Seismic waveform inversion by stochastic optimization

Tristan van Leeuwen, Aleksandr Aravkin and Felix J. Herrmann
Dept. of Earth and Ocean sciences
University of British Columbia
Vancouver, BC, Canada
{*tleeuwen, saravkin, fherrmann*}*@eos.ubc*

ABSTRACT

We explore the use of stochastic optimization methods for seismic waveform inversion. The basic principle of such methods is to randomly draw a batch of realizations of a given misfit function and goes back to the 1950s. A batch in the current setting represents a single random superposition of sources. The ultimate goal of such an approach is to dramatically reduce the number of shots that need to be modeled. Assuming that the computational costs grow linearly with the number of shots, this promises a significant speed-up. Following earlier work, we introduce the stochasticity in the waveform inversion problem in a rigorous way via a technique called *randomized trace estimation*. We then review theoretical results that underlie recent developments in the use of stochastic methods for waveform inversion. We present numerical experiments to illustrate the behavior of different types of stochastic optimization methods and investigate the sensitivity to the batch-size and the noise level in the data. We find that it is possible to reproduce results that are qualitatively similar to the solution of the full problem with modest batch-sizes, even on noisy data. Each iteration of the corresponding stochastic methods requires an order of magnitude fewer PDE solves than a comparable deterministic method applied to the full problem, which may lead to an order of magnitude speed up for waveform inversion in practice.

INTRODUCTION

The use of simultaneous source data in seismic imaging has a long history. So far, the use of incoherent simultaneous sources has been used to increase the efficiency of data acquisition (Beasley et al., 1998; Berkhout, 2008), migration (Romero et al., 2000; Dai et al., 2010) and simulation (Ikelle, 2007; Neelamani et al., 2008; Herrmann et al., 2009). Recently, the use of simultaneous source-encoding has found its way into waveform inversion. Two key factors play a role in this development: *i)* in 3D one is forced to use modeling engines whose cost is proportional to the number of shots (as opposed to 2D frequency-domain methods where one can re-use the LU factorization to cheaply model any number of shots), *ii)* the curse of dimensionality: the number of shots and the number of gridpoints grows by an order of magnitude.

The basic idea of replacing single-shot data by randomly combined 'super-shots' is intuitively pleasing and has lead to several algorithms (Krebs et al., 2009; Moghaddam and Herrmann, 2010; Boonyasiriwat and Schuster, 2010; Li and Herrmann, 2010). All of these aim at reducing the computational costs of full waveform inversion by reducing the number of PDE solves (i.e., the number of simulations). This reduction comes at the cost of

introducing random cross-talk between the shots into the problem. It was observed by Krebs et al. (2009) that it is beneficial to re-combine the shots at every iteration to suppress the random cross-talk and that the approach might be more sensitive to noise in the data. In this paper, we follow Haber et al. (2010a) and introduce randomized source encoding through a technique called *randomized trace estimation* (Hutchinson, 1989; Avron and Toledo, 2010). The goal of this technique is to estimate the trace of a matrix efficiently by sampling its action on a small number of randomly chosen vectors. The traditional least-squares optimization problem can now be recast as a *stochastic* optimization problem. Theoretical developments in this area go back to 1950's and we review them in this paper. In particular, we discuss two distinct approaches to stochastic optimization. The *Stochastic Approximation* (SA) approach consists of a family of algorithms that use a different randomization in each iteration. This idea justifies a key part of the approach described in Krebs et al. (2009). Notably, the idea of averaging the updates over the past is important in this context to suppress the random cross-talk; lack of averaging over the past likely explains the noise sensitivity reported by Krebs et al. (2009). The theory we treat here concerns only first-order optimization methods, though there has been a recent effort to extend similar ideas to methods that exploit curvature information (Byrd et al., 2010).
Another approach, called the *Sample Average Approximation* (SAA), replaces the stochastic optimization problem by an ensemble average over a set of randomizations. The ensemble size should be big enough to suppress the cross-talk. The resulting problem may be treated as a deterministic optimization problem; in particular, one may use any optimization method to solve it.
Most theoretical results in SA and SAA assume that the objective function is convex, which is not the case for seismic waveform inversion. However, in practice one starts from a 'reasonable' initial model and we may be able to converge to the closest local minimum. One would expect SA and SAA to be applicable in the same framework. Understanding the theory behind SA and SAA is then very useful in algorithm design, even though the theoretical guarantees derived under the convexity assumption need not apply.
As mentioned before, the gain in computational efficiency comes at the cost of introducing random cross-talk between the shots into the problem. Also, the influence of noise in the data may be amplified by randomly combining shots. We can reduce the influence of these two types of noise by increasing the batch-size, re-combining the shots at every iteration and averaging over past iterations. We present a detailed numerical study to investigate how these different techniques affect the recovery.
The paper is organized as follows. First, we introduce randomized trace estimation in order to cast the canonical waveform inversion problem as a stochastic optimization problem. We describe briefly how SA and SAA can be applied to solve the waveform inversion problem. In section 3, we review relevant theory for these approaches from the field of stochastic optimization. The corresponding algorithms are presented in section 4. Numerical results on a subset of the Marmousi model are presented in section 5 to illustrate the characteristics of the different stochastic optimization approaches. Finally, we discuss the results and present the conclusions.

WAVEFORM INVERSION AND TRACE ESTIMATION

The canonical waveform inversion problem is to find the medium parameters for which the modeled data matches the recorded data in a least-squares sense (Tarantola, 1984). We

consider the simplest case of constant-density acoustics and model the data in the frequency domain by solving

$$H[m]u \quad = \quad q, \tag{1}$$

where $H[m]$ is the discretized Helmholtz operator $[\omega^2 m + \nabla^2]$ for the squared slowness m (with appropriate boundary conditions), u is the discretized wavefield and q is the discretized source function; both are column vectors. The data are then given by sampling the wavefield at the receiver locations: $d = Pu$. Note that all the quantities are monochromatic. We hide the dependence on frequency for notational simplicity.
We denote the corresponding optimization problem as:

$$\min_m \phi(m, Q, D) = \sum_\omega ||PH[m]^{-1}Q - D||_F^2, \tag{2}$$

where $D = [d_1, d_2, \ldots d_N]$ is a frequency slice of the recorded data and $Q = [q_1, q_2, \ldots, q_N]$ are the corresponding source functions. Note the dependence of H on ω has been suppressed. $||\cdot||_F$ denotes the Frobenuis norm, which is defined as $||A||_F = \sqrt{\text{trace}(A^T A)}$ (here $\cdot^T$ denotes the complex-conjugate transpose. We will use the same notation for the transpose in case the quantity is real). Note that we assume a fixed-spread acquisition where each receiver sees all the sources.
In practice H^{-1} is never computed explicitly, but involves either an LU decomposition (cf. Marfurt, 1984; Pratt, 1999; Operto et al., 2006) or an iterative solution strategy (cf. Erlangga et al., 2006; Riyanti et al., 2006). In the worst case, the matrix has to be inverted separately for each frequency and source position. For 3D full waveform inversion both the costs for inverting the matrix *and* the number of sources increases by an order of magnitude. Recently, several authors have proposed to reduce the computational cost by randomly combining sources (Krebs et al., 2009; Moghaddam and Herrmann, 2010; Boonyasiriwat and Schuster, 2010; Li and Herrmann, 2010; Haber et al., 2010a).
We follow Haber et al. (2010a) and introduce this encoding in a rigorous manner by using a technique called *randomized trace estimation*. This technique was introduced by Hutchinson (1989) as a technique to efficiently estimate the trace of an implicit matrix. Some recent developments and error estimates can be found in Avron and Toledo (2010).
This technique is based on the identity:

$$\text{trace}(A^T A) = \text{E}_w(w^T A^T A w) = \lim_{K\to\infty} \frac{1}{K} \sum_{k=1}^{K} w_k^T A^T A w_k, \tag{3}$$

where E_w denotes the expectation over w. The random vectors w are chosen such that $\text{E}_w(ww^T) = I$ (the identity matrix). The identity can be derived easily by using the cyclic permutation rule for the trace (i.e., $\text{trace}(ABC) = \text{trace}(CAB)$), the linearity of the expectation and the aforementioned property of w. At the end of the section we discuss different choices of the random vectors w. First, we discuss how randomized trace estimation affects the waveform inversion problem.
Using the definition of $\|A\|_F$, we have

$$\phi(m, Q, D) = \text{E}_w \phi(m, Qw, Dw). \tag{4}$$

This reformulation of (2) is a *stochastic* optimization problem. We now briefly outline approaches to solve such optimization problems.

