Preconditioning strategies for truncated Newton full-waveform inversion

Bruno S. Silva*, Jessé C. Costa, UFPA, Brazil

Abstract

Truncated Newton full-waveform inversion is an attractive alternative to conventional gradient-based optimization algorithms. This method accounts for the Hessian within the inversion, however, it is implemented in a "Hessian-free" fashion, without explicit calculation or storage of the Hessian matrix. At each iteration, we obtain the search direction through a conjugate-gradient (CG) solution of the Newton linear system, which requires only evaluations of Hessian-vector products. Due to the additional cost associated with inner CG iterations, it is indispensable applying a preconditioning strategy on CG algorithm to improve its convergence, reducing the number of CG iterations. In this work, we study two preconditioned CG schemes. We propose a scheme based on model reparameterization that adopts a preconditioner operator that combines smoothness and the illumination compensation effect of pseudo-Hessian. We also investigate a more conventional preconditioning scheme that uses only the pseudo-Hessian preconditioner. The numerical experiments show that preconditioning using model reparameterization combining pseudo-hessian compensation followed by a smoothing outperforms the more conventional preconditioning using exclusively pseudo-Hessian operator.

Introduction

Full-waveform inversion (FWI) is recognized as a powerful tool to estimate high-resolution velocity models of the subsurface by iteratively minimizing the misfit between the observed and synthetic seismic data (Virieux and Operto, 2009). This nonlinear optimization problem is commonly solved through first-order gradient-based algorithms, such as the steepest-descent and nonlinear conjugate gradient methods, or the L-BFGS quasi-Newton algorithm.

Recently, second-order optimization algorithms, as the truncated Newton and truncated Gauss-Newton methods, have been explored in FWI (Pan et al., 2017; Matharu and Sacchi, 2019; Liu et al., 2020). These approaches use the gradient and Hessian of the misfit function in the inversion. The Hessian operator has the potential to improve the quality of the model estimation and the convergence, refocusing the information on poorly illuminated parameters. In addition, it better account multiscattered wavefields, providing benefits when highly contrasted media are investigated (Métivier et al., 2017).

The truncated Newton and truncated Gauss-Newton method for FWI has been developed based on the second-order adjoint state formulation (Fichtner and Trampert, 2011; Métivier et al., 2017). At each inversion iteration, the model update is obtained by approximately solving the Newton linear system using a matrix-free conjugate-gradient (CG) algorithm. Thus, only the action of the Hessian operator on an arbitrary vector (i.e., Hessian-vector product) is required, instead of forming the Hessian operator explicitly.

Due to the high computational cost of the evaluation of the Hessian action, it is important to improve the convergence of the CG algorithm, in order to reduce the number of CG iterations. This can be achieved by applying an effective preconditioning strategy on CG algorithm. Preconditioning makes the problem better conditioned, and consequently, it can improve the convergence significantly (Nocedal and Wright, 2006).

In this study, we investigated two preconditioning strategies for truncated Newton and truncated Gauss-Newton FWI. First, we analyze a more conventional preconditioned CG scheme (Golub and Van Loan, 2013), in which we use a pseudo-Hessian preconditioner (Shin et al., 2001). Secondly, we develop a preconditioned CG scheme based on model reparameterization (Harlan, 1995), in which we adopt as preconditioner an operator that combines smoothness and the illumination compensation effect of pseudo-Hessian. The numerical examples compare the performance of both schemes. We find that preconditioning based on model reparameterization outperforms the more conventional preconditioning strategy that uses only the pseudo-Hessian operator.

Method

Acoustic full-waveform inversion attempts to obtain physical parameters of the subsurface by minimizing the misfit between observed and modeled wavefields using the wave equation

$$\frac{1}{c(\mathbf{x})^2}\frac{\partial^2 p(t,\mathbf{x};\mathbf{x_s})}{\partial t^2} - \nabla^2 p(t,\mathbf{x};\mathbf{x_s}) = s(t;\mathbf{x_s}), \qquad (1)$$

where $c(\mathbf{x})$ is the velocity model, $p(t,\mathbf{x};\mathbf{x_s})$ is the pressure field and $s(t;\mathbf{x_s})$ represents the source pulse injected at position $\mathbf{x_s}$.

