4D Full –Waveform Metropolis Hastings Inversion Using a Local Acoustic Solver

Maria Kotsi, Memorial University of Newfoundland, Alison Malcolm, Memorial University of Newfoundland, and Gregory Ely, Massachusetts Institute of Technology*

SUMMARY

Time lapse seismic monitoring usually involves looking for small changes in localized regions. Quantifying the uncertainty related to these changes is important because it affects exploration and production decisions. Traditional methods that use pixel by pixel quantification with large models are computationally infeasible. We use a local acoustic solver in the area of interest, which allows for fast computation of the wavefield solves. This allows us to use a Metropolis Hastings algorithm in a Bayesian inversion to address the uncertainty that is present in the estimation of 4D velocity changes.

INTRODUCTION

Time-lapse (4D) seismic monitoring involves looking for small changes in localized regions. Accurate and high resolution imaging of these changes is crucial for production and drilling decisions. Full–Waveform Inversion (FWI) provides high resolution seismic imaging based on prestack data. In FWI, we iteratively try to match modelled seismic waveforms with raw seismic waveforms collected in the field (Tarantola, 1984; Virieux and Operto, 2009). FWI is extended to the 4D case via several schemes (Watanabe et al., 2005; Zheng et al., 2011; Maharramov and Biondi, 2014; Yang et al., 2014; Asnaashari et al., 2015). In this study we use only Double-Difference FWI (DDFWI). In DDFWI, to estimate a monitor velocity model we start from an inverted baseline model, and instead of the monitor data we use a set of composite data. The composite data are the data difference (monitor - baseline) added to the synthetic baseline data computed using the model recovered from a baseline FWI. In this way, the only signal that is not explained by the starting model in the composite data, is the 4D change. However, all 4D-FWI methods inherently have the same issue, — the non-convexity of the inverse problem— where multiple models can fit the data. The ambiguity of whether or not and to what extent we can trust our result can be addressed with uncertainty quantification. To do so, we need to understand what we know and what we don't know; "there are known unknowns and unknown unknowns" (Ma, 2010). In seismic imaging, there are several sources of uncertainty, varying from acquisition uncertainty to uncertainties in physical properties. Osypov et al. (2013) provide a strategy and overview for uncertainty quantification in seismic tomography that is related to oil and gas exploration and production. Poliannikov and Malcolm (2015) use a Bayesian framework to quantify uncertainty in migrated images considering the effects of velocity model and picking errors. Ely et al. (2018) proposes a bayesian framework with a fast forward solver based on the field expansion method to address uncertainties in seismic images. In (Kotsi and Malcolm, 2017) we studied model uncertainties in different 4D FWI schemes. That work highlights the need for a more robust statistical inversion technique that will allow us more sophisticated sampling to estimate a monitor velocity model. In this study, we setup a Metropolis Hastings algorithm that is appropriate for the 4D case, and is similar in a way to the Double-Difference scheme. We use the Marmousi model (Versteeg, 1994) to perform our numerical tests and we study the uncertainty that is present in the estimates of the 4D velocity change.

THEORY

In seismic imaging we are interested in finding the Earth model that best fits the data. In FWI terminology we use a least squares approach

$$\hat{\mathbf{m}} = \underset{m}{argmin}\{\sum_{x_s}||F(\mathbf{m};x_s)-\mathbf{d}(x_s)||_2^2\}, \tag{1}$$

where $\mathbf{m}$ is the model, $\mathbf{d}$ are the observed data, F is the forward wave solver, and x_s is the source location. For the DDFWI extension, equation 1 becomes

$$\begin{aligned}\hat{\mathbf{m}}_\mathbf{1} =& \underset{m_1}{argmin}\{\sum_{x_s}||(F(\mathbf{m_1};x_s)-F(\mathbf{m_0};x_s)) \\ &+(\mathbf{d_1}(x_s)-\mathbf{d_0}(x_s))||_2^2\}.\end{aligned} \tag{2}$$

where $\mathbf{m_0}$ is the inverted baseline model that is used as the starting model for the monitor inversion, $\mathbf{m_1}$ is the monitor model, $\mathbf{d_0}$ are the observed baseline data, and $\mathbf{d_1}$ are the observed monitor data.