Sample Average Approximation

A natural approach to take is to replace the expectation over w by an ensemble average:

$$\phi_K(m) = \frac{1}{K}\sum_{k=1}^{K}\phi(m, Qw_k, Dw_k). \tag{5}$$

This is often referred to in the literature as the Sample Average Approximation (SAA). The random cross-talk can be controlled by picking a 'large enough' batch size. As long as the required batch size is smaller than the actual number of sources, we reduce the computational complexity.
For a fixed m it is known that the error $|\phi - \phi_K|$ is of order $1/\sqrt{K}$ (cf. Avron and Toledo, 2010). However, it is not the value of the misfit that we are trying to approximate, but the minimizer. Unfortunately the difference between the minimizers of ϕ and ϕ_K is not readily analyzed. Instead, we perform a small numerical experiment to get some idea of the performance of the SAA approach for waveform inversion.
We investigate the misfit along the direction of the negative gradient g_k (defined below):

$$f_K(\alpha) = \phi_K(m - \alpha g_K). \tag{6}$$

The data are generated for the model depicted in figure 1 (a), for 61 co-located, equi-distributed sources and receivers along a straight line at 10 m depth and 7 randomly chosen frequencies between 5 and 30 Hz. The source signature is a Ricker wavelet with a peak frequency of 10 Hz. We use a 9-point discretization of the Helmholtz operator with absorbing boundary conditions and solve the system via a (sparse) LU decomposition (cf. Saad, 1996)[1]. The search direction g_K is the gradient of ϕ_K evaluated at the initial model m_0, depicted in figure 1 (b). The gradient is computed in the usual way via the adjoint-state method (cf. Plessix, 2006). The full gradient as well as the gradients for $K = 1, 5, 10$ are depicted in figure 2. The error between the full and approximated gradient, caused by the cross-talk, is depicted in figure 3. As expected, the error decays as $1/\sqrt{K}$. The misfit as a function of α for various K, as well as the full misfit (no randomization) is depicted in figure 4. This shows that the minimizer of ϕ_K is reasonably close to the minimizer of the full misfit ϕ , even for a relatively small batch-size K.

Stochastic Approximation

A second alternative is to apply specialized stochastic optimization methods to problem (4) directly. This is often referred to as the Stochastic Approximation (SA). The main idea of such algorithms is to pick a new random realization in each iteration and possibly average over past iterations to suppress the resulting stochasticity. In the context of the full waveform inversion problem, this gives an iterative algorithm of the form

$$m^{\nu+1} = m^{\nu} - \gamma_\nu \nabla\phi_{K,\nu}(m^{\nu}),$$

[1]We note that this setup is quite efficient already since the LU decomposition can be re-used for each source. Reduction of the number of sources becomes of paramount importance in 3D where one is forced to use iterative methods whose costs grow linearly with the number of sources.

where batch size K can be as small as 1, $\{\gamma_\nu\}$ represent step sizes taken by the algorithm, and the notation $\phi_{K,\nu}$ emphasizes that a new randomization is used at every iteration ν (in contrast with the SAA approach).
We discuss theoretical performance results and describe SAA and SA in more detail in the next section.

Accuracy and efficiency of randomized trace estimation

Efficient calculation of the trace of a positive semi-definite matrix lies at the heart of our approach. Factors that determine the performance of this estimation include: the random process for the *i.i.d.* w's, the size of the source ensemble K, and the properties of the matrix. Hutchnison's approximation (Hutchinson, 1989), which is based on w's drawn from a Rademacher distribution (i.e., random ± 1), attains the smallest variance for the estimate of the trace. The variance can be used to bound the error via confidence intervals. However, the variance is not the *only* measure of the error. In particular, Avron and Toledo (2010) derive bounds on the batch size in terms of ϵ and δ, defined as follows. A randomized-trace estimator $T_K = K^{-1} \sum w_i^T B w_i$ is an (ϵ, δ)-approximation of $T = \mathrm{trace}\big(B\big)$ if

$$\Pr\left(\frac{|T_K - T|}{|T|} \leq \epsilon \right) \geq 1 - \delta. \tag{7}$$

The expressions for the minimum batch-size K for which the relative error is smaller than ϵ with probability δ are listed in table 1 (adapted from Avron and Toledo (2010)). Smaller ϵ's and δ's lead to larger K, which in turn, leads to more accurate trace estimates with increased probability.
Of course, these bounds depend on the choice of the probability distribution of the *i.i.d.* w's and the matrix B. Aside from obtaining the lowest value for K, simplicity of computational implementation is also a consideration. In Table 1, we summarize the performance of four different choices for the w, namely

1. the Rademacher distribution, i.e., $\Pr(w[i] = \pm 1) = 1/2$, yielding $\mathrm{E}\{w[i]\} = 0$ ($w[i]$ denotes the i^{th} element in the vector w) and $\mathrm{E}\{w[i]^2\} = 1$ for $i = 1 \cdots N$. Aside from the fact that this estimator H_K (see Table 1) leads to minimum variance, the advantage of this choice is that it leads to a fast implementation with a small memory imprint. The disadvantage of this method is that the lower bound depends on the rank of A and requires larger K compared to w's defined by the Gaussian (see Table 1);

2. the standard normal distribution, i.e., $w[i] \in N(0, 1)$ for $i = 1 \cdots N$. While the variance for this estimator G_K (see Table 1) is larger than the variance for H_k, the lower bound for K does not depend on the size or the rank of A and is the smallest of all four methods. This suggests that we can use a fixed value of K for arbitrarily large matrices. However, this method is known to converge slower than Hutchinson's for matrices A that have significant energy in the off-diagonals. This choice also requires a more complex implementation with a larger memory imprint;

3. the fast phase-encoded method where w's selected uniformly from the canonical basis,

i.e., from $\{e_1, \cdots, e_N\}$. This estimator

$$L_K = \frac{N}{K} \sum_{j=1}^{K} w_j^T \mathcal{F} A \mathcal{F}^T w_j,$$

where $\mathcal{F}$ is a unitary (i.e, $\mathcal{F}^T = \mathcal{F}^{-1}$) random mixing matrix. The idea is to mix the matrix B such that its diagonal entries are evenly distributed. This is important since the unit vectors only sample the diagonal of the matrix. The flatter the distribution of the diagonal elements, the faster the convergence (if all the diagonal elements were to be the same, we need only one sample to compute the trace exactly).

Estimator	Distribution of w	Variance of one sample	Bound on K for (ϵ, δ) bound
Hutchinson's $H_K = \frac{1}{K} \sum_{j=1}^{K} w_j^\top A w_j$	$\Pr(w_j = \pm 1) = 1/2$	$2(\|A\|_F^2 - \sum_{i=1}^{N} A_{ii}^2)$	$6\epsilon^{-2} \ln(2 \text{ rank}(A)/\delta)$
Gaussian $G_K = \frac{1}{K} \sum_{j=1}^{K} w_j^\top A w_j$	$w_j \in N(0,1)$	$2\|A\|_F^2$	$20\epsilon^{-2} \ln(2/\delta)$
Phase encoded $L_K = \frac{N}{K} \sum_{j=1}^{K} w_j^T \mathcal{F} A \mathcal{F}^T w_j$	w_j drawn uniformly from $\{e_1, \cdots, e_N\}$	n/a	$2\epsilon^{-2} \ln(4n^2/\delta) \ln(4/\delta)$

Table 1: Summary of bounds, adapted from Avron and Toledo (2010).

The lower bounds summarized in Table 1 tell us that Gaussian w's theoretically require the smallest K and hence the fewest PDE solves. However, this result comes at the expense of more complex arithmetic, which can be a practical consideration (Krebs et al., 2009). Aside from the lowest bound, the estimator based on Gaussian w's has the additional advantage that the bound on K does not depend on the size or rank of the matrix B. Hutchinson's method, on the other hand, depends logarithmically on the rank of B, but has the reported advantage that it performs well for near diagonal matrices (Avron and Toledo, 2010). This has important implications for our application because our matrix B is typically full rank and can be considered near diagonal only when our optimization procedure is close to convergence. At the beginning of the optimization, we can expect the residue to be large and a B that is not necessarily diagonal dominant.

We conduct the following stylized experiment to illustrate the quality of the different trace estimators. We solve the discretized Helmholtz equation at 5Hz for a realistic acoustic model with 301 co-located sources and receivers located at 10 m depth. We compute matrix $B = A^T A$ for a residue A given by the difference between simulation results for the hard and smooth models shown in figure 1. As expected, the resulting matrix B, shown in figure 5 contains significant off-diagonal energy. For the phase-encoded part of the experiment, we use a random mixing matrix based on the DFT, as suggested by Romberg (2009). Such mixing matrices are also commonly found in compressive sensing applications (Candès et al., 2006; Donoho, 2006; Romberg, 2009; Herrmann et al., 2009).

