Geophysical Journal International

Geophys. J. Int. (2015) **200,** 720–744
GJI Seismology

doi: 10.1093/gji/ggu427

Fast full waveform inversion with source encoding and second-order optimization methods

Clara Castellanos,[1] Ludovic Métivier,[2] Stéphane Operto,[3] Romain Brossier[4] and Jean Virieux[4]

[1] *Géoazur, Université de Nice Sophia-Antipolis, Sophia Antipolis, France. E-mail: clarita.castellanos@gmail.com*
[2] *LJK, CNRS, Université de Grenoble Alpes, BP* 53, 38041 *Grenoble Cedex* 09, *France*
[3] *Géoazur, Université de Nice Sophia-Antipolis, CNRS, IRD, OCA, Sophia Antipolis, France*
[4] *ISTerre, Université de Grenoble Alpes, BP* 53, 38041 *Grenoble Cedex* 09, *France*

Accepted 2014 October 29. Received 2014 October 29; in original form 2014 February 25

SUMMARY

Full waveform inversion (FWI) of 3-D data sets has recently been possible thanks to the development of high performance computing. However, FWI remains a computationally intensive task when high frequencies are injected in the inversion or more complex wave physics (viscoelastic) is accounted for. The highest computational cost results from the numerical solution of the wave equation for each seismic source. To reduce the computational burden, one well-known technique is to employ a random linear combination of the sources, rather that using each source independently. This technique, known as source encoding, has shown to successfully reduce the computational cost when applied to real data. Up to now, the inversion is normally carried out using gradient descent algorithms. With the idea of achieving a fast and robust frequency-domain FWI, we assess the performance of the random source encoding method when it is interfaced with second-order optimization methods (quasi-Newton *l*-BFGS, truncated Newton). Because of the additional seismic modelings required to compute the Newton descent direction, it is not clear beforehand if truncated Newton methods can indeed further reduce the computational cost compared to gradient algorithms. We design precise stopping criteria of iterations to fairly assess the computational cost and the speed-up provided by the source encoding method for each optimization method. We perform experiment on synthetic and real data sets. In both cases, we confirm that combining source encoding with second-order optimization methods reduces the computational cost compared to the case where source encoding is interfaced with gradient descent algorithms. For the synthetic data set, inspired from the geology of Gulf of Mexico, we show that the quasi-Newton *l*-BFGS algorithm requires the lowest computational cost. For the real data set application on the Valhall data, we show that the truncated Newton methods provide the most robust direction of descent.

Key words: Inverse theory; Controlled source seismology; Computational seismology; Wave propagation.

1 INTRODUCTION

Full waveform inversion (FWI) is a non-linear ill-posed inverse problem which aims to reconstruct the Earth's parameters such as, for instance, *P*- and *S*-wave velocities, density, attenuation or anisotropic parameters, by fitting seismic data recorded near the surface or at the sea bottom (Lailly 1983; Tarantola 1984; Virieux & Operto 2009). FWI has been shown to be quite successful in 2-D applications under the acoustic (Plessix *et al.* 2012) and elastic (Brossier *et al.* 2009) approximation. More recently, 3-D FWI has also been possible thanks to the increasing development of high performance computing (Plessix 2009; Plessix & Perkins 2010; Sirgue *et al.* 2010; Vigh *et al.* 2013). However, FWI remains computationally intensive, particularly in three dimensions, when high frequencies are injected in the inversion, or complex wave propagation (viscoelastic) is accounted for. Hence, there is a natural interest to reduce this computational burden. There are several existing techniques to achieve this goal such as data decimation (subsample sources, receivers and/or frequencies; Sirgue & Pratt 2004), or stacking the data to reduce the volume [in a deterministic (Gao *et al.* 2010; Habashy *et al.* 2011) or random fashion (Romero *et al.* 2000; Haber *et al.* 2012)]. Data decimation has the setback that it depends

on data redundancy, and not using all the information may in some cases be harmful. On the other hand, stacking the data produces negative cross-talk effects arising from the fact that only the sum of the data is available, and not the original data itself. Random source encoding is one of these techniques that reduces the data volume by replacing the use of individual sources by fewer random linear combinations of sources (Krebs *et al.* 2009; Baumstein *et al.* 2011; Ben Hadj Ali *et al.* 2011; Schuster *et al.* 2011; Choi & Alkhalifah 2012; Haber *et al.* 2012; van Leeuwen & Herrmann 2012).

In order to further improve the computational gain, in the present study we interface random source encoding methods with second-order optimization algorithms, which are expected to have higher convergence rates (Byrd *et al.* 2011; Li *et al.* 2012). In standard FWI, limited memory BFGS (*l*-BFGS) has shown to improve the convergence (Brossier *et al.* 2009). This quasi-Newton method approximates the inverse of the Hessian by performing successive rank-2 updates of an initial estimation from the gradients and the models of the previous l iterations (Byrd *et al.* 1995). The recursive update of the Hessian over iterations in the *l*-BFGS method can be affected by the source encoding method when the random codes are regenerated at each FWI iteration because a regeneration of the code changes the misfit function and hence its gradient. Therefore, we first study how the *l*-BFGS method can be coupled with source encoding strategy based on random encodings, and if a computational gain can be expected from this optimization method.

From a source encoding point of view, the advantage of truncated Newton optimization methods relative to *l*-BFGS is to account for the action of the Hessian only from quantities available at the current iteration. Therefore, the regeneration of the random codes at each non-linear iteration of the inversion is no longer an issue, as it is for the *l*-BFGS method. The drawback is that the truncated Newton approaches require additional seismic modelings per iteration (Métivier *et al.* 2013, 2014). Therefore, we need to assess whether the higher computational cost of one non-linear iteration of the truncated Newton methods can be balanced by an improved convergence rate provided by a more accurate estimation of the Hessian.

In this study, we compare the convergence and the computational efficiency of the above-mentioned optimization methods [non-linear conjugate gradient (*nl*-CG), *l*-BFGS, Gauss–Newton (GN) and full Newton (FN)] when they are implemented in efficient frequency-domain FWI with and without random source encoding. A stopping criterion of iterations is designed allowing for a fair comparison of the optimization methods and a fair assessment of the speed-up provided by the random source encoding method. These work flows are first applied on a realistic synthetic experiment inspired by the geology of the Gulf of Mexico. Then, we assess the benefit provided by random source encoding and second-order optimization methods when applied on a 2-D real ocean-bottom-cable (OBC) data set recorded from the Valhall oil field.

Our paper is organized as follows. We first review the basic principles of FWI in Section 2. We recall the role of the Hessian in FWI and its associated cost in Section 2.2. We describe the preconditioner that we used in Section 2.3, as it plays an important role in the convergence of the optimization methods. The interfacing of source encoding with FWI is described in Section 3. The applications on a synthetic and real case studies are presented in Section 4. Appendix A shows the application on synthetic data with noise. We perform the comparison on two levels. On a first basis we compare the convergence rates and computational costs among different optimization methods with and without source encoding. We measure the convergence rate with the reduction of the misfit function as a function of iterations. On a second instance, for each optimization method we determine the potential computational savings that can be attained (speed-up) with source encoding. For the synthetic example, all the optimization methods converge to a subsurface model of similar accuracy with and without source encoding. We show that the truncated Newton methods have a higher convergence rate than *l*-BFGS and CG, although *l*-BFGS is the fastest method. Although CG is the slowest method, it shows the highest speed-up when source encoding is used because its convergence is less affected by cross-talk noise than Newton-based methods. The real data case study shows that the truncated Newton methods provide the most robust direction of descent leading to subsurface models of similar accuracy, regardless of the encoding. A speed-up of nearly one order of magnitude was attained thanks to a careful design of the stopping criterion of iterations. In Appendix B, we also illustrate that when starting from an inaccurate model, source encoding can help to guide the inversion toward an improved minimum of the misfit function, thanks to a broader exploration of the model space.

2 METHOD

2.1 FWI

Let us define a space $\Omega \subset \mathbb{R}^2$ as a subsurface medium with spatially varying model parameters $m(x)$ which may be, for example, the density $\rho(x)$ and the P-wave velocity $v_p(x)$ in the acoustic approximation. In the frequency domain, the wavefield $u(x, m, \omega)$ satisfies the wave equation,

$$A(x, m, \omega)u(x, m, \omega) = s(x, \omega), \tag{1}$$

where $A(x, m, \omega)$ is the forward modelling operator, which in the acoustic approximation is

$$A(x, m, \omega) = -\frac{1}{\rho(x)v_p(x)^2}\omega^2 - \nabla \cdot \left[\frac{1}{\rho(x)}\nabla\right], \text{ on } \quad \Omega, \tag{2}$$

and the source function is denoted by $s(x, \omega)$. We impose a free surface boundary condition at the surface and absorbing boundary conditions on the other boundaries to simulate an infinite half-space.

The inverse problem thus consists in finding the model m that minimizes the misfit function ϕ that measures the distance from the observed data d to the simulated wavefield u (e.g. Tarantola 1984). The wavefield $u(x)$ is defined on Ω $[u(x) : \Omega \rightarrow \mathbb{R}]$ and $d(x)$ on Ω_r $[d(x) : \Omega_r \rightarrow \mathbb{R}]$, where Ω_r denotes the receiver space. We use as the misfit function the l_2 norm of the difference between the modelled and the recorded wavefields,

$$\min_m \phi(u; m) = \min_m \frac{1}{2}\sum_{i=1}^{N_s} \| Pu_i(m) - d_i \|_2^2, \tag{3}$$

where the misfit ϕ depends explicitly only on u and implicitly on m, and N_s is the number of sources. Since $\Omega \neq \Omega_r$, we use a projection operator P from the whole space Ω to the receiver space, $P : \Omega \rightarrow \Omega_r$. For sake of compactness we shall consider one frequency, although we shall implement multifrequencies inversion in our applications for which we add an external sum over frequencies. The minimization of ϕ is an iterative process, where the subsurface model is updated around an initial model m^n along a direction of descent Δm.

$$m^{n+1} = m^n + \alpha^n \Delta m^n, \tag{4}$$

where n is the iteration index, α is a step length (i.e. the quantity of descent), and the descent direction Δm is given by the

optimization algorithm of choice. The step length satisfies the weak Wolfe conditions (Nocedal & Wright 2006). For steepest-descent (SD) algorithms, the descent direction is opposite to the gradient, $\Delta m = -\mathcal{P}^n \nabla_m \phi(u; m)$, where $\mathcal{P}$ is a pre-conditioner as explained in Section 2.3, that changes in each iteration n. The adjoint-state method allows to compute the gradient solving only two forward problems per source, using the following expression (Lions 1968; Plessix 2006; Chavent 2009),

$$\nabla_m \phi(u; m) = \sum_{i=1}^{N_s} \Re \left(\frac{\partial A}{\partial m} u_i, \lambda_i \right), \tag{5}$$

where (., .) is the scalar product which induces the L^2 norm $\|.\|_2$ used to define the misfit function $\phi(u; m)$, and the back-propagated wavefield λ satisfies the adjoint-state equation

$$A^\dagger \lambda_i = -P^\dagger (P u_i - d_i), \tag{6}$$

where $\dagger$ denotes the adjoint operator. For each iteration n, the gradient (5) requires the solution of one forward problem for $u(m)$ and one for $\lambda(m)$, for each source: the gradient computation in each iteration requires to perform $2 \times N_s$ forward problems.

2.2 Second-order optimization methods

SD algorithms have at best linear convergence rates but are the easiest to implement. Another first order optimization method that, despite having a linear convergence rate has lower constants bounding the errors, is CG and its variants. Nonetheless, to improve even further the convergence rate it may be beneficial to include information about the curvature of the misfit function, known as the Hessian H. Assuming the misfit function close to the starting and current point is quadratic, a Taylor expansion up to second order of the misfit function gives rise to the so-called Newton equations for the model update Δm,

$$H \Delta m = -\nabla \phi. \tag{7}$$

For FWI, the expression of the Hessian is given by (Pratt *et al.* 1998)

$$H = \nabla_m^2 \phi = \sum_{i=1}^{N_s} \Re \left(P \frac{\partial u_i}{\partial m}, P \frac{\partial u_i}{\partial m} \right) + \left(P \frac{\partial^2 u_i}{\partial m^2}, P u_i - d \right). \tag{8}$$

The first term in eq. (8) is known as the GN approximation and employing both terms is referred to as full Newton (FN). Multiplying the gradient by the inverse of the GN approximation operator gives a model perturbation with correct physical units by removing wave-propagation effects such as geometrical spreading and deconvolving the gradient from limited bandwidth effects. The second-order term of the Hessian in eq. (8), aims to correct the gradient for artefacts associated with double-scattering effects not considered in the gradient (Pratt *et al.* 1998).

Despite its importance (Pratt *et al.* 1998), involving the Hessian in the optimization process is often computationally too expensive. The *l*-BFGS method handles this difficulty by storing in memory the previous l computations of the gradient and of the solution, and using them to perform successive rank-2 updates of an initial estimation (Byrd *et al.* 1995; Nocedal & Wright 2006). Specifically, the *l*-BFGS algorithm is quite efficient, as it finds directly the model update through a matrix-free recursive algorithm that computes the multiplication of the *l*-BFGS inverse Hessian by the gradient (algorithm 9.1 in Nocedal & Wright 2006). For the specific case of FWI, *l*-BFGS has shown very good results (Brossier *et al.* 2009) and requires no additional solutions of direct problems.

It is also possible to solve the Newton system in eq. (7) approximately with a matrix-free iterative solver, such as the linear CG method. The maximum number of iterations that are executed to solve the linear system (7) is restricted, giving rise to the name truncated Newton methods. Only the computation of Hessian-vector products is required, which can be performed through the solution of additional direct and adjoint problems, following second-order adjoint strategies. As a result, at each iteration n, the descent direction given by the FN and GN truncated Newton methods requires the solution of $(2 + 2 \times N_{CG}) \times N_s$, where N_{CG} stands for the number of iteration performed by the linear CG solver. The reader is referred to Métivier *et al.* (2013, 2014) for a detailed description of these methods and their application to FWI.

2.3 Pre-conditioner

In order to reduce the number of iterations performed by the linear CG algorithm and hence to reduce the cost of the truncated Newton methods, we apply a left pre-conditioner to the Hessian matrix leading to the preconditioned Newton system:

$$\mathcal{P}^{-1} H \Delta m = -\mathcal{P}^{-1} \nabla \phi. \tag{9}$$

In this study, we follow Métivier *et al.* (2014) and use as pre-conditioner the diagonal elements of the so-called pseudo-Hessian matrix that was introduced by Shin *et al.* (2001) for depth migration. The pseudo-Hessian matrix is formed by the zero-lag correlation of the so-called virtual sources, $(\frac{\partial A}{\partial m} u)$ (Pratt *et al.* 1998), while the GN Hessian is formed by the zero-lag correlation of the partial derivative wavefields at the receiver positions, (8). The advantage of the pseudo-Hessian matrix relative to the GN Hessian matrix is that its expression does not depend on the receiver positions. Therefore, its diagonal elements can be computed with no additional cost once, one the gradient is calculated. Although the wave paths from the receiver to the model parameter are not taken into account in the pseudo-Hessian, the diagonal elements of the pseudo-Hessian provide a suitable scaling of the gradient that make the deep perturbations well balanced relative to the shallower ones. A damping coefficient β, defined as a fraction of the maximum diagonal coefficient, is added to the diagonal elements of the pseudo-Hessian to prevent instabilities resulting from division by very small numbers (Ravaut *et al.* 2004).

The same pre-conditioner is used for the SD algorithm, where the pre-conditioner is multiplied with the gradient to give the descent direction, which resembles a first approximation to a GN step. For the *l*-BFGS method, the pre-conditioner is used as an initial estimation of the Hessian in each iteration, also helping the convergence of the optimization (Métivier *et al.* 2013).

2.4 Tikhonov regularization

A standard Tikhonov regularization term is added to the misfit function so as to deal with the ill-posedness of the FWI, which results from noise and incomplete illumination provided by surface acquisition.

$$\begin{aligned} \phi(u; m) &= \frac{1}{2} \|\Delta d\|^2 + \frac{1}{2} \lambda_x \|\nabla_x m\|^2 + \frac{1}{2} \lambda_z \|\nabla_z m\|^2 \\ &= \frac{1}{2} \|\Delta d\|^2 + \frac{1}{2} \lambda_x \|W_x m\|^2 + \frac{1}{2} \lambda_z \|W_z m\|^2, \end{aligned} \tag{10}$$

where $\Delta d = Pu - d$. The Tikhonov regularization augments the data-space misfit function with smoothing constraints in the horizontal ($\|\nabla_x m\|_2^2$) and vertical ($\|\nabla_z m\|_2^2$) directions, whose partial weights are given by two hyperparameters λ_x and λ_z. It is recalled that the Newton system gives the following descent direction:

$$\Re\left(H + \lambda_x W_x + \lambda_z W_z\right)\Delta m = -\Re\left(\nabla\phi + \lambda_x W_x^\dagger W_x m + \lambda_z W_z^\dagger W_z m\right). \quad (11)$$

A suitable trade-off should be found in the Hessian for the Newton based optimization methods between the action of the second derivative of the data misfit used to improve the descent direction (H) and smoothing regularization constraint ($\lambda_x W_x^\dagger W_x m + \lambda_z W_z^\dagger W_z m$). If the latter is stronger than the former, the second order information of the data misfit is outweighed and will have no impact in the descent direction.

3 SOURCE ENCODING

3.1 Random source encoding

As the solution of the forward problem is commonly the most intensive computational part in FWI due to numerous right-hand sides in the direct problem (2), several authors (Krebs *et al.* 2009; Ben Hadj Ali *et al.* 2011; Schuster *et al.* 2011; Haber *et al.* 2012; van Leeuwen & Herrmann 2012) have explored the possibility of creating a linear combination of the sources into one (or several) supersources $\tilde{s}_k$, defined as

$$\tilde{s}_k = \sum_{i=1}^{N_s} \alpha_i^k s_i, \quad (12)$$

where k labels one supersource, $k = 1, 2, \ldots K$. The quantities $\alpha_i^k \in \mathbb{C}$ are random complex scalar coefficients that satisfy (Haber *et al.* 2012),

$$\mathbb{E}\left[\alpha_i^* \alpha_j\right] = \delta_{i,j}, \quad (13)$$

where $\mathbb{E}$ stands for the expectation over α. As a consequence of the linearity of the solution of the direct problem u with respect to the source s, and the solution of the adjoint problem λ and its source, it follows that the wavefields can be expressed as,

$$\tilde{u}_k = \sum_{i=1}^{N_s} \alpha_i^k u_i \quad (14)$$

$$\tilde{\lambda}_k = \sum_{i=1}^{N_s} \alpha_i^k \lambda_i. \quad (15)$$

These are the new encoded wavefields. Following exactly the same procedure as for the case without encoding, the misfit function and the gradient are

$$\tilde{\phi}(\tilde{u}; m) = \frac{1}{2}\sum_{k=1}^{K} \|P\tilde{u}_k(x, m) - \tilde{d}_k(x)\|_2^2 \quad (16)$$

$$\nabla_m \tilde{\phi}(\tilde{u}; m) = \sum_{k=1}^{K} \Re\left(\frac{\partial A}{\partial m}\tilde{u}_k, \tilde{\lambda}_k\right), \quad (17)$$

where $\tilde{d}_k(x) = \sum_{i=1}^{N_s} \alpha_i^k d_i$ denotes the encoded recorded data set corresponding to the supersource k.

