$. This line search (LS1) uses reduction by half to update trial step lengths.

In combination with the above line searches, we investigate the effectiveness of two search directions:

- Broyen-Fletcher-Goldfarb-Shannon (BFGS), *cf.* [53, p. 140], in the quasi-Newton family (that we further refer to as SD1);
- Polak-Ribière (PR) with restart (referred to as SD2), *cf.* [53, Eqn (5.44), (5.45) p.122-123], in the nonlinear-conjugate gradient (NLCG) family[10].

For general motivation and history of NLCG, we refer to [36], and of quasi-Newton to [26]. Theses search directions are chosen so that as many obstacles as possible can be retrieved while using only

[10]The restart, in which the direction is set to Steepest Descent direction, is needed, since the defined PR search direction ([53, Eqn (5.44) p.122]) is not guaranteed to be a descent direction. On the other hand, BFGS implemented with strong Wolfe line search algorithm (LS3) guarantees descent direction. The initial approximate for the Hessian used in BFG follows [53, Eqn 6.20].

first-order information (*i.e.* the gradient of the cost function), and allowing the initial guesses to be far from the true models.

NLCG and the simpler version, Steepest Descent, are popular in seismic inversion since there is no need for matrix storage, a useful feature for problems which aim to retrieve a large number of parameters. Gradient-based optimization using variants in the NLCG family, have been investigated in seismic inversion, *e.g.* [28] and in the context of inverse scattering, *e.g.* [65, 42]. An extensive comparison between the variants (in NLCG) is carried out in [67]. These references suggest the effectiveness of PR which is thus chosen in our work among others variants in NLCG. Quasi-Newton is less computational intensive than methods in the Newton-like family used in *e.g.* [47, 44, 46, 42, 16] and the references therein, while still offers good convergence rate of Newton's method *cf.* [47].

Remark 6. *In general, $\mu < \frac{1}{2}$, so that the criteria is satisfied for quadratics, cf. discussion in [51]. In our numerical experiments, $\mu = 0.0001$ and $\eta = 0.9$, following the comment in [53, p.142] that these values are 'commonly used'. We also test with $\eta = 0.4$*[11].

Computational complexities Assume the gradient of cost function is already calculated, the cost of search direction, denoted by work_{sd} is

$$\text{work}_{\text{sd}} = \begin{cases} \mathsf{O}(N_{\text{Obs}}^2) & \text{for BFGS,} \\ \mathsf{O}(N_{\text{Obs}}) & \text{for NLCG-PR.} \end{cases} \tag{31}$$

In simple backtracking LS1 and enhanced backtracking LS2, one needs to evaluate the cost of $\mathcal{J}$ at each candidate steplength, while for strong Wolfe line search LS3, one needs to evaluate cost function and its gradient at each linesearch iteration. Thus the cost of each line search iteration, denoted by work_{ls}, is

$$\text{work}_{\text{ls}} = \begin{cases} \text{work}_{\mathcal{J}} = \mathsf{O}(N_{\text{Obs}}^3), & \text{for LS1 or LS2,} \\ \text{work}_{\mathcal{J}} + \text{work}_{\nabla\mathcal{J}} = \mathsf{O}(N_{\text{Obs}}^3) + \mathsf{O}(N_{\text{Obs}}^2), & \text{for LS3.} \end{cases} \tag{32}$$

For a specific choice of search direction and linesearch algorithm, assuming an inversion procedure of n_{run} runs and $n_{\text{iter}}^{(k)}$ iterations per run k, the work load is given as

$$\begin{aligned} \text{Work} &= \sum_{1\le k\le n_{\text{run}}} \sum_{1\le i\le n_{\text{iter}}^{(k)}} \text{work}_{\mathcal{J}} + \text{work}_{\nabla\mathcal{J}} + \text{work}_{\text{sd}} \ + \ n_{\text{ls}}^{(k,i)} \ \times \ \text{work}_{\text{ls}} \\ &\sim \sum_{1\le k\le n_{\text{run}}} \sum_{1\le i\le n_{\text{iter}}^{(k)}} \mathsf{O}(N_{\text{Obs}}^3)\,. \end{aligned} \tag{33}$$

Here $n_{\text{ls}}^{(k,i)}$ is the number of linesearch iterations within iteration i in run k and is set to be ≤ 60 for all tests. We note that the cost of the matrix factorization dominates that of other computational operations.

4 Numerical experiments

For the experiments, the methods are identified with the keyword *SDx-LSy*, where SD refers to the search direction, x is either 1 (for quasi-Newton) or 2 (for nonlinear conjugate gradient). LS stands for the line search method and y can be 1 (simple backtracking), 2 (sufficient descent) or 3 (strong Wolfe condition), see Section 3. To test the robustness of the reconstruction procedure, numerical experiments are carried out in domains of different sizes, with soft or hard scattering obstacles in structured or irregular configurations. For simplicity, we assume that all of the obstacles are of the same radius $\mathsf{r}_I = \mathsf{r}$. The inversion procedure should still work with obstacles of different size, see also Remark 1. We will work with circular obstacles of radius 0.5. The nature of the configuration (periodic or irregular) is not known a priori, this naturally increases the difficulties of the reconstruction.

[11] This was originally chosen so that Fletcher-Reeves (FR) directions [53, Eqn 5.41] are descent, by satisfying hypothesis $\mu < \eta < \frac{1}{2}$ in strong Wolfe linesearch [53, Lemma 5.6]. Due to space limitation, we do not included tests with FR which was found to perform less well than PR. However, we kept $\eta = 0.4$ as another parameter to see how PR performs with a smaller η.

Remark 7. *In figures illustrating the configurations and reconstructions, e.g. Figure 4, the true positions (centers of the obstacles) are represented by red circles (∘). This is not to be mistaken as a representation of the obstacles themselves, which are actually much smaller.*

Recall that the backscattered data are collected according to the restriction mentioned in the introduction (*i.e.*, from at most four angles of incidence, and 128 receiver points per angle and frequency, see Figure 3.) To avoid the 'inverse crime', synthetic data are produced by FSSL order 12 with solver `Lapack` [5] in a separate set of codes. The solver used for the resolution of the forward problem is `Mumps` [1] at lower order of FSSL (≤ 6). For more realistic experiments, complex Gaussian white noise is added by `Matlab` routine `awgn` with the noise level described by signal-to-noise ratio per sample in decibel (dB). The frequencies range is

$$\kappa_{\min} = 0.08 \leq \kappa \leq \kappa_{\max} = 3 \quad ; \quad 0.04 \leq \kappa \mathsf{r} \leq 1.5. \tag{34}$$

These ranges satisfy the requirement for the invertibility of the linear system (10), *cf.* Remark 1.

The inversion codes are written in `Fortran90`. The experiments are run in sequential on personal laptop. In initial runs for each configuration and initial guess, the parameters are set so that convergence is obtained even with high number of iterations. After convergence is obtained, the results are refined by increasing the error tolerance and stagnation criteria and by removing redundant high frequencies, in a way so that the same convergence results are obtained at lower time-cost. This is done in order to reduce the number of iterations. For reproducibility, the actual parameters for each experiment are listed in [55, App. E], for more comments on the implementation see also [55, Section 5].

The quality of a reconstruction is calculated after the inversion has been carried out (outside of the inversion codes), and is based on: the number of frequencies used, the number of iterations taken and the errors in the misfit $\mathcal{J}$ and in position. The error in position is calculated as the distance between reconstructed model $\boldsymbol{p}_{\text{recon}} = (x_1^{(1)}, x_2^{(1)}, \ldots, x_1^{(N_{\text{Obs}})}, x_2^{(N_{\text{Obs}})})$ and the true model $\boldsymbol{p}_{\text{model}} = (\tilde{x}_1^{(1)}, \tilde{x}_2^{(1)}, \ldots, \tilde{x}_1^{(N_{\text{Obs}})}, \tilde{x}_2^{(N_{\text{Obs}})})$,

$$\text{Err. Pos. } \big(\boldsymbol{p}_{\text{recon}}\big) := \text{dist}\big(\boldsymbol{p}_{\text{recon}}, \boldsymbol{p}_{\text{model}}\big) = \sqrt{\sum_{I=1}^{N_{\text{Obs}}} \text{dist}^2(\boldsymbol{x}^{(I)}, \boldsymbol{p}_{\text{model}})}, \tag{35}$$

with

$$\text{dist}(\boldsymbol{x}^{(I)}, \boldsymbol{p}_{\text{model}}) := \min_{1 \leq J \leq N_{\text{Obs}}} \|\boldsymbol{x}^{(I)}, \tilde{\boldsymbol{x}}^{(J)}\|. \tag{36}$$

The relative error position is the scaled distance with respect to the size of the domain. On a domain $[a, b]_x \times [c, d]_y$,

$$\text{Rel. Err. Pos } (\boldsymbol{p}_{\text{recon}}) \ := \ \frac{\text{Err. Pos. } (\boldsymbol{p}_{\text{recon}})}{\min\{|b-a| \ , \ |d-c|\}} \, . \tag{37}$$

Relative error in $\mathcal{J}$ of a reconstruction is calculated as

$$\text{Rel. Err. } \mathcal{J}(\boldsymbol{p}_{\text{recon}}) \ = \ \frac{\sqrt{2\,\mathcal{J}(\boldsymbol{p}_{\text{recon}})}}{\|\mathbf{d}\|} \qquad \text{at final reconstruction frequency} \, . \tag{38}$$

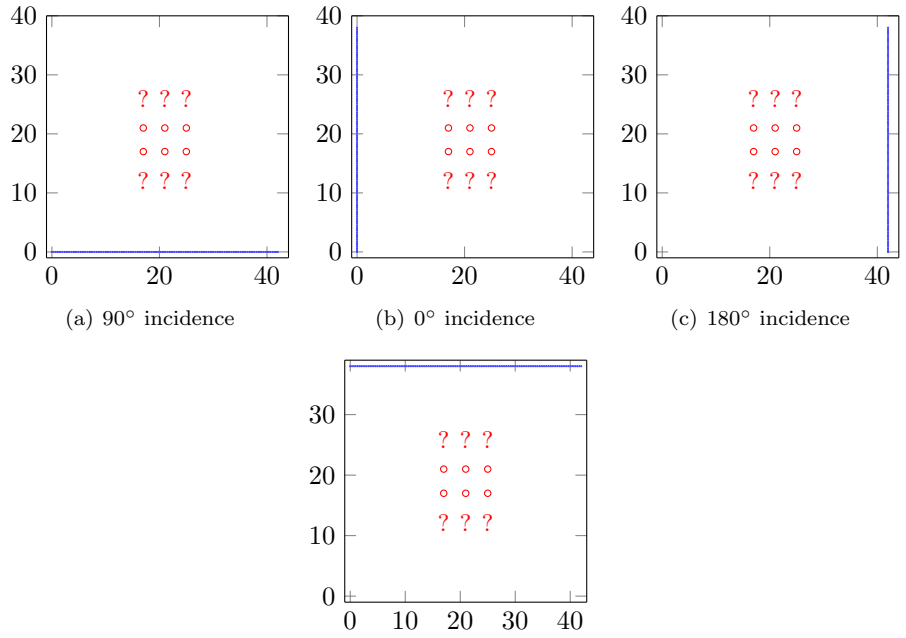
Note that when there is noise, $\mathbf{d}$ is the noisy synthetic data.

4.1 Localization of 6 hard-scattering obstacles

In this experiment, we consider the recovery of the position of six hard-scattering obstacles, their centers are located, inside the domain of interest of size 42×38, at

$$(17,\ 21), \quad (21,\ 21), \quad (25,\ 21), \quad (17,\ 17), \quad (21,\ 17), \quad (25,\ 17).$$

As mentioned, we do not know a priori the periodicity of the true configuration. In Figure 4, we illustrate the obstacles configuration for the domain of interest. We investigate the convergence of all optimization options, first with two specific initial guesses shown Figures 4(a) and 4(b). For more conclusive observations, we then test with 200 initial guesses that are randomly generated.



