On reconstruction of the coefficient in complex Helmholtz’s equation

M Malovichko 1, A Orazbayev 2, Yu Kloss 2,3, N Khokhlov 2

1 Skolkovo Institute of Science and Technology, CDISE, Moscow, Russia
2 Moscow Institute of Science and Technology, Dolgoprudny, Russia
3 National Research Center Kurchatov Institute, Moscow, Russia

E-mail: kl@lokip.ru

Abstract. This note summarizes some preliminary results on the fast solution of the coefficient inverse problem for the Helmholtz equation, given measured pressure in a set of observation points. The Helmholtz equation is the model PDE for the harmonic problem of the linear theory of elasticity, and this work is a move in that direction. The problem has been the primary focus for several research areas, most notably seismic exploration. Still, practical problems are very challenging because they are non-linear and large. In this paper, we develop a novel numerical method for seismic full-waveform inversion based on Newton iterations. Its distinctive future is that it does not require the Jacobian of the target functional. Thus, in certain scenarios, it will perform only a fraction of computations comparing to the conventional Gauss-Newton algorithm. We present some early results on the Helmholtz equation in two dimensions.

1. Introduction

This paper is devoted to the coefficient inverse problem for the Helmholtz equation. In the geophysical community, it is known as the frequency-domain acoustic full-wave inversion (FWI). We refer to [1,2] for a general description.

Many algorithms to perform the FWI have been proposed in the literature, see [2] for a comprehensive review. Most of them fall into three categories: first-order methods, second-order methods (Newtonian), and quasi-Newtonian methods. The first-order methods use only objective function gradients. The most well-known first-order method is the nonlinear conjugate gradients (NLCG). NLCG computes the correction to a given distribution of P-wave velocity combining gradients of the objective function at the two consecutive iterations supplemented with the selection of optimal step size. The second-order methods use the second derivatives of the objective function. A typical representative of this family is the celebrated Gauss-Newton (GN) method. It computes the correction to the model by solving a normal system of linear equations, whose system matrix approximates the Hessian of the objective function. The quasi-Newton methods use information about second derivatives only partially, usually by approximating the Hessian with a diagonal matrix. Typical representatives of this class are L-BFGS and the variant of NLCG, in which a diagonal matrix scales gradients.

First-order and quasi-Newton methods have a common disadvantage. They do not sufficiently account for the nonlinearity of the coefficient inverse problem. In practice, the inversion may converge slowly, making hundreds of nonlinear iterations. The algorithm may break down at complicated velocity distributions. Newton’s methods are mostly free from this drawback and
often converge just in several iterations. The main difficulty is related to the fact that each
Newton’s step requires the solution of the normal system of linear algebraic equations, which
has a large dense and ill-conditioned matrix. This system, due to its size, is solved by an internal
iterative solver. No efficient preconditioner is known for it, and therefore each Newton update
may require thousands of internal (linear) iterations. Thus, thousands of forward solves are
potentially needed.

This paper proposes a new way of calculating Newtonian corrections. The normal system of
equations with a dense matrix, which is difficult to precondition, is replaced by another system
with a larger matrix. This matrix, however, is sparse and thus can be effectively preconditioned.
Thus, the external (Newton) iterations coincide in both cases, but the procedure for calculating
the corrections at each iteration may become much more economical for certain classes of
problems. Here we present some early results which, though positive, should be considered
as a proof of concept. The paper is organized as follows. In Section 2 we formulate the inverse
problems, present the classical Gauss-Newton method and outline our approach. Section 3 is
dedicated to a numerical experiment. Concluding remarks are given in Section 4.

2. Theory
Consider the forward problem
\[-\Delta U - \frac{\omega^2}{c^2} U = F, \text{ in } \Omega,\]
\[U = 0, \text{ on } \partial \Omega.\]

Upon finite-difference discretization, it transforms to the following system of linear equations,
\[Au = f.\]  

In practice, perfectly matched layers (PML) are added to (1) on all sides of \(\Omega\) except the air-
ground interface, for example, [3]. It leads to a complex-valued matrix \(A\). Although in our
numerical experiments we did not add PML layers, we will assume that \(A\) is Hermitian sparse
matrix.