It is possible to see the relation between the deterministic gradient $\nabla_m \phi(u; m)$ and the encoded gradient $\nabla_m \tilde{\phi}(\tilde{u}; m)$ by replacing the expression of the encoded wavefields (14), (15) in (17),

$$\nabla_m \tilde{\phi}(\tilde{u}; m) = \Re \sum_{i=1}^{N_s} \alpha_i \alpha_i^* \left(\frac{\partial A}{\partial m} u_i, \lambda_i\right) + \sum_{i=1}^{N_s}\sum_{j\neq i}^{N_s} \alpha_j \alpha_i^* \left(\frac{\partial A}{\partial m} u_i, \lambda_j\right). \quad (18)$$

Note that the first term in the encoded gradient is a linear combination of the deterministic gradient generated by each individual source, with weights given by $\alpha_i \alpha_i^*$. The second term is proportional to the correlation of incident and backpropagated wavefields u_i and λ_j, generated by different sources $i \neq j$. This second term is therefore commonly referred to as cross-talk because the correlation of wavefileds produced by different sources arises only as a consequence of assembling the sources into supersources and, having no physical meaning in the imaging condition, introduces artefacts in the gradient. When then condition (13) on the random coefficients is satisfied,

$$\mathbb{E}\left[\left(\alpha_i \frac{\partial A}{\partial m} u_i, \alpha_i \lambda_i\right)\right] = \left(\frac{\partial A}{\partial m} u_i, \lambda_i\right) \quad \text{and}$$

$$\mathbb{E}\left[\left(\alpha_i \frac{\partial A}{\partial m} u_i, \alpha_j \lambda_j\right)\right] = 0, \quad (19)$$

meaning the expected value of the encoded gradient is equal to the deterministic gradient, $\mathbb{E}[\nabla_m \tilde{\phi}(\tilde{u}; m)] = \nabla_m \phi(u; m)$.

3.2 Optimization algorithms with source encoding

When encoding the sources, we wish to regenerate the codes as often as possible in order to have an average crosstalk term that tends to zero (Krebs *et al.* 2009). For the first-order optimization methods, we degrade the pre-conditioned *nl*-CG algorithm to a preconditioned SD optimization to use only information of the current iteration, and regenerate the encodings α in every iteration. For the case of *l*-BFGS, we keep the same encoding for intervals of l iterations with which an estimate Hessian is constructed. At the end of the l_{th} iteration we regenerate the random variables, delete the gradients and models stored in memory, and restart the Hessian from a pseudo-Hessian approximation. This restart version of *l*-BFGS, referred to as *l*-BFGS$_r$ in the following, allows to approximate the Hessian using gradients computed with the same encodings, but also to regenerate the random variables throughout the inversion to attenuate the crosstalk terms. Since *l*-BFGS does not require any additional computations of forward problems per non-linear iteration, the potential computational gain using this method or SD with source encoding is $2 \times N_s$ versus $2 \times K$. The potential computational gain using truncated Newton methods with source encoding is $(2 + 2 \times N_{CG}) \times N_s$ versus $(2 + 2 \times N_{CG}) \times K$. The computational cost per iteration for each optimization method with and without source encoding are summarized in Table 1. It should be noted that these costs do not include the differences in memory requirements and input/output overheads, which may be greater with source encoding.

4 NUMERICAL EXAMPLES

We apply frequency-domain FWI on synthetic and real data sets to compare the behaviour of different optimization algorithms when combined with source encoding techniques. The forward problem is

Table 1. Comparison of the total number of forward problems solved in each FWI iteration with and without source encoding for each optimization algorithm. The speed-up S (per cent) defined in eq. (22) represents the computational gain of each method per iteration. For all methods, the computational grain per iteration is $1 - K/S$.

Optimization	*DP* without *SE*	*DP* with *SE*
CG	$2 \times N_s$	$2 \times K$
l-BFGS	$2 \times N_s$	$2 \times K$
GN	$(2 + 2 \times N_{CG}) \times N_s$	$(2 + 2 \times N_{CG}) \times K$
FN	$(2 + 2 \times N_{CG}) \times N_s$	$(2 + 2 \times N_{CG}) \times K$

solved with the 2-D second-order acoustic wave equation for pressure (2), which is discretized with a 9 point mixed-grid finite difference stencil on a regular Cartesian grid of N points. We perform the synthetic experiment with the isotropic wave equation (Hustedt *et al.* 2004), while we introduce anisotropic effects (for vertical transversely isotropic media) in the modelling during the inversion of the real data (Operto *et al.* 2009). The absorbing boundary conditions are implemented with perfectly matched layers (PML; Bérenger 1994). The linear system resulting from the discretization of the frequency-domain wave eq. (1) is solved with the sparse direct solver MUMPS which first performs a LU factorization of the wave-equation operator before computing the monochromatic solutions for multiple right-hand sides (i.e. sources) by substitution (MUMPS-team 2011). Notice that the same LU factorization is valid for all the sources and only depends on the current model m and frequency ω. This is beneficial for the truncated Newton methods because the LU factors can be reused to perform the additional forward problems that allow for the Hessian-vector products during the iterative resolution of the Newton linear system. We fix the density and the Thomsen's parameters (for the anisotropic case) and perform a mono-parameter inversion for the wave speed v_p (the vertical wave speed in the anisotropic case). Thus, in our case, $m = v_p$, leading to N unknowns.

The frequency-domain FWI is decomposed into successive inversions of frequency groups with a limited overlap. Each group is composed of a limited number of discrete frequencies that significantly reduces the intrinsic redundancy of the inverted data. The high-frequency content increases from one frequency group to the next, hence defining a multiscale approach of FWI, which helps to mitigate the non-linearity of the FWI (e.g. Bunks *et al.* 1995; Sirgue & Pratt 2004). We consider fixed-spread acquisitions for both the synthetic and real data case studies, for which source encoding methods are suitable.

A suitable stopping criterion of iterations should be defined, as a fair assessment of the speed-up provided by source encoding methods when combined with different optimization methods. The maximum number of FWI iterations that are performed during each frequency-group inversion is controlled mainly by the relative reduction of the misfit function below a predefined threshold ϵ_1

$$\phi(m_n)/\phi(m_0) < \epsilon_1. \tag{20}$$

Note that, even when source encoding is applied, we pay the price to compute periodically the deterministic misfit function using all the sources independently to test whether this stopping criterion of iterations is reached. In addition, we impose a pre-defined maximum value of forward problems, in case the expected relative reduction of the misfit function is not attained.

The optimization methods considered when using all the sources independently are the pre-conditioned *nl*-CG, *l*-BFGS, pre-conditioned truncated GN and the pre-conditioned truncated FN approximation. For *l*-BFGS the pre-conditioner $\mathcal{P}$ is used an initial guess for the Hessian in each iteration. When the sources are encoded, the four optimization methods used are pre-conditioned SD, limited memory BFGS with periodic restart (*l*-BFGS$_r$) and pre-conditioned truncated GN and FN. To encode the sources and the data we use a Gaussian distribution for the random variables,

$$\gamma_i \backsim \mathcal{N}(0, 1) \tag{21}$$

which satisfy the desired properties (13). We tested encoding with other distributions satisfying the imposed conditions, but we did not observe any significant differences to be reported.

For a fair comparison of the different optimization algorithms, we use the same value of the hyperparameters (λ_x, λ_z) and threshold value in the pre-conditioner β for all optimization methods. When using source encoding, the same number of supersources is used for all optimization methods. Indeed, the hyperparameters and threshold value in the pre-conditioner shared by the four optimization methods are adapted during each experiment to the problem at hand, in particular to the signal-to-noise ratio in the data. A suitable trade-off should be found in the Hessian for the Newton and quasi-Newton optimization methods between the action of H used to improve the descent direction and smoothing regularization constraint ($\lambda_x W_x^\dagger W_x m + \lambda_z W_z^\dagger W_z m$). If the latter is stronger than the former, the second order information of ϕ is outweighed and will have no impact in the descent direction. High values of the hyperparameters λ_x, λ_z may however be necessary to mitigate the effect of noise to follow a robust direction of descent and prevent convergence towards a local minimum. In the inversion with source encoding, the crosstalk term can be seen as an increase in the noise level in the data-space misfit function. This implies that depending on the cross-talk level, FWI with source encoding can require to increase the hyperparameters λ_x and λ_z relative to those of FWI without source encoding to account for the increased level of noise, thus degrading the precision of the Hessian approximation.

4.1 The quantities used for comparison: convergence rate, computational efficiency and speed up

For each data set we perform the inversion with and without source encoding, to compare the differences in the computational costs and in the number of iterations required to attain a desired relative reduction of the misfit value. We compare these values using four different optimization algorithms.

(i) We measure the 'convergence rate' by looking at how $\phi(m_n)$ tends to $\phi(m^*)$ as a function of the iterations, where $\phi(m^*)$ is the value of the misfit evaluated at the optimal model m^*. The optimization method requiring the fewest number of iterations n to reach a predefined $\phi(m^*)$ means it has the highest convergence rate.

(ii) We also measure the 'computational efficiency' through the number of direct problems required to attain a given misfit function reduction. The method solving the fewest number of forward problems provides the highest computational efficiency.

(iii) Using the measure of the number of direct problems solved, we provide 'speed up' values that compare the computational efficiency of the same optimization method, with and without source encoding. We define the speed up as a function of the ratio of

the number of forward problems solved with and without source encoding,

$$S = \left(1 - \frac{DP_s}{DP_d}\right) \times 100 \text{ per cent}, \tag{22}$$

where DP_d denotes the number of forward (direct) problems when using the full set of sources independently and DP_s the number of forward problems with source encoding. This includes the number of forward problems solved during the line search. Since the speed up is a measure that compares the number of forward problems solved with and without source encoding for *one* specific method, the speed up measure can not be used to compare among different optimization methods. Rather, the speed up measure is used to determine how much the efficiency of a certain optimization method is improved when combining it with source encoding.

(iv) When source encoding is applied, random variables α need to be drawn from a probability distribution. Different realizations of the random variables may lead to different solutions. Therefore, when source encoding is used, the convergence rate curves and the computational efficiency curves represent averages over an ensemble of 50 independent realizations.

To quantify the 'statistical stability' we measure the variability of the final models with source encoding by computing their sample variance,

$$\text{Var}(m) = \frac{1}{MC}\sum_{j=1}^{MC}(m_j - \bar{m})^2 \tag{23}$$

$$\bar{m} = \frac{1}{MC}\sum_{j=1}^{MC} m_j, \tag{24}$$

where MC is the number of realizations (in this case $MC = 50$), m_j is the final velocity model of realization j.

4.2 Experimental protocol

We define the relative reduction of the misfit function for each frequency group ϵ_1, the number of supersources K to be used, the maximum number of iterations allowed to solve the truncated system N_{CG} and the values of the hyperparameters λ_x and λ_z. The convergence and computational efficiency curves are determined for each optimization method, with and without source encoding. A speed up measure can then be deduced for each optimization method.

We apply this protocol initially to the BP-2004 salt model using data without noise and observe that without source encoding the convergence rate is notably higher for optimization methods that take into account the effect of the Hessian. On the contrary, when source encoding is applied, the difference in convergence rates is similar for all optimization methods, indicating that including the Hessian information does not improve considerably the descent direction. We observe that *l*-BFGS is the most computationally efficient method, solving the fewest number of direct problems. The method with the highest speed up (the method that has the largest difference when used with and without source encoding) is CG. The final velocity models provided by all optimization methods, with and without encoding, are the same.

When using the Valhall real data set, the comparison between optimization methods is more difficult because the final value of the misfit value is not the same for all optimization methods. In particular, we observe that *l*-BFGS terminates the inversion early due to failed line searches. It is not straightforward to determine the most computationally efficient method, or the one with the highest convergence rate because the terminal point of the optimization is not the same. The average final velocity models are also not the same for all optimization methods. We therefore analyse the statistical stability of each method and conclude that the truncated Newton methods are the most reliable when using source encoding and provide final data misfits that are lower than other optimization methods.

4.3 Synthetic example

The BP-2004 salt velocity model is a benchmark for seismic imaging whose key attribute is that it has a considerable difference in the velocity of P waves between the water ($\approx$1500 m s^{-1}) and the salt ($\approx$4899 m s^{-1}; Fig. 1a). This sharp contrast generates high-amplitude primary reflection arrivals from the salt as well as energetic multiples between the salt and the free surface. This makes the recovery of the subsalt structures difficult because the sharp velocity contrast on top and on bottom of the salt hampers the transmission of a significant amount of seismic energy below salt and the information in the seismograms that constrains these parts of the model can be overprinted by high-amplitude multiples. We consider a limited target of the BP-2004 velocity model of horizontal and vertical dimensions 6.2 km $\times$ 4.2 km, respectively. The sources and receivers are deployed all along the surface at 25 m in depth below the water level. There are 62 sources and 248 receivers with a horizontal spacing of 100 and 25 m, respectively. The initial velocity model is a smoothed version of the true model (Fig. 1b). The velocity in the water layer is kept unchanged during inversion (we set the gradient to zero), since we do not want to update the velocity in this area. We use two frequency groups without overlap, with a frequency interval of 1 Hz: [1, 2, 3, 4] Hz, [5, 6, 7, 8, 9, 10] Hz. For

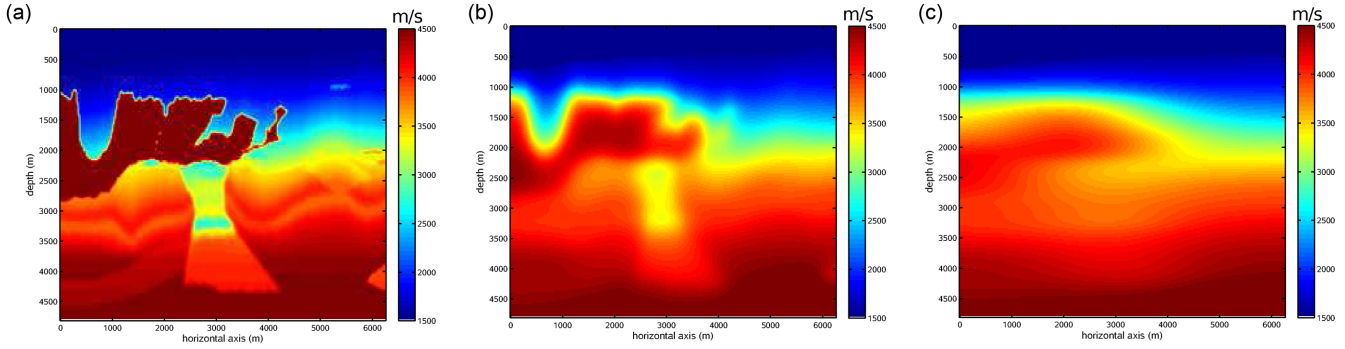

Figure 1. BP-2004 Salt v_p model. (a) True model. (b) Initial model. (c) Smoother initial model.

all of the tests, we fix the maximum number of forward problems to 10^5.

4.3.1 Illustration of the problem: convergence rates of stochastic versus deterministic algorithms

To introduce the issues related with source encoding we compare the convergence rates between deterministic and stochastic methods, and illustrate with a numerical example that deterministic methods have higher convergence rates. As a consequence, the potential computational gain provided by source encoding depends on the point where the inversion is terminated.

The convergence rate of deterministic gradient-based algorithms depends on the properties of the misfit function, such its convexity and smoothness. For a simple gradient algorithm, when the misfit function is convex and smooth, it can attain a linear convergence rate $O(1/I)$ and can be even improved with accelerated gradient algorithms to $O(1/I^2)$, where I is the iteration number (Boyd & Vandenberghe 2009). However, the convergence rate of stochastic gradient algorithms is $O(1/\sqrt{I})$ and Nemirovsky & Yudin (1983) showed it can not be easily improved. To the present, there are no general theoretical proofs for the convergence for convex second-order stochastic optimization methods (Bottou & Le Cun 2005; Bottou & Bousquet 2011). Under certain conditions on how the approximate Hessian converges towards the Hessian, a proof of convergence for second-order stochastic methods can be obtained (Bottou & Le Cun 2005). In particular, *l*-BFGS does not satisfy the conditions on the Hessian and thus there is no convergence proof for this stochastic optimization algorithm (Schraudolph *et al.* 2007). Moreover, even in the case where the conditions on the convergence on the approximate Hessian are satisfied and second-order stochastic methods converge, the convergence rate is not improved and only the constants bounding the errors are decreased (Bottou & Bousquet 2011). Nonetheless, Schraudolph *et al.* (2007) implemented a stochastic *l*-BFGS algorithm and showed that it outperforms standard stochastic gradient algorithms for convex functions for some large scale applications.

These results are well established in the machine-learning community which mainly treat convex misfit functions. However, it is not clear how these results extend for large-scale non-convex optimization problems. We perform a numerical test in the FWI context to understand the convergence properties of stochastic versus deterministic optimization algorithms. Using the BP-2004 salt model and without noise in the data, we solve the inverse problem using only the first frequency group (1–4 Hz). The only criterion to stop the inversion is when it reaches the maximum number of forward problems (10^5), or when the line search fails to satisfy the Wolfe conditions.

The misfit function versus the iteration number using *l*-BFGS is plotted in Fig. 2(a). To extract the information on the convergence rates for different optimization methods, we plot the reduction of the misfit function on a log–log plot, as shown in Fig. 3(a). The coefficients 1.26, 1.89 and 3.56 of the linear interpolation correspond to the asymptotic convergence rate, for CG, *l*-BFGS and GN, respectively. Notice that the convergence rate increases as the Hessian estimation becomes more accurate. We compare the convergence rates of the deterministic methods to those with source encoding, shown in Fig. 3(b). The convergence rates are approximately 1.02, 1.19 and 1.46 for SD, *l*-BFGS and GN. Again, the convergence rates respect the expected order. However, notice that the convergence rates with source encoding are now comparable for different optimization methods, and are always lower than the convergence with deterministic methods. This points in the same direction as the theoretic estimates of Bottou & Bousquet (2011), suggesting that the Hessian information does not significantly improve the asymptotic convergence rate. Note that for stochastic gradient we have not obtained the theoretical convergence rate. The difference may come from the non linearity of our problem, or the different conditions imposed on the step length (Robbins & Monro 1951).