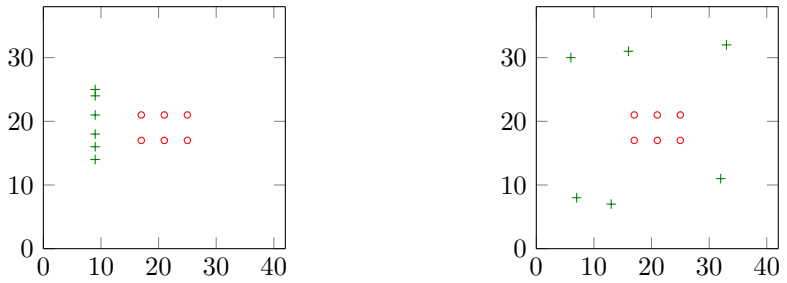
(a) 90° incidence (b) 0° incidence (c) 180° incidence

(d) 270° incidence

Figure 3: For each angle of incidence, synthetic data are collected at 128 receivers (in blue, ·) equally spaced and lying on a line perpendicular to the angle of incidence, on the side of the domain. The goal is to retrieve the positions of the obstacles indicated by the red circles (◦).

4.1.1 Inversion using data with noise with initial guess 1 and 2

For these two guesses, we work with noisy data at 23dB obtained from three angles of acquisition: 90°, 0° and 180°, illustrated in Figure 3. This noise level corresponds to total relative L^2 error (polluted observed data compared to true observed data) between 6% and 8%, in L^∞ norm between 10% and 21%, depending on the angle of incidence and κ. For this reconstruction, we use a set of frequencies: 0.08, 0.09, and from 0.1 to $\kappa_{\max}$ with step 0.1 and $\kappa_{\max} \leq 0.8$. We illustrate the observed data (with and without noise) as well as the data corresponding to both initial guesses at frequencies $\kappa = 0.08$ and $\kappa = 1$ in Figure 5.

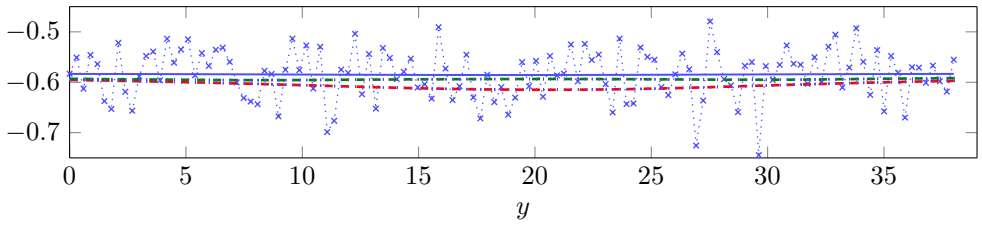


(a) Initial Guess 1 (IG1): Error Position is 20.5, relative Error Position is 54%

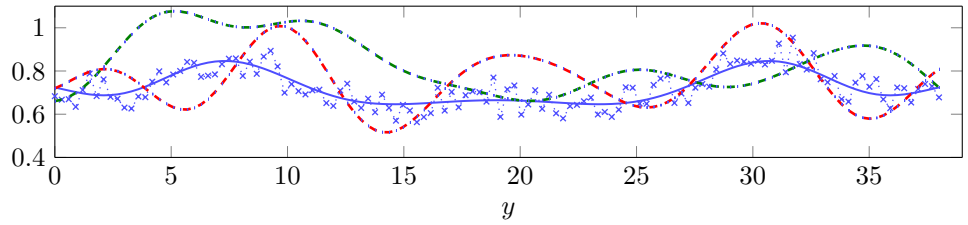
(b) Initial Guess 2 (IG2): Error Position is 29.5, relative Error Position is 77.6%

Figure 4: True positions (red circles, ◦) and initial guesses (green pluses, +) in the reconstruction of 6 hard-scattering obstacles.

We see that low frequencies give poor information about the interaction between the obstacles (showing little oscillation) and that there is a small gap between the synthetic data (both noise-free



(a) Real part of the data for an incidence angle of 0° at frequency $\kappa = 0.08$. Relative error between observed and noisy data is 3% with L^2-norm and 6.5% with L^∞ norm.



(b) Real part of the data for an incidence angle of 0° at frequency $\kappa = 1$. Relative error between observed and noisy data is 7% with L^2-norm and 13.3% with L^∞ norm.

Figure 5: Data obtained at the 128 receivers at different frequencies. The blue line (——) represents the synthetic data, the blue dashed line with crosses (·····) the data incorporating 23dB noise, the red dashed line (- - -) simulated data associated to the initial guess 1 (see Figure 4(a)) and the green dashed line (- - -) simulated data associated to initial guess 2 (see Figure 4(b)).

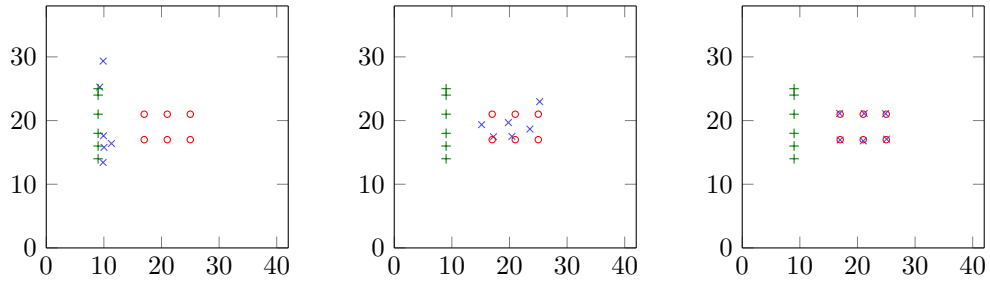
and polluted) and the simulated data of the initial guesses (see Figure 5(a)). This helps explain intuitively how beginning the reconstruction at low frequencies allows the initial guess to be far from the true model and of different nature. At higher frequency, the data collected for the different configurations provide a much larger contrast (see Figure 5(b)).

Results for initial guess 1 For all methods, $\kappa_{\max} = 0.7$ with the exception of SD1-LS2 (Search Direction method 1, Line Search method 2) for which $\kappa_{\max} = 0.6$. In Table 1, we provide the information for the iterative reconstruction and illustrate a reconstruction in Figure 6. Figure 7 shows the evolution of the cost function for five selected frequencies. Figure 8 shows the evolution of the error of the reconstructed position with iterations. We have the following observations.

- Optimization at lower frequencies stagnates faster than at higher frequencies, see Figure 7.
- In this configuration, the two search directions perform similarly well. Although they might take different numbers of iterations to convergence, all methods are able to arrive at a reconstruction of similar accuracy, see Table 1. SD2-LS1 is actually the fastest method here.

Table 1: Results for the 6 obstacles reconstruction using noisy data from three angles of radiation and the initial guess 1 (Figure 4(a)).

Method	♯ freq	$\kappa_{\max}$	Final rel. error $\mathcal{J}$	♯ iterations	Final rel. position error	Run time (s)
SD1-LS1	9	0.7	7.00%	88	0.7%	0.64
SD1-LS2	8	0.6	6.77%	67	1.0%	0.4
SD1-LS3	9	0.7	7.00%	72	0.8%	0.6
SD2-LS1	9	0.7	7.00%	42	1.0%	0.31
SD2-LS2	9	0.7	7.00%	131	0.6%	0.72
SD2-LS3	9	0.7	7.00%	59	0.8%	0.82



(a) Iteration 2, $\kappa = 0.08$; $\mathcal{J} = 1$; position error is 19.9; scaled position error is 52%.

(b) Iteration 21, $\kappa = 0.3$; $\mathcal{J} = 0.9$; position error is 4.3; scaled position error is 11%.

(c) Iteration 42, $\kappa = 0.7$; $\mathcal{J} = 0.8$; position error is 0.4; scaled position error is 1%.

Figure 6: Six obstacles position recovery from noisy data with method SD2-LS1 and starting from initial guess 1 (Figure 4(a)). The true positions of the obstacles are the red circles (∘), initial guesses the green pluses (+) and current reconstruction the blue crosses (×).

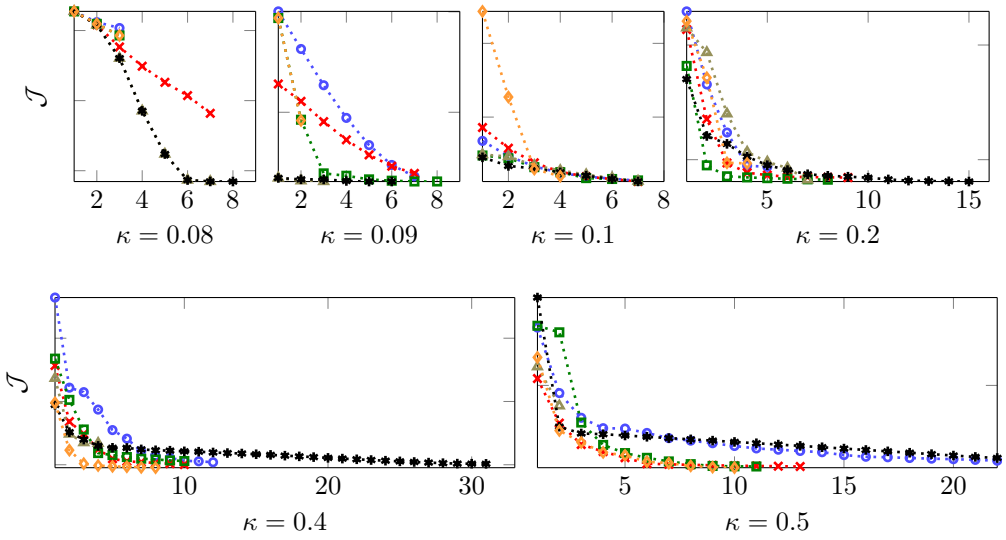


Figure 7: Evolution of the cost function in the reconstruction of 6 obstacles with noisy data of 23dB and starting from the initial guess 1 (Figure 4(a)). The blue circles (···○···) indicate method SD1-LS1; the red crosses (···×···) method SD1-LS2; the green squares (···□···) method SD1-LS3; the yellow triangles (···△···) method SD2-LS1; the black stars (···∗···) method SD2-LS2; the orange diamonds (···◇···) method SD2-LS3. The results of the reconstruction are given Table 1. All methods do not have the same number of iterations per frequency due to the stagnation criteria in the iterative algorithm.

Results for initial guess 2 The convergence results are given in Table 2. In Figure 10, we plot the evolution of the position error of the obstacles with iterations. In Figure 9, we illustrate the accuracy of the reconstruction results of SD2-LS2 method. We have the following observations.

- Overall, while taking different numbers of iterations, all methods are able to provide similar accuracy in the reconstruction, with less than 1% position error. In particular, here, all methods based on SD1 and SD2-L2 perform equally well.
- SD1 (quasi-Newton) seems less affected by starting model. While SD2-LS1 using initial guess 1 performs the fastest, with initial guess 2, the convergence of SD2-LS1 and SD2-LS3 requires twice the number of iterations compared to SD2-LS2. On the other hand, the behavior of the line searches coupled with SD1 remains the same for both initial guesses; in particular, the convergence pattern among SD1-LS1, SD1-LS2 and SD1-LS3 remain the same for both initial guesses.