Let us by \(m\) denote a discretized inverse squared velocity. Matrix \(A\) depends on \(m\), so we
will sometimes write \(A(m)\) to emphasize this dependency. We introduce the so-called forward
operator \(\mathcal{F}\), which computes data \(d\) (e.g. measured pressure at a set of receivers due to a given
source) for a given \(m\). Operator \(\mathcal{F}\) is simply
\[\mathcal{F}(m) = QA(m)^{-1}.\]

The standard approach to the FWI (in fact, to any geophysical coefficient problem) is to
minimize the following Tikhonov’s functional
\[\min_m \frac{1}{2} W \left( d - \mathcal{F}(m) \right)^2_2 \frac{\varepsilon}{2} \left| L^{1/2} m \right|^2_2.\]

In the Gauss-Newton framework, \(m\) is iterated as \(m_{n+1} = m_n + p\) with \(p\) being the update to
\(m_n\), satisfying the following minimization problem
\[\min_p \frac{1}{2} W \left( r - Jp \right)^2_2 \frac{\varepsilon}{2} \left| L^{1/2} (mn + p) \right|^2_2.\]

Here \(r = d - \mathcal{F}(m_n)\) is data residual at \(n\)-th iteration. Matrix \(J\) is the matrix of Fréchet
operator at \(m_n\), \(J = \frac{\partial \mathcal{F}(m_n)}{\partial m} \bigg|_{m_n}\). Due to the fact that (1) is a linear PDE, it can be shown that
the Jacobian is given by the following formula,
\[J = WQA^{-1}P,\]
where $P$ is a diagonal matrix containing forward solution at $m_n$. In other words,

$$P = \omega^2 \text{diag}(u),$$

where $u$ is the solution of the problem (2) with $m = m_n$. The gradient of the target functional with respect to $m$, which we will denote by $g$, can be expressed through the Jacobian,

$$g = -J^*W^TWr + \varepsilon Lm_n.$$

These formulations are well known in the geophysical community, see [4].

In the standard (Gauss-Newton) approach, update $p$ is the solution of the normal system of linear equations,

$$(J^*W^TWJ + \varepsilon L)p = -g. \quad (3)$$

Matrix $J$ is dense and so large that the use of direct factorization for 3D problems is untractable even on modern supercomputers. The largest singular value of $J$ depends on the mesh step size through $A$. Thus, for a fixed $\varepsilon$, the condition number of the systems matrix in (3) grows rapidly with problem size, which is very unfavorable for iterative solvers.

One of the few practical options to mitigate computational burden is the use of “background” Jacobian. In this approach, one fixes a background velocity distribution, $c_b$, such that the forward problem is easier to compute. Then the system matrix in (3) is replaced with matrix

$$J_b^*W^TWJ_b + \varepsilon L \quad (4)$$

with $J_b$ being the Jacobian at the background velocity distribution $c_b$. The computation load decreases considerably when $c_b$ is a constant value or a layered medium, for which the forward problem is fast. Unfortunately, as velocity distribution is updating, the efficiency of Newton’s step deteriorates.

It can be shown that (3) is equivalent to the following system of linear equations,

$$\begin{bmatrix}
\varepsilon L & -P^* & O & A \\
-P & O & A^* & Q^TW^TWQ \\
0 & A^* & Q^TW^TWQ \\
\end{bmatrix}
\begin{bmatrix}
p \\
\lambda \\
v \\
\end{bmatrix}
= \begin{bmatrix}
-\varepsilon Lm_n \\
0 \\
Q^TW^TWr \\
\end{bmatrix}, \quad (5)$$

where matrix $P$ corresponds to $u$ computed for $m = m_n$. Let us show that (5) is indeed equivalent to (3). First, we exclude $\lambda$ using the first equation in (5). The second and the third equations become

$$-Pp + Av = 0,$$

$$\varepsilon A^*(P^*)^{-1}L(p + m_n) + Q^TW^TWQv = Q^TW^TWr, \quad (6)$$

Now, we express $v$ from the first equation of (6) and substitute it to the second one,

$$\varepsilon L(p + m_n) + P^*(A^*)^{-1}Q^TW^TWQA^{-1}Pp = P^*(A^*)^{-1}Q^TW^TWr.$$

Using definition (4), we rewrite it as

$$\varepsilon L(p + m_n) + J^*Jp = J^*Wr,$$

which is (3).