The computational costs for *l*-BFGS shown in Fig. 2(b) indicate the region where a speed-up is possible. Although the cost of stochastic methods is less per iteration, the difference in convergence rates creates an intersection between the two curves thus bounding the region of potential computational gain. We should suspect that the higher the convergence rate of the deterministic method, the smaller the room for computational gain because the convergence rate of stochastic methods does not improve proportionally, as shown in Fig. 3(b).

The result in Fig. 2(b), indicates that the potential gain in computational cost (speed up) is related to the iteration number where the optimization is terminated. If the inversion is stopped somewhere in the region highlighted by the box in Fig. 2(b), the stochastic methods will solve less forward problems than deterministic ones to attain a desired value of misfit, and thus a speed-up will be attained. Outside the bounded region, deterministic methods are more efficient. Therefore, in the numerical experiments that follow, we predefine stopping criteria that allow to attain computational gain, while leading to a sufficient accuracy of the subsurface model.

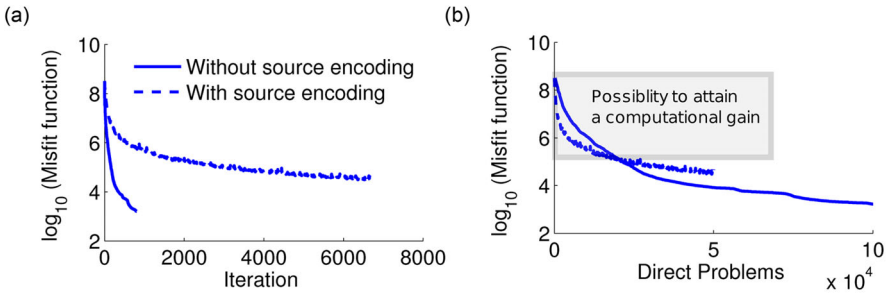

Figure 2. BP case study without noise: comparison between the convergence rates (a) and the computational cost (b) of stochastic (dotted lines) versus deterministic (solid lines) algorithms. The optimization methods are *l*-BFGS and *l*-BFGS$_r$. FWI is performed for the first frequency group. The only stopping criterion is the maximum number of forward problems, which is set to 10^5. This criterion was not reached by the stochastic approach because of line search failure. (a) The convergence rate is higher for the deterministic methods than for the stochastic ones, as shown in Fig. 3. (b) The computational cost (measured by the number of forward problems) for the stochastic methods is lower at the beginning of the inversion. However, the deterministic inversion will eventually catch up with the stochastic inversion because deterministic methods have higher convergence rates.

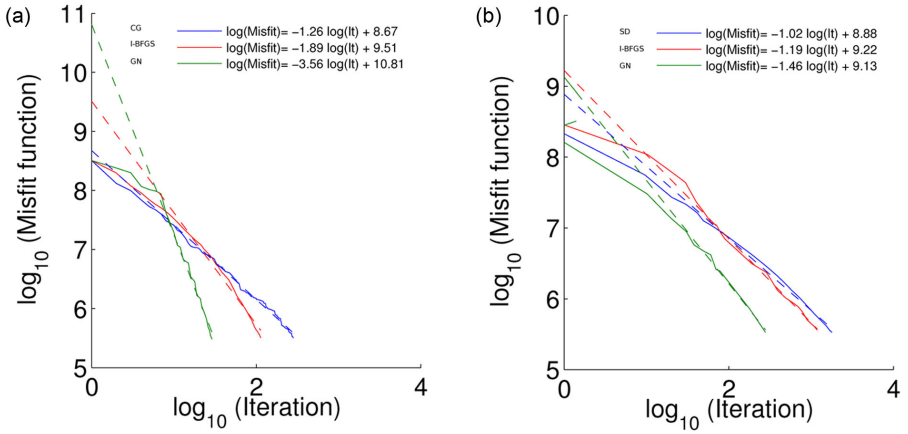

Figure 3. BP case study without noise: reduction of the misfit function as a function of the iteration on a log–log plot to extract the information on convergence rate. Blue: CG, Red: *l*-BFGS, green: GN. (a) Without source encoding. (b) With source encoding. The convergence rates of deterministic methods are higher than those of stochastic methods.

We now proceed to compare the computational gains (speed-ups) using source encoding with various optimization algorithms. We find that, even though with source encoding the convergence rates are similar for different optimization algorithms (and lower than deterministic methods), in the early part of the inversion, stochastic second-order optimization algorithms require lower computational costs and thus outperform the standard stochastic gradient descent, as was shown numerically by Schraudolph *et al.* (2007) for the convex case.

4.3.2 Synthetic data without noise

We assemble the sources with random coefficients following a Gaussian distribution (21), and create three supersources ($K = 3$). The role of the number of supersources is explained in more detail in Appendix C. We use the same values for the tuning parameters (β, λ_x and λ_z) as those used with all the sources processed independently and *l*-BFGS is restarted every five iterations. The values for the free parameters in the optimization are summarized in Table 2. Unless otherwise mentioned, the four optimization methods applied with this experimental set-up attain the same final value of the misfit function, whether source encoding is used or not.

Convergence rate. The convergence rates for the first frequency group were analysed previously in Figs 3(a) and (b). In Figs 4(a)–(c), we represent this same information, and complement with Figs 4(b)–(d) where the convergence for the second frequency group is illustrated, with and without source encoding.

In addition to what was mentioned in Section 4.3.1, we see that among the truncated Newton methods, GN outperforms FN probably because the GN approximation leads to a positive definite Hessian, while the additional second-order term in the FN Hessian approximation may not be. This may cause an earlier termination of the linear CG iterations during the resolution of the Newton system as the positivity condition is violated (Métivier *et al.* 2013). The difference in convergence rates between the different optimization methods is clearly less pronounced when using source encoding, in particular for the inversion of the second frequency group. We conclude from this statement that the Newton methods are more penalized than the SD method by the source encoding method and this suggests that the action of the Hessian in the Newton methods is hampered by the cross-talk noise injected in the gradient of the misfit function.

The accuracy of the final velocity models obtained for all methods is similar whether source encoding is used or not. Therefore, we only show the final FWI model obtained with GN and its error when all the sources are processed independently and when source encoding is used (Fig. 5). The small body in the sedimentary cover $(x, z) = (5.5$ km, 1 km) is well reconstructed as well as the contours of the salt body. The sub-salt structures are well identified in particular the small body at $(x, z) = (1.7$ km, 2.2 km). The focusing of the deep reflectors could have been improved by allowing more iterations (i.e. by using a smaller value of ϵ_1). Qualitative comparison between Figs 5(a) and (b) and Figs 5(c) and (d) shows that the accuracy of the final FWI models inferred from the GN optimization method was not significantly hampered by the source encoding.

Computational efficiency and speed-up. Now that we checked that all the optimization methods converge to the same solution, we can compare on the one hand the computational efficiency of each optimization method and on the other hand the speed-up provided by the source encoding method for each of these methods. We compare the reduction of the misfit function as a function of the number of forward problems for the two frequency groups when all the sources are processed independently and when source encoding is used in Fig. 6. This comparison is shown separately for each optimization method for sake of clarity. From top to bottom in Fig. 6, comparison between the number of forward problems performed by each optimization method to reach the desired value of the misfit function informs us about the computational efficiency of each optimization method in an absolute sense. This comparison can be performed when all the sources are processed independently (Fig. 6, solid lines) or when source encoding is used (Fig. 6, dash lines).

Table 2. BP-2004 case study. Tuning parameters for optimization algorithms. The same parameters are used for all of the optimization methods. β: damping factor of the Hessian pre-conditioner. λ_x, λ_z: weighting factors applied to the Tikhonov regularization in the misfit function. ϵ_1: relative reduction of the misfit function used as a stopping criteria. The number of memory models in *l*-BFGS is 5. The maximum number of inner iterations (N_{CG}) in truncated Newton methods is 30. The number of supersource K equals to 3. The maximum number of forward problems is 10^5. For this case study, the same tuning is used when source encoding is used or not.

Tuning parameters for the inverse problem			
β	$\lambda_x = \lambda_z$	ϵ_1 (first frequency group)	ϵ_1 (second frequency group)
10^{-2}	10^{-8}	10^{-3}	10^{-2}

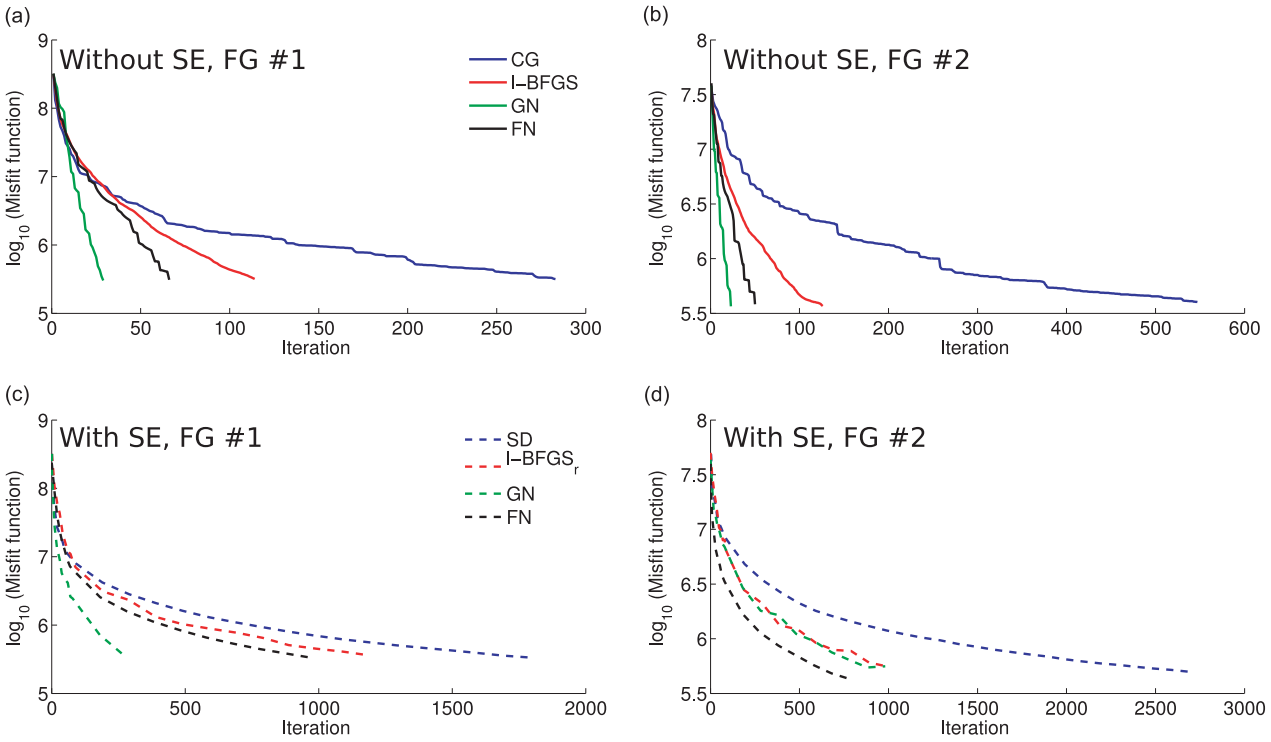

Figure 4. BP case study without noise: convergence rate. Reduction of the misfit function as a function of the iteration number without (a–b) and with (c–d) source encoding. (a, c) First frequency group (FG #1). (b, d) Second frequency group (FG #2). The curves are shown for the four optimization methods. Blue lines: *nl*-CG/SD; red lines: *l*-BFGS($_r$); green lines: GN; black lines: FN.

Second, in each panel of Fig. 6, the ratio between the number of forward problems with and without source encoding (eq. 22) for a given value of the misfit function value represents the computational gain (speed-up) provided by the source encoding method for one optimization method (Fig. 7).

Although we have shown that the truncated Newton methods have the highest convergence rate, *l*-BFGS has the lowest computational cost, followed closely by GN, whether source encoding is used or not. This results because *l*-BFGS performs a more limited number of forward problem per non-linear iteration than truncated Newton methods. The *nl*-CG/SD method has the highest computational cost due to its poor convergence rate compared to the quasi-Newton and truncated Newton methods. Although *nl*-CG/SD has the highest computational cost, it shows the best speed-up (Figs 7a and b and Table 3). This reflects the fact that, among all the optimization methods, it is the one whose convergence rate has been less affected by the cross-talk noise. This might result because SD is the only optimization method that does not account for the Hessian, whose action is affected by the cross-talk noise introduced by source encoding. Note that, for the realization shown in Fig. 6, the GN optimization methods does not show any speed-up for the second frequency group.

Independent of the optimization method, a general conclusion is that the speed-up decreases as the misfit function gets closer to a minimum, that is as the convergence rate of the optimization slows down to reach a plateau (Figs 2, 7a and b). Therefore, a trade-off must be found between the speed-up provided by source encoding and the quality of the final FWI model. We also observe that the convergence rates of different optimization methods tend to be more homogeneous with source encoding. We interpret this as the penalizing effect of the cross-talk noise (that can be interpreted as noise in the data) on the action of the Hessian approximation. Therefore, as the Hessian approximation becomes less accurate, the convergence rate of all optimization methods tends to be levelled down.

Similar conclusions derive when the inversion is performed with noisy data, as can be found in Appendix A. We observe that when noise is added to the data, the speed-up decreases more rapidly (Figs 7c and d). In addition, the performance of all the optimization methods tend to be levelled down as noise is added to the data and the action of the Hessian is damped. Even though *l*-BFGS is the fastest method with and without source encoding, it is not very robust when noise is added to the data.

In addition to the benefits regarding the computational cost, we illustrate in Appendix B that employing source encoding may find more satisfactory local minimum. We perform a numerical test where we degrade the initial model to that depicted in Fig. 1(c), the final FWI velocity model obtained when all the sources are processed independently is less accurate than the one inferred from the stochastic optimization. Therefore, we conclude that, with source encoding, we may not only reduce the computational cost but we may also steer the solution towards another local minimum thanks to a broader exploration of the model space. For the example presented in Appendix B, the local minimum attained is better than with deterministic methods. However, there is no guarantee that this will always be the case and that the solution with source encoding will always be a more adequate local minimum.

4.4 Real data example

To validate that source encoding techniques have a true interest in real data applications, we use a 2-D OBC data set from the Valhall

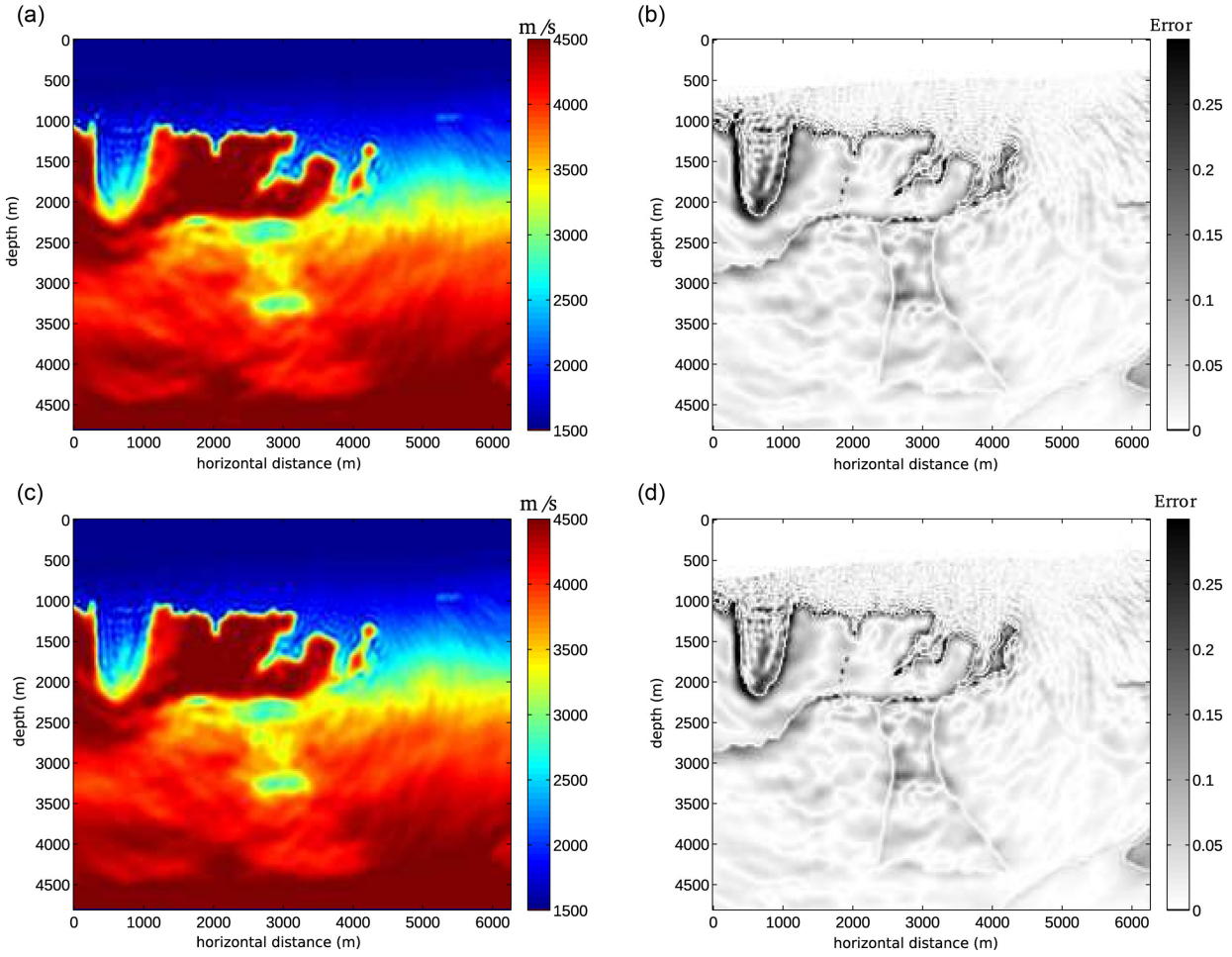

Figure 5. BP case study without noise. (a, c) Final FWI model for the GN optimization method without (a) and with (c) source encoding ($K = 3$). (b, d) Velocity model error (difference between the final FWI model and the true subsurface model).

oil field in the North Sea. This data set contains 320 sources that are located 5 m beneath the water level, with a spacing of 50 m, and are recorded by 210 hydrophone receivers on the sea bottom at 68 m below the water level, also with a 50 m spacing. The dimensions of the subsurface model are 16 km $\times$ 4 km. Several studies have already been conducted using this data set (Prieux *et al.* 2011, 2013; Gholami *et al.* 2013a). The subsurface model is mainly characterized by soft quaternary sediments below the sea level, low-velocity gas layers between 1.5 and 2.5 km in depth above the reservoir which delineates a sharp positive velocity contrast. These structures are highlighted in a reverse time migrated image computed in a background subsurface model that will be used as initial model for FWI in the following of this study (Fig. 8). Anisotropy, which is significant and can reach a maximum value of 15 per cent, is taken into account in the seismic modelling performed during FWI. The initial models for the vertical wave speed and the Thomsen parameters δ and ϵ were developed by reflection traveltime tomography (courtesy of BP) and are shown in Gholami *et al.* (2013a). The background density model is inferred from the initial vertical wave speed model by Gardner's law and the quality factor is fixed at a constant value of 200. We perform a monoparameter FWI for the vertical velocity v_{P0} keeping the Thomsen's parameters δ and ϵ, the density and the quality factor fixed. The relevance of the VTI parametrization (v_{P0}, δ, ϵ) for monoparameter FWI is discussed in Gholami *et al.* (2013b), Gholami *et al.* (2013a) and Operto *et al.* (2013). The source wavelet estimation is updated in each iteration (Pratt 1999).