Solving the forward problem in the entire domain while we are interested in a small subdomain (i.e. reservoir) is inefficient. Willemsen et al. (2016) develop a frequency based local domain solver that calculates wavefields identical to the ones produced by a full domain solver, and apply it to a salt boundary inversion example. Malcolm and Willemsen (2017) successfully apply the local domain solver to Double Difference FWI on a Marmousi test case. There, they observe that the local domain solver improves the computational cost and also uses fewer iterations in the inversion. In this study, we use that local solver in order to efficiently compute our statistical distributions.

Bayes' theorem describes the relationship between a hypothesis ($\mathbf{H}$) and given evidence ($\mathbf{E}$). In seismic imaging, this is expressed in terms of velocity models ($\mathbf{m}$) and observed data ($\mathbf{d}$),

$$p(\mathbf{m}|\mathbf{d}) = \frac{p(\mathbf{d}|\mathbf{m})p(\mathbf{m})}{p(\mathbf{d})}, \tag{3}$$

where $p(\mathbf{m}|\mathbf{d})$ is the quantity of interest for any probabilistic inversion, the probability that you obtain the model ($\mathbf{m}$) given the data ($\mathbf{d}$). $p(\mathbf{m}|\mathbf{d})$ is also called the posterior. $p(\mathbf{d}|\mathbf{m})$ is the

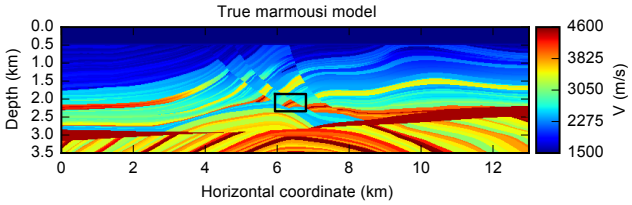

Figure 1: True baseline marmousi model, with the black box indicating the area of the local domain.

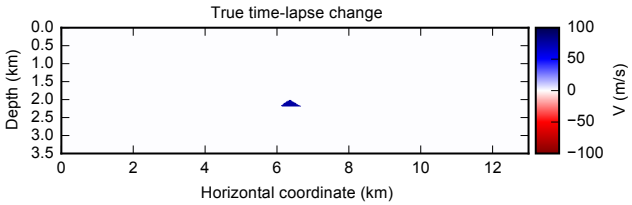

Figure 2: True time-lapse velocity change with a magnitude of 75 m/s

likelihood function and it is calculated by (Tarantola, 2005)

$$L(\mathbf{m}) \equiv p(\mathbf{d}|\mathbf{m}) \propto exp\left[-\frac{1}{2}(f(\mathbf{m})-\mathbf{d})^T\Sigma^{-1}(f(\mathbf{m})-\mathbf{d})\right], \quad (4)$$

which involves the forward wave solver $f(\mathbf{m})$. $p(\mathbf{m})$ is the model distribution that is used as an input information and $p(\mathbf{d})$ is considered to be a normalization constant. For an accurate posterior calculation, a large number of samples need to be generated, and then the first half are discarded in order to reduce the impact of the starting model (Brooks et al., 2011). Here, we use a Metropolis Hastings algorithm that doesn't require the computation of $p(\mathbf{d})$, rather than uses the history of the process to tune the proposal distribution (Haario et al., 2001). See (Ely et al., 2018) for a recent application of the Adaptive Metropolis Hastings algorithm in seismic imaging. Here we use the Metropolis Hastings algorithm to invert for 4D model changes from 4D seismic data. The pseudocode in Algorithm 1 provides a brief explanation of the algorithm's steps. The acceptance of the new proposal is assessed by the ratio of the likelihood functions of the proposal and current samples. If it is accepted, the proposal becomes the new current, and if it is rejected the current proposed model is reused. Since a large number of samples need to be evaluated, this means that a lot of forward wave solves needs to be evaluated for the computation of the likelihood functions (equation 4). To make this computationally feasible, we use the local domain solver.

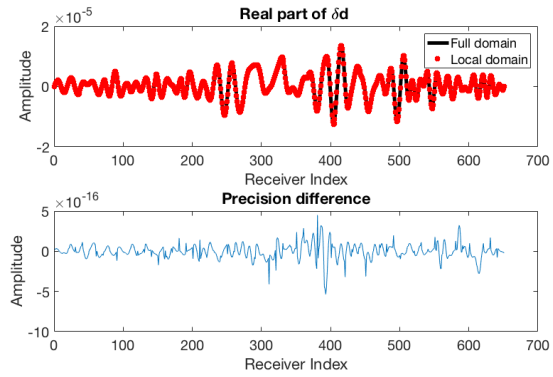

Figure 3: Comparison of the the data residuals created from one shot, at the frequency of 8.0 Hz in the full and local domain (top panel). The bottom panel is the difference between the the two data residuals from the top panel.