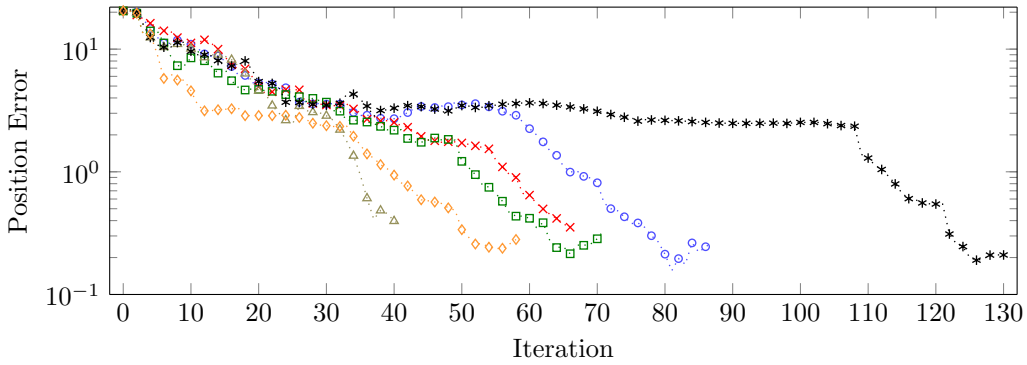
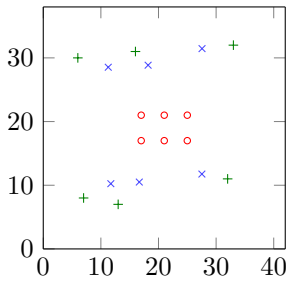


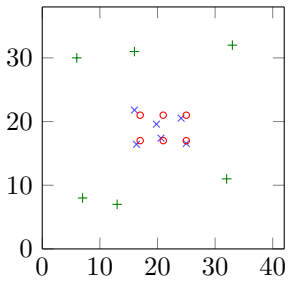
Figure 8: Progression of the position error in the reconstruction of 6 obstacles with noisy data of 23dB and starting from the initial guess 1 (Figure 4(a)). The markers are indicated one iteration over two for clarity. The blue circles (···○···) indicate method SD1-LS1; the red crosses (···×···) method SD1-LS2; the green squares (···□···) method SD1-LS3; the yellow triangles (···△···) method SD2-LS1; the black stars (···∗···) method SD2-LS2; the orange diamonds (···◇···) method SD2-LS3. The results of the reconstruction are given Table 1.

Table 2: Results for the 6 obstacles reconstruction using noisy data from three angles of radiation and the initial guess 2 (Figure 4(b)).

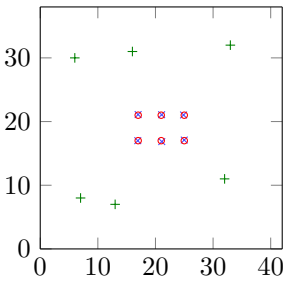
Method	♯ freq	$\kappa_{\max}$	Final rel. error $\mathcal{J}$	♯ iterations	Final rel. position error	Run time (s)
SD1-LS1	9	0.7	7.00%	72	0.8%	0.67
SD1-LS2	8	0.6	6.77%	79	1.0%	0.48
SD1-LS3	8	0.6	6.77%	75	0.8%	0.68
SD2-LS1	10	0.8	6.88%	181	0.7%	2.64
SD2-LS2	9	0.7	7.00%	79	0.75%	0.46
SD2-LS3	8	0.6	6.77%	155	0.73%	1.40



(a) Iteration 2, $\kappa = 0.08$; $\mathcal{J} = 0.96$; position error is 20; scaled position error is 54%.



(b) Iteration 39, $\kappa = 0.4$; $\mathcal{J} = 1$; position error is 2.7; scaled position error is 7%.



(c) Iteration 79, $\kappa = 0.7$; $\mathcal{J} = 0.8$; position error is 0.3; scaled position error is 0.8%.

Figure 9: Six obstacles position recovery from noisy data with method SD2-LS1 and starting from initial guess 2 (Figure 4(b)). The true positions of the obstacles are the red circles (○), initial guesses the green pluses (+) and current reconstruction the blue crosses (×).

4.1.2 Statistics from two hundred different initial guesses

We have seen from previous experiments that the convergence of the optimization methods depends on the initial guess. In order to have more sound observations, we test with 200 different initial guesses randomly generated in the domain. For these tests, we employ all four angles for the data acquisition and the following frequencies: 0.08, 0.09 and from 0.1 to 1 with a step of 0.1. For consistency, the robustness is analyzed for each method keeping the exact same set parameters for all initial guesses

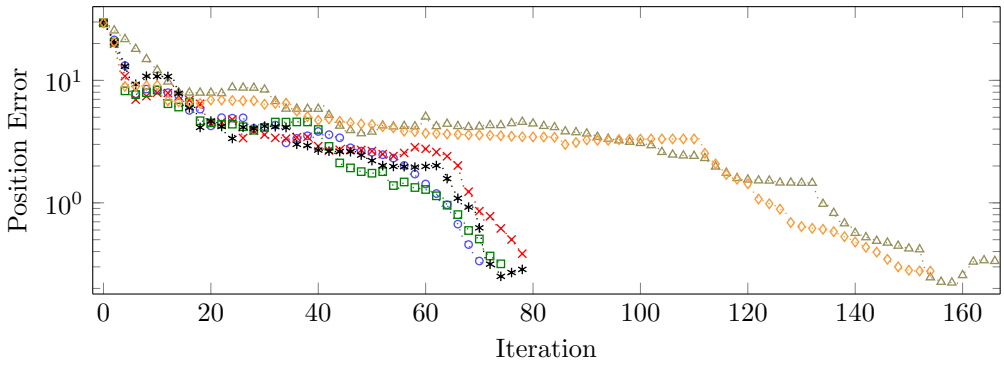


Figure 10: Progression of the position error in the reconstruction of 6 obstacles with noisy data of 23dB and starting from the initial guess 2 (Figure 4(b)). The markers are indicated one iteration over two for clarity. The blue circles (···○···) indicate method SD1-LS1; the red crosses (···×···) method SD1-LS2; the green squares (···□···) method SD1-LS3; the yellow triangles (···△···) method SD2-LS1; the black stars (···∗···) method SD2-LS2; the orange diamonds (···◇···) method SD2-LS3. The results of the reconstruction are given Table 2.

(*i.e.*, maximal number of iterations, initial step length for the line search, parameters μ and η) so that no 'tuning' is performed (which is important for users). Because we do not seek the optimal convergence speed, the stagnation criteria are set low and the maximal number of iterations high: slow convergence is accepted. Because of this, the run time is higher compared to the previous experiments. Table 3 provides the results for all methods, with the average number of iterations, run time and final cost.

Table 3: Results for the 6 obstacles reconstruction using four angles of acquisition for the data and 200 different initial guesses. The average and standard deviation given correspond only with experiments that have successfully recovered the obstacles.

Method	SD1-LS1	SD1-LS2	SD1-LS3	SD2-LS1	SD2-LS2	SD2-LS3
Success rate	91%	83%	99%	85%	75%	92%
Average run time (s)	7.6	2.1	2.0	16.0	4.3	2.8
Standard deviation run time (s)	1.1	0.7	0.5	6.0	1.4	0.6
Average iteration number	221	291	150	417	655	234
Standard deviation iteration number	39	114	31	149	237	45
Average final $\mathcal{J}$	3×10^{-10}	5×10^{-11}	5×10^{-11}	7×10^{-9}	8×10^{-8}	7×10^{-11}
Standard deviation final $\mathcal{J}$	6×10^{-10}	1×10^{-10}	2×10^{-11}	2×10^{-8}	5×10^{-7}	9×10^{-11}

We draw the following observations.

- It seems that the LS option has more influence than the choice of search direction.
- The LS3 option gives the best success rate for both search directions. Namely, LS3 performs better than LS1 which itself performs better than LS2. More precisely, SD1-LS3 is the best method as it always converges to the solution (99% success rate, 92% for SD2-LS3) and takes the smallest run time (2s). SD1-LS1 is also a good candidate with high success rate (91%) but it takes much more time than the methods using LS3 (almost four times more than SD1-LS3). SD1-LS2 and SD2-LS1 performs well (83% and 85% of success rate) but SD2-LS1 takes the largest time of all methods (with an average of 16 s). Finally, SD2-LS2, performs the worst with only 75% of success. Note that it is possible to obtain the convergence for all methods and initial guesses by increasing the number of frequency used.

- The simplest method to implement (LS1) is actually the more time consuming in applications.
- Quasi-Newton (SD1) performs better than nonlinear conjugate gradient (SD2).

In short, the most robust and effective methods in this configuration are SD1-LS3, and SD2-LS3 slightly behind. For Quasi-Newton, the efficiency of using Wolfe condition (*i.e.* SD1-LS3) over other methods (SD1-LS1 and SD1-LS2) can actually be anticipated, as indicated in [53, p.142].

4.2 Localization of 12 soft-scattering obstacles

In this subsection, we consider a structured configuration of twelve soft-scattering obstacles centered at

$$(22,\ 32),\quad (26,\ 32),\quad (30,\ 32),\quad (34,\ 32),\quad (22,\ 36),\quad (26,\ 36),$$
$$(30,\ 36),\quad (34,\ 36),\quad (22,\ 40),\quad (26,\ 40),\quad (30,\ 40),\quad (34,\ 40),$$

inside a domain of size 56×72. We use four angles of radiation for the data, sending planewaves from all four sides of the domain. In Figure 11, we show the initial guess taken to start the reconstruction algorithm.

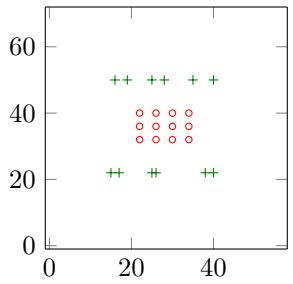


Figure 11: True positions (red circles, ◦) and initial guesses (green pluses, +) in the reconstruction of 12 soft-scattering obstacles.

We incorporate noise in the generated data, using a signal-to-noise ratio of 30dB. It leads to a relative L^2-norm error between 2.8% and 3.5% and L^∞-norm between 4.1% and 8.1% compared to the synthetic, noise-free data. We restrict ourselves to two methods for the reconstruction: quasi-Newton search direction using strong Wolfe condition for line search (SD1-LS3) and nonlinear conjugate gradient search direction, also using strong Wolfe condition for line search (SD2-LS3). These are the most robust combination as shown in previous experiments. Both methods use the same line search parameters with $\mu = 1.0 \times 10^{-4}$ and $\eta = 0.4$ (see (30)). The frequency varies from $\kappa = 0.09$ to 0.7.

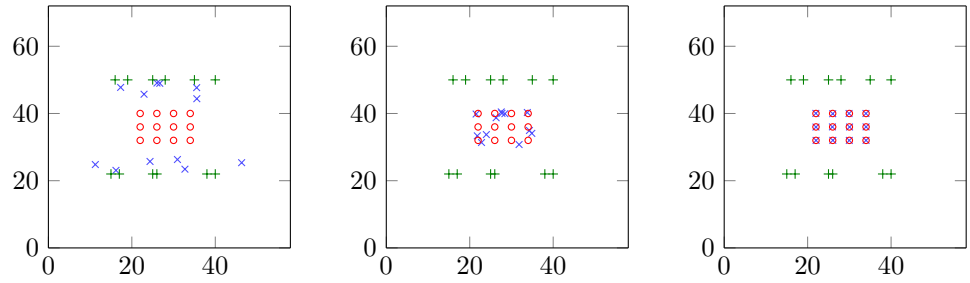
The information for the iterative algorithm are given in Table 4. In Figure 12, we show the obstacle position recovery at three iterations, for the SD1-LS3 method. The progression of the cost function with iterations is plotted in Figure 13 and of the position error with iterations in Figure 14.

Table 4: Results for the 12 obstacles reconstruction using noisy data starting from initial guess Figure 11. The number of matrix factorizations takes into account the ones realized during the line search algorithm.

Method	♯ freq	$\kappa_{\max}$	Final rel. error $\mathcal{J}$	♯ iterations	Final rel. position error	Run time (s)	♯ facto.
SD1-LS3	8	0.7	3.18%	423	0.3%	11	1167
SD2-LS3	8	0.7	3.18%	401	0.4%	11	902

One can observe the following,

- the convergence for low frequencies is more rapid with method SD2-LS3, while at higher frequency, it is faster with SD1. However, the slow convergence of SD1-LS3 at low frequencies is compensated by the rapid convergence at higher frequencies.