We evaluated the different trace estimators 1000 times for batch sizes ranging from $K =$

	$\epsilon = 10^{-1}$	$\epsilon = 10^{-2}$	$\epsilon = 10^{-3}$
Gauss	$6 \cdot 10^3$	$6 \cdot 10^5$	$6 \cdot 10^7$
Hutchinson	$4 \cdot 10^3$	$4 \cdot 10^5$	$4 \cdot 10^7$
Phase	$9 \cdot 10^3$	$9 \cdot 10^5$	$9 \cdot 10^7$

Table 2: This table shows the theoretical lower bounds (see table 1) on the batch size K for $\delta = 10^{-1}$ for the matrix shown in figure 5.

$1 \ldots 1000$. The probability for the error level ϵ is estimated by counting the number of times we were able to achieve that error level for each K. The results for the different trace estimators and error levels are summarized in figure 6. For this particular example we see little difference in performance between the different estimators. The corresponding *theoretical* bounds on the batch size, as given by table 1 are listed in table 2. Clearly, these bounds are overly pessimistic in this case. In our experiments, we observed that we get similar reconstruction behavior if we use a finer source/receiver sampling. This suggests that the gain in efficiency will increase with the data size, since we can use larger batch sizes for a fixed downsampling ratio. We also noticed, in this particular example, little or no change in behavior if we change the frequency.

OPTIMIZATION

Sample Average Approximation

The Sample Average Approximation (SAA) is used to solve the following class of stochastic optimization problems:

$$\min_{x \in X} \{f(x) = \mathrm{E}_w\{F(x, w)\}\} \tag{8}$$

where $X \subset \mathbb{R}^n$ is the set of admissible models (assumed to be a compact convex set, e.g., box constraints $x_{\min} \leq x \leq x_{\max}$), w is a random vector with distribution supported on $W \subset \mathbb{R}^d$, $F : X \times W \to \mathbb{R}$, and the function $f(x)$ is convex (Nemirovski et al. (2009)). The last important assumption is the Law of Large Numbers (LLN), i.e. $\hat{f}_K(x) \to f(x)$ with probability 1 as $K \to \infty$. These assumptions are required for most of the known theoretical results about convergence of SAA methods. The convexity assumption and LLN assumption can be relaxed in the case when $F(\cdot, w)$ is continuous on X for almost every $w \in W$ and $F(x, w)$ is dominated by an integrable function $G(w)$ so that $|f(x)| \leq \mathrm{E}_w\{G(w)\}$ for every $x \in X$ (Shapiro (2003)). Given an optimization problem of type (8), the SAA approach (Nemirovski et al. (2009)) is to generate a random sample $w_1, \ldots, w_K$ and solve the approximate (or sample average) problem

$$\min_{x \in X} \left\{ \hat{f}_K(x) = \frac{1}{K} \sum_{j=1}^{K} F(x, w_j) \right\}. \tag{9}$$

When these assumptions are satisfied, the optimal value of (9) converges to the optimal value of the full problem (8) with probability 1. Moreover, under more technical assumptions on the distribution of the random variable w, conservative bounds have been derived on the batch size K necessary to obtain a particular accuracy level ϵ (Shapiro and Nemirovsky,

2005, eq. (22)). These bounds do not require the convexity assumptions but instead require assumptions on local behavior of $F(\cdot, w)$. It is worth underscoring that 'accuracy' here of solution $\bar{x}$ with respect to the optimal solution x^* is defined with respect to the function value difference $f(x) - f(x^*)$, rather than in terms of $\|x - x^*\|_2$ or other measure in the space of model parameters. From a practical point of view, the SAA approach is appealing because it allows flexibility in the choice of algorithm for the solution of (9). This works on two levels. First, if a faster algorithm becomes available for the solution of (9), it can immediately impact (8). Second, having fixed a large $\bar{K}$ and $\hat{f}_{\bar{K}}$ to obtain reasonable accuracy in the solution of (8), one is free to approximately solve a sequence of smaller problems ($K_i << \bar{K}$) with warm starts on the way to solving $\hat{f}_{\bar{K}}$ (Haber et al., 2010a). In other words, SAA theory guarantees the existence of an K large enough for which the approximate problem is close to the full problem; however, the algorithm for solving the approximate problem (9) is left completely to the practitioner, and in particular may require the evaluation of very few samples at early iterations.

Stochastic Approximation

Stochastic Approximation (SA) methods go back to Robbins and Monro (1951), who considered the root-finding problem

$$g(x) = g_0,$$

in the case where $g(x)$ cannot be evaluated directly. Rather, one has access to a function $G(x, w)$ for which $\mathrm{E}_w\{G(x, w)\} = g(x)$. The approach can be translated to optimization problems of the form

$$\min f(x)$$

by considering g to be the gradient of f and setting $g_0 = 0$. Again, we cannot evaluate $f(x)$ directly, but we have access to $F(x, w)$ for which $\mathrm{E}_w\{F(x, w)\} = f(x)$. More generally, for problems of type (8), Bertsekas and Tsitsiklis (1996) and Betrsekas and Tsitsiklis (2000) consider iterative algorithms of the form

$$x^{\nu+1} = x^\nu - \gamma_\nu s^\nu,$$

where γ_ν are a sequence of step sizes determined a priori that satisfy certain properties, and s^ν can be thought of as noisy unbiased estimates of the gradient (i.e., $\mathrm{E}_w s^\nu = \nabla f(x^\nu)$). Note that right away we are forced into an algorithmic framework, which never appears in the SAA discussion. The positive step sizes γ_ν are chosen to satisfy

$$\sum_{\nu=0}^{\infty} \gamma_\nu = \infty \ , \quad \sum_{\nu=0}^{\infty} \gamma_\nu^2 < \infty. \tag{10}$$

The main idea is that the step sizes go to zero, but not too fast. A commonly used example of such a sequence of step sizes is

$$\gamma_\nu \propto \frac{1}{\nu}.$$

The main result of Bertsekas and Tsitsiklis (1996) is that if ∇f satisfies the Lipshitz condition with constant L

$$\|\nabla f(x) - \nabla f(y)\| \leq L\|x - y\|$$

i.e., the changes in the gradient are bounded in norm by changes in the parameter space, and if the directions s^ν on average point 'close to' the gradient and are not too noisy, then the sequence $f(x^\nu)$ converges, and every limit point $\bar{x}$ of $\{x^\nu\}$ is a stationary point of f (i.e. $\nabla f(\bar{x}) = 0$). Under stronger assumptions that the level sets of f are bounded and the minimum is unique, this guarantees that the algorithms described above will find it. A similar family of algorithms was studied by Polyak and Juditsky (1992), who considered larger step sizes γ_ν but included averaging model estimates into their algorithm. In the context discussed above, the step size rule 10 is replaced by

$$\frac{\gamma_\nu - \gamma_{\nu+1}}{\gamma_\nu} = o(\gamma_\nu). \tag{11}$$

A particular example of such a sequence cited by the paper is

$$\gamma_\nu \propto \nu^{-\beta},\ 0 < \beta < 1.$$

The iterative scheme is then given by

$$\begin{aligned} x^{\nu+1} &= x^\nu - \gamma_\nu s^\nu \\ \bar{x}^\nu &= \tfrac{1}{\nu}\textstyle\sum_{i=0}^{\nu-1} x^i. \end{aligned}$$

Under assumptions similar in spirit to the ones in Bertsekas and Tsitsiklis (1996), there is a result for the convergence of the iterates x^ν to the true estimate x^*, namely $\bar{x}^\nu \to x^*$ almost surely and

$$\sqrt{\nu}(\bar{x}^\nu - x^*) \to_D \mathbf{N}(0, V),$$

where the convergence is in distribution, and the matrix V is in some sense optimal and is related to the Hessian of f at the solution x^*. A more recent report (Nesterov and Vial, 2000) also considers averaging of model iterates in the context of optimizing (not necessarily smooth) convex functions of the form

$$f(x) = \mathrm{E}_w\{F(x, w)\}$$

over a convex set X. When f is smooth, this situation reduces to the previous discussion. Nesterov and Vial (2000) choose a finite sequence of N step sizes a priori, and consider the error in the expected value function

$$\mathrm{E}_{\bar{x}^\nu}\{f(\bar{x}^\nu)\} - f(x^*)$$

after N iterations. This is similar to the SAA analysis but is much easier to interpret, because now the desired accuracy in the objective value directly translates to the number of iterations of a particular algorithm:

$$\begin{aligned} x^{\nu+1} &= \pi_X(x^\nu - \gamma_\nu s^\nu), \\ \bar{x} &= \textstyle\sum_{\nu=0}^{N-1} \gamma_\nu x^\nu / \sum_{\nu=0}^{N-1} \gamma_\nu, \end{aligned}$$

where π_X is projection onto the convex set of admissible models X. Unfortunately, the error is $O(L^2 \frac{\sum \gamma_\nu^2}{\sum \gamma_\nu} + R^2 \frac{1}{\sum \gamma_\nu})$, where R is the diameter of the set X (related to the bounds on x from our earlier example) and L is a uniform bound on $\|\nabla f\|$, and so the estimate may

be overly conservative. If all the γ_ν are chosen to be uniform, the optimal size is $\gamma = \frac{R}{L\sqrt{N}}$, and then the result is simply

$$\mathrm{E}_{\bar{x}^\nu}\{f(\bar{x}^\nu)\} - f(x^*) \leq \frac{LR}{\sqrt{N}}.$$

For a recent survey of stochastic optimization and new robust SA methods, please see Nemirovski et al. (2009).