Experimental set-up. We use four overlapping frequency groups, ranging from 3.5 Hz to 6 Hz: [3.5, 3.78, 4], [4, 4.3, 4.76], [4.76, 5, 5.25], [5.25, 5.6, 6] Hz. We did not see significant reduction in the misfit function and no improvement in the velocity models for higher frequencies, and the data is too noisy for inversion at lower frequencies. For each frequency group, the stopping criterion of non-linear iterations is controlled by the relative reduction of the misfit function ($\epsilon_1 = 0.7$ for each frequency group) with and without source encoding. When each source is processed independently, we also set the maximum number of non-linear iterations to 20. For the truncated Newton methods, we use $N_{\mathrm{CG}} = 3$ because the real data is considered to be noisy (Métivier *et al.* 2013). When source encoding is used, we perform the inversion with one supersource ($K = 1$). This choice is described in more detail in Appendix C. We use the spatial reciprocity of Green functions to process receivers as sources during FWI. Therefore, 210 sources are stacked to form a supersource. This number of sources is significantly higher than the one used during the BP experiment (62) and this difference must be taken into account in the speed-up analysis. The free parameters for the optimization are summarized in Table 4.

Convergence of FWI. We first compare the convergence of the different optimization methods without source encoding for the four

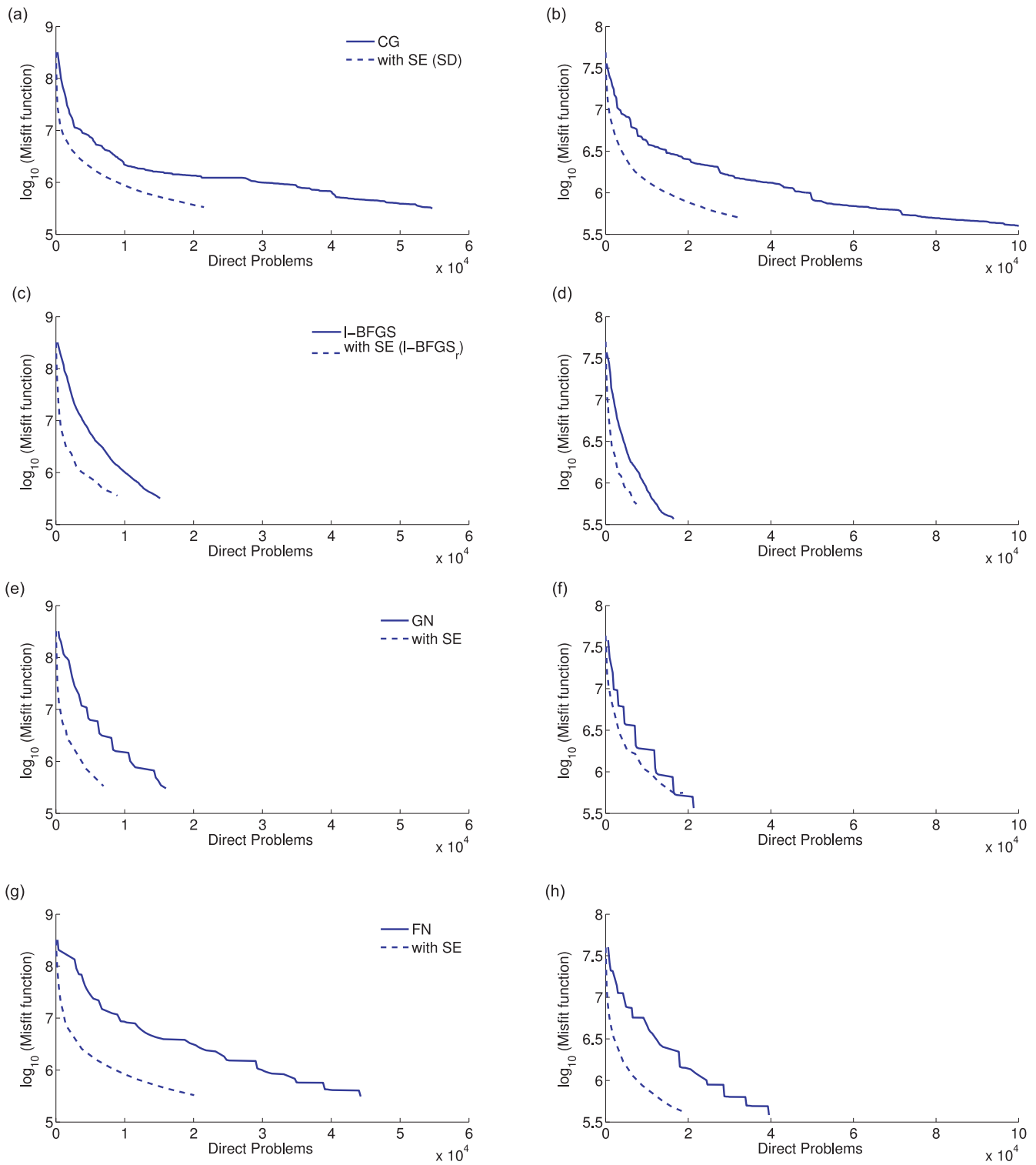

Figure 6. BP case study without noise. Assessment of the computational efficiency and computational savings (speed-up) provided by source encoding: reduction of the misfit function as a function of the forward problems, for the first (a, c, e, g) and second (b, d, f, h) frequency groups. (a and b) *nl*-CG optimization method. (c and d) *l*-BFGS optimization method. (e and f) GN optimization method. (g and h) FN optimization method. The computational gain is provided by the difference between the number of forward problems performed with (dash lines) and without (solid lines) source encoding for a given misfit function value.

frequency groups (Fig. 9a). For the first two frequency groups, the two truncated Newton methods converge to the same misfit function value, which suggests that the second-order term in the truncated FN is smaller than the regularization term and thus has no significant effect in the inversion. For the third and fourth frequency groups, however, the two methods follow different optimization paths and GN attains a lowest misfit function value for the fourth frequency group. For this particular choice of free parameters, *l*-BFGS does

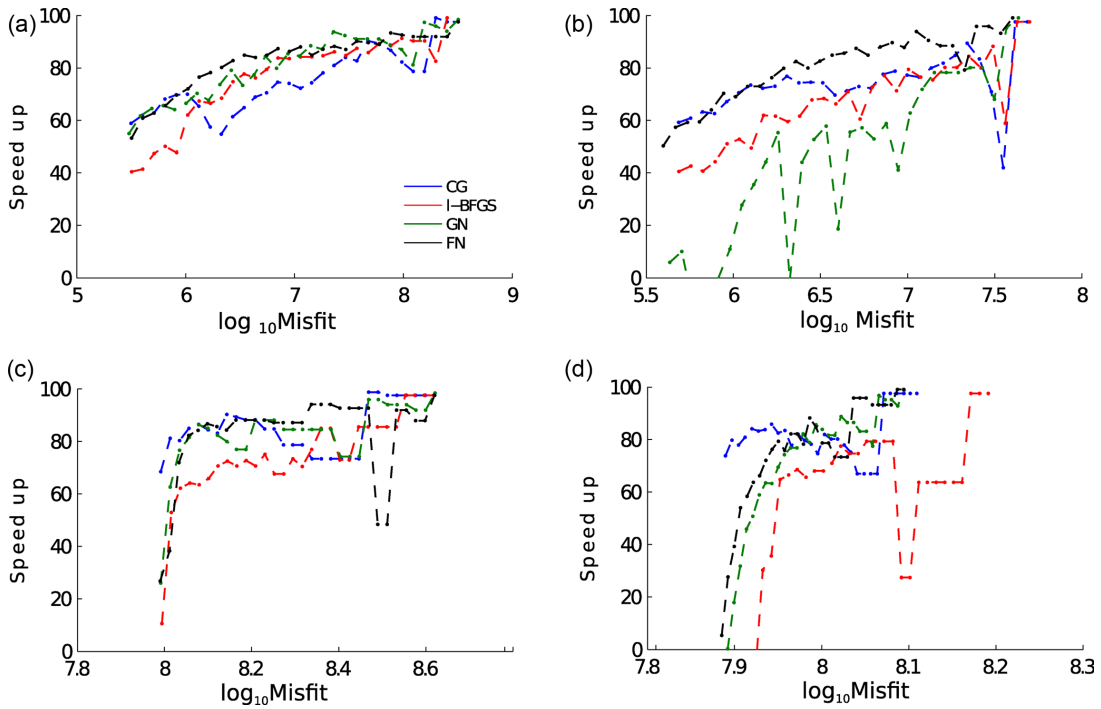

Figure 7. BP case study. Speed-up versus the average data misfit function for the two frequency groups for the case without noise (a–b) and with noise (c–d). Panels (a) and (b) summarize the results for the speed-up without noise for each optimization method shown in Fig. 6, for the first and second frequency group. Panels (c) and (d) summarize the results for the speed-up with noise for each optimization method shown in Fig. A3, for the first and second frequency group. Note how the speed-up drops abruptly in the case of noisy data, as the convergence rate slows down near the minimum of the misfit function. Blue lines: *nl*-CG/SD. Red lines: *l*-BFGS. Green lines: GN. Black lines: FN.

not decrease the misfit function as much as the other optimization methods. In particular, it fails to update the model during the second frequency group. The *nl*-CG method performs a misfit function reduction close to the one achieved by the truncated Newton methods during the inversion of the first two frequency groups, unlike for the third and fourth frequency groups for which the convergence rate of the *nl*-CG method is poorer and the minimum of the misfit function that was reached is higher. Overall for all frequency groups, *l*-BFGS performs the fewest number of iterations but with a poorer convergence level. Failed line search in *l*-BFGS causes the inversion to terminate early in some frequency groups. This can be improved by increasing the relative weight of regularization. However, since we want to perform a fair comparison among different optimization methods using the same parameters, we do not increase here the value of the hyperparameters. We conclude from these results that the truncated Newton methods clearly provide the most robust direction of descent relative to quasi-Newton and CG methods.

The final velocity models that were obtained with each optimization method at the end of the fourth frequency group are shown in Figs 10(a)–(d). Comparison between a sonic log at 9.5 km in distance and the corresponding profiles of each FWI model is shown in Figs 11(a)–(d). The FWI models obtained with the truncated Newton methods clearly provide the best trade-off between signal-to-noise ratio and resolution. For example, shallow artefacts near the end of the model inferred from the *nl*-CG method (Fig. 10a) are not present in the truncated Newton models (Figs 10c and d). Moreover, the deep reflector below the reservoir level shown in Fig. 8 is far less contrasted in the *l*-BFGS model (Figs 10b) than in the truncated-Newton models (Figs 10c and d), although the geometry of this reflector seems well reconstructed in the *l*-BFGS model. The weaker amplitudes of the velocity perturbations retrieved by *l*-BFGS may result from the more limited number of iterations performed by this optimization method.

Table 3. BP-2004 case study without noise. Comparison of the total number of forward problems solved in the two frequency groups with and without source encoding for each optimization algorithm. The speed-up S (per cent) represents the computational gain when the inversion is performed with source encoding.

Optimization	*DP* without *SE*	*DP* with *SE*	Total S (per cent)
CG	154 752	56 058	64
l-BFGS	31 620	17 256	45
GN	37 324	27 720	27
FN	83 824	40 338	52

When source encoding is used, a similar hierarchy among the different optimization methods is shown with superior results achieved by the truncated Newton methods both in terms of convergence rate and convergence level (Fig. 9b). The quasi-Newton *l*-BFGS$_r$ method failed to converge during a frequency group as shown in Fig. 9(b). Note also that using the SD method, the misfit function increases in the last frequency group. This explains some artefacts in the variance of the final velocity models that will be later discussed. The increase of the misfit function in the last frequency groups for SD and *l*-BFGS suggests that a higher regularization weight and/or a higher number of supersources is needed compared to the one required by Newton methods. The initial value of the misfit function in Figs 9(a) and (b) is the same with and without encoding, for all optimization methods ($\approx 2 \times 10^6$). However, for the second frequency group and those that follow, the initial misfit values are no longer the same with and without encoding, and with all optimization methods. This is due to the fact that the final models and

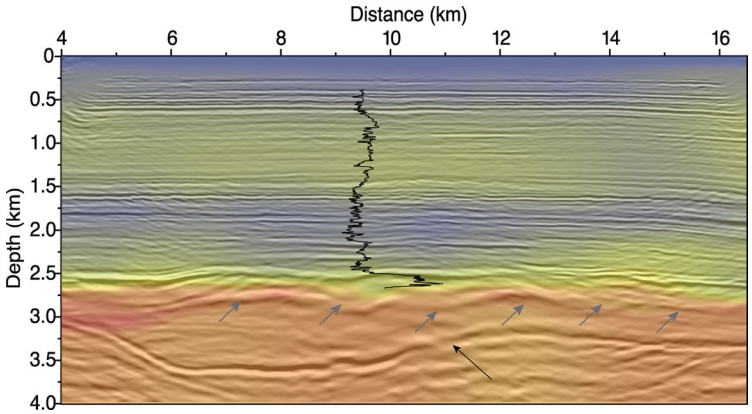

Figure 8. Valhall case study. Reverse time migrated image computed in the initial vertical velocity model, which is superimposed with a transparency. A sonic log located at 9.5 km in distance is also superimposed on the migrated image. The grey arrows delineate the base of the reservoir. The black arrow points a location in depth where the image of a deep reflector lacks continuity.

Table 4. Valhall case study. Tuning parameters for optimization. Maximum number of iterations for the Newton methods is limited to $N_{\mathrm{CG}} = 3$. The number of memory models in *l*-BFGS is 5. The maximum number of FWI iterations is limited to 20. The number of supersources is $K = 1$. See Table 2 for the nomenclature.

	β	λ_x	λ_z	ϵ_1	$N_{\mathrm{nlit}}^{\mathrm{max}}$
No source encoding	10^{-2}	10^{-3}	2.5×10^{-4}	0.7	20
With source encoding	10^{-2}	10^{-2}	2.5×10^{-4}	0.7	–

misfits in each frequency group are not the same in this real data application.

The final velocity models obtained with source encoding are shown in Figs 10(e)–(h). The FWI model obtained with *nl*-CG shows significant artefacts along the shallow reflector at around 0.6 km in depth. The deep reflector is also reconstructed with weak amplitudes in the *nl*-SD model (Fig. 10e). The footprint of the cross-talk noise is clearly visible in the shallow part of the *l*-BFGS$_r$ model, while the deep reflector is better reconstructed relative to the one obtained without source encoding (compare Fig. 10b and f). The truncated Newton methods show a more robust behaviour with respect to source encoding in the sense that the velocity models inferred from these methods with and without source encoding are quite consistent [compare panels (c–d) and (g–h) in Fig. 10].

Computational efficiency and speed-up. The misfit function as a function of the number of forward problems are shown for the four optimization methods when no source encoding is used in Fig. 12. The truncated Newton methods are around two times more expensive than *nl*-CG when all the sources are processed independently. When source encoding is applied, the misfit function is shown in Fig. 13, for all optimization methods. The red bars denoting the sample variance of the misfit function over the 50 realizations, show that it is large for SD and *l*-BFGS. As we established earlier, this suggests that a higher regularization weight or more supersources are needed. We chose one random realization to compare the computational costs summarized in Table 5. Since the cost of truncated Newton methods and quasi Newton methods is similar when source encoding is used, and since it was very different with the sources processed independently, this lead to a higher speed-up of the truncated Newton methods relative to the *nl*-CG/SD method (around 96 per cent against 92 per cent). The speed-up is quite significant and represents almost one order of magnitude in terms of computational saving. However, recall that this speed-up is highly sensitive to the choice of the stopping criterion of iteration ϵ_1 (Fig. 2). If a value of 60 per cent instead of 70 per cent would have been chosen for ϵ_1, almost no speed-up would have been shown because we would have let the inversion to perform many iterations without significant decrease of the misfit function whether source encoding is used or not.

Quality control: statistical stability. The convergence curves plotted as a function of the number of forward problems for the four frequency groups confirm that the truncated Newton methods are

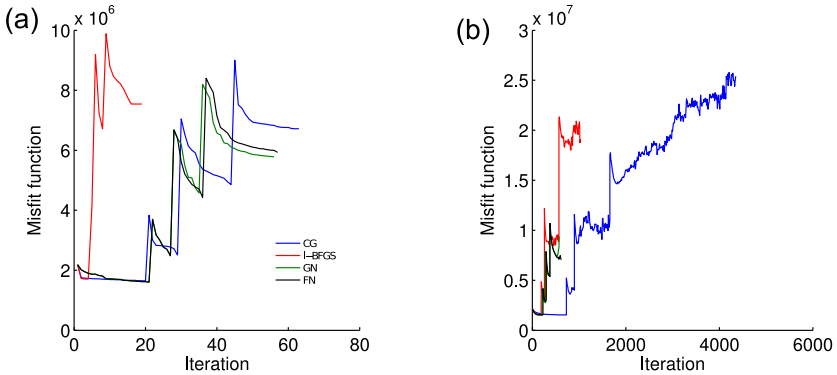

Figure 9. Valhall case study. Convergence of FWI. (a) Misfit function versus iteration number for the four frequency groups without SE. (b) Average over 50 independent realizations of the misfit function versus iteration number, for the four frequency groups with SE. Blue line: *nl*-CG optimization method. Red line: *l*-BFGS method. Green line: GN optimization method. Black line: FN optimization method.