(a) Iteration 2, $\kappa = 0.1$; $\mathcal{J} = 100$; position error is 31; scaled position error is 56%.

(b) Iteration 60, $\kappa = 0.2$; $\mathcal{J} = 0.25$; position error is 5.5 ; scaled position error is 10%.

(c) Iteration 424, $\kappa = 0.7$; $\mathcal{J} = 0.25$; position error is 0.15 ; scaled position error is 0.25%.

Figure 12: Twelve obstacles position recovery from noisy data with method SD1-LS3 and starting from initial guess of Figure 11. The true positions of the obstacles are the red circles (∘), initial guesses the green pluses (+) and current reconstruction the blue crosses (×).

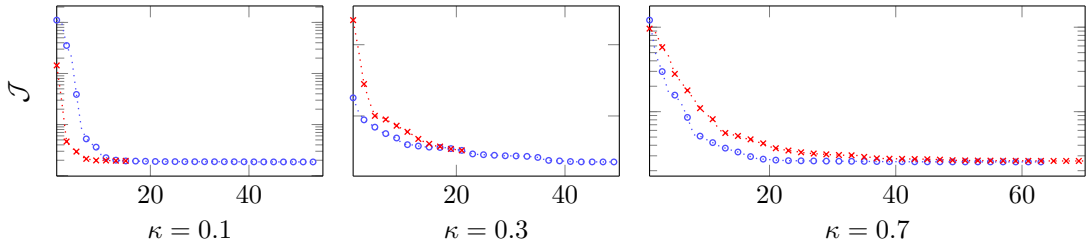


Figure 13: Evolution of the cost function in the reconstruction of 12 obstacles with noisy data of 30dB and starting from the initial guess of Figure 11. The markers are indicated one iteration over two for clarity. The blue circles (···∘···) indicate method SD1-LS3 and the red crosses (···×···) method SD2-LS3. The results of the reconstruction are given Table 4. The methods do not have the same number of iterations per frequency due to the stagnation criteria in the iterative algorithm.

- Eventually, the difference in performance is imperceptible between the two methods. In particular, the number of iterations and their run time are comparable.

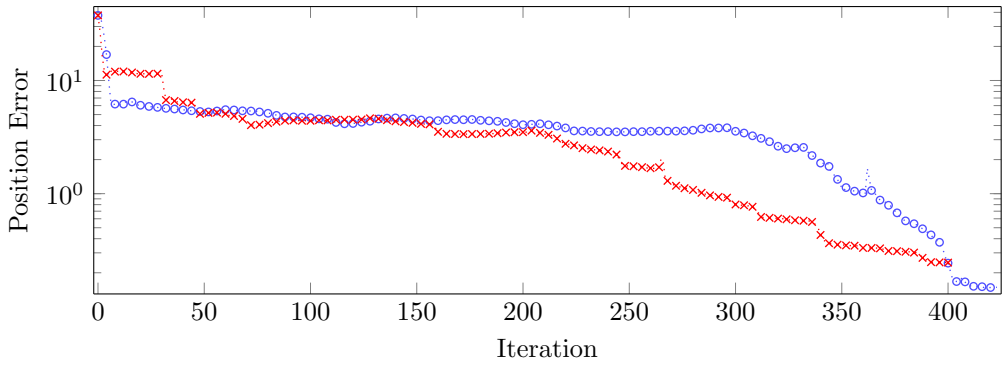


Figure 14: Progression of the position error in the reconstruction of 12 obstacles with noisy data of 30dB and starting from the initial guess of Figure 11. The markers are indicated one iteration over four for clarity. The blue circles (···∘···) indicate method SD1-LS3 and the red crosses (···×···) method SD2-LS3. The results of the reconstruction are given Table 4.

4.3 Localization of 12 hard-scattering obstacles in irregular configuration

As another validation of our reconstruction method, we design an experiment with twelve hard-scattering obstacles, now positioned irregularly in the domain of interest of size 50×60. The obstacles are centered at

$$(25.5,\ 42.0),\quad (27.0,\ 31.0),\quad (24.0,\ 38.0),\quad (15.4,\ 39.9),\quad (32.0,\ 33.9),\quad (23.0,\ 29.0),$$
$$(20.0,\ 26.0),\quad (27.0,\ 25.2),\quad (17.0,\ 31.2),\quad (31.0,\ 28.2),\quad (20.0,\ 35.0),\quad (30.5,\ 39.8).$$

The distance between the centers of two obstacles is between 4.24 and 19.5, with corresponding ratio $(\mathsf{d_{IJ}}/\mathsf{r})$ between 8.5 and 39. The domain and the obstacles configuration are illustrated in the Figure 15, together with the initial guess for the reconstruction. Similar to previous experiment, the initial guess is of very different nature compared to the true model. The data are generated using the four angles of illumination. For simplicity, we only test the reconstruction with SD1-LS3 method, which appears as the most reliable combination from previous experiments.

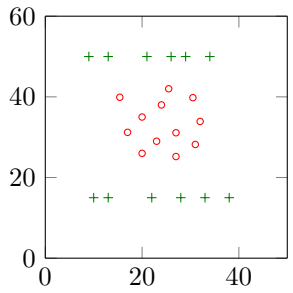


Figure 15: True positions (red circles, ◦) and initial guesses (green pluses, +) in the reconstruction of 12 obstacles irregularly positioned. The position error for the initial guess if of 40, and the underlying relative error of 79.4%.

We incorporate noise in data with 30dB signal-to-noise ratio. The L^2-norm error with the noise-free data is between 2.7% and 3.6%; the L^∞-norm is between 3.9% and 8.8%. We illustrate the data captured at the receivers location in Figure 16. Table 5 provides the parameters of the reconstruction. Here, 10 frequencies are used, from $\kappa = 0.08$ to 0.8. Figure 18 shows the evolution of the cost function with iterations for three frequencies, Figure 19 shows the evolution of the obstacle position error. In this figure, we observe that despite the decrease in the cost function, the error in the obstacle position may increase along with the iteration. This is perhaps due to the randomness of the configuration. However, the final reconstruction remains very accurate, with 0.3% error in the position, as illustrated in the visualization of Figure 17.

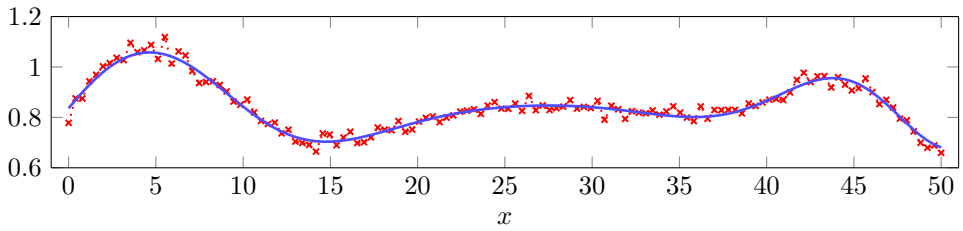
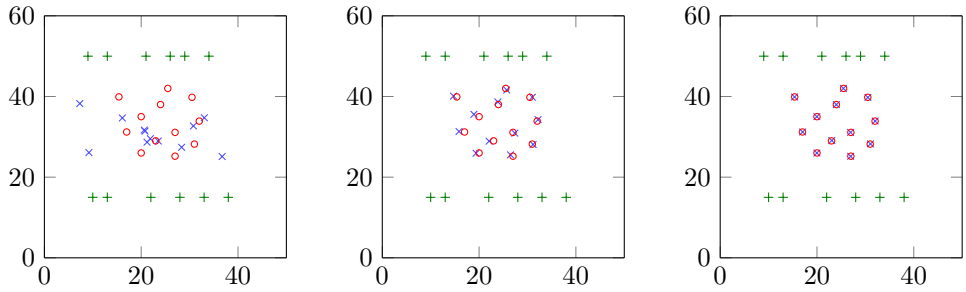


Figure 16: Real part of the data obtained at the 128 receivers at frequency $\kappa = 0.08$ for incidence angle of 90°. The blue line (——) represents the synthetic data and the red dashed line with crosses (···×···) the data incorporating 30dB noise. Relative error between observed and noisy data is 3.3% with L^2-norm and 6.8% with L^∞ norm.

We observe that the periodic configuration in Subsection 4.2 is harder to retrieve than the current irregular one. Both start from the same type of initial guess, use the same method of optimization SD1–LS3, the same type of data at the same level of noise. However, the periodic case takes more



(a) Iteration 8, $\kappa = 0.08$; $\mathcal{J} = 0.3$; position error is 16; scaled position error is 32%.

(b) Iteration 120, $\kappa = 0.4$; $\mathcal{J} = 0.3$; position error is 2.5; scaled position error is 5%.

(c) Iteration 317, $\kappa = 0.8$; $\mathcal{J} = 0.17$; position error is 0.17 ; scaled position error is 0.3%.

Figure 17: Twelve obstacles in irregular configuration recovery from noisy data with method SD1-LS3 and starting from initial guess Figure 15. The true positions of the obstacles are the red circles (∘), initial guesses the green pluses (+) and current reconstruction the blue crosses (×).

Table 5: Results for the 12 obstacles irregularly-positioned reconstruction starting from initial guess of Figure 15 and using noisy data. SD1-LS3 corresponds with quasi-Newton search direction imposing a strong Wolfe condition for line search. Line search parameters are $\mu = 10^{-4}$ and $\eta = 0.4$. The number of matrix factorizations takes into account the ones realized during the line search algorithm.

Method	♯ freq	$\kappa_{\max}$	Final rel. error $\mathcal{J}$	♯ iterations	Final rel. position error	Run time (s)	♯ facto.
SD1-LS3	10	0.8	3%	317	0.3%	8	584

iterations and thus longer to converge, *cf.* Table 4 (it uses around 1200 factorizations, double the number needed by the irregular configuration). In Figure 14, which shows the position error for periodic configuration, after the first initial sharp drop, the error decreases but very slowly before iteration 300. On the other hand, for the irregular case, *cf.* Figure 17, the drop is steady and by iteration 300, all true positions are almost located. Intuitively, the distribution of obstacles is much more dense in the periodic case, with more obstacles obstructed from view (in fact all but those on the boundary layer of the configuration). These locations, insensitive to low frequencies, are much more difficult and require a greater number of iterations to retrieve.

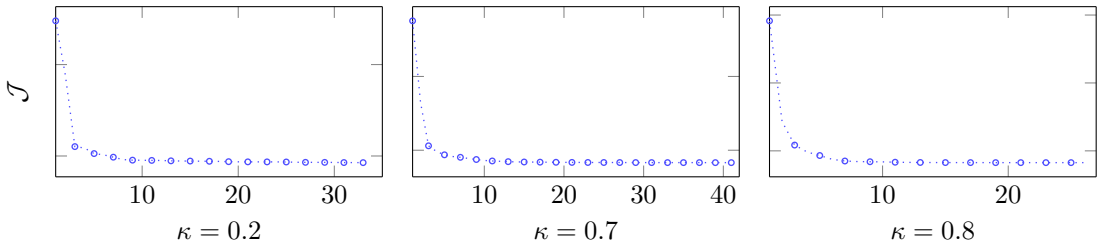


Figure 18: Evolution of the cost function in the reconstruction of 12 obstacles in irregular configuration with noisy data of 30dB, starting from the initial guess Figure 15 and using method SD1-LS3. The markers are indicated one over two iterations for clarity. The results of the reconstruction are given Table 5.