Note that the error rate in the objective values is $O\left(1/\sqrt{N}\right)$, where the constant depends in a straightforward way on the size of the set X and the behavior of $\|\nabla f\|$. Compare this to the $O(1/\sqrt{K})$ error bound for the SAA approach. In contrast to the SAA, the SA approach translates directly into a particular algorithm. This makes it easier to implement for full waveform inversion, but also leaves less freedom for algorithm design than in SAA, where any algorithm can be used to solve the deterministic ensemble average problem.

ALGORITHMS

To test the performance of the SAA approach, we chose to use a steepest descent method with an Armijo line search (cf. Nocedal and Wright, 1999). Although one could in principle use a second order method (such as L-BFGS), we chose to use a first order method to allow for better comparison to the SA results. The pseudo-code is presented in Algorithm 1.
The SA methods are closely related to the steepest descent method. The main difference is that for each iteration a new random realization is drawn form a prescribed distribution and that the result is averaged over past iterations. We chose to implement a few modifications to the standard SA algorithms. First, we use an Armijo line search to determine the step size instead of using a prescribed sequence such as discussed in the previous section. This assures some descent at each iteration with respect to the current realization of ϕ_K, and we found that this greatly improved the convergence. Second, we allow for averaging over the past n iterations instead of the full history. This prevents the method from stalling. The pseudo-code is presented in Algorithm 2.

Algorithm 1 Steepest descent

while not converged **do**
  $s \leftarrow -\nabla\phi[m_i]/||\nabla\phi[m_i]||_2$
  find λ s.t. $\phi[m_i + \lambda s] \leq \phi[m_i] + c\lambda\nabla\phi[m_i]^T s$
  $m_{i+1} \leftarrow m_i + \lambda s$
  $i \leftarrow i + 1$
end while

Algorithm 2 Stochastic descent

while not converged **do**
  draw w from a pre-scribed distribution
  $s \leftarrow -\nabla\phi[m_i, w]/||\nabla\phi[m_i, w]||_2$
  find λ s.t. $\phi[m_i + \lambda s, w] \leq \phi[m_i, w] + c\lambda\nabla\phi[m_i, w]^T s$
  $m_{i+1} \leftarrow \frac{1}{n+1}\left(\sum_{i-n}^{i} m_i + \lambda s\right)$
  $i \leftarrow i + 1$
end while

RESULTS

For the numerical experiments we use the true and initial squared-slowness models depicted in figure 1. The data are generated for 61 equi-spaced, co-located sources and receivers at 10 m depth and 7 randomly chosen (but fixed) frequencies between 5 and 30 Hz. The latter strategy is inspired by results from Compressive Sensing (cf. Hennenfent and Herrmann, 2008; Herrmann et al., 2009; Lin and Herrmann, 2007). The basic idea is to turn aliases that are introduced by sub-Nyquist sampling into random noise.
The Helmholtz operator is discretized on a grid with 10 m spacing, using a 9-point finite difference stencil and absorbing boundary conditions. The point-sources are represented as narrow Gaussians. As a source signature, we use a Ricker wavelet with a peak-frequency of 10 Hz. The noise is Gaussian with a prescribed SNR.
We run each of the optimization methods for 500 iterations and compare the performance for various batch sizes and noise levels to the result of steepest descent on the full problem. Remember that by using small batch-sizes, the iterations are very cheap so we can afford to do more. The random vectors are drawn from a Gaussian distribution with zero mean and unit variance. We chose to use the Gaussian because the theoretical bounds on K do not depend on properties of the residual matrix. Although the matrix will change constantly during the optimization, we can at least expect a uniform quality of the approximation.
In a realistic application one might want to add a regularization term. In particular, this would prevent the over-fitting that we observe in the noisy case. Note that limiting the amount of iterations also serves as a form of regularization (Hansen, 1998).

Sample Average Approximation

We choose a set of K Gaussian random vectors with zero mean and unit variance and run the steepest descent algorithm presented previously on the resulting deterministic optimization problem. The results after 500 iterations on data without noise are shown in the first column of figure 7. The error between the recovered and true model is shown in figure 8 (a). As reference, the error between the true and recovered model for the inversion with *all* the sequential sources is also shown. As expected, the recovery is better for larger batch-sizes. The recovered models for data with noise are shown in the second column (SNR = 20 dB) and third (SNR = 10 dB) columns of figure 7. The corresponding recovery error is shown in figure 8 (b) and (c), respectively. It shows that the SAA approach starts over-fitting in an earlier stage than the full inversion. Also, we are not able to reach the same model-error as the full inversion.

Stochastic Approximation

We run the stochastic descent algorithm for varying batch sizes ($K = 1, 5, 10$) and history sizes ($n = 0, 10, 500$).
The results obtained without averaging are shown in figure 9. The columns represent different batch sizes while the rows represent different noise levels. The recovery errors for the different batch-sizes and noise levels are shown in figure 10. In the noiseless case, we are able to achieve the same recovery error as the full inversion with only one simultaneous source.

When noise is present in the data one simultaneous source is not enough, however. Still, we can achieve the same recovery error as the full problem with only 10 simultaneous sources. This yields an order of magnitude improvement in our computation, since the *total* number of iterations needed by the stochastic method to achieve a given level of accuracy is roughly the same as required by a deterministic first order method used on the full system, but *each* stochastic iteration requires ten times fewer PDE solves than a deterministic iteration on the full system.
Results obtained with averaging over the past 10 iterations are shown in figure 11. The rows represent different batch sizes while the columns represent different noise levels. The corresponding recovery errors are shown in figure 12. It shows that averaging helps to overcome some of the noise sensitivity and we are now able to achieve a good reconstruction with only 5 simultaneous sources. Also, the averaging damps the irregularity of the convergence somewhat.
Finally, we show the result obtained by averaging over the full history in figure 13. The corresponding recovery error is shown in figure 14. It shows that too much averaging slows down the convergence.

CONCLUSIONS AND DISCUSSION

Following Haber et al. (2010b), we reduce the dimensionality of full waveform inversion via *randomized trace estimation*. This reduction comes at the cost of introducing random cross-talk between the sources into the updates. The resulting optimization problem can be treated as a *stochastic optimization problem*. Theory for such methods goes back to the 1950s and justifies the approach presented by Krebs et al. (2009). In particular, we use theoretical results by Avron and Toledo (2010) on randomized trace estimation to get bounds for the batch size needed to approximate the misfit to a given accuracy level with a given probability. Numerical tests show, however, that these bounds may be overly pessimistic and that we get reasonable approximations for modest batch sizes.
Theory from the field of stochastic optimization suggests several approaches to tackle the optimization problem and reduce the influence of the cross-talk introduced by the randomization. The first approach, the *Sample Average Approximation*, dictates the use of a fixed set of random sources and relies solely on increasing the batch-size to get rid of the cross-talk. The *Stochastic Approximation*, on the other hand, dictates that we redraw the randomization each iteration and average over the past in order to suppress the stochasticity of the gradients.
We note that, as opposed to *randomized* dimensionality reduction, several authors have proposed methods for *deterministic* dimensionality reduction (Haber et al., 2010b; Symes, 2010). These techniques are related to optimal experimental design and try to determine the source combination that somehow optimally illuminates the target. It is not quite clear how such methods compare to the randomized approach discussed here. It is clear, however, that by using random superpositions we have access to powerful results from the field of compressive sensing to further improve the reconstruction. Li and Herrmann (2010) use sparse recovery techniques instead of Monte Carlo sampling to get rid of the cross-talk.
In our experiments, we were able to obtain results that are comparable to the full optimization with a small fraction of the number of sources. In the noiseless case we needed only *one* simultaneous source for the SA approach. Even with noisy data, *five* simultaneous sources

proved sufficient. This is a very promising result, since using five simultaneous sources for the SA method means that every iteration requires 20 times fewer PDE solves, which directly translates to a 20x computational speedup compared to a first order deterministic method. The key point is that both SA and the full deterministic approach require roughly the same number of iterations to achieve the same accuracy.