The inverse problem consists in finding the model parameter $\mathbf{m}$ that minimizes the least-squares misfit

functional

$$J(\mathbf{m}) = \frac{1}{2} \sum_s \sum_r \left\| p^{obs}(t, \mathbf{x_r}; \mathbf{x_s}) - p(t, \mathbf{x_r}; \mathbf{x_s}; \mathbf{m}) \right\|^2, \quad (2)$$

where $p^{obs}(t, \mathbf{x_r}; \mathbf{x_s})$ is the observed wavefield at receivers position $\mathbf{x}_r$ and $p(t, \mathbf{x_r}; \mathbf{x_s})$ is the modeled wavefield at position $\mathbf{x}_r$. We consider as model parameterization the squared slowness $\mathbf{m} = \frac{1}{c(\mathbf{x})^2}$.

The Newton optimization approach is derived from the second-order Taylor series expansion of the misfit functional

$$J(\mathbf{m} + \Delta\mathbf{m}) = J(\mathbf{m}) + \Delta\mathbf{m}^\mathsf{T}\mathbf{g}(\mathbf{m}) + \frac{1}{2}\Delta\mathbf{m}^\mathsf{T}\mathbf{H}(\mathbf{m})\Delta\mathbf{m}, \quad (3)$$

in which $\mathbf{g}(\mathbf{m}) = \frac{\partial J}{\partial \mathbf{m}}$ denotes the gradient and $\mathbf{H}(\mathbf{m})$ denotes the Hessian matrix of misfit functional. The search direction $\Delta\mathbf{m}$ is obtained by solving the linear system

$$\mathbf{H}(\mathbf{m})\Delta\mathbf{m} = -\mathbf{g}(\mathbf{m}). \quad (4)$$

Thus, the solution that minimizes equation (2) is iteratively computed updating an initial model along a direction $\Delta\mathbf{m}$:

$$\mathbf{m}^k = \mathbf{m}^{k-1} + \gamma^k \Delta\mathbf{m}^k, \quad (5)$$

where k denotes the iteration number and γ is a scalar step length computed through a line-search method that satisfies the weak Wolfe conditions (Nocedal and Wright, 2006).

The truncated Newton method computes an approximate solution of the linear system (4) using a conjugate-gradient (CG) algorithm. This approach requires only the action of the Hessian operator on an arbitrary vector in the model space (i.e., Hessian-vector product), instead of forming the Hessian operator explicitly.

In order to implement second-order optimization algorithms, we need the gradient of the objective function (2) and the action of the Hessian operator on an arbitrary vector. Using the adjoint-state method (Plessix, 2006; Chavent, 2010) we can compute the gradient by solving firstly the forward wave equation (1) and then the adjoint equation backward in time

$$\frac{1}{c(\mathbf{x})^2} \frac{\partial^2 \Lambda(t, \mathbf{x}; \mathbf{x_s})}{\partial t^2} - \nabla^2 \Lambda(t, \mathbf{x}; \mathbf{x_s}) = e(t, \mathbf{x_r}; \mathbf{x_s}), \quad (6)$$

where $\Lambda(t, \mathbf{x}; \mathbf{x_s})$ is the adjoint wavefield and the source term, $e(t, \mathbf{x_r}; \mathbf{x_s}) = \sum_r [p^{obs}(t, \mathbf{x}; \mathbf{x_s}) - p(t, \mathbf{x}; \mathbf{x_s})] \delta(\mathbf{x} - \mathbf{x_r})$, is the data residual. Having computed the wavefields, the gradient with respect to the model parameter $\mathbf{m}$ (squared slowness) is obtained by cross-correlation

$$\mathbf{g}(\mathbf{m}) = \sum_s \int_0^T dt \, \Lambda(t, \mathbf{x}; \mathbf{x_s}) \frac{\partial^2 p(t, \mathbf{x}; \mathbf{x_s})}{\partial t^2}. \quad (7)$$

The Hessian-vector product can be derived through the second order adjoint-state method (Fichtner and Trampert, 2011). Constructing the Hessian-vector product $\mathbf{H}(\mathbf{m})\mathbf{u}$ requires the computation of four different wavefields: the incident field $p(t, \mathbf{x})$ by solving equation (1), the adjoint field $\Lambda(t, \mathbf{x})$ by solving equation (6), and two scattered

wavefields, $\Gamma_1(t, \mathbf{x})$ and $\Gamma_2(t, \mathbf{x})$, by solving additional forward and adjoint problems. The scattered forward field $\Gamma_2(t, \mathbf{x})$ is the solution of

$$\frac{1}{c^2(\mathbf{x})} \frac{\partial^2 \Gamma_2(t, \mathbf{x}; \mathbf{x_s})}{\partial t^2} - \nabla^2 \Gamma_2(t, \mathbf{x}; \mathbf{x_s}) = -u(\mathbf{x}) \frac{\partial^2 p(t, \mathbf{x}; \mathbf{x_s})}{\partial t^2}, \quad (8)$$