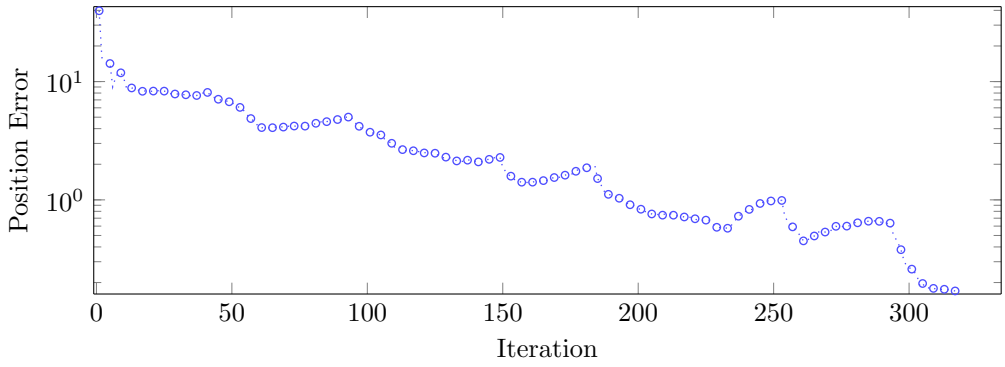


Figure 19: Progression of the position error in the reconstruction of 12 obstacles in irregular configuration with noisy data of 30dB, starting from the initial guess Figure 15 and using method SD1-LS3. The markers are indicated one iteration over four for clarity. The results of the reconstruction are given Table 5.

4.4 Localization of 24 hard-scattering obstacles in irregular configuration

We increase the number of obstacles and consider an irregular configuration with twenty-four hard-scattering obstacles centered at

$$
\begin{array}{llllllll}
(39.0, 57.7), & (40.0, 25.0), & (26.3, 22.5), & (43.1, 56.4), & (15.9, 13.4), & (17.4, 25.0), & (11.0, 39.6), & (58.2, 20.2), \\
(58.5, 30.0), & (15.0, 49.9), & (33.4, 32.0), & (42.8, 42.5), & (24.5, 49.8), & (47.7, 21.7), & (37.9, 47.0), & (31.4, 15.6), \\
(23.4, 35.8), & (47.7, 51.9), & (54.9, 56.0), & (46.4, 34.9), & (30.3, 23.9), & (56.9, 42.6), & (22.8, 55.9), & (36.7, 35.5),
\end{array}
$$

inside a square domain of size 70×70. This case is the most difficult compared to previous ones. We choose a configuration with obstacles equally distributed in the four angles. This configuration is illustrated Figure 20. However, this is more challenging than starting from an irregular configuration in which obstacles are dispersed inside the domain. For the reconstruction, we work with noise-free data from four angles of radiation. The iterative reconstruction is conducted using quasi-Newton search direction with strong Wolfe condition for line search (SD1-LS3), which is the most robust method, as established before.

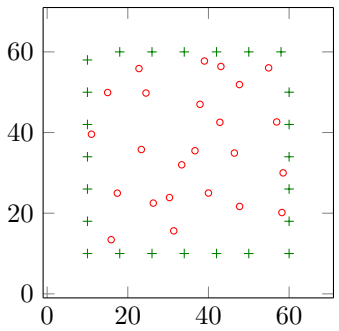


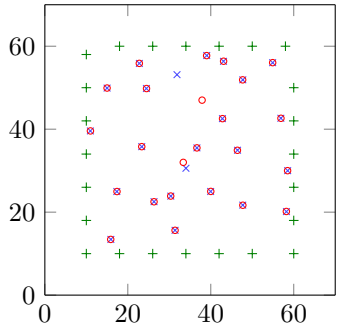
Figure 20: True positions (red circles, ∘) and initial guesses (green pluses, +) in the reconstruction of 24 hard-scattering obstacles. Its relative position error is of 48%.

The reconstruction is conducted in two stages, the first uses a set of frequency from 0.08 to 3 (with 0.08, 0.09 and from 0.1 to 3 with a step of 0.1). The algorithm run for 2533 iterations and the current reconstruction is shown in Figure 21(a). We see that most of the obstacles are accurately recovered, with the exception of two which are in the interior of the configuration. However, at this configuration, the value of the cost function is still quite high, with $\mathcal{J} = 2.28$ at frequency $\kappa = 3$, see Table 6 and Figure 21(a). In the next stage, instead of continuing frequency marching (from low to high), we recycle nine medium-ranged frequencies: $\{0.1, 0.2, 0.3, 0.5, 1, 1.5, 2, 2.5, 3\}$. This is due to

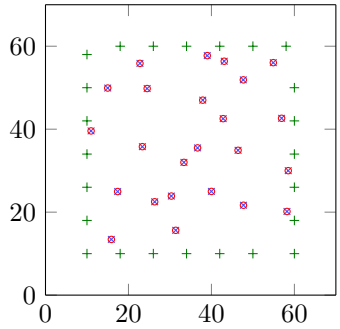
the loss of efficiency of higher frequencies, *cf.* Remark 8. The results after the recycling are shown Figure 21(b), the cost function is significantly reduced to $\mathcal{J} = 1.4 \times 10^{-7}$ (*cf.* Table 6). Figure 22 shows the progression in the cost function at three (relatively low) frequencies and compare the first and second stages in the procedure. Recycling allows a faster and stronger decrease in the cost function, as it starts with more information. From this experiment, we observe that the cost function is able to capture the difference and allows the user to identify if recycling should be used.

Table 6: The iterative reconstruction for the 24 obstacles follows two sets of frequencies. The second set comprises of recycled frequencies and starts from the final guess given by the first set. Overall, there is a total of 2893 iterations and the run time is of 4min7sec. The number of matrix factorizations takes into account the ones realized during the line search algorithm.

Stage	♯ freq	$\kappa_{\max}$	Final $\mathcal{J}$	♯ iter.	Final rel. position error	Run time (s)	♯ facto.
1	32	3	2.28	2533	11.8%	200	5854
2	9	3	1.4×10^{-7}	360	2×10^{-4}%	47	971



(a) Before frequency cycle, iteration 2533, $\kappa = 3.0$; $\mathcal{J} = 2.28$; scaled position error is 12%, run time is 200 s.



(b) Final reconstruction after frequency cycle for a total of 2893 iterations, $\kappa = 3.0$; $\mathcal{J} = 1.4 \times 10^{-7}$; scaled position error is 2×10^{-4}%, total run time including the first cycle is 260 s.

Figure 21: Twenty-four obstacles position recovery with method SD1-LS3, the true positions of the obstacles are the red circles (∘), initial guesses the green pluses (+) and current reconstruction the blue crosses (×).

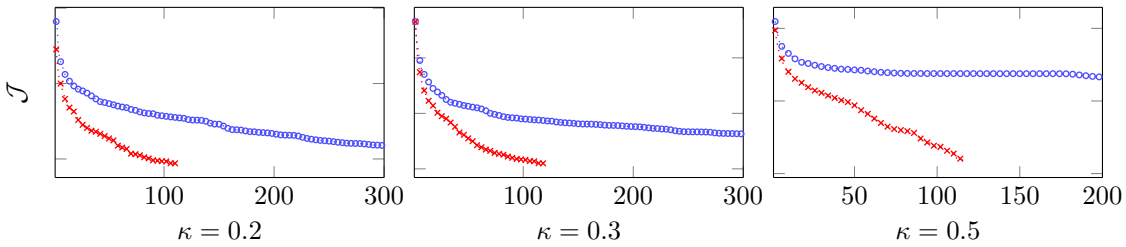


Figure 22: Evolution of the cost function in the reconstruction of 24 obstacles. The markers are indicated one iteration over four for clarity. The blue circles (···∘···) indicate the first stage in the reconstruction and the red crosses (···×···) the second stage. The methods do not have the same number of iterations per frequency due to the stagnation criteria in the iterative algorithm.

Remark 8 (Sampling of receivers per wavelength)**.** *In our experiments, we are limited to 128 receivers per angle of incidence; this is motivated by physical experiments, cf. Footnote 1. This prevents*

iterating over higher and higher frequencies, due to the decreasing number of receivers point per wavelength. Figure 23 shows the number of points per wavelength corresponding to 128 *receivers. We see that wavenumber* $\kappa > 3$ *gives less than four points per wavelength and carries too few information for inversion purposes*[12]*. Because of this, we resort to recycling over medium ranged frequencies.*

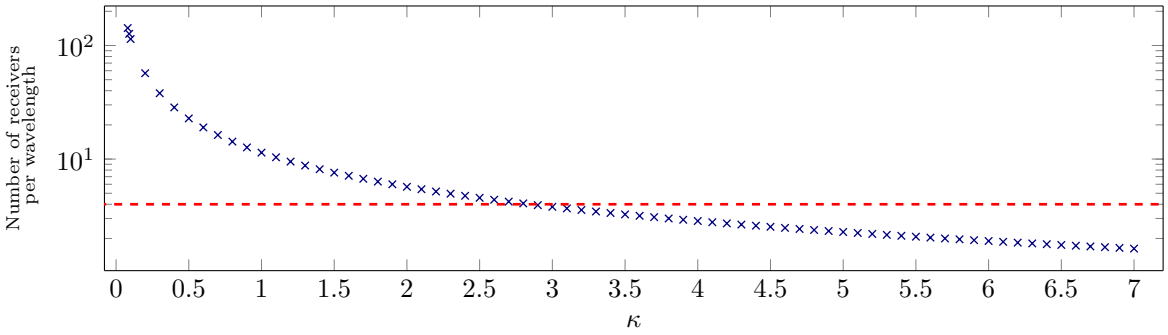


Figure 23: Receivers sampling per wavelength for varying wavenumber κ. Our experiment use 128 fixed receiver points to capture the data, they are equally spaced on a line between 0 and 70. The red dashed line (- - -) corresponds with 4 points per wavelength (usually recommended to obtain enough information on the wave propagation).

Remark 9. *The localization problem becomes increasingly difficult as the number of obstacles grows. This is evidenced by an extremely steep increase in the number of iterations and hence the number of factorizations (of forward problems). For 6 obstacles, this is about 150 factorizations, for 12 obstacles, this increases to 580 in irregular configuration, and 1200 in periodic configurations; for 24 obstacles, the number of factorization is 6800. This translates to a steep increase in inversion time. For configuration of 6 obstacles, most reconstructions take less than 1*s*. For 12 obstacles, the inversion results take around 11*s*. For 24 obstacles, it takes around* 4 min.

In a reconstruction, it is usually the locations on the outer rim of the true configuration that get retrieved first. The later iterations are usually spent on locating the positions in the interior, which require higher frequencies and more iterations per position. This factor is also determined by the denseness of obstacle distribution in a configuration, see also the observations made in subsection 4.3, comparing between the periodic configuration and the irregular one with 12 obstacles. With 6 obstacles, even in a periodic configuration, all are on the boundary layer of the configuration. In short, a configuration with more obstacles which are hidden-from-view are more difficult and take more iterations and thus time to retrieve.

5 Conclusion

In this paper, we have implemented FWI procedure with gradient based line search optimization strategy to locate soft or hard-scattering obstacles inside a homogeneous medium. Restricted to backscattered data from at most four angles of incidence and with fixed data resolution, we use low-to-high frequency progression and frequency recycling, and have been able to recover up to 24 obstacles with high-precision, without a priori information on the true locations. Despite the large number of factorizations needed, the inversion time is kept reasonably low by FSSL (10s for 6 obstacles, 11s for 12 and 4min for 24 obstacles in sequential execution).

We have also investigated several methods of optimization, including two search directions: quasi-Newton BFGS (SD1) and nonlinear conjugate gradient (SD2), combined with three line search algorithms imposing different criteria for choosing a step length: simple backtracking (LS1), sufficient descent (LS2) and strong Wolfe conditions (LS3). Our comparisons with 200 randomly generated initial guesses for six obstacles show that SD1 and LS3 are the most reliable methods. This is again

[12]Note that Shannon sampling theorem would require two points per wavelength, we double this value per safety.

confirmed in the experiment with twelve obstacles, which shows a slight edge towards SD1-LS3. The latter option also allows us to locate twenty-four obstacles. We also note that, SD2 performs better at lower frequencies, giving steeper descent in the value of the cost function, right from the start. However, at higher frequencies, it is SD1 that gives faster convergence. This means that mixed algorithms could be investigated to optimize the iterations number.