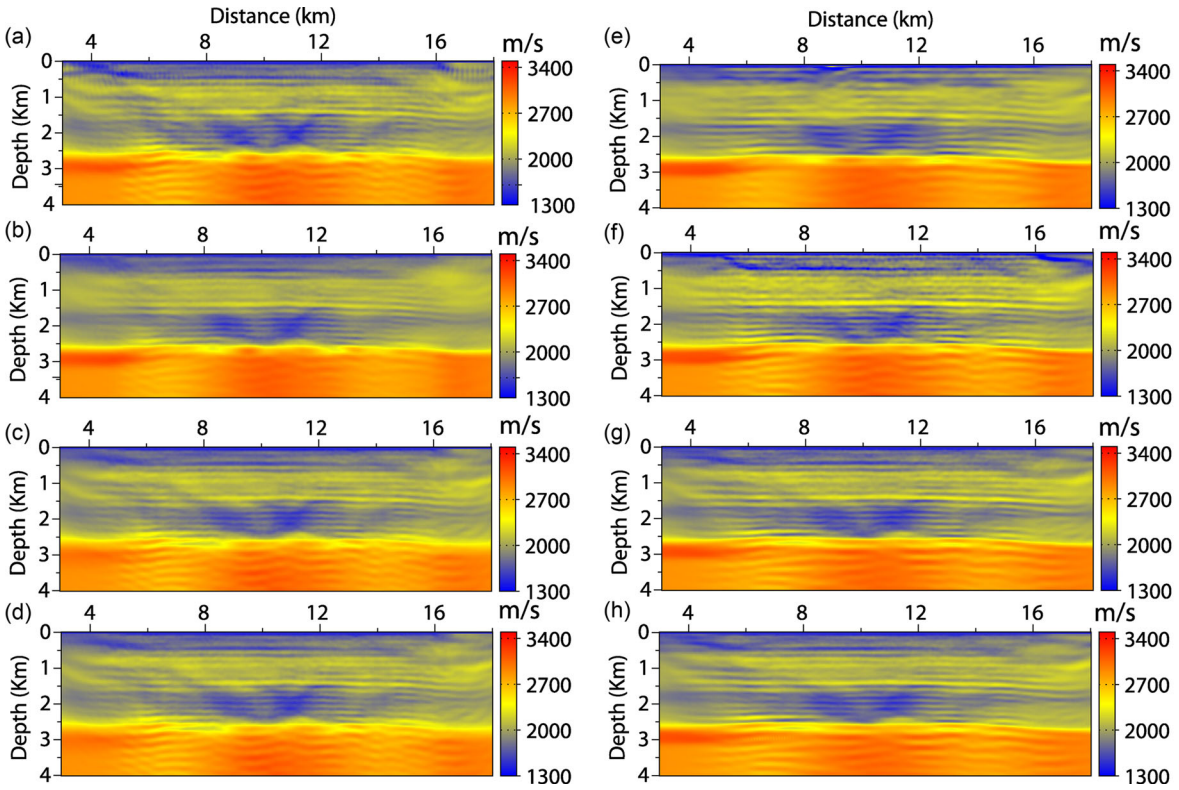

Figure 10. Valhall case study. Final FWI velocity models obtained without (a–d) and with (e–h) source encoding. (a, e) *nl*-CG/SD optimization method. (b, f) *l*-BFGS/BFGS$_r$ optimization method. (c, g) GN optimization method. (d, h) FN optimization method.

the most robust relative to SD and *l*-BFGS$_r$ optimization methods (Fig. 13). Accordingly, the velocity models built by the truncated Newton models have the smaller variance (Fig. 14). The higher values are shown in the shallow part near the ends of the receiver cable where inversion has more degrees of freedom to perturb the subsurface model (Figs 14c–d). Significant values of the variance are also shown at the reservoir level at around 2.5 km in depth near the ends of the reflector segment that was imaged by FWI, still in relation with a more limited illumination. It is worth noting that the variance is almost zero in the bottom right and bottom left of the model where a strong deficit of illumination exists. This is consistent with the fact that the (damped) regularized Hessian prevents the updating of the subsurface model where the sensitivity of the inversion to the information contained in the data is below some given threshold. The variance in the SD model reaches the highest values in the shallow part, which is consistent with the shallow artefacts highlighted in Fig. 10(e). The variance of the *l*-BFGS$_r$ realizations reflects the imprint of the cross-talk noise in the shallow part of the model already highlighted in Fig. 10(f) as well as shallow artefacts near the ends of the cable (Fig. 10b).

Quality control: reverse time migration. We apply anisotropic reverse time migration to the Valhall data using the FWI models inferred from the FN optimization method with and without source encoding as background models (Fig. 15). The experimental set-up to perform reverse time migration is outlined in Prieux *et al.* (2011). The two migrated images (Fig. 15) can be compared with the one computed in the initial model (Fig. 8). We superimposed in transparency on each migrated image the background velocity model that was used to perform migration to check the consistency between the reflectors mapped by migration and the velocity variations built by FWI. The accuracy of the migrated images can be assessed by the flatness of the reflectors in the common image gathers (CIGs; Fig. 16). As it is highlighted in Prieux *et al.* (2011), it is quite challenging to improve the migrated images computed in the background model built by reflection traveltime tomography, because reflection traveltime tomography is designed to optimally focus reflection energy. However, FWI has improved the imaging of the reflectors in the shallow part (down to 600 m in depth) where reflection traveltime tomography can encounter difficulty to pick traveltimes. This is highlighted by an improved focusing of the shallow reflectors between 0.4 and 0.6 km in depth in Fig. 15. The highest resolution of the FWI velocity models relative to the traveltime tomography model is also highlighted by the closer correlation between the reflectors mapped by migration and the sharper velocity variations imaged by FWI. This improvement was also shown in some close-up of the CIGs centred on the shallow reflectors in Prieux *et al.* (2011, their fig. 10). Aside the shallow reflectors, other local improvements of the migrated images inferred from the FWI background models are highlighted in Fig. 15, grey and black arrows) and in the CIGs (Fig. 16, shaded area). The most obvious one is that the base of the reservoir is more continuous in the FWI-based migrated images than in the tomography-based migrated image (compare Figs 8 and 15, grey arrows). The deep reflector below the reservoir is also more continuous in particular where this reflector has more significant dip (compare Figs 8 and 15, black arrows). We do not see any imprint of the cross-talk noise in the RTM image computed in the FWI model obtained with the source encoding method (Fig. 15b). This RTM image generally shows more continuous and focused reflectors, in particular at the reservoir level, than the RTM image computed in the FWI model obtained without source encoding (compare the two panels in Fig. 15). This probably results because we use a stronger horizontal regularization weight

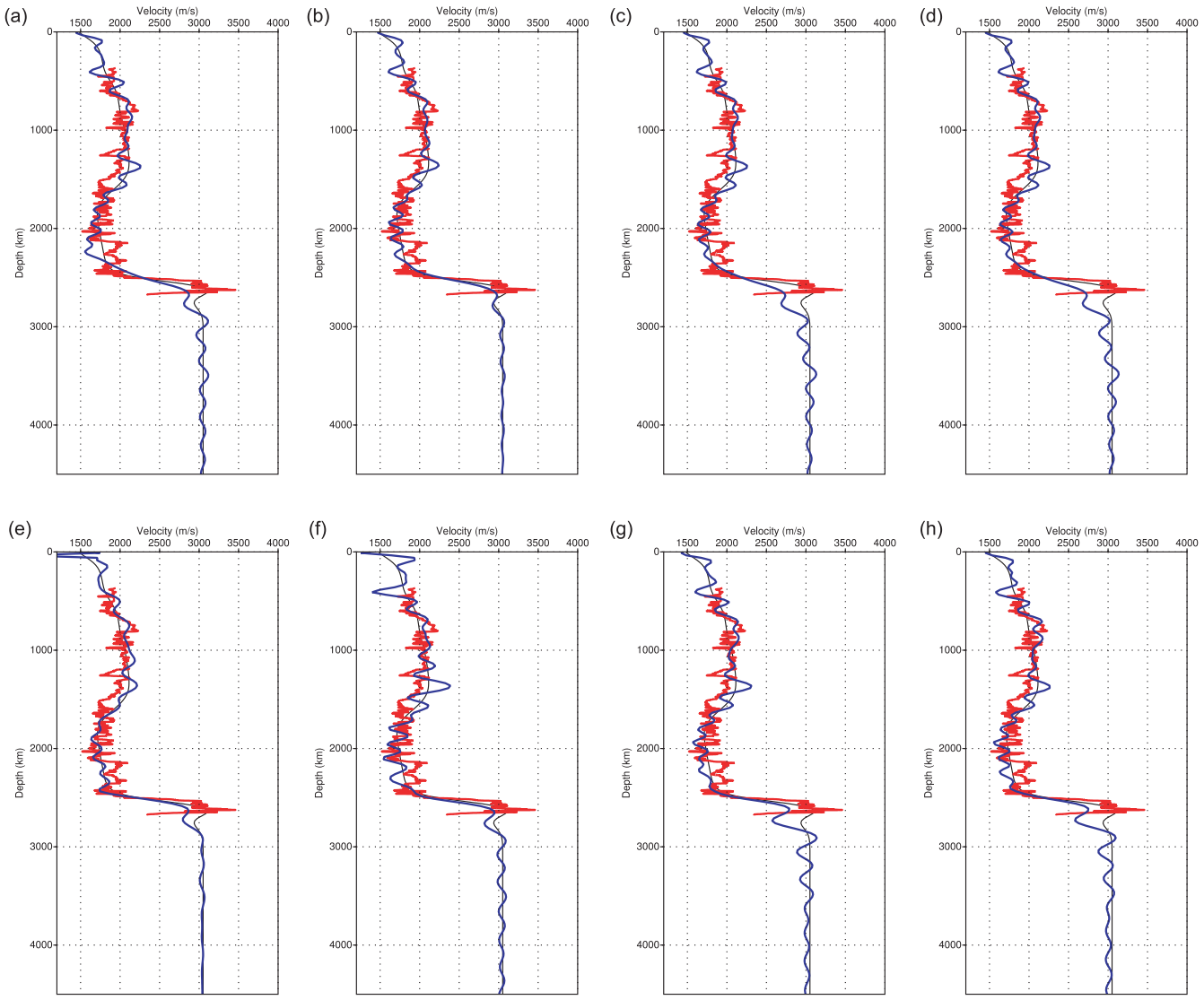

Figure 11. Valhall case study. (a–d) Logs of the final velocity models (blue lines) obtained without source encoding at a horizontal distance $x = 9.5$ km for (e–h) Logs of the final velocity models (blue lines) when source encoding is used during FWI. (a, e) *nl*-CG/SD, (b, f) *l*-BFGS/BFGS$_r$, (c, g) GN and (d, h) and FN. The sonic log is plotted with a red lines and the log of the initial model is plotted with a black line.

during FWI when source encoding is used. This statement reflects the trade-off between resolution and error and how the errors that are accumulated over the non-linear iterations of FWI models are mapped in the migrated image.

5 CONCLUSION

We have applied 2-D efficient frequency-domain FWI on synthetic and real data when random source encoding is interfaced with different optimization methods in order to determine the best strategy to perform a fast FWI in a robust manner. We numerically asses how second order methods perform with stochastic optimization. A stopping criteria of iterations and the use of the same values for the free parameters in the inversion allows for a fair assessment of the computational efficiency of each optimization method and the speed-up provided by the source encoding method.

Without using source encoding we found that, in an ideal noise-free data scenario with frequency groups that determine an approximately convex misfit function, truncated Newton methods have the highest convergence rate, and thus require less iterations to attain a desired relative reduction of the misfit function. However, truncated Newton methods remain more computationally expensive than the quasi-Newton method *l*-BFGS as truncated Newton methods require additional forward problems per non-linear iteration. As noise is added to the synthetic data and more aggressive regularization is used, the action of the Hessian becomes less effective and the convergence rate of the Newton-based methods is thus degraded. This contributes to level down the convergence rate of Newton-based methods relative to SD method. All optimization methods when combined with source encoding were shown to be statistically stable, meaning that, when several independent inversions are carried out, each inversion converges to approximately the same model. However, the truncated Newton methods have a more robust behaviour showing a smaller variance in the final solution.

The speed up (gain in computational cost) provided by source encoding is measured by the ratio of the number of forward problems that have to be solved with and without source encoding. During

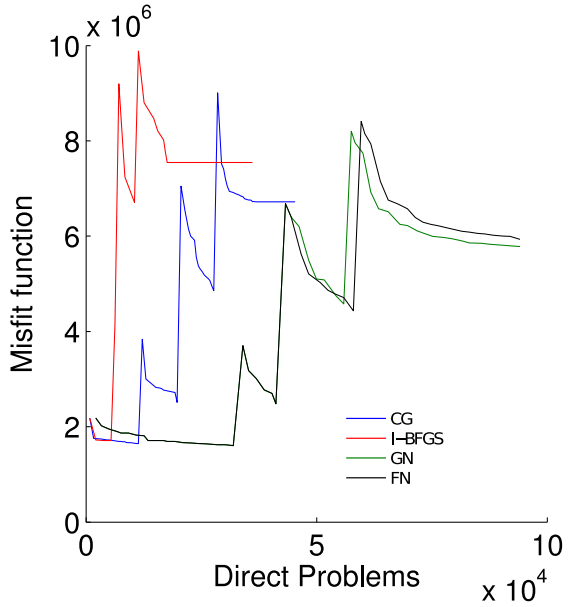

Figure 12. Valhall case study. Computational cost. Misfit function versus number of forward problems for each optimization method and for the four frequency groups when all the sources are processed independently. Blue lines: SD. Red lines: *l*-BFGS. Green lines: GN. Black lines: FN.

Table 5. Valhall field data. Statistics of the FWI with and without source encoding. See Table 3 for the nomenclature.

Optimization algo.	*DP* without *SE*	*DP* with *SE*	*S* (per cent)
CG	45 360	3464	92 per cent
l-BFGS	36 120	1714	95 per cent
GN	94 080	3072	96 per cent
FN	94 080	3572	96 per cent

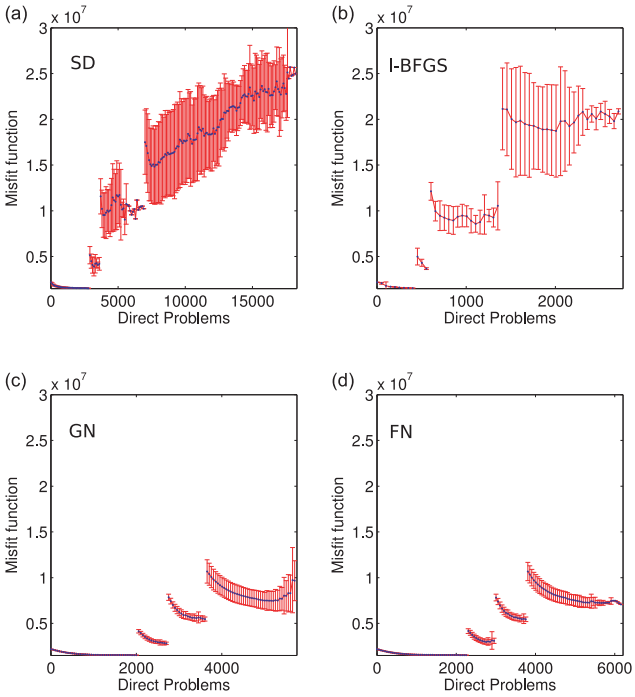

Figure 13. Valhall case study. FWI with source encoding. Mean value over 50 realizations of the reduction of the misfit function for the four frequency groups versus the number of direct problems. The variance of the misfit value is delimited with the red bars. (a) SD. (b) *l*-BFGS$_r$, (c) GN, (d) FN. Random variables follow a normal distribution.

the early iterations of the inversion, the misfit function with source encoding decreases sufficiently fast to make the speed-up large. As the iterations proceed and the convergence slows down, stochastic methods require more iterations than deterministic methods to decrease the error by the same amount and the speed-up decreases accordingly. Therefore, we showed that the speed-up strongly depends on the value of the relative reduction of the misfit function for which iterations are stopped. To that end a suitable stopping criterion of iterations should be designed such that the best trade-off between computational efficiency and quality of the subsurface model is found. For the stopping criteria we used in the Valhall real data set application, we obtained speed-ups of around 90 per cent.

Besides the computational gain that can be attained, we illustrate in Appendix B that source encoding techniques may provide advantages when minimizing non-convex misfit functions. Indeed, stochastic optimization methods allow for the exploration of regions in the model space that are never accessible in the deterministic case, because stochastic approaches allow the misfit function to increase. We present a numerical test where we inverted nine frequencies in a single frequency group to render the misfit function non-convex, and we found that the final model is more accurate when we use random source encoding relative to the case where we use all the sources independently. There is, however, always the possibility that the final solution found with source encoding provides a poorer quality.

While all of the optimization methods generate subsurface models of similar accuracy for the synthetic example, application on real data from the Valhall field shows that the truncated Newton methods attain a lower misfit function value than the other optimization methods, hence suggesting a more robust behaviour to noise and other source of errors such as incomplete wave physics. A speed-up of nearly one order of magnitude was reached for the selected stopping criterion of iterations. The accuracy of the subsurface models that was achieved for this stopping criterion of iteration was validated against published previous works, a sonic log and reverse time migration.

To further improve the convergence rates, other hybrid strategies may be implemented. For example, it seems feasible to use a few number of supersources until the misfit function starts reaching a plateau. At this point, one could use all the sources independently and switch to a deterministic optimization problem, converging at a higher rate.

Currently, it is required to store in memory or on disk the direct and backpropagated wavefields for each source in the truncated Newton methods to build the sources of the adjoint-state equations during the computation of the Hessian vector product. This may not be feasible or too computationally expensive on a 3-D perspective, in particular if the inversion is performed in the time domain. However, this may be once more viable from an implementation point of view when source encoding is used with a few number of supersources.