where $u(\mathbf{x})$ represents the arbitrary vector $\mathbf{u}$ in the model space, note this acts as scattering source. The scattered adjoint field $\Gamma_1(t, \mathbf{x})$ is obtained by solving backward in time

$$\frac{1}{c^2(\mathbf{x})} \frac{\partial^2 \Gamma_1(t, \mathbf{x}; \mathbf{x_s})}{\partial t^2} - \nabla^2 \Gamma_1(t, \mathbf{x}; \mathbf{x_s}) = -\left(u(\mathbf{x}) \frac{\partial^2 \Lambda(t, \mathbf{x}; \mathbf{x_s})}{\partial t^2} \right.$$
$$\left. + \sum_r \delta(\mathbf{x} - \mathbf{x_r}) \Gamma_2(t, \mathbf{x}; \mathbf{x_s}) \right). \quad (9)$$

Using the computed wavefields, the Hessian-vector product with respect to squared slowness model $\mathbf{m}$ is given by

$$\mathbf{H}(\mathbf{m})\mathbf{u} = \sum_s \left[\int_0^T dt \, \Gamma_1(t, \mathbf{x}; \mathbf{x_s}) \frac{\partial^2 p(t, \mathbf{x}; \mathbf{x_s})}{\partial t^2} \right.$$
$$\left. + \int_0^T dt \, \Gamma_2(t, \mathbf{x}; \mathbf{x_s}) \frac{\partial^2 \Lambda(t, \mathbf{x}; \mathbf{x_s})}{\partial t^2} \right]. \quad (10)$$

The computational cost of this operation is twice the gradient cost, since two extra wave-propagation problems are solved.

Algorithm 1: full Newton Hessian-vector product computation

---

for $t = 1$ *to* Nt **do**
   update incident field $p(t, \mathbf{x})$;
   update scattered field $\Gamma_2(t, \mathbf{x})$;
end
for $t = Nt$ *to* 1 **do**
   update adjoint field $\Lambda(t, \mathbf{x})$;
   update scattered field $\Gamma_1(t, \mathbf{x})$;
   reconstruct fields $p(t, \mathbf{x})$ and $\Gamma_2(t, \mathbf{x})$;
   build the Hessian-vector product $\mathbf{H}(\mathbf{m})\mathbf{u}$ in
     time-domain;
end

Alternatively, truncated Gauss-Newton method is based on an approximation of the full Hessian operator. The Gauss-Newton Hessian approximation neglects the second-order terms of full Hessian, furthermore, it is always semipositive definite, whereas the full Hessian may not be semipositive definite (Métivier et al., 2017). In practice, this approach has the advantage of having a lower computational cost compared to truncated Newton method, as the action of Gauss-Newton Hessian requires the computation of three different wavefields: the incident field $p(t, \mathbf{x})$ by solving equation (1), the scattered field $\Gamma_2(t, \mathbf{x})$ by solving equation (8), and the scattered field $\Gamma_1(t, \mathbf{x})$ by the solving backward in time

$$\frac{1}{c^2(\mathbf{x})} \frac{\partial^2 \Gamma_1(t, \mathbf{x}; \mathbf{x_s})}{\partial t^2} - \nabla^2 \Gamma_1(t, \mathbf{x}; \mathbf{x_s}) = -\sum_r \delta(\mathbf{x} - \mathbf{x_r}) \Gamma_2(t, \mathbf{x}; \mathbf{x_s}). \quad (11)$$

Then the Gauss-Newton Hessian-vector product can be given by

$$\mathbf{H}_{GN}(\mathbf{m})\mathbf{u} = \sum_s \int_0^T dt \, \Gamma_1(t, \mathbf{x}; \mathbf{x}_s) \frac{\partial^2 p(t, \mathbf{x}; \mathbf{x}_s)}{\partial t^2}. \quad (12)$$

Note that the expression for the Hessian-vector product reduced to only one term, therefore, there is no need to recompute field $\Gamma_2(t, \mathbf{x}; \mathbf{x}_s)$ for the cross-correlation as in expression (10). Furthermore this operation does not require the computation of adjoint field $\Lambda(t, \mathbf{x})$.