Our results serve as a satisfactory initial feasibility assessment for the procedure. We observe that the difficulty of the localization problem increases sharply with the number of obstacles, especially when there are more hidden-from-view positions. This is the subject of future work. Problem with higher number of obstacles would require an improved computational framework (parallel code, which is in progress) and an analysis to treat the increased ill-posedness (*e.g.*, with the frequency cycling as proposed here). In addition, we could work with second order optimization, and start from an initial guess given by a direct image method, as was done in [7] for medium reconstruction.

6 Acknowledgment

The authors would like to thank the referees for their constructive comments, which have greatly helped improve the paper. This project has received funding from the European Union's Horizon 2020 research and innovation program under the Marie Sklodowska-Curie grant agreement Number 777778 (Rise action Mathrocks). The research of Ha Pham is supported by the E2S CHICkPEA project. The research of Florian Faucher is supported by the Inria-TOTAL strategic action DIP.

A Calculation of the gradient by adjoint-state method

As announced in Section 3, the main ingredient of these first-order methods is the Fréchet derivatives of the cost function $\mathcal{J}$ with respect to the position of the obstacles $\boldsymbol{p}$. The adjoint-state method allows to calculate $\mathcal{J}'$ without having to calculate the whole Jacobian $\partial_{\boldsymbol{p}}\Phi$. For simplicity, in the following we take $N_{\rm Acq} = 1$. In case of multiple angles of incidence, the final result of Proposition 1 is summed over all angles of illumination.

A.1 The adjoint-state method

Step 1 We first explore the structure of $\partial_{\boldsymbol{p}}\ \Phi$.

$$\partial_{\boldsymbol{p}}\,\Phi \;=\; \left(\partial_{\boldsymbol{p}} \sum_{J=1}^{N_{\rm Obs}} \sum_{l=-\mathbf{m}}^{\mathbf{m}} (\mathcal{S}_J\,\mathbf{w}_{J,l})(\mathsf{R}_k)\,\mathsf{S}_{Jl}\right)_{1\le k\le N_{\rm rec}} \;=\; \mathfrak{R}\,\partial_{\boldsymbol{p}}\,\mathsf{S} \;+\; (\partial_{\boldsymbol{p}}\,\mathfrak{R}) \underset{(J,l)}{\cdot} \mathsf{S}\,, \tag{39}$$

where we have written $\partial_{\boldsymbol{p}}\ \Phi$ as the sum of two matrices,

$$\begin{aligned} \mathfrak{R}\ \partial_{\boldsymbol{p}}\,\mathsf{S} \;&=\; \left(\sum_{J=1}^{N_{\rm Obs}} \sum_{l=-\mathbf{m}}^{\mathbf{m}} \Big(\mathcal{S}_J\,\mathbf{w}_{J,l}\Big)(\mathsf{R}_k)\,\Big[\partial_{\boldsymbol{p}}\,\mathsf{S}_{Jl}\Big]\right)_{1\le k\le N_{\rm rec}}\,; \\ (\partial_{\boldsymbol{p}}\,\mathfrak{R}) \underset{(J,l)}{\cdot} \mathsf{S} \;&:=\; \left(\sum_{J=1}^{N_{\rm Obs}} \sum_{l=-\mathbf{m}}^{\mathbf{m}} \Big[\partial_{\boldsymbol{p}}\,(\mathcal{S}_J\,\mathbf{w}_{J,l})(\mathsf{R}_k)\Big]\,\mathsf{S}_{Jl}\right)_{1\le k\le N_{\rm rec}}. \end{aligned} \tag{40}$$

Recall that the single-layer operator $\mathcal{S}_J$ and the Fourier nodes $\mathbf{w}_{J,l}$ are defined in (7) and (6). At each index k, the quantities in the parentheses of (40) are row vectors of size $N_{\rm par}$. The contraction $\underset{(J,l)}{\cdot}$ is described in Step 3 in Subsection 3.1. An explicit expression of the tensor $\partial_{\boldsymbol{p}}\,\mathfrak{R}$ is given by (26).

To calculate the second matrix in (39), one simply needs the Fréchet derivative of $\mathfrak{R}$ which in essence requires the Fréchet derivatives of the single layer $\mathcal{S}$. An explicit calculation of this is given by in [55, Eqn (66) App. B1], with the expression evaluated at $V = \mathsf{S}(\boldsymbol{p})$. It remains to study the first matrix in (39). We write

$$\mathbf{A}(\boldsymbol{p})\,\mathsf{S}(\boldsymbol{p}) \;=\; F(\boldsymbol{p}) \quad\Rightarrow\quad \mathbf{W}(\boldsymbol{p}) \;+\; \mathbf{A}(\boldsymbol{p})\,\partial_{\boldsymbol{p}}\left[\mathsf{S}(\boldsymbol{p})\right] \;=\; \partial_{\boldsymbol{p}}F(\boldsymbol{p})\,.$$

Here, $\mathbf{W}(\boldsymbol{p})$ is matrix of size $N \times N_{\rm par}$,

$$\mathbf{W} \;:=\; \partial_{\boldsymbol{p}}\mathbf{A} \underset{(J,l)}{\cdot} \mathsf{S} \tag{41}$$

with its (I,l) row given by

$$\mathbf{W}((I,l),:) \;=\; \Big(\sum_{\tilde{J}=1}^{N_{\mathbf{Obs}}} \sum_{m=-\mathbf{m}}^{\mathbf{m}} \big(\partial_{\boldsymbol{p}}\, \mathbf{A}_{I\tilde{J};lm}\big)\, \mathsf{S}_{\tilde{J}m} \Big)_{1\le I\le N_{\mathbf{Obs}}\,,\, -\mathbf{m}\le l\le \mathbf{m}} \,. \tag{42}$$

As a result, we obtain

$$\partial_{\boldsymbol{p}}\, \mathsf{S}(\boldsymbol{p}) \;=\; \mathbf{A}(\boldsymbol{p})^{-1}\, \big(-\mathbf{W}(\boldsymbol{p}) \;+\; \partial_{\boldsymbol{p}}\, F(\boldsymbol{p})\big)\,. \tag{43}$$

Step 2 We now compute the action of $\partial_{\boldsymbol{p}}\Phi$ on the vector $\Phi(\boldsymbol{p}) - \mathbf{d}$. Using the decomposition (39), we obtain

$$(\Phi - \mathbf{d_{obs}})^\star\, \partial_{\boldsymbol{p}}\Phi \;=\; (\Phi - \mathbf{d_{obs}})^\star\, \mathfrak{R}\, \partial_{\boldsymbol{p}}\mathsf{S} \;+\; (\Phi - \mathbf{d_{obs}})^\star\, (\partial_{\boldsymbol{p}}\, \mathfrak{R}) \underset{(J,l)}{\cdot} \mathsf{S}. \tag{44}$$

Rewrite further the first term by substituting in (43) (by convention, all vectors are column),

$$\begin{aligned} (\Phi - \mathbf{d_{obs}})^\star\, \mathfrak{R}\, \partial_{\boldsymbol{p}}\, \mathsf{S} &= (\Phi - \mathbf{d_{obs}})^\star\, \mathfrak{R}\; \mathbf{A}^{-1}(-\mathbf{W} + \partial_{\boldsymbol{p}} F) \\ &= \big(\, \mathbf{A}^{-\star}\, \mathfrak{R}^\star\, (\Phi - \mathbf{d_{obs}})\big)^\star\, (-\mathbf{W} + \partial_{\boldsymbol{p}} F) \\ &= \boldsymbol{\gamma_1}^\star\, (-\mathbf{W} + \partial_{\boldsymbol{p}} F)\,. \end{aligned} \tag{45}$$

Here, we have denoted the expression in the first parentheses by $\boldsymbol{\gamma_1}$,

$$\boldsymbol{\gamma_1} \;=\; -\mathbf{A}^{-\star}\, \mathfrak{R}^\star\, (\Phi - \mathbf{d_{obs}}) \;\Leftrightarrow\; \mathbf{A}^\star\, \boldsymbol{\gamma_1} \;=\; -\mathfrak{R}^\star (\Phi - \mathbf{d_{obs}})\,. \tag{46}$$

Now using (22), (44)–(46), we arrive at the following formula to calculate $\mathcal{J}'$.

Proposition 1. *With adjoint state $\boldsymbol{\gamma_1}$ given by* (46)*, $\mathsf{S}(\boldsymbol{p})$* (12)*, and $\mathbf{W}$* (41)–(42)*, the derivative of the cost function is given by the following expressions,*

$$\begin{aligned} \mathcal{J}'(\boldsymbol{p}) &= \mathrm{Re}\,\big[(\mathfrak{R}V - \mathbf{d}_{\boldsymbol{obs}})^\star\, \partial_{\boldsymbol{p}}(\mathfrak{R}\, V) \;+\; \gamma_1^\star\, \big(\partial_{\boldsymbol{p}}\, (\mathbf{A}\, V) \;-\; \partial_{\boldsymbol{p}} F)\big]\big|_{V=\mathsf{S}(\boldsymbol{p})\,,\,\gamma_1=\boldsymbol{\gamma_1}} \\ &= \mathrm{Re}\,\big[(\Phi(\boldsymbol{p}) - \mathbf{d}_{\boldsymbol{obs}})^\star\, \partial_{\boldsymbol{p}}\mathfrak{R} \underset{(J,l)}{\cdot} \mathsf{S}(\boldsymbol{p}) \;+\; \boldsymbol{\gamma_1}^\star\, (\partial_{\boldsymbol{p}}\, \mathbf{A} \underset{(J,l)}{\cdot} \mathsf{S}(\boldsymbol{p}) \;-\; \partial_{\boldsymbol{p}} F)\big] \\ &= \mathrm{Re}\,\big[(\Phi(\boldsymbol{p}) - \mathbf{d}_{\boldsymbol{obs}})^\star\, \partial_{\boldsymbol{p}}\mathfrak{R} \underset{(J,l)}{\cdot} \mathsf{S}(\boldsymbol{p}) \;+\; \boldsymbol{\gamma_1}^\star\, (\mathbf{W} \;-\; \partial_{\boldsymbol{p}} F)\big]\,. \end{aligned} \tag{47}$$

A.2 The adjoint-state method by the Lagrangian

We now the results of Prop. 1 from the Lagrangian point of view. Introduce the cost function $\mathcal{G}$ defined as

$$\mathcal{G} \,:\, \mathbb{R}^{N_{\rm par}} \times \mathbb{C}^N \;\longrightarrow\; \mathbb{R} \quad ; \quad \mathcal{G}(\boldsymbol{p}, V) \,:=\, \|\mathfrak{R}(\boldsymbol{p})\, V \,-\, \mathbf{d}\,\|^2_{\mathbb{C}^{N_{\rm rec}}}\,. \tag{48}$$

By the definition of the forward map Φ in (17), the cost function $\mathcal{G}$ evaluated at $V = \mathsf{S}(\boldsymbol{p})$, defined in (12), gives the reduced cost function $\mathcal{J}$,

$$\mathcal{J}(\boldsymbol{p}) \quad :=\quad \mathcal{G}\,\big(\boldsymbol{p}\;,\; \mathsf{S}(\boldsymbol{p})\big) \;=\; \|\,\Phi(\boldsymbol{p}) \;-\; \mathbf{d}\,\|^2_{\mathbb{C}^{N_{\rm rec}}}. \tag{49}$$