4D PROBLEM SETUP AND POSTERIOR CALCULATION

We now explain how we are going to setup our problem for the Metropolis Hastings algorithm. Let $\mathbf{m}$ be the velocity model and n be zero mean Gaussian noise with a covariance matrix Σ, then

$$\mathbf{d} = G(\mathbf{m}) + n, \quad (5)$$

where G is the forward modeling solver (i.e. the local domain solver here). In the 4D case, we have $\mathbf{d_1} = G(\mathbf{m_1}) + n_1$ for the baseline model, and $\mathbf{d_2} = G(\mathbf{m_2}) + n_2$ for the monitor. If we let $\delta\mathbf{m} = \mathbf{m_2} - \mathbf{m_1}$ and $\delta\mathbf{d} = \mathbf{d_2} - \mathbf{d_1}$, then our goal is to find the probability of the model difference given the two datasets,

$$p(\delta\mathbf{m}|\mathbf{d_1}, \mathbf{d_2}). \quad (6)$$

In this expression there is a hidden variable, $\mathbf{m_1}$. In order to calculate the distribution over $\delta\mathbf{m}$, we will first need to calculate the joint distribution $p(\mathbf{m_1}, \delta\mathbf{m}|\mathbf{d_1}, \mathbf{d_2})$ and then integrate over $\mathbf{m_1}$ to get the distribution on only $\delta\mathbf{m}$. This, however, would be computationally expensive since we have to sample for both $\delta\mathbf{m}$ and $\mathbf{m_1}$, where $\mathbf{m_1}$ is the baseline velocity model. We cannot recover $\mathbf{m_1}$ from the local solver because it computes the velocity only within the local domain whereas the baseline velocity model obviously exists throughout the full domain. Therefore, we will need an expression that is only in terms of $\delta\mathbf{m}$. To do so, we rewrite the forward modeling expression in equation 1 as,

$$\begin{aligned}\delta\mathbf{d} &= G(\mathbf{m_2}) + n_2 - G(\mathbf{m_1}) - n_1 \\ &= G(\mathbf{m_2}) - G(\mathbf{m_1}) + (n_2 - n_1) \\ &= G(\mathbf{m_1} + \delta\mathbf{m}) - G(\mathbf{m_1}) + (n_2 - n_1). \end{aligned} \quad (7)$$

The sum or difference of two zero mean Gaussians (n_1, n_2) is equal to a single Gaussian (n_3) with a covariance $\Sigma_3 = \Sigma_1 + \Sigma_2$. From this, we can rewrite equation 7 as

$$\delta\mathbf{d} = G(\mathbf{m_1} + \delta\mathbf{m}) - G(\mathbf{m_1}) + n_3. \quad (8)$$

If we let $F(\mathbf{m_1}, \delta\mathbf{m}) = G(\mathbf{m_1} + \delta\mathbf{m}) - G(\mathbf{m_1})$, equation 8 becomes

$$\delta\mathbf{d} = F(\mathbf{m_1}, \delta\mathbf{m}) + n_3. \quad (9)$$

This expression now is almost entirely in terms of $\delta\mathbf{m}$. If we now assume that $F(\mathbf{m_1}, \delta\mathbf{m})$ is independent of the initial model $\mathbf{m_1}$, then at any model update

$$F(\mathbf{m_{1a}}, \delta\mathbf{m}) = F(\mathbf{m_{1b}}, \delta\mathbf{m}), \quad (10)$$

where $\mathbf{m_{1a}} \approx \mathbf{m_{1b}}$. Then, the forward model is only dependent on the difference of models, difference of observed data, and the sum of some known covariance matrices,

$$\delta\mathbf{d} = F(\delta\mathbf{m}) + n_3. \quad (11)$$

Bayes' theorem from equation 3 now becomes

$$p(\delta\mathbf{m}|\delta\mathbf{d}) = \frac{p(\delta\mathbf{d}|\delta\mathbf{m})p(\delta\mathbf{m})}{p(\delta\mathbf{d})}, \quad (12)$$

and the likelihood function from equation 4 is

$$L(\delta\mathbf{m}) \equiv p(\delta\mathbf{d}|\delta\mathbf{m}) \propto \quad (13)$$

$$exp\left[-\frac{1}{2}(F(\delta\mathbf{m}) - \delta\mathbf{d})^T {\Sigma_3}^{-1}(F(\delta\mathbf{m}) - \delta\mathbf{d})\right]. \quad (14)$$