3-D viscoelastic FWI is one of the main challenge of seismic imaging for the next decade. Taking into account the Hessian in multiparameter FWI is crucial to manage the cross-talk between parameters of different nature. Owing that elastic seismic modelling is two to three orders of magnitude more expensive than acoustic modelling, the combination of random source encoding with truncated Newton methods should be of particular interest to

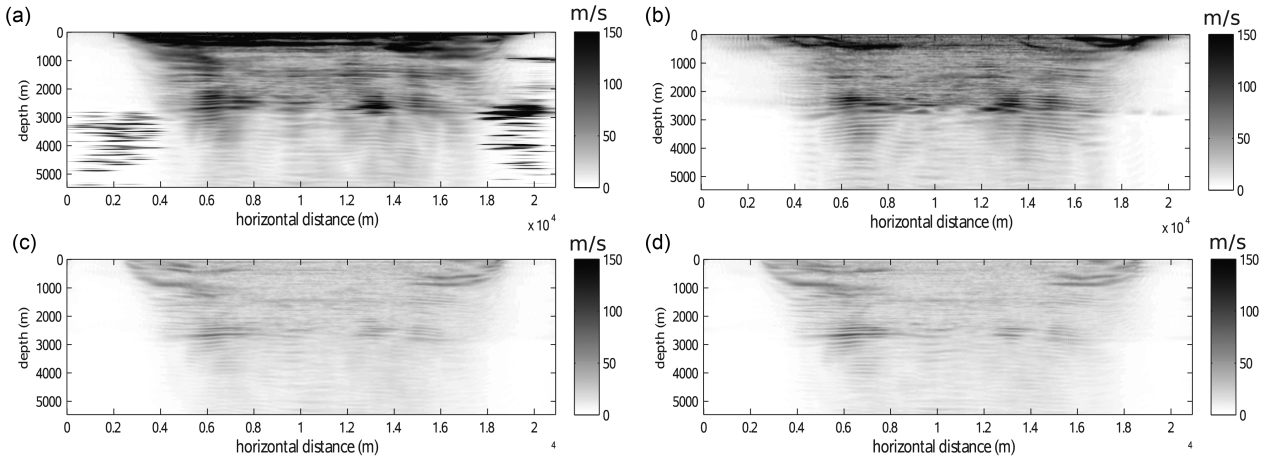

Figure 14. Valhall case study. Standard deviation of the final velocity model for 50 realizations of FWI with source encoding. (a) SD. (b) *l*-BFGS$_r$, (c) GN, (d) FN. Random variables follow a normal distribution.

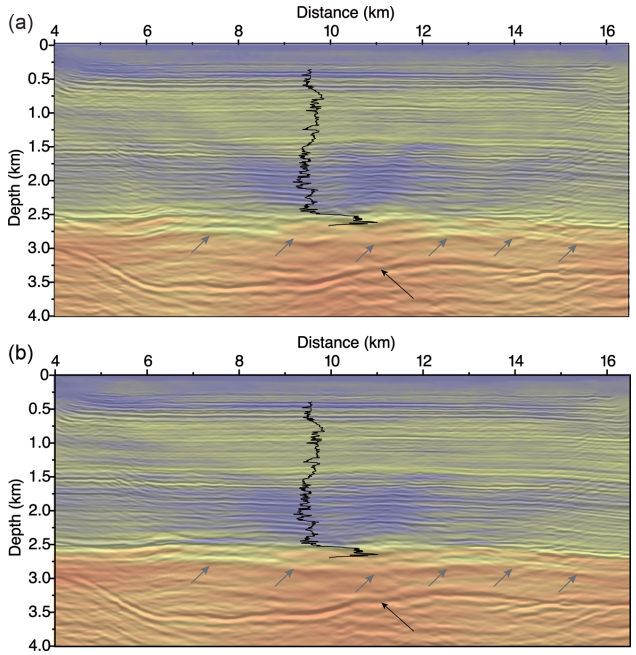

Figure 15. Valhall case study. Reverse time migrated image computed in the FWI model inferred from the FN optimization method without (a) and with (b) source encoding. The vertical velocity model within which the reverse time migrated image is computed is displayed with a transparency allowing one to check the consistency between the background velocities built by FWI and the reflectors mapped by the migration. The sonic log as well as the black and grey arrows shown in Fig. 8 are superimposed.

manage both the computational burden and the ill-posedness of 3-D viscoelastic FWI.

ACKNOWLEDGEMENTS

This study was partly funded by the petroleum SEISCOPE consortium (http://seiscope.oca.eu,http://seiscope2.osug.fr) and the Université de Nice Sophia Antipolis. The linear systems were solved with the MUMPS package, available on http://graal.ens-lyon.fr/MUMPS/index.html and http://mumps.enseeiht.fr. This study was granted access to the high-performance computing facilities of the SIGAMM (Observatoire de la Côte d'Azur) and we gratefully acknowledge these facilities and the support of their staff. We thank BP Norge AS and their Valhall partner Hess Norge AS, for allowing access to the Valhall data set as well as the well-log velocities. We also thank the anonymous reviewer and the associate editor for comments and suggestions to improve the paper.

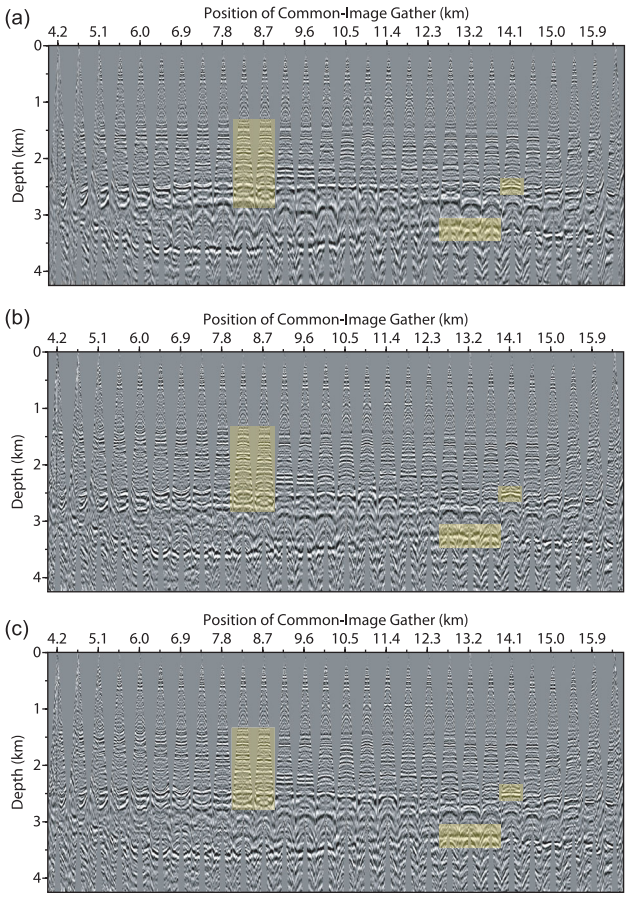

Figure 16. Valhall case study. Common image gathers in the offset-depth domain computed by reverse time migration performed in the initial model (a) and in the final FWI models inferred from the FN optimization method without (b) and with (c) source encoding. Reflectors which are flatter in the CIGs computed in the FWI model obtained with source encoding are highlighted.

REFERENCES

Baumstein, A., Ross, W. & Lee, S., 2011. Simultaneous source elastic inversion of surface waves, in *Proceedings of the 73rd EAGE Conference & Exhibition,* p. C040, European Association of Geoscientists and Engineers, Expanded Abstracts.

Ben Hadj Ali, H., Operto, S. & Virieux, J., 2011. An efficient frequency-domain full waveform inversion method using simultaneous encoded sources, *Geophysics,* **76**(4), R109–R124.

Bérenger, J.-P., 1994. A perfectly matched layer for absorption of electromagnetic waves, *J. Comput. Phys.,* **114,** 185–200.

Bottou, L., 1991. Stochastic gradient learning in neural networks, in *Proceedings of Neuro-Nîmes 91,* EC2, Nimes, France.

Bottou, L. & Bousquet, O., 2011. The tradeoffs of large-scale learning, in *Optimization for Machine Learning,* pp. 351, eds Sra, S., Nowizin, S. & Wright, S.J., MIT Press.

Bottou, L. & Le Cun, Y., 2005. On-line learning for very large data sets, *Appl. Stoch. Models Business Ind.,* **21**(2), 137–151.

Boyd, S. & Vandenberghe, L., 2009. *Convex Optimization,* Cambridge Univ. Press.

Brossier, R., Operto, S. & Virieux, J., 2009. Seismic imaging of complex onshore structures by 2D elastic frequency-domain full-waveform inversion, *Geophysics,* **74**(6), WCC105–WCC118.

Bunks, C., Salek, F.M., Zaleski, S. & Chavent, G., 1995. Multiscale seismic waveform inversion, *Geophysics,* **60**(5), 1457–1473.

Byrd, R.H., Lu, P. & Nocedal, J., 1995. A limited memory algorithm for bound constrained optimization, *SIAM J. Scient. Stat. Comput.,* **16,** 1190–1208.

Byrd, R.H., Chin, G.M., Neveitt, W. & Nocedal, J., 2011. On the use of stochastic hessian information in optimization methods for machine learning, *SIAM J. Opt.,* **21**(3), 977–995.

Chavent, G., 2009. *Nonlinear Least Squares for Inverse Problems,* Springer.

Choi Alkhalifah, T., 2012. Multi-source waveform inversion of marine streamer data using the normalized wavefield, in *Proceedings of the EAGE,* Expanded Abstracts.

Gao, F., Atle, A. & Williamson, P., 2010. Full waveform inversion using deterministic source encoding, *SEG Tech. Prog. Expanded Abstracts,* **29**(1), 1013–1017.

Gholami, Y., Brossier, R., Operto, S., Prieux, V., Ribodetti, A. & Virieux, J., 2013a. Which parametrization is suitable for acoustic VTI full waveform inversion? Part 2: application to Valhall, *Geophysics,* **78**(2), R107–R124.

Gholami, Y., Brossier, R., Operto, S., Ribodetti, A. & Virieux, J., 2013b. Which parametrization is suitable for acoustic VTI full waveform inversion? Part 1: sensitivity and trade-off analysis, *Geophysics,* **78**(2), R81–R105.

Habashy, T.M., Abubakar, A., Pan, G. & Belani, A., 2011. Source-receiver compression scheme for full-waveform seismic inversion, *Geophysics,* **76**(4), R95–R108.

Haber, E., Chung, M. & Herrmann, F., 2012. An effective method for parameter estimation with PDE constraints with multiple right-hand sides, *SIAM J. Opt.,* **22**(3), 739–757.

Hustedt, B., Operto, S. & Virieux, J., 2004. Mixed-grid and staggered-grid finite difference methods for frequency domain acoustic wave modelling, *Geophys. J. Int.,* **157,** 1269–1296.

Krebs, J., Anderson, J., Hinkley, D., Neelamani, R., Lee, S., Baumstein, A. & Lacasse, M.D., 2009. Fast full-wavefield seismic inversion using encoded sources, *Geophysics,* **74(6),** WCC105–WCC116.

Lailly, P., 1983. The seismic problem as a sequence of before-stack migrations, in *Proceedings of the Conference on Inverse Scattering: Theory and Applications,* ed. Bednar, J., SIAM.

Li, X., Aravkin, A., van Leeuwen, T., Burke, J. & Herrmann, F., 2012. Fast randomized full-waveform inversion with compressive sensing, *Geophysics,* **77(3),** A13–A17.

Lions, J.L., 1968. *Contrôle optimal de systèmes gouvernés par des équations aux dérivées partielles,* Dunod.

Métivier, L., Brossier, R., Virieux, J. & Operto, S., 2013. Full waveform inversion and the truncated Newton method, *SIAM J. Scient. Comput.,* **35**(2), B401–B437.

Métivier, L., Bretaudeau, F., Brossier, R., Operto, S. & Virieux, J., 2014. Full waveform inversion and the truncated Newton method: quantitative imaging of complex subsurface structures, *Geophys. Prospect.,* doi:10.1111/1365-2478.12136.

MUMPS-team 2011. *MUMPS—MUltifrontal Massively Parallel Solver users' guide - version 4.10.0 (May 10, 2011)*, ENSEEIHT-ENS Lyon, http://www.enseeiht.fr/apo/MUMPS/ or http://graal.ens-lyon.fr/MUMPS.

Nemirovsky, A.S. & Yudin, D.B., 1983. *Problem Complexity and Method Efficiency in Optimization,* Wiley.

Nocedal, J. & Wright, S.J., 2006. *Numerical Optimization,* 2nd edn, Springer.

Operto, S., Virieux, J., Ribodetti, A. & Anderson, J.E., 2009. Finite-difference frequency-domain modeling of visco-acoustic wave propagation in two-dimensional TTI media, *Geophysics,* **74**(5), T75–T95.

Operto, S., Gholami, Y., Prieux, V., Ribodetti, A., Brossier, R., Métivier, L. & Virieux, J., 2013. A guided tour of multiparameter full waveform inversion for multicomponent data: from theory to practice, *Leading Edge,* **32**(9), 1040–1054.

Plessix, R.E., 2006. A review of the adjoint-state method for computing the gradient of a functional with geophysical applications, *Geophys. J. Int.,* **167**(2), 495–503.

Plessix, R.E., 2009. Three-dimensional frequency-domain full-waveform inversion with an iterative solver, *Geophysics,* **74**(6), WCC53–WCC61.

Plessix, R.E. & Perkins, C., 2010. Full waveform inversion of a deep water ocean bottom seismometer dataset, *First Break,* **28,** 71–78.

Plessix, R.-E., Baeten, G., de Maag, J.W. & ten Kroode, F., 2012. Full waveform inversion and distance separated simultaneous sweeping: a study with a land seismic data set, *Geophys. Prospect.,* **60,** 733–747.

Pratt, R.G., 1999. Seismic waveform inversion in the frequency domain. Part I: theory and verification in a physic scale model, *Geophysics,* **64,** 888–901.

Pratt, R.G., Shin, C. & Hicks, G.J., 1998. Gauss-Newton and full Newton methods in frequency-space seismic waveform inversion, *Geophys. J. Int.,* **133,** 341–362.

Prieux, V., Brossier, R., Gholami, Y., Operto, S., Virieux, J., Barkved, O. & Kommedal, J., 2011. On the footprint of anisotropy on isotropic full waveform inversion: the Valhall case study, *Geophys. J. Int.,* **187,** 1495–1515.

Prieux, V., Brossier, R., Operto, S. & Virieux, J., 2013. Multiparameter full waveform inversion of multicomponent OBC data from valhall, Part 1: imaging compressional wavespeed, density and attenuation, *Geophys. J. Int.,* **194**(3), 1640–1664.

Ravaut, C., Operto, S., Improta, L., Virieux, J., Herrero, A. & dell'Aversana, P., 2004. Multi-scale imaging of complex structures from multi-fold wide-aperture seismic data by frequency-domain full-wavefield inversions: application to a thrust belt, *Geophys. J. Int.,* **159,** 1032–1056.

Robbins, H. & Monro, S., 1951. A stochastic approximation method, *Ann. Math. Stat.,* **22**(3), 400–407.

Romero, L.A., Ghiglia, D.C., Ober, C.C. & Morton, S.A., 2000. Phase encoding of shot records in prestack migration, *Geophysics,* **65, (2),** 426–436.

Schraudolph, N.N., Yu, J. & Günter, S., 2007. A stochastic quasi-Newton method for online convex optimization, in *Proceedings of 11th International Conference on Artificial Intelligence and Statistics (AIstats),* Society for Artificial Intelligence and Statistics, San Juan, Puerto Rico, pp. 433–440.

Schuster, G.T., Wang, X., Huang, Y., Dai, W. & Boonyasiriwat, C., 2011. Theory of multisource crosstalk reduction by phase-encoded statics, *Geophys. J. Int.,* **184,** 1289–1303.

Shin, C., Jang, S. & Min, D.J., 2001. Improved amplitude preservation for prestack depth migration by inverse scattering theory, *Geophys. Prospect.,* **49,** 592–606.

Sirgue, L. & Pratt, R.G., 2004. Efficient waveform inversion and imaging: a strategy for selecting temporal frequencies, *Geophysics,* **69**(1), 231–248.

Sirgue, L., Barkved, O.I., Dellinger, J., Etgen, J., Albertin, U. & Kommedal, J.H., 2010. Full waveform inversion: the next leap forward in imaging at Valhall, *First Break,* **28,** 65–70.

Tarantola, A., 1984. Inversion of seismic reflection data in the acoustic approximation, *Geophysics,* **49**(8), 1259–1266.
van Leeuwen, T. & Herrmann, F., 2012. Fast waveform inversion without source-encoding, *Geophys. Prospect.,* **61**(s1), 10–19.
Vigh, D., Moldoveanu, N., Jiao, K., Huang, W. & Kapoor, J., 2013. Ultralong-offset data acquisition can complement full-waveform inversion and lead to improved subsalt imaging, *Leading Edge,* **32**(9), 1116–1122.
Virieux, J. & Operto, S., 2009. An overview of full waveform inversion in exploration geophysics, *Geophysics,* **74**(6), WCC1–WCC26.

APPENDIX A: BP-2004 SALT MODEL SYNTHETIC DATA WITH NOISE

We perform FWI using the BP-2004 salt model adding 25 per cent of mean zero additive Gaussian noise to the data to test the sensibility to the noise. Regarding the set-up for the numerical test, we increase the value of they hyperparameters ($\lambda_x = \lambda_z = 10^{-6}$) and we reduce the maximum number of iterations performed in the resolution of the Newton system $N_{CG} = 5$, compared to the noise-free experiment (see Table A1). As the regularization damps the action of the Hessian (11), we can anticipate that this more aggressive regularization will penalize the convergence rate of the Newton methods. For the stopping criteria, we decrease the expected relative reduction of the misfit function to $\epsilon_1 = 25$ and 70 per cent for the frequency groups 1 and 2, respectively, as it is expected to be much less compared to the previous case because of the noise in the data. We add an additional termination criteria where we stop the inversion if the average of the misfit function over the previous 30 iterations has not changed more than 10 per cent ($\epsilon_2 = 0.1$). For this test, the ϵ_1 criterion was generally triggered during the inversion of the first frequency group, while the ϵ_2 criterion was generally triggered during the inversion of the second frequency group before the misfit function reaches a value corresponding to ϵ_1. The tuning parameters are outlined in Table A1 and can be compared with those used for the experiment performed without noise. When we apply source encoding, we use three encoded sources ($K = 3$).

Table A1. BP-2004 case study with noise. Tuning parameters for optimization algorithms. The same parameters are used for all of the optimization methods. β: damping factor of the Hessian pre-conditioner. λ_x, λ_z: weighting factors applied to the Tikhonov regularization in the misfit function. ϵ_1, ϵ_2: stopping criteria of non-linear iterations (see text for more details). The number of memory models in *l*-BFGS is 5. The maximum number of CG iterations, N_{CG}, in truncated Newton methods is 30. The number of supersource K equals to 3 when source encoding is used. The maximum number of forward problems is 10^5. For this case study, the same tuning is used when source encoding is used or not. Note that the weight of the Tikhonov regularization increased with respect to the case without noise in the data.