Averaging over a limited number of past iterations improved the results for a fixed batch size and allows for the use of fewer simultaneous sources. However, too much averaging slows down the convergence.

The results of the SA approach, where a new realization of the random vectors are drawn at every iteration, are superior to the SAA results, where the random vectors are fixed. However, one could use a more sophisticated (possibly black-box) optimization method for the SAA approach to get a similar result with fewer iterations. The trade-off between using a smaller batch size and first order methods (i.e., more iterations) versus using a larger batch size and second order methods (i.e., less iterations) needs to be investigated further.

Random superposition of shots only makes sense if those shots are sampled by the same receivers. In particular, this hampers straightforward application to marine seismic data. One way to get around this is to partition the data into blocks that are fully sampled. However, this would not give the same amount of reduction in the number of shots because only shots that are relatively close to each other can be combined without losing too much data.

The type of encoding used will most likely affect the behavior of both SA and SAA methods. It remains to be investigated which encoding is most suitable for waveform inversion.

ACKNOWLEDGMENTS

We thank Eldad Haber and Mark Schmidt for insightful discussions on trace estimation and stochastic optimization. This work was in part financially supported by the Natural Sciences and Engineering Research Council of Canada Discovery Grant (22R81254) and the Collaborative Research and Development Grant DNOISE II (375142-08). This research was carried out as part of the SINBAD II project with support from the following organizations: BG Group, BP, Chevron, ConocoPhillips, Petrobras, Total SA, and WesternGeco.

REFERENCES

Avron, H., and S. Toledo, 2010, Randomized algorithms for estimating the trace of an implicit symmetric positive semi-definite matrix: to appear in Journal of the ACM.

Beasley, C. J., R. E. Chambers, and Z. Jiang, 1998, A new look at simultaneous sources: SEG Technical Program Expanded Abstracts, **17**, 133–135.

Berkhout, A. J. G., 2008, Changing the mindset in seismic data acquisition: The Leading Edge, **27**, 924–938.

Bertsekas, D. P., and J. Tsitsiklis, 1996, Neuro-dynamic programming: Athena Scientific.

Betrsekas, D. P., and J. N. Tsitsiklis, 2000, Gradient convergence in gradient methods with errors: Siam Journal of Optimization, **10**, 627–642.

Boonyasiriwat, C., and G. T. Schuster, 2010, 3D multisource full-waveform inversion using dynamic random phase encoding: SEG Technical Program Expanded Abstracts, **29**, 1044–1049.

Byrd, R., G. Chin, W. Neveitt, and J. Nocedal, 2010, On the use of stochastic Hessian information in unconstrained optimization: Technical report, Optimization Center, Northwestern University.

Candès, E., J. Romberg, and T. Tao, 2006, Stable signal recovery from incomplete and inaccurate measurements: **59**, 1207–1223.

Dai, W., C. Boonyasiriwat, and G. T. Schuster, 2010, 3D multi-source least-squares reverse time migration: SEG Technical Program Expanded Abstracts, **29**, 3120–3124.

Donoho, D. L., 2006, Compressed sensing: **52**, 1289–1306.

Erlangga, Y. A., C. W. Oosterlee, and C. Vuik, 2006, A novel multigrid based preconditioner for heterogeneous Helmholtz problems: SIAM Journal on Scientific Computing, **27**, 1471–1492.

Haber, E., M. Chung, and F. J. Herrmann, 2010a, An effective method for parameter estimation with PDE constraints with multiple right hand sides: Technical Report TR-2010-4, UBC-Earth and Ocean Sciences Department.

Haber, E., L. Horesh, and L. Tenorio, 2010b, Numerical methods for the design of large-scale nonlinear discrete ill-posed inverse problems: Inverse Problems, **26**, 025002.

Hansen, P. C., 1998, Rank-deficient and discrete ill-posed problems: Numerical aspects of linear inversion: SIAM.

Hennenfent, G., and F. J. Herrmann, 2008, Simply denoise: wavefield reconstruction via jittered undersampling: Geophysics, **73**, no. 3.

Herrmann, F. J., Y. A. Erlangga, and T. Lin, 2009, Compressive simultaneous full-waveform simulation: Geophysics, **74**, A35.

Hutchinson, M., 1989, A stochastic estimator of the trace of the influence matrix for Laplacian smoothing splines: Communications in Statistics - Simulation and Computation, **18**, 1059–1076.

Ikelle, L., 2007, Coding and decoding: Seismic data modeling, acquisition and processing: SEG Technical Program Expanded Abstracts, **26**, 66–70.

Krebs, J. R., J. E. Anderson, D. Hinkley, R. Neelamani, S. Lee, A. Baumstein, and M.-D. Lacasse, 2009, Fast full-wavefield seismic inversion using encoded sources: Geophysics, **74**, WCC177–WCC188.

Li, X., and F. J. Herrmann, 2010, Fullwaveform inversion from compressively recovered model updates: SEG Expanded Abstracts, **29**, 1029–1033.

Lin, T. T. Y., and F. J. Herrmann, 2007, Compressed wavefield extrapolation: Geophysics, **72**, no. 5, SM77–SM93.

Marfurt, K. J., 1984, Accuracy of finite-difference and finite-element modeling of the scalar and elastic wave equations: Geophysics, **49**, 533–549.

Moghaddam, P. P., and F. J. Herrmann, 2010, Randomized full-waveform inversion: a dimensionality-reduction approach: SEG Technical Program Expanded Abstracts, **29**, 977–982.

Neelamani, N., C. Krohn, J. Krebs, M. Deffenbaugh, and J. Romberg, 2008, Efficient seismic forward modeling using simultaneous random sources and sparsity: Presented at the SEG International Exposition and 78th Annual Meeting.

Nemirovski, A., A. Juditsky, G. Lan, and A. Shapiro, 2009, Robust stochastic approximation approach to stochastic programming: Siam J. Optim., **19**, 1574–1609.

Nesterov, Y., and J.-P. Vial, 2000, Confidence level solutions for stochastic programming: CORE Discussion Papers 2000013, Universit catholique de Louvain, Center for Operations Research and Econometrics (CORE).

Nocedal, J., and S. Wright, 1999, Numerical optimization: Springer. Springer Series in

Operations Research.

Operto, S., J. Virieux, P. Amestoy, L. Giraud, and J. Y. L'Excellent, 2006, 3D frequency-domain finite-difference modeling of acoustic wave propagation using a massively parallel direct solver: a feasibility study: SEG Technical Program Expanded Abstracts, **25**, 2265–2269.

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.

Polyak, B. T., and A. B. Juditsky, 1992, Acceleration of stochastic approximation by averaging: Siam Journal of Control and Optimzation, **30**, 838–855.

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

Riyanti, C., Y. Erlangga, R.-E. Plessix, W. Mulder, C. Vuik, and C. Oosterlee, 2006, A new iterative solver for the time-harmonic wave equation: Geophysics, **71**, E57–E63.

Robbins, H., and S. Monro, 1951, Robust stochastic approximation approach to stochastic programming: Annals of Mathematical Statistics, **22**, 400–407.

Romberg, J., 2009, Compressive sensing by random convolution: SIAM J. Imaging Sciences.

Romero, L. A., D. C. Ghiglia, C. C. Ober, and S. A. Morton, 2000, Phase encoding of shot records in prestack migration: Geophysics, **65**, 426–436.

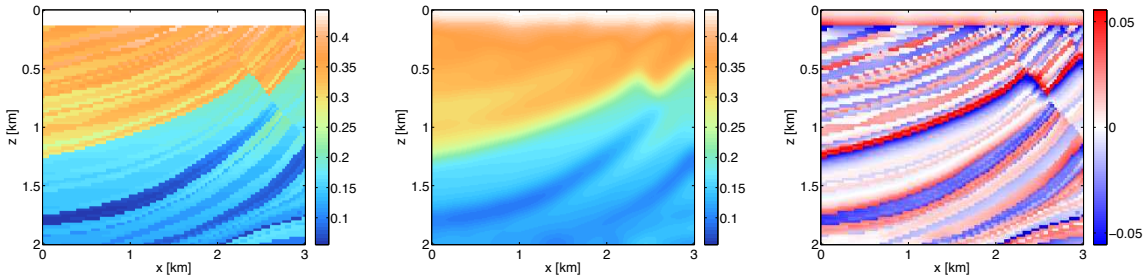
Saad, Y., 1996, Iterative methods for sparse linear systems: PWS publishing company.