At each iteration i, we get a new proposal $\delta\mathbf{m}_*$ by adding a zero mean perturbation to the current ($\delta\mathbf{m_{i-1}}$). We evaluate the likelihood $L(\mathbf{m}_*)$ and if $L(\mathbf{m}_*) > L(\mathbf{m_{i-1}})$ we accept the new proposal and $\delta\mathbf{m_i} = \delta\mathbf{m}_*$. If $L(\mathbf{m}_*) < L(\mathbf{m_{i-1}})$ then we reject the proposal and $\delta\mathbf{m_i} = \delta\mathbf{m_{i-1}}$. In this study, we are changing only the perturbation $\delta\mathbf{m}$, therefore we have only one degree of freedom. We are also using a fixed step-size σ throughout the process, and hence the covariance used to get the new proposal is $C = \sigma I_d$ where d are the degrees of freedom; here $d = 1$ hence $C = \sigma$. The pseudocode in Algorithm 1 provides a summary of our methodology.

Algorithm 1 4D Metropolis Hastings algorithm

Require: $\delta\mathbf{m_0}$ ▷ initial perturbation
Require: N ▷ maximum number of iterations
Require: C ▷ covariance matrix
$L(\delta m_0)$
for $i = 1, ..., N$ **do**
  $n \leftarrow Normal(0, C)$ ▷ proposed jump
  $\delta m_* \leftarrow \delta m_{i-1} + n$ ▷ proposed model perturbation
  $L(m_*)$ ▷ get the likelihood of the proposal
  $\alpha_i = \frac{L(m_*)}{L(m_{i-1})}$ ▷ acceptance probablility
  $u \leftarrow U[0, 1]$
  if $u < \alpha_i$ **then**
    $\delta m_i \leftarrow \delta m_*$ ▷ accept proposal
  else
    $\delta m_i \leftarrow \delta m_{i-1}$ ▷ reject proposal
  end if
end for

NUMERICAL EXAMPLE

To perform our numerical example we use the standard marmousi model (Versteeg, 1994) as the baseline true model (Figure 1). We create the monitor true model by adding a perturbation of 75 m/s to one of the layers (Figure 2). We generate synthetic data at 8 Hz with a single shot and 651 equispaced receivers. We validate the accuracy of the local solver by computing the residual $\delta\mathbf{d}$ in the full and local domain (Figure 3). To keep the problem simple we add noise to the true residual $\delta\mathbf{d}$. We generate uncorrelated Gaussian noise in the frequency domain, and we define the noise lever r as the energy of the noise signal $\delta\mathbf{d_n}$ relative to the noiseless signal $\delta\mathbf{d}$,

$$r = \frac{\sum {\delta\mathbf{d_n}}^2}{\sum \delta\mathbf{d}^2}. \quad (15)$$

To assess the effect of noise in the convergence of the 4D Metropolis Hastings algorithm we compare two different noise levels. In the first case we consider a relatively high noise level where $r = 1.28$ (Figure 4), and in the second case a very low noise level of $r = 0.01$. In order to be able to use the