	Tuning parameters			
β	$\lambda_x = \lambda_z$	ϵ_1 (first frequency group)	ϵ_1 (second frequency group)	ϵ_2
10^{-2}	10^{-4}	0.25	0.7	0.1

Table A2. BP-2004 with 25 per cent of noise in the data. The same nomenclature than for Table 3 is used.

Optimization algo.	*DP* without *SE*	*DP* with *SE*	*S* (per cent)
CG	63 364	24 864	61
l-BFGS	26 784	12 252	54
GN	62 248	50 736	18
FN	57 040	40 848	28

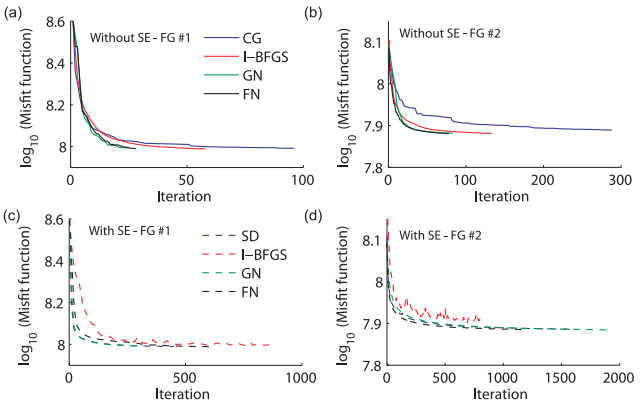

Figure A1. BP case study with noise: convergence rate. Reduction of the misfit function as a function of the iteration number without (a–b) and with (c–d) source encoding. (a, c) First frequency group (FG #1). (b, d) Second frequency group (FG #2). The curves are shown for the four optimization methods. Blue lines: *nl*-CG; red lines: *l*-BFGS; green lines: GN; black lines: FN.

A1 Convergence rate

A key difference with the noise free experiment is that the relative reduction of the misfit function is much smaller because the data noise level is higher. Once the noise level has been reached, the FWI may continue to significantly update the model without a perceptible decrease in the data misfit. This occurs because improvements in the model have a small weight in the data misfit compared to the high noise energy. As a consequence we do not immediately terminate the inversion when the misfit function is flat, but rather proceed to measure the relative change of the average value over a certain number of iterations, giving rise to stopping criterion 2 controlled with ϵ_2. Finding a suitable value of ϵ_2 that provides the best-trade-off between computational efficiency and quality of the subsurface model is not obvious because the risk is either to stop the iterations too early (before a sufficient accuracy of the subsurface model is reached) or too late (iterations do not lead to significant update of the subsurface model). The speed-up estimation is always sensitive to the stopping criteria but in this context it is even more critical.

This trend in the convergence curves is illustrated in Fig. A1, which shows the slowly decreasing misfit functions as a function of the number of iterations for the four optimization methods with and without source encoding. These curves can be compared with those obtained when the data does not contain noise (Fig. 4; note that the horizontal and vertical scales in Fig. A1 spans over a much narrow range of misfit function values than in Fig. 4). As for the noise-free case, the truncated Newton methods reach the stopping criterion of iteration with a smaller number of iteration than *nl*-CG and to a lesser extent to *l*-BFGS when all the sources are processed independently (Figs A1a and b) and when source encoding is used (Figs A1c and d). However, the difference between the convergence speed of the different optimization methods is less pronounced than for the noise-free case and this levelling down of the convergence speed is still accentuated when source encoding is used. This levelling down of the performances results because the convergence of the truncated Newton methods now reaches a plateau before satisfying the stopping criterion of iterations when a significant amount of noise is added to the data, unlike in the case of noise-free data.

The final FWI velocity models inferred from the GN optimization with and without source encoding are similar and compare well with the subsurface models inferred from the noise-free experiment (compare Figs 5 and A2).

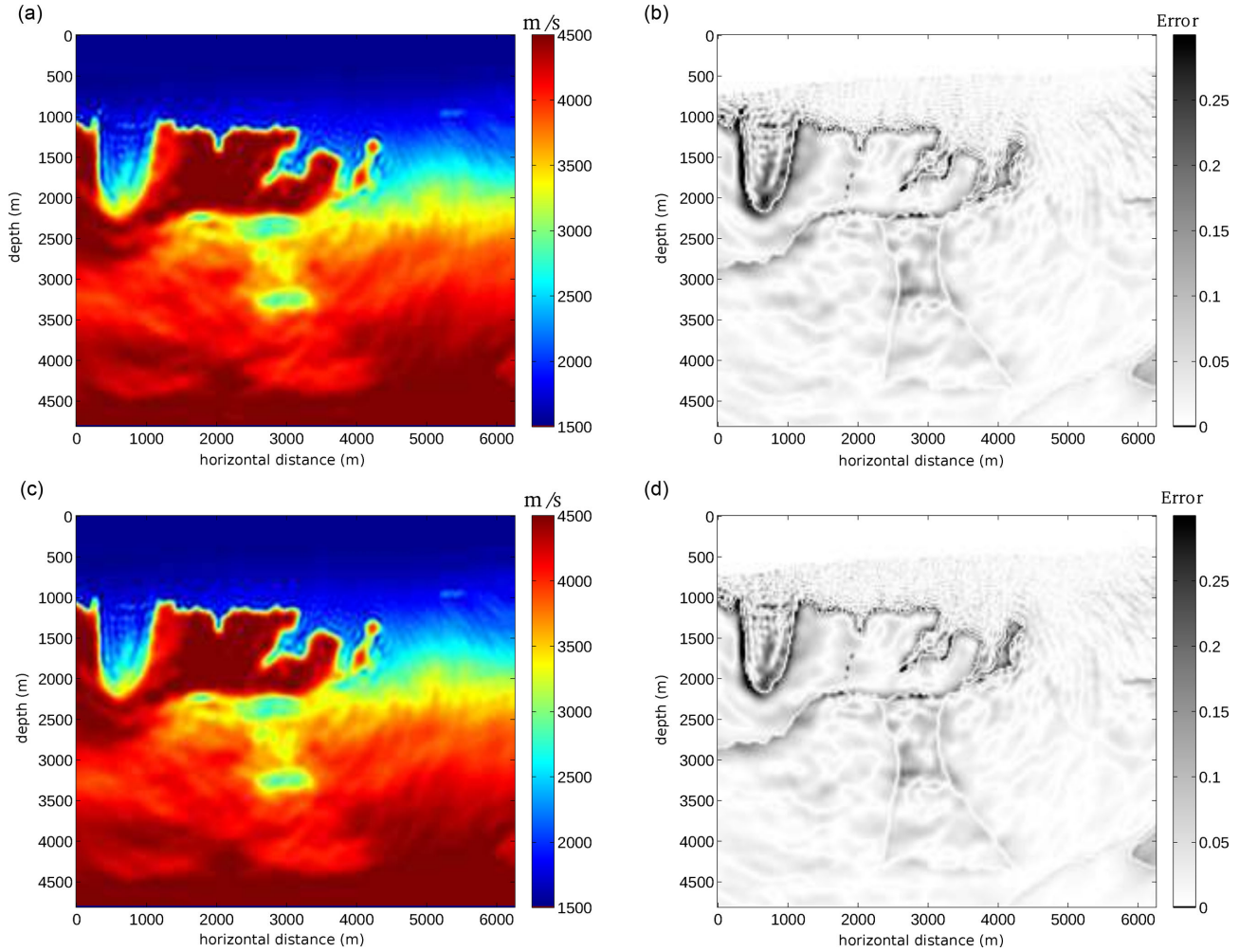

Figure A2. BP case study with noise. (a, c) Final FWI model for the GN optimization method used without (a) and with (c) source encoding ($K = 3$). (b, d) Velocity model error (difference between the final FWI model and the true subsurface model).

A2 Computational efficiency and speed-up

The reduction of the misfit function as a function of the number of forward problems, when all the sources are processed independently and when source encoding is used, is shown in Fig. A3 for each optimization method. These curves can be compared with those inferred from noise-free data (Fig. 6). The speed-up of each optimization method as a function of the misfit-function value is synthesized in Figs 7(c) and (d) for noisy data.

The *nl*-CG/SD method still has the best speed-up. Moreover, the difference with the other optimization methods has even increased compared to the case of noise-free data (see also Table 3). This results because, as already mentioned, the convergence rate of the truncated Newton methods was affected more than the *nl*-CG/SD optimization by both the Gaussian noise and the cross-talk noise. The effect of noise on the speed-up is clearly illustrated by the comparison of the speed-up curves inferred from noise-free and noisy data (Fig. 7). In the case of noise-free data, the speed-up decreases slowly as the value of the misfit function decreases from right to left in the figure, while it decreases much more rapidly in the case of noisy data as the convergence curves start reaching a plateau. As the slope of the speed-up curves increases near the smallest values of the misfit function in Figs 7(c) and (d), differences between the speed-up of each optimization methods are emphasized.

The average and variance of fifty convergence curves of the misfit function for the first and second frequency group are plotted in Fig. A4. In general all realizations tend towards the same minimum, for all the optimization methods, although *l*-BFGS shows a less robust behaviour (Fig. A4b). The variance shows that there is a maximum variability for the *l*-BFGS$_r$ method, but confirms that the inversion of noisy data with source encoding methods is statistically stable (Fig. A5). The maximum of the variance is shown on top of the salt body near the end of the model where a more limited illumination is available. GN has the smallest variance (Fig. A5).

In conclusion, we observe that when noise is added to the data, the speed-up with respect to the value of the misfit function decreases more rapidly, and the convergence of the optimization near the global minimum of the misfit function slows down (Fig. 7). In addition, the performance of all the optimization methods tend to be levelled down as noise is added to the data and the action of the Hessian is damped. Even though *l*-BFGS is the fastest method with and without source encoding, it is not very robust, specially when noise is added to the data. We observed the same behaviour with the application on real data.

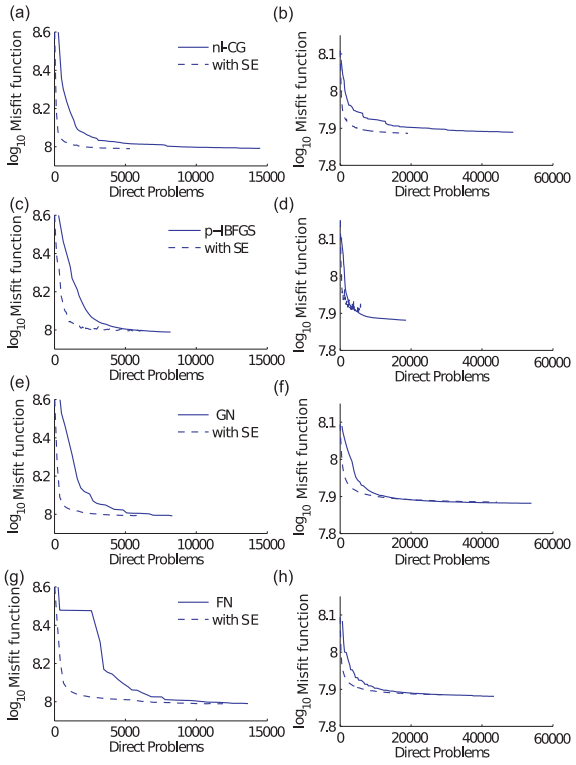

Figure A3. BP case study with noise. Assessment of the computational saving provided by source encoding: reduction of the misfit function as a function of the forward problems, for the first (a, c, e, g) and second (b, d, f, h) frequency groups. (a and b) *nl*-CG optimization method. (c and d) *l*-BFGS optimization method. (e and f) GN optimization method. (g and h) FN optimization method. The computational gain is provided by the difference between the number of forward problems performed with (dash lines) and without (solid lines) source encoding for a given misfit function value.

APPENDIX B: STOCHASTIC GRADIENT AND SOURCE ENCODING

Implementation of source encoding in a SD method mimics a stochastic gradient algorithm (Robbins & Monro 1951) that is widely used in many applications. In machine learning, for example, a very large set of training examples are available and instead of using all of them at once to compute the gradient, each training example is individually employed to find a descent direction. Although the descent direction inferred from one training example is not as accurate as the one inferred from all the training examples at once, it has been shown to be more efficient to do a sweep using one at a time. This way, many low cost inaccurate iterations are performed that converge to the global minimum.

In source encoding, the randomness is not generated by choosing a different source at each iteration, which would be the analogous of the machine-learning setting, but rather the randomness is introduced by changing the weights in the linear combination that create the supersources. In addition to the benefits in computational cost, stochastic gradient has advantages inherent to stochastic optimization techniques, such as simulated annealing, that aid to find a global minimum of a misfit function that may possess several local minima. Bottou (1991) highlights the analogy between the temperature in simulated annealing and the step length (referred to as learning rate) in the stochastic gradient method. In the stochastic gradient method, we allow the misfit function to increase for when we change the encoding (Fig. A4), allowing us to explore regions of the model space that would never be accessible with deterministic methods. This relaxation of the explored model space, which is illustrated by the small fluctuations of the convergence curves in Fig. A4(b), may allow to overcome small local minima and we observe this advantage through the following synthetic numerical test.

We apply FWI on the noise-free data without the multi-scale strategy by inverting nine frequencies ranging from 1 Hz to 9 Hz in one group, such as to render the misfit function non-convex with many local minima. Using the initial model shown in Fig. 1(b), the inversion with and without source encoding converge to equivalent final solutions (Figs A6a and b). When we degrade the initial model to that depicted in Fig. 1(c), the final FWI velocity model obtained when all the sources are processed independently is less accurate than the one inferred from the stochastic optimization (Figs A6c and d). Therefore, we conclude that, with source encoding, we may not only reduce the computational cost but we may also steer the solution towards another local minimum thanks to a broader exploration of the model space. For the example presented here, the local minimum attained is better than with deterministic methods, and it is statistically stable, as shown in the variance of the final model in Fig. A7. However, there is no guarantee that this will always be the case and that the solution with source encoding will always be a more adequate local minimum.

APPENDIX C: NUMBER OF SUPERSOURCES IN SOURCE ENCODING

Currently, there is no theoretical result to determine the number of supersources that will allow to obtain the highest computational gain and guarantee convergence. Consequently, in the numerical experiments with synthetic and real data, we selected the lowest possible quantity of supersources that converge to an acceptable final velocity model. In this section, we illustrate how the computational cost and the statistical stability change as the number of supersources is increased. Generally we find there is a trade-off : the computational cost is lower with fewer number of supersources, and the statistical stability increases as the number of supersources increases. However, as was illustrated in Section 4.4, the Newton methods have a lower variance, allowing therefore to benefit from a higher statistical variance with a fewer number of supersources.

Using the synthetic BP-2004 salt model presented in Section 4.3, we perform an inversion with noiseless data using $K = 3$, 21 and $K = 31$ supersources. Fig. A8(a) shows the mean value of the reduction of the misfit function over 50 independent realizations as a function of the forward problems solved using *l*-BFGS for both frequency groups. The computational cost with $K = 21$ and 31 is very similar, and lower to the one obtained with $K = 1$. On the other hand, the lowest computational cost using GN is obtained with $K = 3$, as in shown in Fig. A8(b). All the inversions for different values of supersources converge to the same value of misfit function and the statistical variance of the final solutions is similar for all inversions, as shown in Fig. A9. We choose to work with $K = 3$ that provides the lowest computational cost for GN, even though for *l*-BFGS it is not optimal. However, since the computational cost for GN is greater (as can be seen by comparing the length of the horizontal axes in Fig. A8), *l*-BFGS will still be the most efficient (as shown in Section 4.3). All the results in Section 4.3 and A are thus performed using $K = 3$.

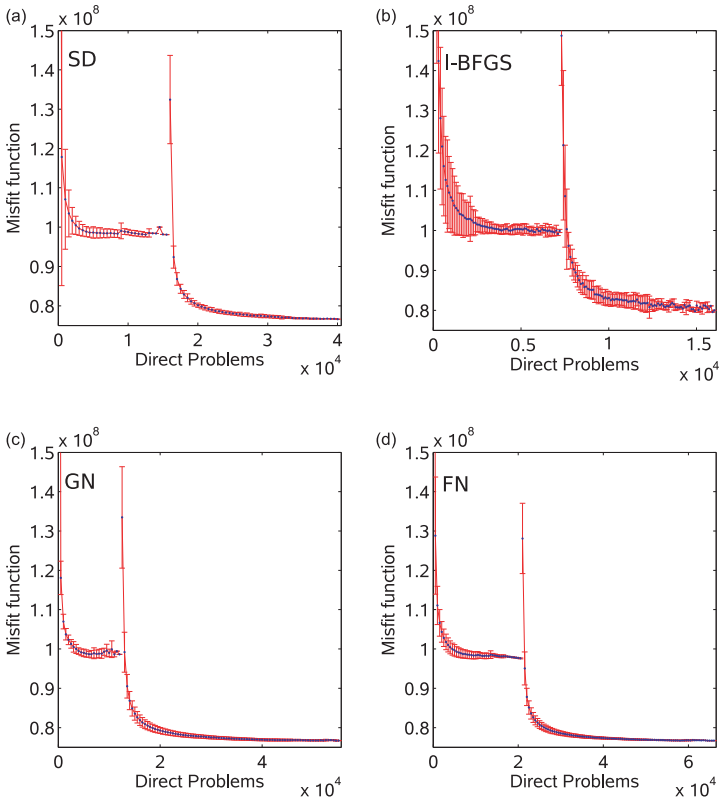

Figure A4. BP case study with noise. Mean value and variance of the misfit function value as a function of the number of forward problems for the first and second frequency group, during 50 independent realizations, using source encoding. (a) SD, (b) *l*-BFGS, (c) GN, (d) FN.