System (4) is 3 times bigger than (3). It, however, has considerable advantages over the normal system: (a) it is sparse, and (b) its blocks correspond to the discretized PDE (Helmholtz’s equation). Collectively, it means that an efficient iterative solver can be devised.
Figure 1. Initial velocity distribution (left) and the result of inversion (right) at a frequency of 16 Hz

Figure 2. The history of outer iterations. The numbers denote angular frequency $\omega$ in rad/s.

3. Numerical experiments

To verify the presented formulas, we created a Newton-based inversion in the two-dimensional case. Critical parts of the code, including the forward problem, were implemented in C++. The optimization part was programmed in Python. Here we solve the forward problem by a sparse direct solver.

A part of the Marmousi model has been taken as a true model; see Figure 1. We placed 40 sources and 80 receivers at a depth of 60 m. A uniform finite-difference grid with a step of 20 m was employed. The step of the computational grid was 20 m. We simulate input data in the frequency range from 1.6 Hz to 16 Hz and supply them to the inversion as input data.

We solve inverse problems progressively, from low to high frequencies. The algorithms tried to make as many updates at a single frequency as possible, switching to the next frequency when the misfit was below 1%. Figure 2 depicts history of outer (Newton) iterations. The final model, reconstructed by the Gauss-Newton method, is shown in Figure 1.

The two velocity models have a good match, which is expected since it was a purely synthetic
Figure 3. Comparison of Newton’s update to $c^{-2}$ for a single source computed with (3) (left) and with (5) (right). The source located at $x=100$ with the frequency $1.6$ Hz.

example with precise data and with no noise. To verify formula (5), we compared the Newtonian steps calculated in two ways for individual sources. The comparison for a single source is shown in Figure 2. The two images are almost identical apart from a slight difference in amplitude visible in the bottom-left corner. This difference is likely to be related to the difference in the accuracy to which linear systems were solved.

Conclusions
We have proposed a new method for accelerating Newton seismic full-wave inversion. In contrast with the standard approach with a solution to the normal system, the problem is reduced to a $3 \times 3$ block sparse matrix, which can, at least in principle, be more easily treated with iterative solvers. Our numerical example demonstrated that the two approaches produce almost equivalent updates as predicted theoretically. Based on the results obtained, we are very optimistic that the new approach will accelerate Newton-based seismic data inversions.

Acknowledgements
This research was supported by the Russian Science Foundation, project no. 21-11-00139. The authors acknowledge computational resources granted by Complex for Simulation and Data Processing for Mega-Science Facilities at NRC “Kurchatov Institute”, http://ckp.nrcki.ru/; and Skoltech CDISE’s high-performance computing cluster, Zhores [5].

References
[1] J Virieux and S Operto 2009 Geophysics 74 WCC1–WCC26.
[2] L Metivier, R Brossier, Operto, and J Virieux 2017 SIAM Review 59(1) 153–195
[3] N Yavich, N Khokhlov, M Malovichko, and M S Zhdanov 2021 Computers and Mathematics with Applications 86 63–72.
[4] R E Plessix 2006 Geophysical Journal International 167 495–503.
[5] I Zacharov, R Arslanov, M Gunin, D Stefonishin, A Bykov, S Pavlov, O Panarin, A Malutin, S Rykovanov, and M Fedorov 2019 Zhores—Petaflops supercomputer for data-driven modeling, machine learning and artificial intelligence installed in Skolkovo Institute of Science and Technology. Open Eng. (9) 512–520