Algorithm 2: Gauss-Newton Hessian-vector product computation

for $t = 1$ *to* Nt **do**
  update incident field $p(t, \mathbf{x})$;
  update scattered field $\Gamma_2(t, \mathbf{x})$;
end
for $t = Nt$ *to* 1 **do**
  update scattered field $\Gamma_1(t, \mathbf{x})$;
  reconstruct fields $p(t, \mathbf{x})$;
  build the Hessian-vector product $\mathbf{H}_{GN}(\mathbf{m})\mathbf{u}$ in
    time-domain;
end

Solving the linear system by preconditioned CG algorithm

The conjugate-gradient (CG) method can be used to solver the linear system which determines the search direction $\Delta \mathbf{m}$. However, the Hessian matrix is often ill-conditioned and it is not a positive-definite operator. In order to ensure the positive-definiteness and stabilize the solution, we apply a damping term to the Hessian in eq. (4), this leads to a damped linear system

$$(\mathbf{H} + \lambda \mathbf{D})\Delta \mathbf{m} = -\mathbf{g}, \quad (13)$$

where λ is a scalar damping parameter, which ideally should be the smallest value that ensures the stability of the solution; $\mathbf{D}$ represents a weight matrix, which the simplest choice is the identity matrix $\mathbf{I}$. The solution to system of equations (13) corresponds to the minimization of a quadratic function with a penalty term

$$\phi(\Delta \mathbf{m}) = \frac{1}{2}\Delta \mathbf{m}^\mathsf{T} \mathbf{H} \Delta \mathbf{m} + \Delta \mathbf{m}^\mathsf{T} \mathbf{g} + \frac{\lambda}{2}\Delta \mathbf{m}^\mathsf{T} \mathbf{D} \Delta \mathbf{m}. \quad (14)$$

The CG method may suffer from slow convergence. In order to improve the conditioning of linear system and accelerate the convergence, we can use preconditioning. Commonly, preconditioning is introduced by multiplying both sides of the linear system by a suitable preconditioner $\mathbf{P}$ (Golub and Van Loan, 2013). Choosing $\mathbf{D} = \mathbf{I}$ in equation (13), the preconditioned linear system can be written as

$$\mathbf{P}(\mathbf{H} + \lambda \mathbf{I})\Delta \mathbf{m} = -\mathbf{P}\mathbf{g}. \quad (15)$$

For the effectiveness of this approach, the operator $\mathbf{P}$ is chosen so that the condition number of $\mathbf{PH}$ is less than the condition number of $\mathbf{H}$. Ideally, $\mathbf{P}$ should be an approximation of the inverse Hessian, so one convenient

preconditioner is based on pseudo-Hessian operator (Shin et al., 2001) that can be cast as

$$\mathscr{H}(\mathbf{m}) = \sum_s \int_0^T dt \left(\frac{\partial^2 p(t, \mathbf{x}; \mathbf{x}_s)}{\partial t^2}\right)^2. \quad (16)$$

The pseudo-Hessian has shown to be useful in FWI since it is widely used to preconditioning gradient-based algorithms. Using pseudo-Hessian we can obtain a diagonal approximation of the inverse Hessian

$$\mathbf{P}^{PH} = \text{diag}\left(\frac{1}{\mathscr{H} + \theta \max\{\mathscr{H}\}}\right), \quad (17)$$

where $\theta \in [0, 1]$ is a threshold parameter that is chosen to avoid division by very small numbers.

Another approach to preconditioning a linear system is based on model reparameterization (Harlan, 1995; Claerbout and Fomel, 2008). From linear system with damping term (13), we can introduce preconditioning via a change of variable

$$\Delta \mathbf{m} = \mathbf{P}\mathbf{v}, \quad (18)$$

where $\mathbf{v}$ is a new variable and $\mathbf{P}$ is the preconditioner. Choosing as weight matrix $\mathbf{D} = \mathbf{P}^{-1}$, we get a alternative preconditioned linear system

$$(\mathbf{HP} + \lambda \mathbf{I})\mathbf{v} = -\mathbf{g}. \quad (19)$$

In this approach, the preconditioner can be chosen to enforce a desirable behavior in the solution.

We investigate the effect of two preconditioning strategies in truncated Newton and truncated Gauss-Newton FWI. In the first investigated strategy, we apply the pseudo-Hessian preconditioner, $\mathbf{P}^{PH}$, to the linear system (15). Accordingly, the search direction is obtained using the preconditioned CG algorithm displayed in Algorithm 3. This represents a more conventional preconditioning scheme applied to truncated Newton FWI (Pan et al., 2017; Métivier et al., 2017).

The second preconditioning strategy estimates the search direction by solving the linear system (19). In this case, we propose as preconditioner a combined operator

$$\mathbf{P}^S = \mathbf{S}\mathbf{P}^{PH}, \quad (20)$$

in which the pseudo-Hessian preconditioner $\mathbf{P}^{PH}$ acts to compensate for subsurface illumination and $\mathbf{S}$ behaves like a smoothing filter that acts to enforces smoothness in the search direction. As shown by Fomel and Claerbout (2003), smoothing preconditioning can provide faster convergence at early iterations by focusing on the low-wavenumbers of the expected solution of the linear system. Algorithm 4 shows the corresponding preconditioned CG scheme.