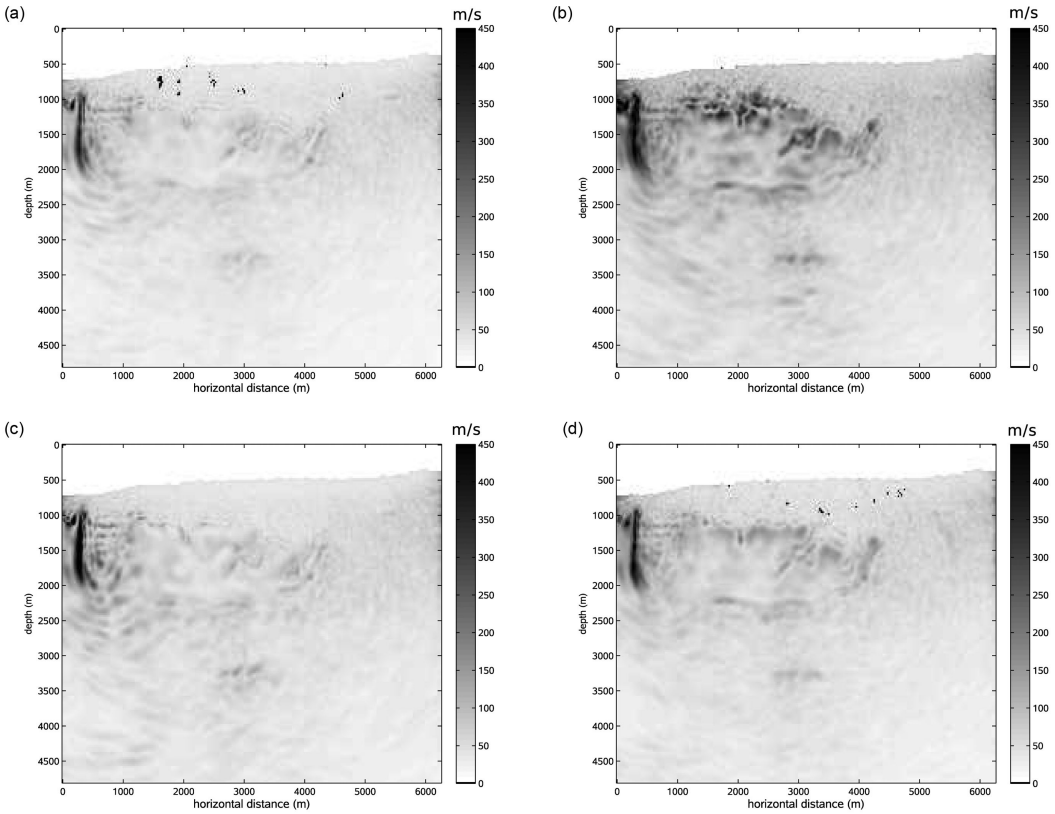

Figure A5. BP case study with noise. Standard deviation of the final velocity model (in m s^{-1}) for 50 realizations using source encoding. (a) SD, (b) *l*-BFGS, (c) GN, (d) FN.

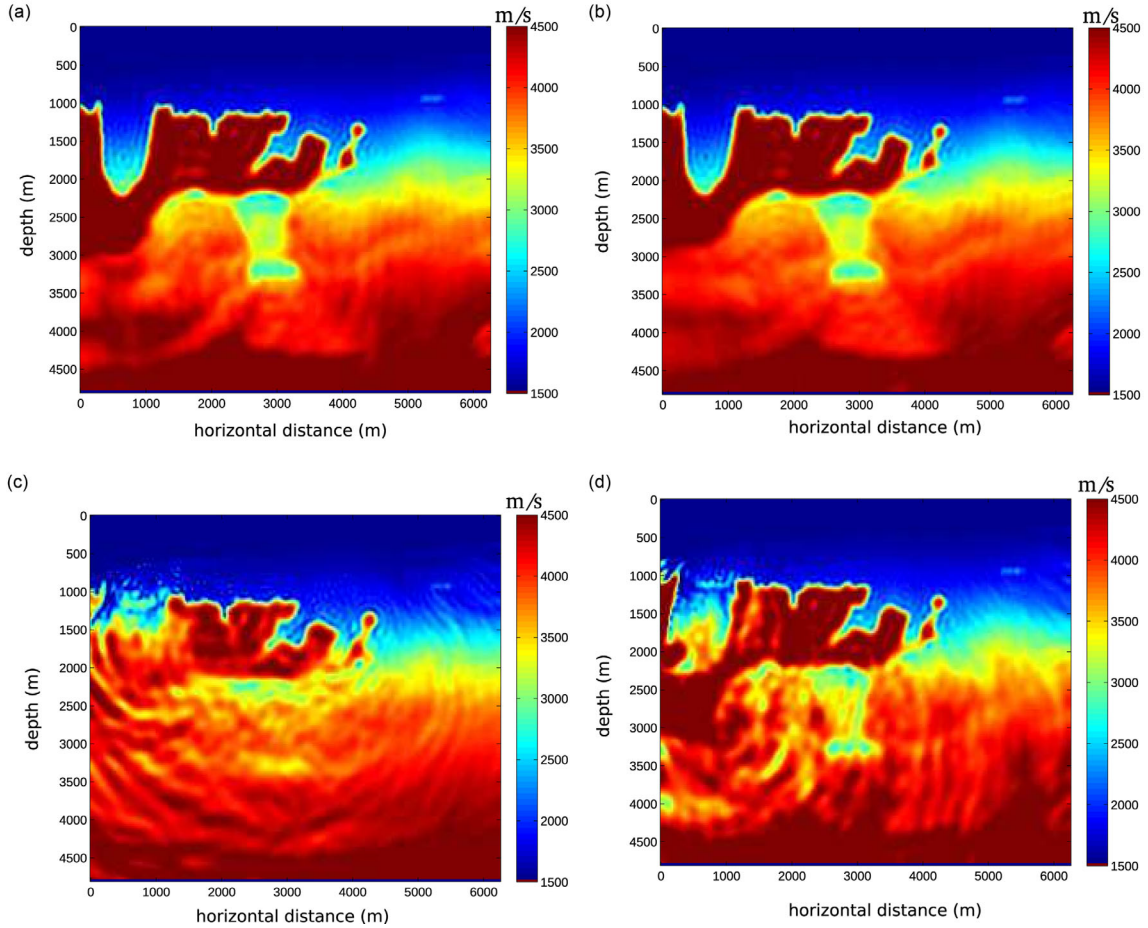

Figure A6. BP case study without noise. (a and b) Final velocity models obtained by inverting a single frequency group containing nine frequencies between 1 and 9 Hz without (a) and with source encoding (b) and with the initial velocity model shown in Fig. 1(b). (c and d) Same as (a and b) with the smoother initial velocity model shown in Fig. 1(c). Note how source encoding allows to reach an improved local minimum.

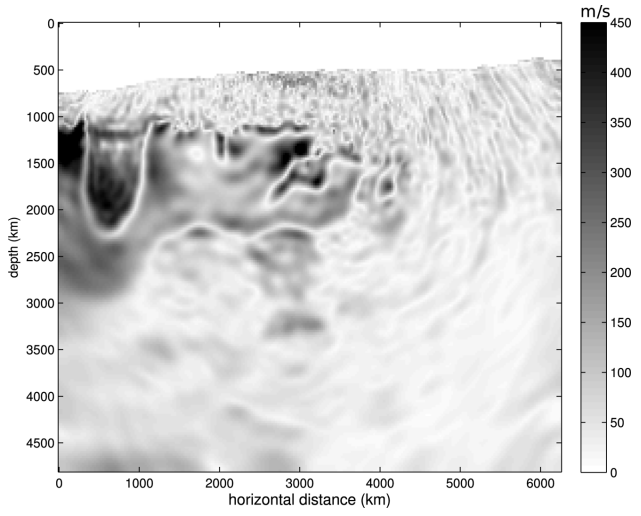

Figure A7. BP case study without noise. Standard deviation (in m s^{-1}) of the final velocity model over 50 realizations using source encoding associated to model shown in Fig. A6(d).

When using source encoding with noisy or real data, the descent direction provided by *l*-BFGS and SD is less robust. Using the Valhall data set presented in Section 4.4, we illustrate in Fig. A10(a) the mean value of the reduction of the misfit function for four frequency

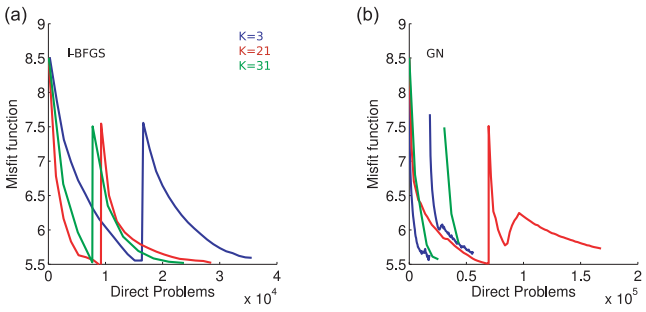

Figure A8. BP case study without noise. Average reduction of the misfit function versus number of forward problems with source encoding for different values of supersources K, over 50 realizations. Blue line: $K = 3$. Red line: $K = 21$. Green line: $K = 31$. (a) *l*-BFGS, (b) GN. There are originally 62 sources.

groups over 50 independent realizations with *l*-BFGS, using $K = 1$, 11, 21 and $K = 53$ supersources. For all the values of K shown, there are some frequency groups where, in average, *l*-BFGS fails to converge. As can be seen, the computational cost increases as the number of supersources increases. However, the convergence curves for a higher number of supersources ($K = 21$, 53) show a more monotonic behaviour. This is in agreement with the reduction of the statistical variance of the final velocity models with an increasing number of supersources, shown in Fig. A11. When

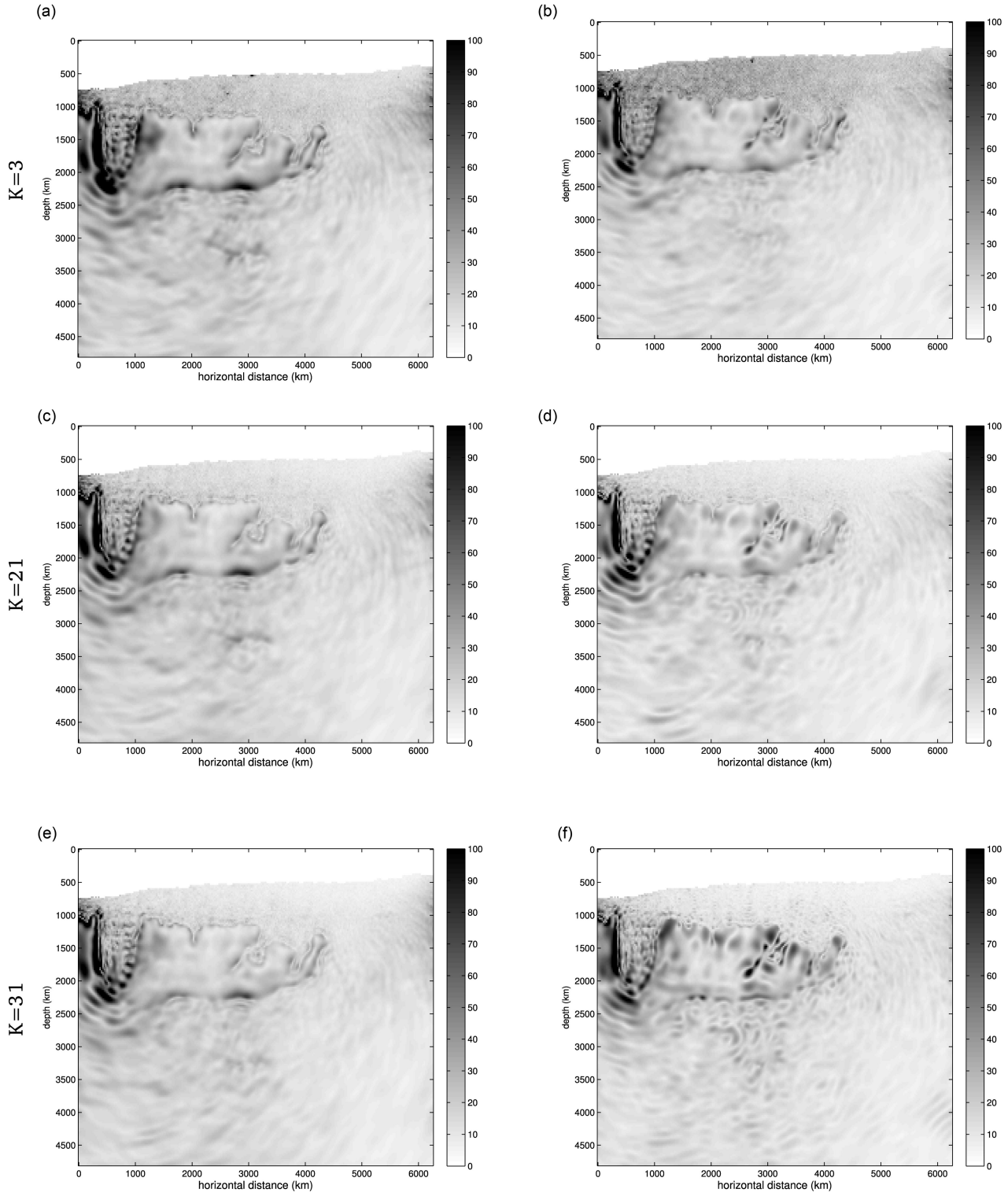

Figure A9. BP case study without noise. Variance of the final velocity model with source encoding for different values of supersources K, over 50 realizations. Panels (a), (c) and (e) correspond to an inversion with l-BFGS. Panels (b), (d) and (f) correspond to an inversion with GN. First row: $K = 3$. Second row: $K = 21$. Third row: $K = 31$. There are originally 62 sources.

the inversion is performed with different number of supersources, but interfaced GN, the overall behaviour is the same: the lowest computational cost is attained with the fewest number of supersources (Fig. A10b), and the statistical variance is inversely proportional to the number of supersources (Fig. A12). However, the reduction of the misfit function is more monotonous for all values of K and the variations of the final velocity models are smaller. Because we seek the highest computational gain, we choose to work with $K = 1$ for all the numerical experiments with real data presented in Section 4.4.

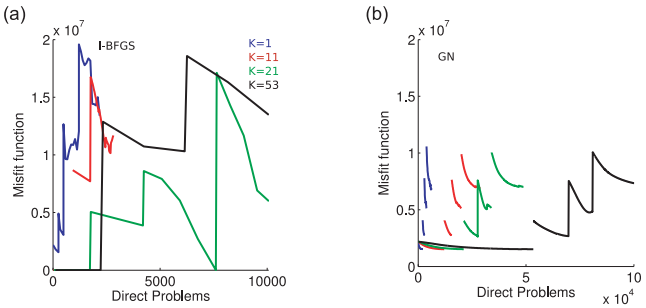

Figure A10. Valhall case study. Average reduction of the misfit function versus number of forward problems with source encoding for different values of supersources K, over 50 realizations. Blue line: $K = 1$. Red line: $K = 11$. Green line: $K = 21$. Black lines: $K = 53$. (a) l-BFGS, (b) GN. There are originally 210 sources.

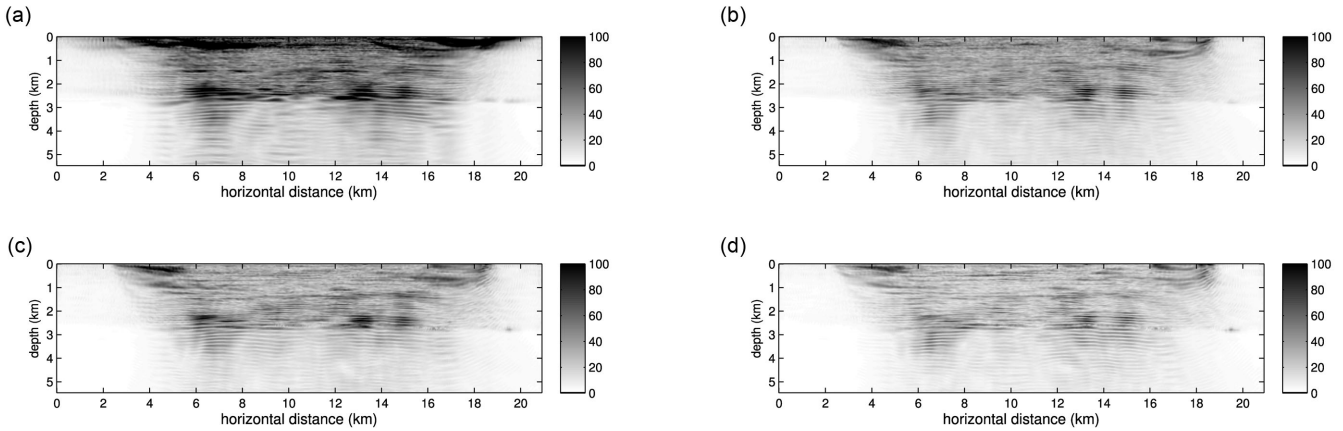

Figure A11. Valhall case study. Variance of the final velocity model (in m s^{-1}) with source encoding using the l-BFGS optimization method for different values of supersources K. (a) K = 1, (b) $K = 11$, (c) $K = 21$ and (d) $K = 53$.

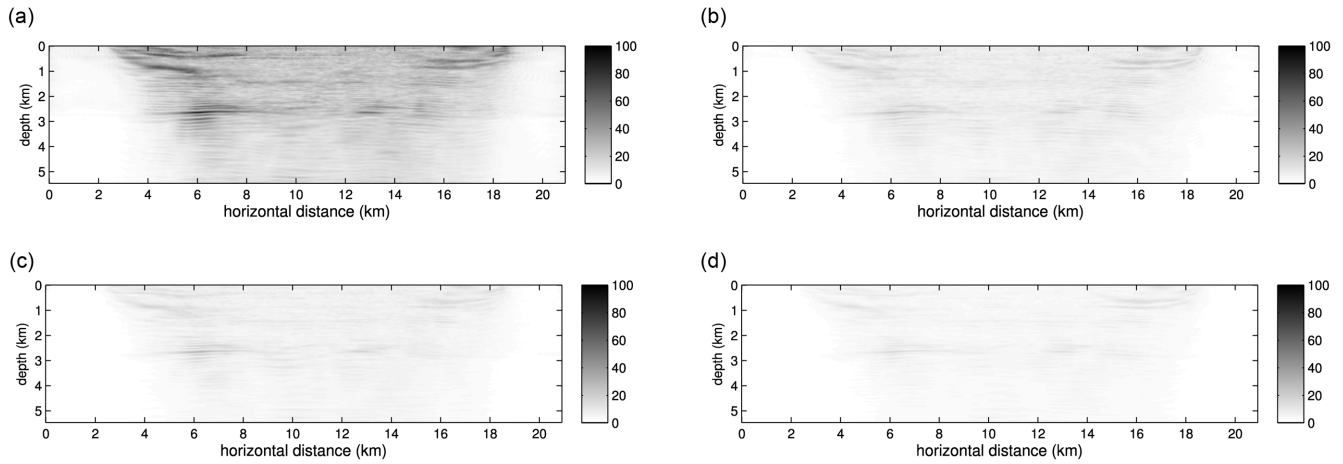

Figure A12. Valhall case study. Variance of the final velocity model (in m s^{-1}) with source encoding using the Gauss–Newton optimization method for different values of supersources K. (a) $K = 1$, (b) $K = 11$, (c) $K = 21$ and (d) $K = 53$.