Lagrangian To calculate $\mathcal{J}'$, we consider the following optimization problem

$$\min_{\boldsymbol{p}\,\in\,\mathbb{R}^{N_{\rm par}}\,,\,V\in\mathbb{C}^N} \;\; \mathcal{G}(\boldsymbol{p}, V) \quad \text{subject to} \quad \mathbf{A}(\boldsymbol{p})\, V - F(\boldsymbol{p}) = 0\,,$$

with $\boldsymbol{p}$ and V treated as independent variables, with the linear equation (10) (also called state equation) posed as a constraint. For this purpose, we introduce the Lagrangian $\mathcal{L}$ and the adjoint state variable $\gamma_1 \in \mathbb{C}^N$

$$\begin{aligned} &\mathcal{L} \,:\, \mathbb{R}^{N_{\rm par}} \times \mathbb{C}^N \times \mathbb{C}^N \;\longrightarrow\; \mathbb{C} \\ &\mathcal{L}(\boldsymbol{p}, V, \gamma_1) \,:=\, \mathcal{G}(\boldsymbol{p}, V) \,+\, \mathfrak{l}(\boldsymbol{p}, V, \gamma_1)\,, \text{with } \mathfrak{l}(\boldsymbol{p}, V, \gamma_1) \,:=\, \gamma_1^\star \left(\mathbf{A}(\boldsymbol{p})\, V - F(\boldsymbol{p})\right). \end{aligned} \tag{50}$$

Here * is the transposed complex conjugate. In terms of Lagrangian $\mathcal{L}$, the cost function is given by,

$$\mathcal{J}(\boldsymbol{p}) \,=\, \mathrm{Re}\, \mathcal{L}\big(\boldsymbol{p}\,,\, \mathsf{S}(\boldsymbol{p})\,,\, \gamma_1\big)\;,\; \forall\, \gamma_1\,,\, \boldsymbol{p} \quad ; \quad \mathcal{J}'(\boldsymbol{p}) \,=\, \partial_{\boldsymbol{p}} \left[\, \mathrm{Re}\, \mathcal{L}\big(\boldsymbol{p}\,,\, \mathsf{S}(\boldsymbol{p})\,,\, \gamma_1\big)\right].$$

By [55, Eqn 56 in App. A],

$$\begin{aligned} \mathcal{J}'(\boldsymbol{p})\, \delta_{\boldsymbol{p}} \;=\; \mathrm{Re}\Big[&\big(\partial_{\boldsymbol{p}} \mathcal{L}\big)(\boldsymbol{p}, V, \gamma_1) \,+\, \big(\partial_V^{\mathsf{CR}} \mathcal{L}\big)(\boldsymbol{p}, V, \gamma_1)\; \partial_{\boldsymbol{p}} \mathsf{S}(\boldsymbol{p}) \\ &+\, \big(\partial_{\overline{V}}^{\mathsf{CR}} \mathcal{L}\big)(\boldsymbol{p}, V, \gamma_1)\; \partial_{\boldsymbol{p}}\, \overline{\mathsf{S}(\boldsymbol{p})}\Big]\Big|_{V = \mathsf{S}(\boldsymbol{p})}\, \delta_{\boldsymbol{p}}\,. \end{aligned} \tag{51}$$

Here, $\partial_{\boldsymbol{p}} \mathcal{L}$ is the usual partial derivative of $\mathcal{L}$ with respect to $\boldsymbol{p}$, while $\partial_V^{\mathsf{CR}} \mathcal{L}$, and $\partial_{\overline{V}}^{\mathsf{CR}} \mathcal{L}$ are the complex partial derivatives defined in [55, Def 1 in App. A].

The adjoint-state problem We would like to avoid calculating $\partial_{\boldsymbol{p}}\, \overline{\mathsf{S}(\boldsymbol{p})}$. For this purpose, we can and will choose γ_1 so that the terms in (51) containing this quantity and its conjugate vanish, *i.e.*

$$\big(\partial_V^{\mathsf{CR}} \mathcal{L}\big)(\boldsymbol{p}, V, \boldsymbol{\gamma_1})\; \partial_{\boldsymbol{p}} \mathsf{S}(\boldsymbol{p}) \;+\; \big(\partial_{\overline{V}}^{\mathsf{CR}} \mathcal{L}\big)(\boldsymbol{p}\,,\, V\,,\, \boldsymbol{\gamma_1})\; \partial_{\boldsymbol{p}} \overline{\mathsf{S}(\boldsymbol{p})} \;=\; 0\,. \tag{52}$$

To arrive at a more explicit equation, we now calculate $\mathcal{L}_V$, and $\mathcal{L}_{\overline{V}}$. We have

$$\partial_V^{\mathsf{CR}}\, \mathcal{L} \;=\; \partial_V^{\mathsf{CR}}\, \mathcal{G} \;+\; \partial_V^{\mathsf{CR}}\, \mathfrak{l} \quad ; \quad \partial_{\overline{V}}^{\mathsf{CR}}\, \mathcal{L} \;=\; \partial_{\overline{V}}^{\mathsf{CR}}\, \mathcal{G} \;+\; \partial_{\overline{V}}^{\mathsf{CR}}\, \mathfrak{l}\,.$$

By its definition, $\mathfrak{l}$ is linear in S which is itself linear in V, thus $\mathfrak{l}$ is Fréchet differentiable with respect to V and

$$\partial_V^{\mathsf{CR}}\, \mathfrak{l} \;=\; \partial_V^{\mathsf{CR}} \Big[\boldsymbol{\gamma_1}^\star (\mathbf{A}V - F)\Big] \;=\; \boldsymbol{\gamma_1}^\star \mathbf{A} \quad ; \quad \partial_{\overline{V}}^{\mathsf{CR}}\, \mathfrak{l} \;=\; 0\,.$$

By [55, Def 1 in App. A], we have

$$\partial_V^{\mathsf{CR}}\, \mathcal{G} \;=\; \tfrac{1}{2} (\boldsymbol{\mathfrak{R}}\, V - \mathbf{d_{obs}})^\star\, \boldsymbol{\mathfrak{R}} \quad ; \quad \partial_{\overline{V}}^{\mathsf{CR}}\, \mathcal{G} \;=\; \tfrac{1}{2} (\boldsymbol{\mathfrak{R}}\, V - \mathbf{d_{obs}})^t\, \overline{\boldsymbol{\mathfrak{R}}}\,.$$

LHS of condition (52) is

$$\begin{aligned} &\big(\partial_V^{\mathsf{CR}}\, \mathcal{L}\big)(\boldsymbol{p}\,,\, V\,,\, \gamma_1)\; \partial_{\boldsymbol{p}} \mathsf{S}(\boldsymbol{p}) \;+\; \big(\partial_{\overline{V}}^{\mathsf{CR}} \mathcal{L}\big)(\boldsymbol{p}\,,\, V\,,\, \gamma_1)\; \partial_{\boldsymbol{p}} \overline{\mathsf{S}(\boldsymbol{p})} \\ &= \tfrac{1}{2} \Big(\boldsymbol{\mathfrak{R}} V - \mathbf{d_{obs}}\Big)^\star\, \boldsymbol{\mathfrak{R}}\; \partial_{\boldsymbol{p}} \mathsf{S}(\boldsymbol{p}) \;+\; \tfrac{1}{2} \Big(\boldsymbol{\mathfrak{R}}\, V - \mathbf{d_{obs}}\Big)^t\; \overline{\boldsymbol{\mathfrak{R}}}\; \partial_{\boldsymbol{p}} \overline{\mathsf{S}(\boldsymbol{p})} \;+\; \gamma_1^\star \mathbf{A}\; \partial_{\boldsymbol{p}} \mathsf{S}(\boldsymbol{p})\,. \end{aligned}$$

This then gives

$$\begin{aligned} &\mathrm{Re}\, \Big[\big(\partial_V^{\mathsf{CR}}\, \mathcal{L}\big)(\boldsymbol{p}, V, \gamma_1)\; \partial_{\boldsymbol{p}} \mathsf{S}(\boldsymbol{p}) \;+\; \big(\partial_{\overline{V}}^{\mathsf{CR}} \mathcal{L}\big)(\boldsymbol{p}, V, \gamma_1)\; \partial_{\boldsymbol{p}} \overline{\mathsf{S}(\boldsymbol{p})}\Big] \\ &\qquad =\; \mathrm{Re}\, \Big[\Big(\big(\boldsymbol{\mathfrak{R}} V - \mathbf{d_{obs}}\big)^\star\, \boldsymbol{\mathfrak{R}} \;+\; \gamma_1^\star\, \mathbf{A}\Big)\, \partial_{\boldsymbol{p}}\, \mathsf{S}(\boldsymbol{p})\Big]\,. \end{aligned}$$

Condition (52) is satisfied, if γ_1 solves the following equation

$$\big(\boldsymbol{\mathfrak{R}} V - \mathbf{d_{obs}}\big)^\star\, \boldsymbol{\mathfrak{R}} \,+\, \gamma_1^\star\, \mathbf{A} \,=\, 0 \;\Leftrightarrow\; \mathbf{A}^\star \gamma_1 \;=\; -\boldsymbol{\mathfrak{R}}^\star\, (\boldsymbol{\mathfrak{R}} V - \mathbf{d_{obs}})$$

evaluated at $V = \mathsf{S}(\boldsymbol{p})$. We have thus arrived at the adjoint problem (46).

Final step We use the adjoint state $\gamma_1 = \boldsymbol{\gamma_1}$ as defined in (46) to simplify (51) to

$$\hat{J}'(\boldsymbol{p})\ \delta_{\boldsymbol{p}}\ =\ \mathrm{Re}\left[\left(\partial_{\boldsymbol{p}}\mathcal{L}\right)(\boldsymbol{p},V,\gamma_1)\,\big|_{V=\mathcal{S}(\boldsymbol{p})\,,\,\gamma_1=\boldsymbol{\gamma_1}}\ \delta_{\boldsymbol{p}}\right].$$

It remains to calculate $\partial_{\boldsymbol{p}}\mathcal{L} = \partial_{\boldsymbol{p}}\mathcal{G} + \partial_{\boldsymbol{p}}\mathfrak{l}$. By using [55, Prop 4,App. A], one obtains, for details see [55, p.19]

$$(\partial_{\boldsymbol{p}}\mathcal{G})(\boldsymbol{p},V,\gamma_1)\ =\ 2\,\mathrm{Re}\left[\tfrac{1}{2}\left(\mathfrak{R}V-\mathbf{d_{obs}}\right)^{\star}\ \partial_{\boldsymbol{p}}(\mathfrak{R}\,V)\right].$$

On the other hand,$(\partial_{\boldsymbol{p}}\mathfrak{l})(\boldsymbol{p},V,\gamma_1)\ =\ \gamma_1^{\star}\left[\partial_{\boldsymbol{p}}(\mathbf{A}V)\ -\ \partial_{\boldsymbol{p}}F\right]$. As a result, we arrive at results of Prop 1.

References

[1] Patrick R Amestoy, Iain S Duff, Jean-Yves L'Excellent, and Jacko Koster. A fully asynchronous multifrontal solver using distributed dynamic scheduling. *SIAM Journal on Matrix Analysis and Applications*, 23(1):15–41, 2001.

[2] Habib Ammari, Mark Asch, Lili Guadarrama Bustos, Vincent Jugnon, and Hyeonbae Kang. Transient wave imaging with limited-view data. *SIAM Journal on Imaging Sciences*, 4(4):1097–1121, 2011.

[3] Habib Ammari, Ekaterina Iakovleva, and Dominique Lesselier. A MUSIC algorithm for locating small inclusions buried in a half-space from the scattering amplitude at a fixed frequency. *Multiscale Modeling & Simulation*, 3(3):597–628, 2005.

[4] Habib Ammari and Hyeonbae Kang. *Reconstruction of small inhomogeneities from boundary measurements*. Springer, 2004.

[5] E. Anderson, Z. Bai, C. Bischof, S. Blackford, J. Demmel, J. Dongarra, J. Du Croz, A. Greenbaum, S. Hammarling, A. McKenney, and D. Sorensen. *LAPACK Users' Guide*. Society for Industrial and Applied Mathematics, Philadelphia, PA, third edition, 1999.