Numerical examples

In this section, we apply truncated Newton (FN) and truncated Gauss-Newton (GN) FWI on a simple Gaussian-anomaly model, followed by a modified Marmousi II model. We also illustrate the effect of two different preconditioning

Algorithm 3: Preconditioned CG scheme to solve $\mathbf{P}\left(\mathbf{H}+\lambda\mathbf{I}\right)\Delta\mathbf{m}=-\mathbf{P}\mathbf{g}$

Input:
current model $\mathbf{m}$;
gradient $\mathbf{g}(\mathbf{m})$;
pseudo-Hessian preconditioning operator $\mathbf{P}=\mathbf{P}^{PH}$;
damping parameter λ;
Output: search direction $\Delta\mathbf{m}$
Initialization:
$\Delta\mathbf{m}^0=0$, $\mathbf{r}^1=-\mathbf{g}$, $\mathbf{d}^1=\mathbf{P}\mathbf{r}^1$, $j=1$;
while $j\leq N_{max}$ & $\varepsilon>\varepsilon_{min}$ **do**

$\quad\quad\mathbf{s}^j=\mathbf{H}\left(\mathbf{m}\right)\mathbf{d}^j$

$\quad\quad\alpha=\dfrac{\langle\mathbf{r}^j,\mathbf{P}\mathbf{r}^j\rangle}{\langle\mathbf{d}^j,\mathbf{s}^j\rangle+\lambda\langle\mathbf{d}^j,\mathbf{d}^j\rangle}$

$\quad\quad\Delta\mathbf{m}^j=\Delta\mathbf{m}^{j-1}+\alpha\mathbf{d}^j$

$\quad\quad\mathbf{r}^{j+1}=\mathbf{r}^j-\alpha\left(\mathbf{s}^j+\lambda\mathbf{d}^j\right)$

$\quad\quad\beta=\dfrac{\langle\mathbf{r}^{j+1},\mathbf{P}\mathbf{r}^{j+1}\rangle}{\langle\mathbf{r}^j,\mathbf{P}\mathbf{r}^j\rangle}$

$\quad\quad\mathbf{d}^{j+1}=\mathbf{P}\mathbf{r}^{j+1}+\beta\mathbf{d}^j$

$\quad\quad\varepsilon^j=\dfrac{\|\mathbf{r}^{j+1}\|^2}{\|\mathbf{g}\|^2}$

$\quad\quad j=j+1$

end

Algorithm 4: Preconditioned CG scheme to solve $\left(\mathbf{H}\mathbf{P}+\lambda\mathbf{I}\right)\mathbf{v}=-\mathbf{g}$

Input:
current model $\mathbf{m}$;
gradient $\mathbf{g}(\mathbf{m})$;
combined preconditioning operator $\mathbf{P}=\mathbf{S}\mathbf{P}^{PH}$;
damping parameter λ;
Output: search direction $\Delta\mathbf{m}$
Initialization:
$\mathbf{v}^0=0$, $\mathbf{r}^1=\mathbf{d}^1=-\mathbf{g}$, $j=1$;
while $j\leq N_{max}$ & $\varepsilon>\varepsilon_{min}$ **do**

$\quad\quad\mathbf{s}^j=\mathbf{H}\left(\mathbf{m}\right)\mathbf{P}\mathbf{d}^j$

$\quad\quad\alpha=\dfrac{\langle\mathbf{r}^j,\mathbf{r}^j\rangle}{\langle\mathbf{d}^j,\mathbf{s}^j\rangle+\lambda\langle\mathbf{d}^j,\mathbf{d}^j\rangle}$

$\quad\quad\mathbf{v}^j=\mathbf{v}^{j-1}+\alpha\mathbf{d}^j$

$\quad\quad\mathbf{r}^{j+1}=\mathbf{r}^j-\alpha\left(\mathbf{s}^j+\lambda\mathbf{d}^j\right)$

$\quad\quad\beta=\dfrac{\langle\mathbf{r}^{j+1},\mathbf{r}^{j+1}\rangle}{\langle\mathbf{r}^j,\mathbf{r}^j\rangle}$

$\quad\quad\mathbf{d}^{j+1}=\mathbf{r}^{j+1}+\beta\mathbf{d}^j$

$\quad\quad\varepsilon^j=\dfrac{\|\mathbf{r}^{j+1}\|^2}{\|\mathbf{g}\|^2}$

$\quad\quad j=j+1$

end
$\Delta\mathbf{m}=\mathbf{P}\mathbf{v}$

schemes (Algorithm 3 and Algorithm 4) in these methods and compare the results with the steepest-descent (SD) and L-BFGS method. In the experiments, the stopping criteria for the CG algorithms were $N_{max}=10$ and $\varepsilon_{min}=0.1$. The step lengths γ were computed through line-search algorithm from Moré and Thuente (1994).