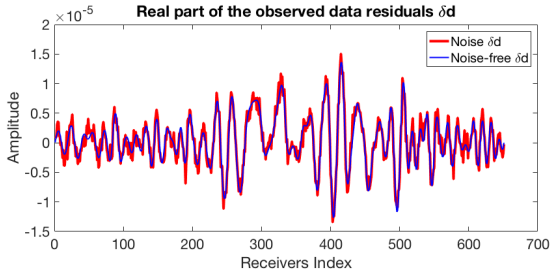

Figure 4: Comparison of the noisy signal with r = 1.28 with the noiseless signal

local solver, we first require the Green's functions computed on the background model in the full domain. Here, we are interested in exploring the uncertainty related to the estimation of the 4D velocity change. Therefore, we keep everything else exact, and we compute the background Green's functions on the true baseline model. Once these functions are available, we can compute the wavefield inside the local domain for any perturbations $\delta\mathbf{m}$ as many times as we want for very little computational cost. In our case here, for one shot at a single frequency we can have a wave solve at approximately 0.8 seconds. We start three different Markov chains with starting models $\delta\mathbf{m_0} = 0m/s$, $\delta\mathbf{m_0} = 75m/s$, and $\delta\mathbf{m_0} = 400m/s$. In the first case we start from the baseline model, in the second case we start from the monitor model, and in the third case we start with a perturbation far from the true. For each chain, we run the Metropolis Hastings algorithm for 50,000 iterations and we discard the first half. We do this for both noise levels. Figure 5 and Figure 6 show the resulting histograms. In the case of $r = 1.28$ we see that all chains have converged to a Gaussian distribution, however we notice that the chains have not converged to the true values. To understand why this might happen, we plot the likelihood values in terms of a range of perturbations (Figure 7). We see that the likelihood doesn't change significantly between perturbations of 40 and 90 m/s. This means that a range of values fit the data equally well, and hence the algorithm converges to these values. We see both chains starting $\delta\mathbf{m_0} = 75m/s$ and $\delta\mathbf{m_0} = 75m/s$ converge to the same range 80-100 m/s.

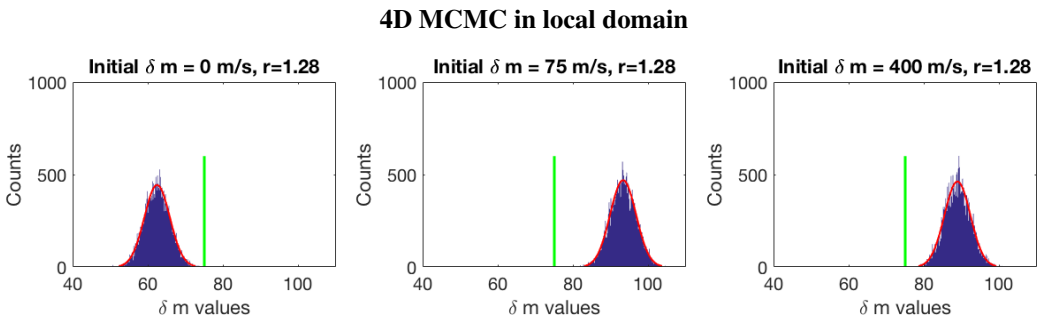

Figure 5: Histograms of the three Markov chains starting from a different initial guess. Here we plot only the second half of the chain, 25000, to reduce the dependency on the starting model. The red curve is the distribution that better fits the data (δm (m/s) values) and the green line indicates the true model. The noise level to all of them is r=1.28.

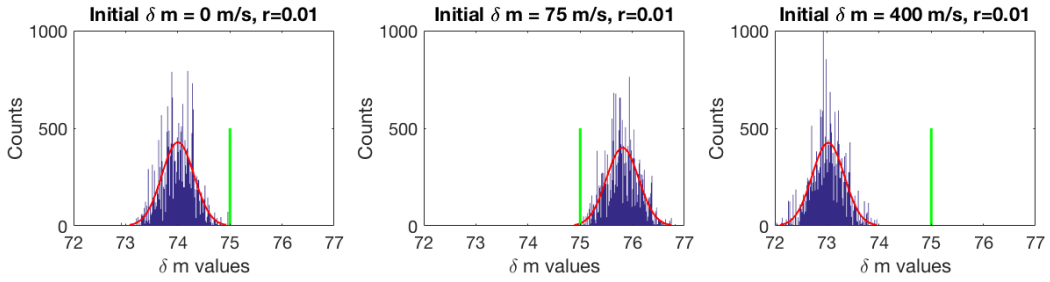

Figure 6: Histograms of the three Markov chains from Figure 5 with the noise level of r = 0.01. The red curve is the distribution that better fits our data and the green line shows where the true model is.

However we need to study further why this is not the case when $\delta\mathbf{m_0} = 0m/s$. When we repeat the same procedure for half a million iterations (Figure 8) we notice that one of the chains converges to values closer to the true, while the other one seems to converge to similar ranges as before. Our future efforts will be focused on understanding the source of this discrepancy and investigating the effects of noise into the starting model $\delta\mathbf{m_0}$. In the case when $r = 0.01$ there is a slightly different pattern. All chains have converged to similar values, with a deviation of $\pm 2m/s$ from the true value.

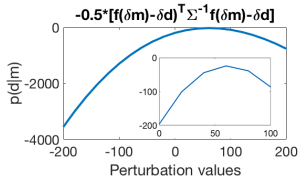

Figure 7: Likelihood evolution in terms of different perturbations when noise is present $r = 1.28$.