Shapiro, A., 2003, Monte carlo sampling methods, *in* Stochastic Programming, Volume 10 of Handbooks in Operation Research and Management Science: North-Holland.

Shapiro, A., and A. Nemirovsky, 2005, On complexity of stochastic programming problems, *in* Continuous Optimization: Current Trends and Applications: Springer, New York.

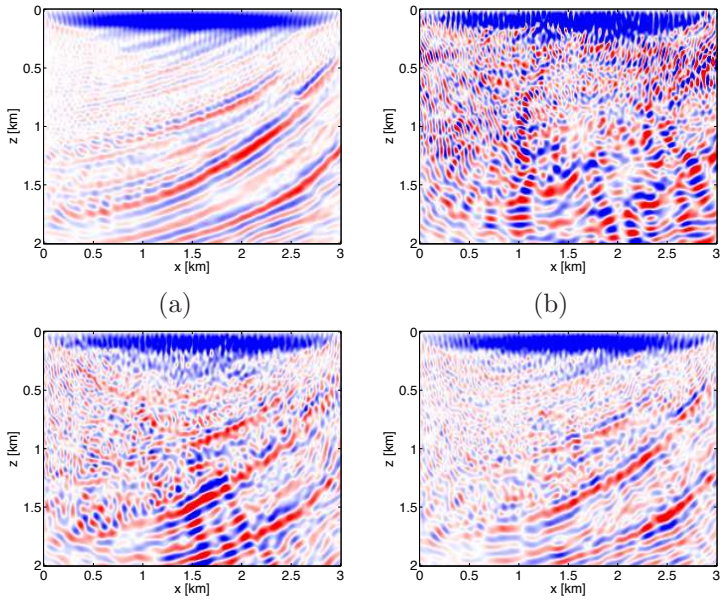
Symes, W., 2010, Source synthesis for waveform inversion: SEG Expanded Abstracts, **29**, 1018–1022.

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



(a) (b) (c)

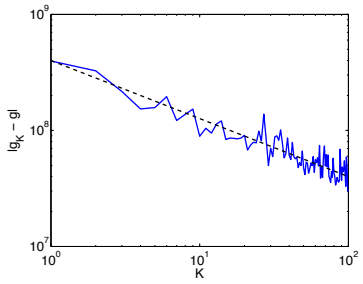
Figure 1: True (a) and initial (b) squared-slowness models (s^2/km^2) and the true reflectivity.



(a) (b)

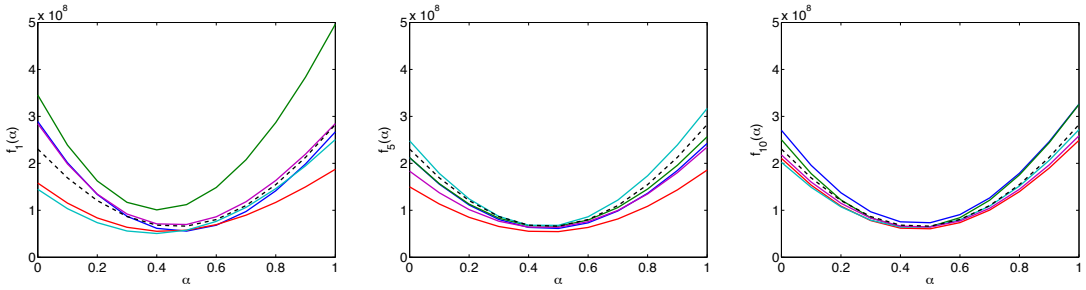
(c) (d)

Figure 2: The full gradient is depicted in (a). The approximate gradients for various K are depicted in (b) $K = 1$, (c) $K = 5$ and (d) $K = 10$. For a relatively small batch size, the approximate gradients already show the main features.



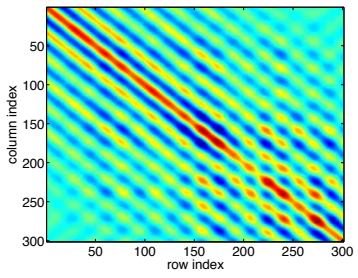
(a)

Figure 3: Error in gradient for as a function of the batch-size K. As expected, the error goes down as $1/\sqrt{K}$ (dashed line).



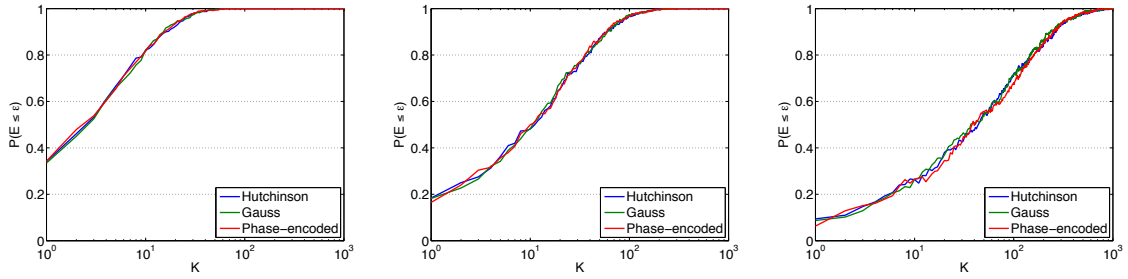
(a) (b) (c)

Figure 4: Behavior of misfit for various K. Shown are five different stochastic realizations and the true misfit (dashed line) for (a) $K = 1$, (b) $K = 5$ and (c) $K = 10$. The stochastic misfits approximate the true misfit fairly well for relatively small batch sizes.



(a)

Figure 5: Residual matrix $A = S^T S$, where S is the data residual corresponding to the smooth model depicted in figure 1 (a) at 5 Hz.



(a) (b) (c)

Figure 6: Reconstruction as a function of K for various methods and error levels: (a) $\epsilon = 10^{-1}$, (b) $\epsilon = 10^{-2}$ and (c) $\epsilon = 10^{-3}$.

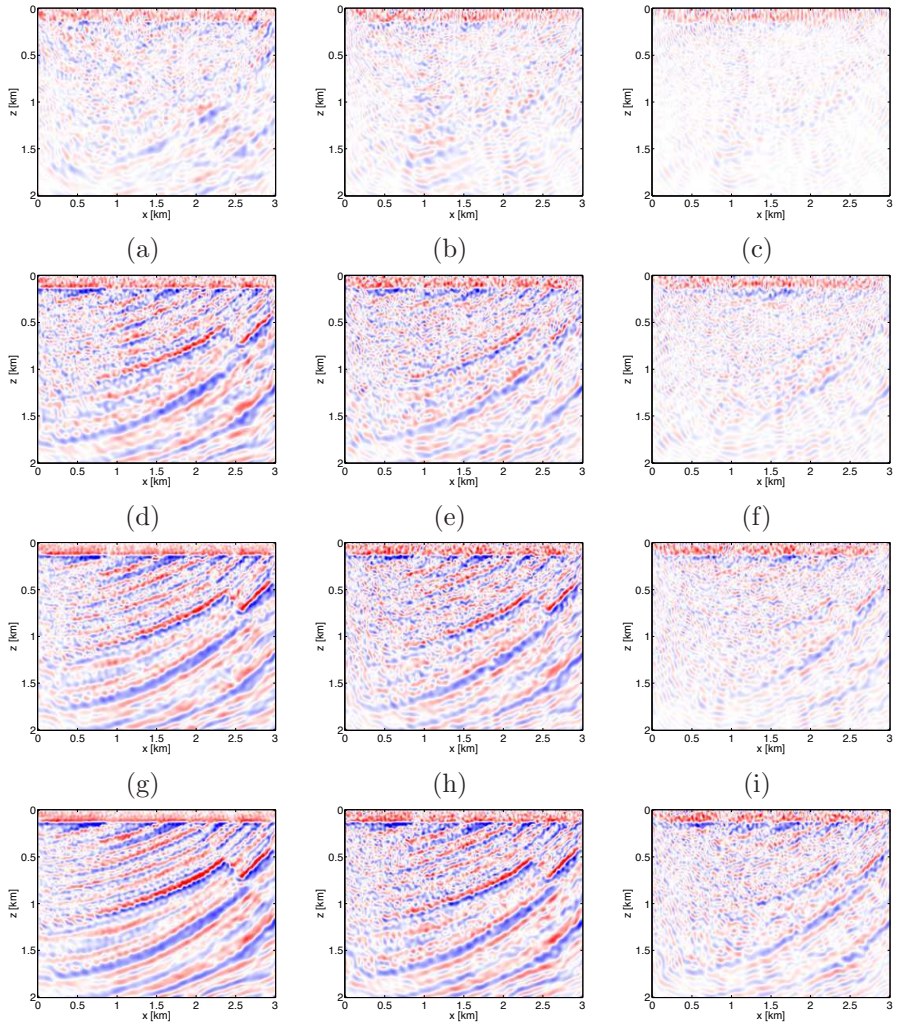
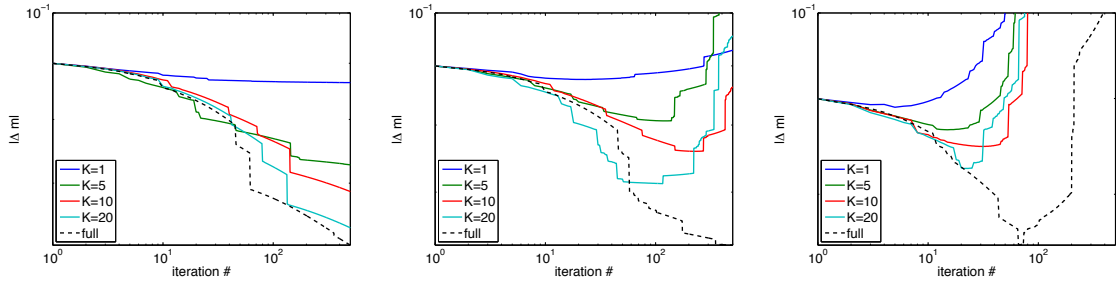