The first example is a Gaussian-anomaly model (Fig. 1a) that consists of 51 x 100 grid, with a grid interval of 10 m in the horizontal and vertical directions. The data set were modeled using 49 shots with an interval of 20 m and a depth of 20 m; 100 receivers were arranged from 10 to 1000 m every 10 m at the depth of 20 m. The source function was a Ricker wavelet with a dominant frequency of 10 Hz. The initial model was a constant velocity of 2000 m/s.

Figure 1 shows the models recovered in 40 iterations with SD (Fig. 1b), L-BFGS (Fig. 1c), FN using preconditioning schemes described in Algorithm 3 (Fig. 1d) and Algorithm 4 (Fig. 1e), GN using preconditioning schemes described in Algorithm 3 (Fig. 1f) and Algorithm 4 (Fig. 1g). We can see that the recovered anomalies with second order optimization methods, FN and GN, are closer to the true model than results obtained with SD and L-BFGS.

The convergence rate and computational cost of these optimization algorithms are analyzed through two types of convergence curves, one in terms of nonlinear iterations and another in terms of direct problems solved. Note that gradient computation requires the solution of two direct problems, full Hessian action and Gauss-Newton Hessian action requires the solution of four and three direct problems, respectively. In addition, for each nonlinear iteration of FN and GN, there are inner CG linear iterations.

Figure 2a shows that FN and GN have considerably higher convergence rates than SD and L-BFGS. Nonetheless, Figure 2b illustrates the high computational cost associated with these second-order optimization methods. Comparing the performance of two preconditioning schemes applied to FN and GN, we note that preconditioning based on model reparameterization (Algorithm 4) has a better convergence rate and lower computational cost. We also observe that GN experiments provide better performance compared to the FN experiments. The second example corresponds to a portion of Marmousi II (Martin et al., 2002). The model, shown in Figure 3a, consists of 151 x 401 grid with a grid interval of 10 m in the horizontal and vertical directions. The data set was modeled using 49 shots with an interval of 80 m and a depth of 10 m. A total of 401 receivers were arranged from 0 to 4000 m every 10 m at the depth of 10 m. The source function was a Ricker wavelet with a dominant frequency of 8 Hz.

Figure 3b shows the initial model, which is a smoothed version of the true model. We compare the inversion results obtained after 20 nonlinear iterations with L-BFGS (Fig. 3c), FN using preconditioning scheme described in Algorithm 3 (Fig. 3d) and Algorithm 4 (Fig. 3e), GN using preconditioning schemes described in Algorithm 3 (Fig. 3f) and Algorithm 4 (Fig. 3g). Note that the quality of the recovered models obtained with all methods is similar.

We compare the behavior of optimization methods through convergence curves shown in Figure 4. We can see that after 20 iterations the data misfit for FN and GN is significantly smaller compared to L-BFGS. The FN experiments show a slightly better convergence rate compared to the GN experiments, however, the computational cost of FN is higher.

The performance difference between preconditioning schemes is more evident in terms of number of direct problems solved (see Fig. 4b). We note that the preconditioning based on model reparameterization

(Algorithm 4) provides lower computational cost for both FN and GN.

Conclusions

We investigate preconditioning strategies in truncated Newton and truncated Gauss-Newton FWI. In these methods, the search direction is determined by solving the Newton linear system via CG algorithm. We present two preconditioning strategies that lead to different preconditioned CG schemes (Algorithm 3 and Algorithm 4).

Inversion experiments were carried out on a Gaussian-anomaly model and a modified Marmousi II model. The results obtained with second-order optimization methods (truncated Newton and truncated Gauss-Newton) were compared with gradient-based methods like steepest descent and L-BFGS. The experiments demonstrated that second-order methods provide considerably faster convergence, but they are computationally expensive. This is due to the additional cost of inner linear CG iterations per outer nonlinear iteration. Furthermore, the truncated Gauss-Newton appears to be more advantageous due to its lower computational cost compared to truncated Newton.