[6] Xavier Antoine, Karim Ramdani, and Bertrand Thierry. Wide frequency band numerical approaches for multiple scattering problems by disks. *Journal of Algorithms & Computational Technology*, 6(2):241–259, 2012.

[7] Gang Bao, Songming Hou, and Peijun Li. Inverse scattering by a continuation method with initial guesses from a direct imaging algorithm. *Journal of Computational Physics*, 227(1):755–762, 2007.

[8] Gang Bao, Kai Huang, Peijun Li, and Hongkai Zhao. A direct imaging method for inverse scattering using the generalized Foldy-Lax formulation. *Contemp. Math*, 615:49–70, 2014.

[9] Gang Bao and Peijun Li. Inverse medium scattering problems for electromagnetic waves. *SIAM Journal on Applied Mathematics*, 65(6):2049–2066, 2005.

[10] Gang Bao and Peijun Li. Shape reconstruction of inverse medium scattering for the Helmholtz equation. *Computational methods for applied inverse problems.*, 2012.

[11] Gang Bao, Peijun Li, Junshan Lin, and Faouzi Triki. Inverse scattering problems with multi-frequencies. *Inverse Problems*, 31(9):093001, 2015.

[12] Gang Bao and Jun Liu. Numerical solution of inverse scattering problems with multi-experimental limited aperture data. *SIAM Journal on Scientific Computing*, 25(3):1102–1117, 2003.

[13] H. Barucq, J. Chabassier, H. Pham, and S. Tordeux. A study of the numerical robustness of single-layer method with Fourier basis for multiple obstacle scattering in homogeneous media. Research Report RR-8988, Inria Bordeaux Sud-Ouest, December 2016.

[14] Hélène Barucq, Juliette Chabassier, Ha Pham, and Sébastien Tordeux. Numerical robustness of single-layer method with Fourier basis for multiple obstacle acoustic scattering in homogeneous media. *Wave Motion*, 2017.

[15] Carlos Borges, Adrianna Gillman, and Leslie Greengard. High resolution inverse scattering in two dimensions using recursive linearization. *SIAM Journal on Imaging Sciences*, 10(2):641–664, 2017.

[16] Carlos Borges and Leslie Greengard. Inverse obstacle scattering in two dimensions with multiple frequency data and multiple angles of incidence. *SIAM Journal on Imaging Sciences*, 8(1):280–298, 2015.

[17] Ovidio Mario Bucci, Lorenzo Crocco, Tommaso Isernia, and Vito Pascazio. Inverse scattering problems with multifrequency data: reconstruction capabilities and solution strategies. *IEEE Transactions on Geoscience and Remote Sensing*, 38(4):1749–1756, 2000.

[18] Tan Bui-Thanh and Omar Ghattas. Analysis of the hessian for inverse scattering problems: Ii. inverse medium scattering of acoustic waves. *Inverse Problems*, (5):055002, 2012.

[19] Carey Bunks, Fatimetou M. Saleck, S. Zaleski, and G. Chavent. Multiscale seismic waveform inversion. *Geophysics*, 60(5):1457–1473, 1995.

[20] Fioralba Cakoni and David Colton. *Qualitative methods in inverse scattering theory: An introduction*. Springer Science & Business Media, 2005.

[21] Fioralba Cakoni and Jacob D Rezac. Direct imaging of small scatterers using reduced time dependent data. *Journal of Computational Physics*, 338:371–387, 2017.

[22] Durga Prasad Challa and Mourad Sini. Inverse scattering by point-like scatterers in the Foldy regime. *Inverse Problems*, 28(12):125006, 2012.

[23] Guy Chavent. Identification of functional parameters in partial differential equations. In R. E. Goodson and M. Polis, editors, *Identification of Parameters in Distributed Systems*, pages 31–48. ASME, New York, 1974.

[24] Guy Chavent. *Nonlinear least squares for inverse problems: theoretical foundations and step-by-step guide for applications*. Springer Science & Business Media, 2010.

[25] Yu Chen. Inverse scattering via Heisenberg's uncertainty principle. *Inverse problems*, 13(2):253, 1997.

[26] John E Dennis, Jr and Jorge J Moré. Quasi-Newton methods, motivation and theory. *SIAM review*, 19(1):46–89, 1977.

[27] Anthony J Devaney, Edwin A Marengo, and Fred K Gruber. Time-reversal-based imaging and inverse scattering of multiply scattering point targets. *The Journal of the Acoustical Society of America*, 118(5):3129–3138, 2005.

[28] Florian Faucher. *Contributions to seismic full waveform inversion for time harmonic wave equations: stability estimates, convergence analysis, numerical experiments involving large scale optimization algorithms*. PhD thesis, Université de Pau et Pays de l'Ardour, 2017.

[29] Roohallah Fazli and Mansor Nakhkash. An analytical approach to estimate the number of small scatterers in 2d inverse scattering problems. *Inverse Problems*, 28(7):075012, 2012.

[30] A Fichtner. *Full seismic waveform modelling and inversion*. Springer Science & Business Media, 2011.

[31] Kamyar Firouzi and Butrus T Khuri-Yakub. Localization of weak objects in reverberant fields using waveform inversion. *The Journal of the Acoustical Society of America*, 142(2):1088–1097, 2017.

[32] Zydrunas Gimbutas and Leslie Greengard. Fast multi-particle scattering: A hybrid solver for the Maxwell equations in microstructured materials. *Journal of Computational Physics*, 232(1):22–32, 2013.

[33] Leslie Greengard and Vladimir Rokhlin. A fast algorithm for particle simulations. *Journal of computational physics*, 73(2):325–348, 1987.

[34] Roland Griesmaier and Christian Schmiedecke. A multifrequency MUSIC algorithm for locating small inhomogeneities in inverse scattering. *Inverse Problems*, 33(3):035015, 2017.

[35] Semion Gutman and Michael Klibanov. Two versions of quasi-newton method for multidimensional inverse scattering problem. *Journal of Computational Acoustics*, 1(02):197–228, 1993.

[36] William W Hager and Hongchao Zhang. A survey of nonlinear conjugate gradient methods. *Pacific journal of Optimization*, 2(1):35–58, 2006.

[37] Frank Hettlich. Fréchet derivatives in inverse obstacle scattering. *Inverse problems*, 11(2):371, 1995.

[38] Frank Hettlich. Fréchet derivatives in inverse obstacle scattering. *Inverse Problems*, 14(1):209, 1998.

[39] Kai Huang, Peijun Li, and Hongkai Zhao. An efficient algorithm for the generalized Foldy–Lax formulation. *Journal of Computational Physics*, 234:376–398, 2013.

[40] Masaru Ikehata, Esa Niemi, Samuli Siltanen, et al. Inverse obstacle scattering with limited-aperture data. *Inverse problems and imaging.*, 2012.

[41] Michel Kern. *Numerical Methods for Inverse Problems*. John Wiley & Sons, 2016.

[42] RE Kleinman and PM den Berg. Two-dimensional location and shape reconstruction. *Radio Science*, 29(4):1157–1169, 1994.

[43] Aleksandr E Kolesov, Michael V Klibanov, Loc H Nguyen, Dinh-Liem Nguyen, and Nguyen T Thanh. Single measurement experimental data for an inverse medium problem inverted by a multi-frequency globally convergent numerical method. *Applied Numerical Mathematics*, 2017.

[44] Rainer Kress. Newtons method for inverse obstacle scattering meets the method of least squares. *Inverse Problems*, 19(6):S91, 2003.

[45] Rainer Kress. Uniqueness and numerical methods in inverse obstacle scattering. In *Journal of Physics: Conference Series*, volume 73, page 012003. IOP Publishing, 2007.

[46] Rainer Kress and Kuo-Ming Lee. A second degree newton method for an inverse obstacle scattering problem. *Journal of Computational Physics*, 230(20):7661–7669, 2011.

[47] Rainer Kress and William Rundell. A quasi-newton method in inverse obstacle scattering. *Inverse Problems*, 10(5):1145, 1994.

[48] Rainer Kress and William Rundell. Inverse obstacle scattering using reduced data. *SIAM Journal on Applied Mathematics*, 59(2):442–454, 1998.

[49] Jacques Louis Lions and Sanjoy K Mitter. *Optimal control of systems governed by partial differential equations*, volume 1200. Springer Berlin, 1971.

[50] Dhairya Malhotra and George Biros. Pvfmm: A parallel kernel independent fmm for particle and volume potentials. *Communications in Computational Physics*, 18(3):808–830, 2015.

[51] Jorge J Moré and David J Thuente. Line search algorithms with guaranteed sufficient decrease. *ACM Transactions on Mathematical Software (TOMS)*, 20(3):286–307, 1994.

[52] Dinh-Liem Nguyen, Michael V Klibanov, Loc H Nguyen, Aleksandr E Kolesov, Michael A Fiddy, and Hui Liu. Numerical solution of a coefficient inverse problem with multi-frequency experimental raw data by a globally convergent algorithm. *Journal of Computational Physics*, 345:17–32, 2017.

[53] Jorge Nocedal and Stephen Wright. *Numerical optimization*. Springer Science & Business Media, 2006.

[54] S Operto, A Miniussi, R Brossier, L Combe, L Métivier, V Monteiller, Alessandra Ribodetti, and J Virieux. Efficient 3-d frequency-domain mono-parameter full-waveform inversion of ocean-bottom cable data: Application to valhall in the visco-acoustic vertical transverse isotropic approximation. *Geophysical Journal International*, 202(2):1362–1391, 2015.

[55] Ha Pham, Hélène Barucq, and Florian Faucher. Quantitative localization of small obstacles with single-layer potential fast solvers. Research report, Inria Bordeaux Sud-Ouest ; Magique 3D ; Universite de Pau et des Pays de l'Adour, December 2017.

[56] Roland Potthast. Fréchet differentiability of the solution to the acoustic neumann scattering problem with respect to the domain. *Journal of Inverse and Ill-Posed Problems*, 4(1):67–84, 1996.

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

[58] R. Gerhard Pratt, Changsoo Shin, and G. J. Hick. Gauss–newton and full newton methods in frequency–space seismic waveform inversion. *Geophysical Journal International*, 133(2):341–362, 1998.

[59] Changsoo Shin and Young Ho Cha. Waveform inversion in the Laplace Fourier domain. *Geophysical Journal International*, 177(3):1067–1079, 2009.

[60] Mourad Sini and Nguyen Trung Thành. Regularized recursive newton-type methods for inverse scattering problems using multifrequency measurements. *ESAIM: Mathematical Modelling and Numerical Analysis*, 49(2):459–480, 2015.

[61] Laurent Sirgue and R. Gerhard Pratt. Efficient waveform inversion and imaging: A strategy for selecting temporal frequencies. *Geophysics*, 69(1):231–248, 2004.

[62] A. Tarantola. Inversion of seismic reflection data in the acoustic approximation. *Geophysics*, 49:1259–1266, 1984.

[63] A. Tarantola. Linearized inversion of seismic reflection data. *Geophysical Prospecting*, 32:998–1015, 1984.

[64] Bertrand Thierry. *Analyse et simulations numériques du retournement temporel et de la diffraction multiple*. PhD thesis, Université Henri Poincaré-Nancy I, 2011.

[65] PM Van Den Berg, RE Kleinman, et al. Gradient methods in inverse acoustic and electromagnetic scattering. *IMA Volumes in Mathematics and its Applications*, 92:173–194, 1997.

[66] Jean Virieux and Stéphane Operto. An overview of full waveform inversion in exploration geophysics. *Geophysics*, 74(6):WCC1–WCC26, 2009.

[67] Xiaodong Zhang, Shira L Broschat, and Patrick J Flynn. A numerical study of conjugate gradient directions for an ultrasound inverse problem. *Journal of Computational Acoustics*, 12(04):587–604, 2004.

[68] A Zinn. On an optimisation method for the full-and the limited-aperture problem in inverse acoustic scattering for a sound-soft obstacle. *Inverse Problems*, 5(2):239, 1989.