Figure 7: Inversion result for the SAA approach with various batch sizes and noise levels. The rows represent different batch sizes $K = 1, 5, 10, 20$ while the columns represent different noise levels: no noise, SNR=20 dB and SNR=10 dB. The reconstruction with $K = 20$ for noiseless data (j) is qualitatively comparable to the full reconstruction. The quality deteriorates quickly for small batch sizes and noisy data.



(a) (b) (c)

Figure 8: Error between the inverted and true model for the SAA approach with various batch sizes and the full problem, (a) without noise, (b) with noise (SNR=20 dB) and (c) with noise (SNR=10 dB). On noiseless data, we achieve a qualitatively comparable result with $K = 20$, as can be seen from (a). For noisy data, however, the largest batch size is not enough to prevent over-fitting.

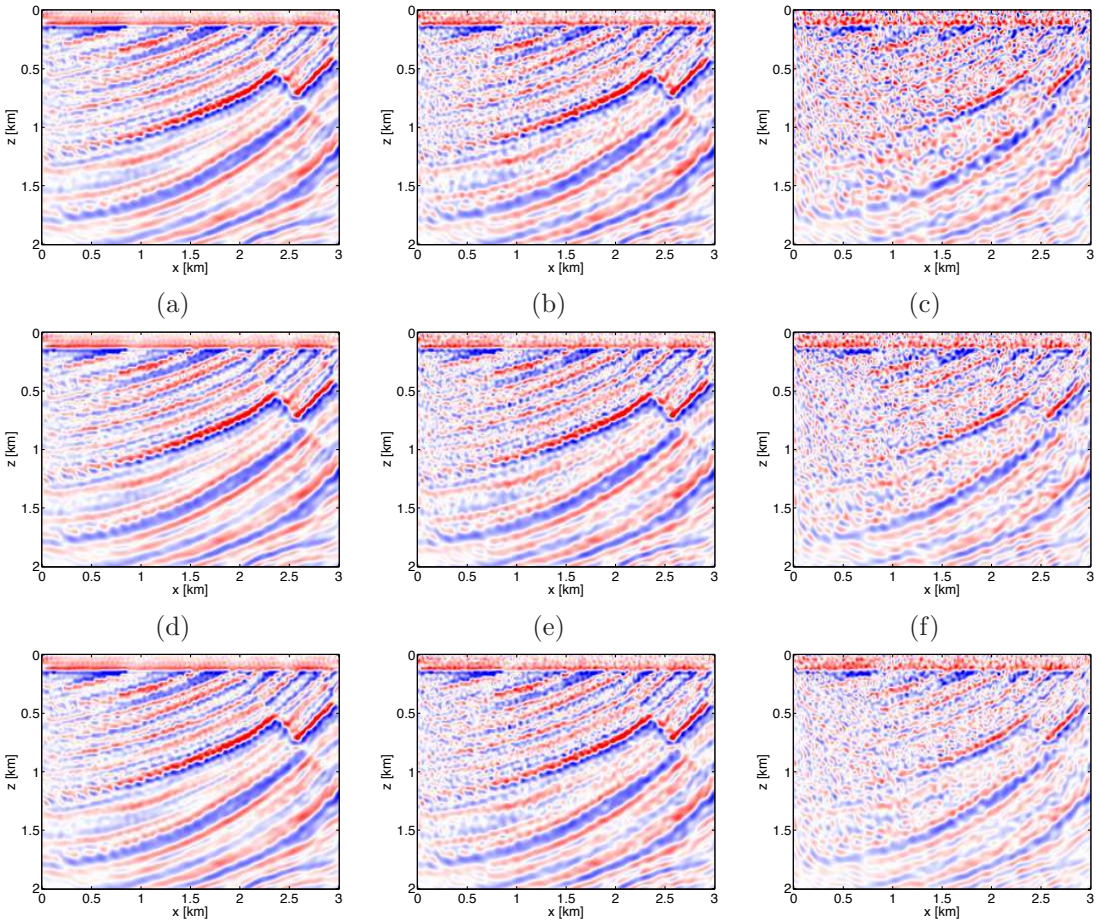


Figure 9: Inversion result for the SA approach without averaging for various batch sizes and noise levels. The rows represent different batch sizes $K = 1, 5, 10$ while the columns represent different noise levels: no noise, SNR=20 dB and SNR=10 dB. We obtain good results with $K = 1$, and the quality does not improve dramatically for larger batch sizes, except for the highest noise level.

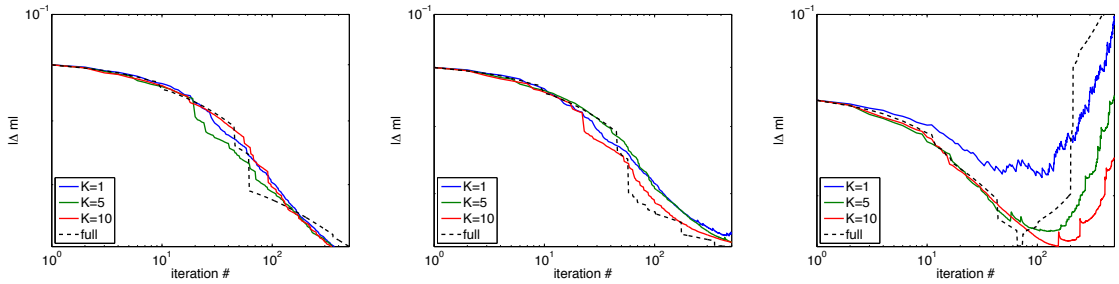


Figure 10: Error between the inverted and true model for the SA approach without averaging for various batch-sizes and the full problem, (a) without noise, (b) with noise (SNR=20 dB) and (c) with noise (SNR=10 dB). We get qualitatively similar results, compared to the full inversion, with $K = 1$ for noiseless data and data with 10 dB of noise. For very noisy data (20 dB) we need a larger batch size. Although the SA approach requires roughly the same number of iterations as the full inversion, the iterations are much cheaper. For $K = 1$, we model the data for only one simultaneous source per iteration, compared to 61 for the full inversion.