CONCLUSIONS

We propose a local acoustic solver for a fast 4D Metropolis Hastings inversion. In our model, one wave solve takes approximately 0.8 seconds in the local domain compared to approximately 1 hour in the full domain. Calculating the full posterior pixel by pixel would be impossible. We have created a framework to calculate the uncertainty in a targeted way that is computationally feasible. We assessed the effect of noise in the convergence of the algorithm, and we saw that in our case, noise results in a biased result. Our future efforts will be into a more comprehensive understanding of the convergence of the method, and eventually be able to assess the impact of more sources of uncertainty and additional model parameters.

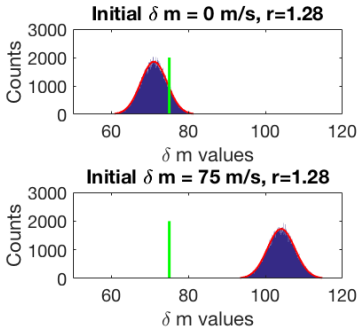

Figure 8: Resulted histograms from half a million runs of the Metropolis Hastings algorithm when $r = 1.28$, after discarding the first half.

ACKNOWLEDGMENTS

This work is supported by Chevron and with grants from the Natural Sciences and Engineering Research Council of Canada Industrial Research Chair Program and the InnovateNL and by the Hibernia Management and Development Corporation.

REFERENCES

Asnaashari, R., R. Brossier, S. Garambois, F. Audebert, P. Thore, and J. Virieux, 2015, Time-lapse seismic imaging using regularized full-waveform inversion with a prior model: which strategy?: Geophysical Prospecting, **63**, 78–98.

Brooks, S., A. Gelman, G. Jones, and X. L. Meng, 2011, Handbook of markov chain monte carlo: CRC Press.

Ely, G., A. Malcolm, and O. V. Poloannikov, 2018, Assessing uncertainties in velocity models and images with a fast nonlinear uncertainty quantification method: Geophysics, **83**, R63–R75.

Haario, H., E. Saksman, and J. Tamminen, 2001, An adaptive metropolis algorithm: Bernouli, **7**, 223–242.

Kotsi, M., and A. Malcolm, 2017, A statistical comparison of three 4d full-waveform inversion schemes: SEG International Exposition and 87th Annual Meeting, Expanded Abstracts, 1434–1438.

Ma, Y. Z., 2010, Uncertainty analysis in reservoir characterization and management: How much should we know about what we don't know?: In: Uncertainty analysis and reservoir modeling. AAPGMemoir, **96**, 1–15.

Maharramov, M., and B. Biondi, 2014, Joint full waveform inversion of time-lapse seismic data sets: SEG Denver Annual Meeting, 1–5.

Malcolm, A., and B. Willemsen, 2017, Rapid 4d fwi using a local solver: The Leading Edge, **35**, 1053–1059.

Osypov, K., Y. Yang, A. Fournier, N. Ivanova, R. Bachrach, C. E. Yarman, Y. You, D. Nichols, and M. Woodward, 2013, Model-uncertainty quantification in seismic tomography: method and applications: Geophysical Prospecting, **61**, 1114–1134.

Poliannikov, O. V., and A. E. Malcolm, 2015, The effect of velocity uncertainty on migrated reflectors: Improvements from relative-depth imaging: Geophysics, **81**, S21–S29.

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

———, 2005, Inverse problem theory and methods for model parameter estimation: SIAM.

Versteeg, R., 1994, The marmousi experience: Velocity model determination on a synthetic complex data set: The Leading Edge, **13**, 927–936.

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

Watanabe, T., S. Shimizu, E. Asakawa, and T. Matsuoka, 2005, Differential waveform tomography for time-lapse crosswell seismic data with application to gas hydrate production monitoring: SEG Technical Program Expanded Abstracts.

Willemsen, B., A. Malcolm, and W. Lewis, 2016, A numerically exact local solver applied to salt boundary inversion in seismic full waveform inversion: Geophysical Journal International, **204**, 1703–1720.

Yang, D., A. Malcolm, and M. Fehler, 2014, Time-lapse full waveform inversion and uncertainty analysis with different survey geometries: 76th EAGE Conference and Exhibition.

Zheng, Y., P. Barton, and S. Singh, 2011, Strategies for elastic full waveforms inversion of time-lapse ocean bottom cable (obc) seismic data: SEG San Antonion Annual Meeting, 4195–4200.