Regarding the investigated preconditioning strategies, the convergence curves show that the preconditioning scheme based on model reparameterization (Algorithm 4) provides lower computational cost since the number of direct problems solved is smaller compared to another scheme (Algorithm 3). This indicates a reduced number of inner CG iterations. Hence, our numerical experiments show
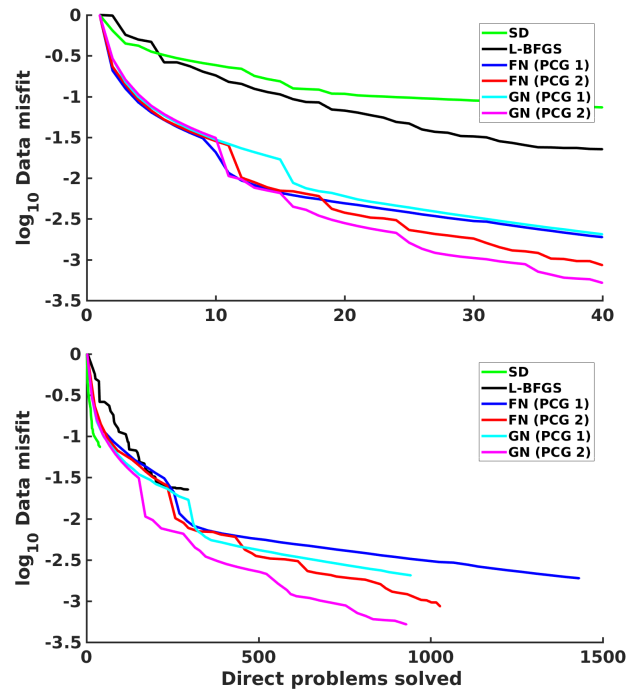
Figure 2: *Convergence curves for SD, L-BFGS, FN and GN. Preconditioning schemes summarized in Algorithm 3 and Algorithm 4 are referred to as PCG-1 and PCG-2, respectively.*
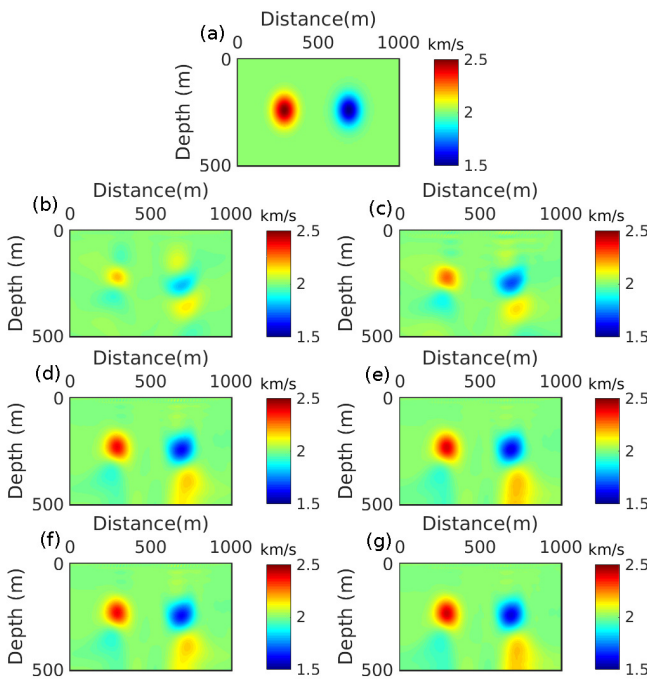
Figure 1: *(a) True Gaussian-anomaly model. Models recovered in 40 iterations with (b) SD, (c) L-BFGS, FN using preconditioning schemes from (d) Algorithm 3 and (e) Algorithm 4, GN using preconditioning scheme from (f) Algorithm 3 and (g) Algorithm 4.*
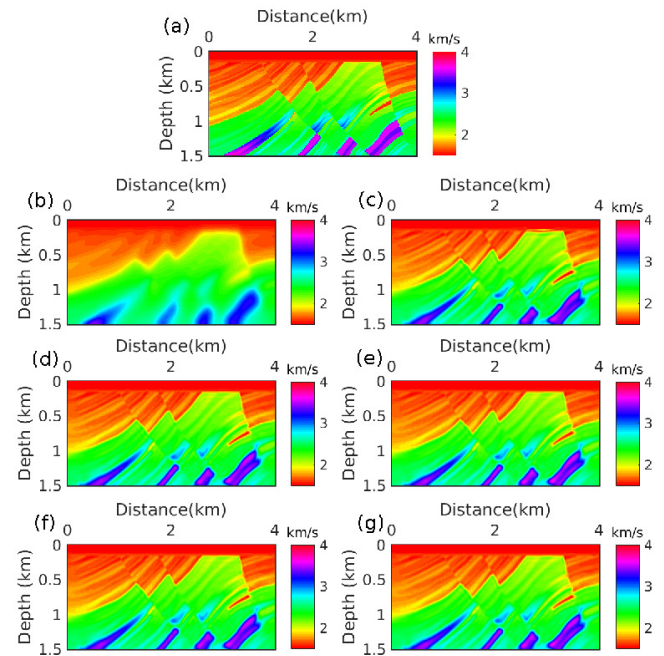
Figure 3: *(a) Modified Marmousi II model.(b) Initial model. Inversion results obtained with (c) L-BFGS, FN using preconditioning schemes from (d) Algorithm 3 and (e) Algorithm 4, GN using preconditioning scheme from (f) Algorithm 3 and (g) Algorithm 4.*
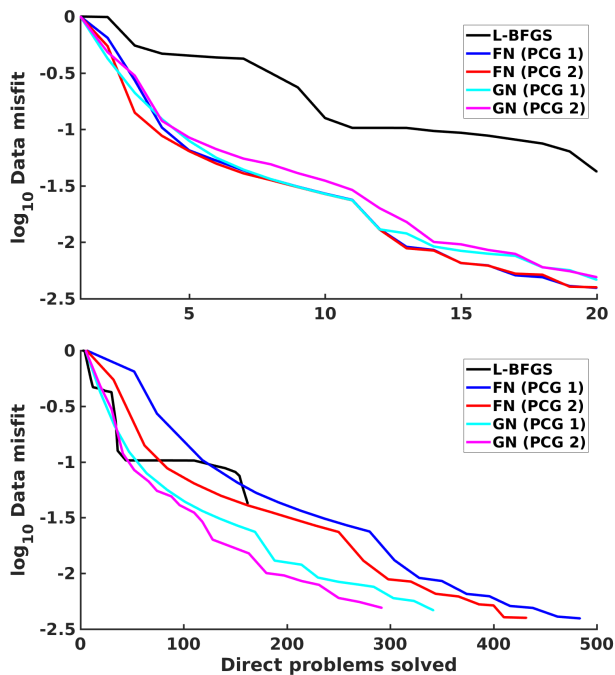
Figure 4: *Convergence curves for L-BFGS, FN and GN. Preconditioning schemes summarized in Algorithm 3 and Algorithm 4 are referred to as PCG-1 and PCG-2, respectively.*