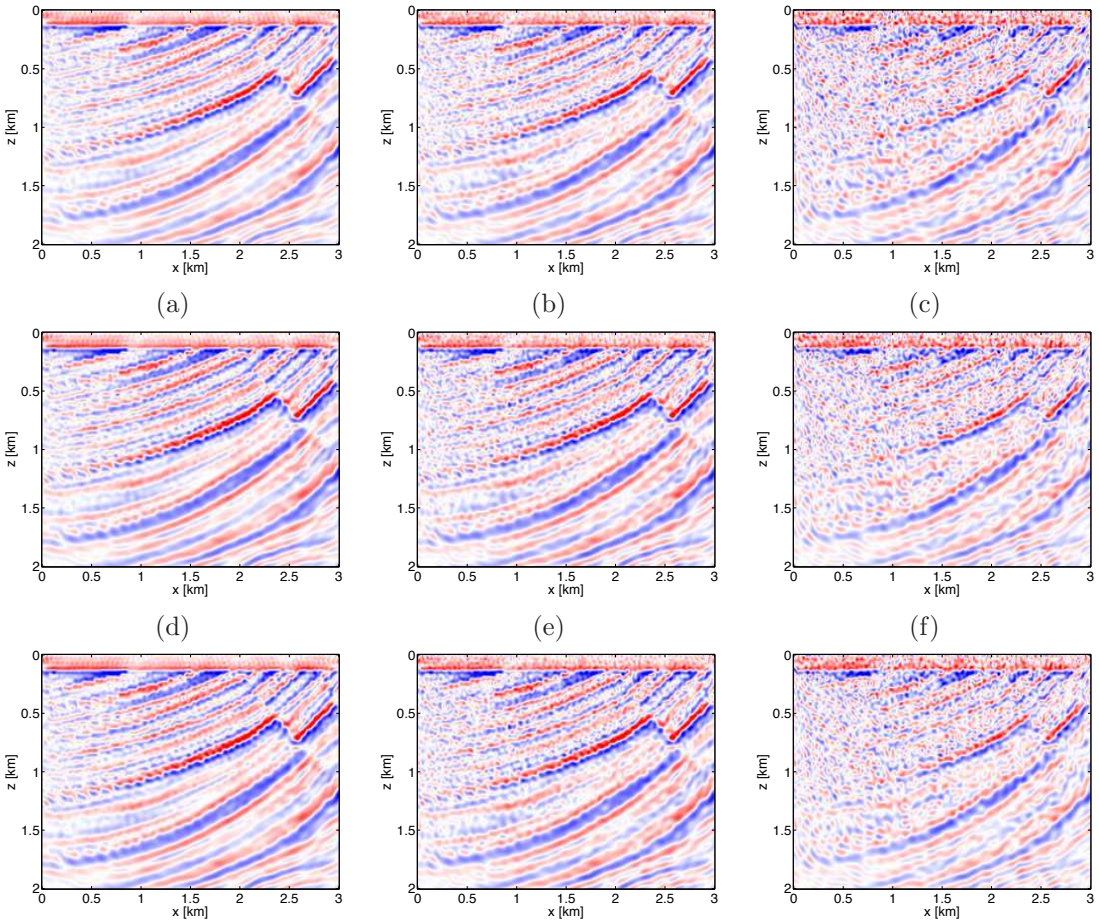
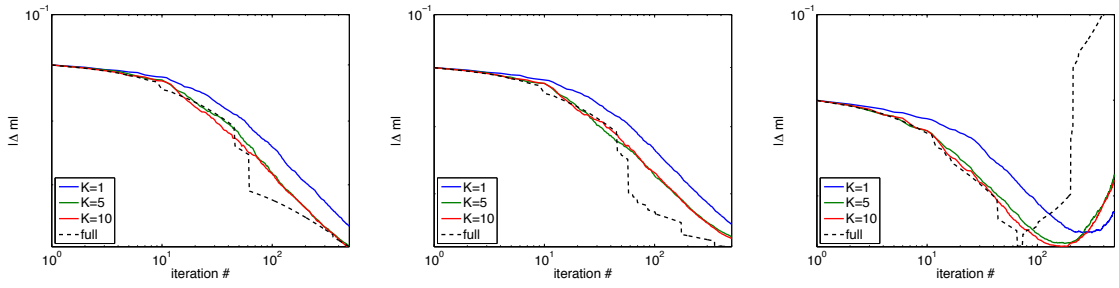


Figure 11: Inversion result for the SA approach with limited averaging ($n = 10$) for various batch sizes and noise levels. The rows represent different batch sizes $K = 1, 5, 10$ while the columns represent different noise levels: no noise, SNR=20 dB and SNR=10 dB. We obtain good results with $K = 1$, and the quality does not improve dramatically for larger batch sizes, except for the highest noise level.



(a) (b) (c)

Figure 12: Error between the inverted and true model for the SA approach with limited averaging for various batch sizes and the full problem, (a) without noise, (b) with noise (SNR=20 dB) and (c) with noise (SNR=10 dB). The convergence is smoother than that of SA without averaging, especially when the data is very noisy (10 dB). The averaging seems to slow down the convergence slightly, however, and we need a batch size $K = 5$ for the best results.

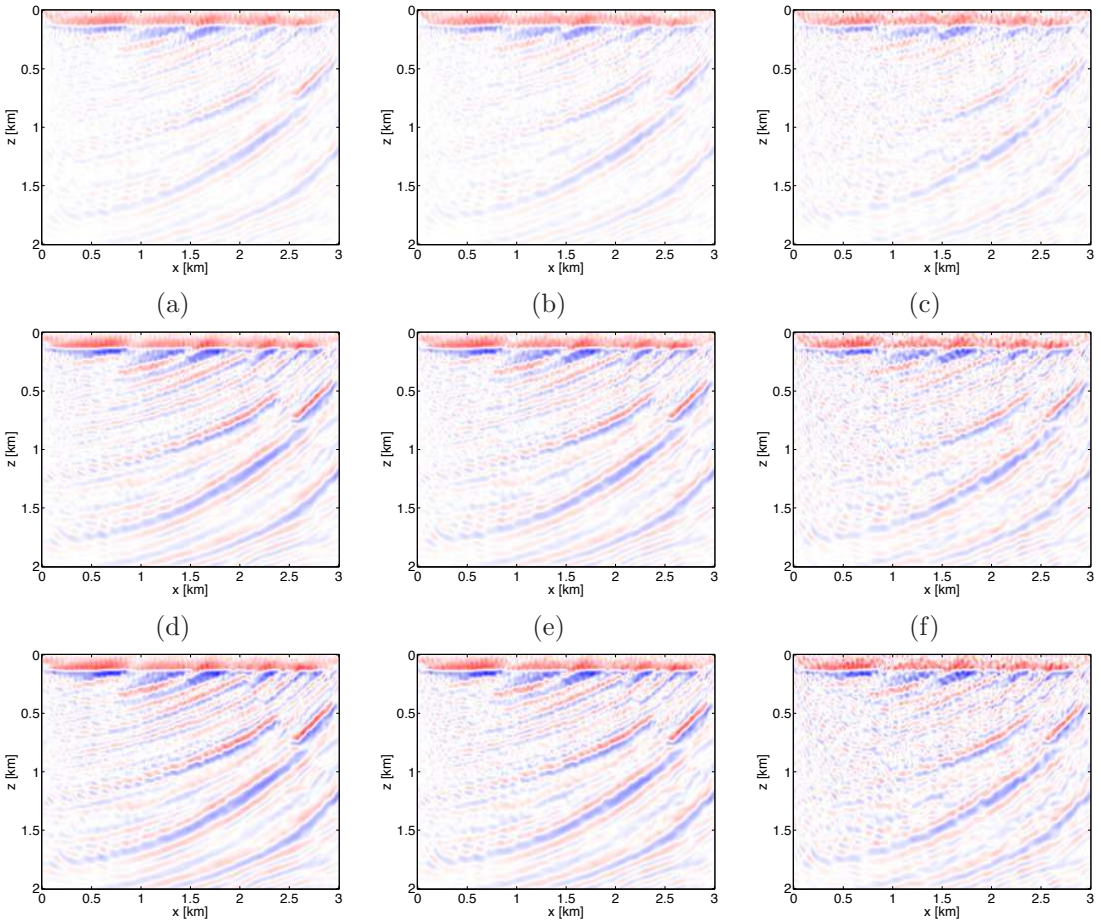
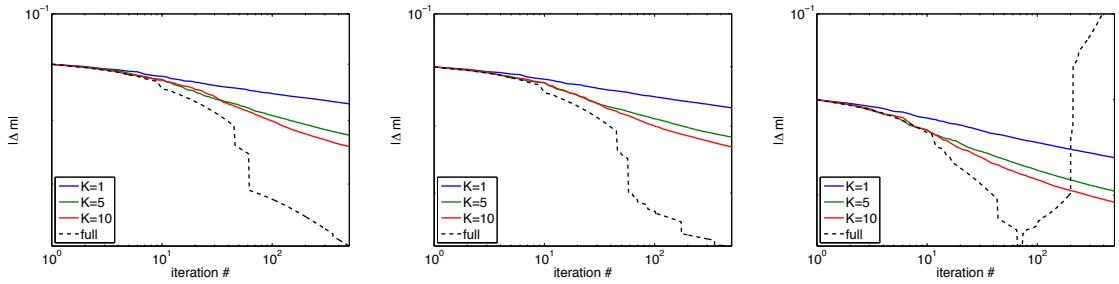


Figure 13: Inversion result for the SA approach with full averaging ($n = 500$) for various batch sizes and noise levels. The rows represent different batch sizes $K = 1, 5, 10$ while the columns represent different noise levels: no noise, SNR=20 dB and SNR=10 dB. Averaging over the full past dramatically deteriorates the reconstrunction.



(a) (b) (c)

Figure 14: Error between the inverted and true model for the SA approach with full averaging for various batch sizes and the full problem, (a) without noise, (b) with noise (SNR=20 dB) and (c) with noise (SNR=10 dB). Averaging over the full past slows down the convergence dramatically.