that preconditioning based on model reparameterization clearly outperforms the more conventional pseudo-hessian preconditioning on truncated Newton FWI.

Acknowledgments

References

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

Claerbout, J. F., and S. Fomel, 2008, Image estimation by example: geophysical soundings image construction: multidimensional autoregression: Citeseer.

Fichtner, A., and J. Trampert, 2011, Hessian kernels of seismic data functionals based upon adjoint techniques: Geophysical Journal International, **185**, 775–798.

Fomel, S., and J. F. Claerbout, 2003, Multidimensional recursive filter preconditioning in geophysical estimation problems: Geophysics, **68**, 577–588.

Golub, G. H., and C. F. Van Loan, 2013, Matrix computations: Johns Hopkins University Press.

Harlan, W. S., 1995, Regularization by model reparameterization: Citeseer.

Liu, Y., Z. Wu, H. Kang, and J. Yang, 2020, Use of prismatic waves in full-waveform inversion with the exact hessian: Geophysics, **85**, R325–R337.

Martin, G. S., K. J. Marfurt, and S. Larsen, 2002, Marmousi-2: An updated model for the investigation of avo in structurally complex areas, *in* SEG Technical Program Expanded Abstracts 2002: Society of Exploration Geophysicists, 1979–1982.

Matharu, G., and M. Sacchi, 2019, A subsampled truncated-newton method for multiparameter full-waveform inversion: Geophysics, **84**, R333–R340.

Métivier, L., R. Brossier, S. Operto, and J. Virieux, 2017, Full waveform inversion and the truncated newton method: SIAM Review, **59**, 153–195.

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

Nocedal, J., and S. Wright, 2006, Numerical optimization: Springer Science & Business Media.

Pan, W., K. A. Innanen, and W. Liao, 2017, Accelerating hessian-free gauss-newton full-waveform inversion via l-bfgs preconditioned conjugate-gradient algorithm: Geophysics, **82**, R49–R64.

Plessix, R.-E., 2006, A review of the adjoint-state method for computing the gradient of a functional with geophysical applications: Geophysical Journal International, **167**, 495–503.

Shin, C., S. Jang, and D.-J. Min, 2001, Improved amplitude preservation for prestack depth migration by inverse scattering theory: Geophysical prospecting, **49**, 592–606.

Virieux, J., and S. Operto, 2009, An overview of full-waveform inversion in exploration geophysics: Geophysics, **74**, WCC1–WCC26.