Geophysical Journal International

Geophys. J. Int. (2013) **193,** 1432–1446
Advance Access publication 2013 March 08

doi: 10.1093/gji/ggt055

Three-dimensional parallel distributed inversion of CSEM data using a direct forward solver

A. V. Grayver, R. Streich* and O. Ritter

GFZ German Research Centre for Geosciences, Telegrafenberg, 14473 *Potsdam, Germany. E-mail: agrayver@gfz-potsdam.de*

Accepted 2013 February 6. Received 2013 February 1; in original form 2012 November 4

SUMMARY

For 3-D inversion of controlled-source electromagnetic (CSEM) data, increasing availability of high-performance computers enables us to apply inversion techniques that are theoretically favourable, yet have previously been considered to be computationally too demanding. We present a newly developed parallel distributed 3-D inversion algorithm for interpreting CSEM data in the frequency domain. Our scheme is based on a direct forward solver and uses Gauss–Newton minimization with explicit formation of the Jacobian. This combination is advantageous, because Gauss–Newton minimization converges rapidly, limiting the number of expensive forward modelling cycles. Explicit calculation of the Jacobian allows us to (i) precondition the Gauss–Newton system, which further accelerates convergence, (ii) determine suitable regularization parameters by comparing matrix norms of data- and model-dependent terms in the objective function and (iii) thoroughly analyse data sensitivities and interdependencies. We show that explicit Jacobian formation in combination with direct solvers is likely to require less memory than combinations of direct solvers and implicit Jacobian usage for many moderate-scale CSEM surveys. We demonstrate the excellent convergence properties of the new inversion scheme for several synthetic models. We compare model updates determined by solving either a system of normal equations or, alternatively, a linear least-squares system. We assess the behaviour of three different stabilizing functionals in the framework of our inversion scheme, and demonstrate that implicit regularization resulting from incomplete iterative solution of the model update equations helps stabilize the inversion. We show inversions of models with up to two million unknowns in the forward solution, which clearly demonstrates applicability of our approach to real-world problems.

Key words: Inverse theory; Numerical approximations and analysis; Electromagnetic theory.

1 INTRODUCTION

Controlled-source electromagnetic (CSEM) surveying in the frequency domain is an established tool for hydrocarbon exploration and near-surface investigations in marine (Constable 2010), airborne (Siemon *et al.* 2009) and cross-well (Newman & Alumbaugh 1997) configurations. Interpretation of CSEM data requires multidimensional modelling and inversion techniques. Considerable effort has been devoted recently to developing algorithms that accurately recover the electrical conductivity distribution and, at the same time, are executable on existing computer hardware (Newman & Alumbaugh 1997; Zhang 2003; Abubakar *et al.* 2008; Commer & Newman 2008; Plessix & Mulder 2008; Cox *et al.* 2010; Schwarzbach & Haber 2012).

Generally, CSEM inversion iteratively minimizes a penalty functional by successively updating a model of subsurface electrical conductivity. Model updates are based on comparison of observed and forward-modelled data for the assumed conductivity model, and on *a priori* constraints (e.g. Constable *et al.* 1987; Candansayar 2008; Lelièvre & Oldenburg 2009). Computations of the required forward solutions typically constitute the most expensive part of the inversion algorithm (Newman & Boggs 2004; Oldenburg *et al.* 2008; Brossier *et al.* 2010; Egbert & Kelbert 2012). Many existing 3-D EM inversion algorithms use iterative forward solvers (e.g. Haber *et al.* 2007; Commer & Newman 2008; Plessix & Mulder 2008). Iterative techniques require little memory and are fast for computing single-source solutions, yet very time consuming if many solutions are required, as is the case in 3-D inversion. They are also difficult to apply for solving the Helmholtz equation occurring in EM modelling (Mulder 2006; Ernst & Gander 2011; Um *et al.* 2012), and solutions may fail to converge.

Forward modelling can also be carried out using direct solvers, which are typically more robust than iterative solvers for numerically difficult cases, such as models containing air or strongly non-uniform grids (Gould *et al.* 2007), and solution times are far less dependent on grid configuration and conductivity than for

* Now at: Shell Global Solutions International BV, Kesslerpark 1, 2288 GS Rijswijk, The Netherlands.

iterative solvers. Direct solvers separate the solution of linear equation systems into a resource-demanding system matrix factorization and comparatively inexpensive forward-backward substitution steps. For computing multiple solutions, only the forward-backward substitution needs to be repeated. This makes direct solvers particularly efficient for multisource problems and inversions that require many additional forward solutions for computing data sensitivities. Direct solvers have recently been used for 2.5-D inversion (Abubakar *et al.* 2008; Key & Ovall 2011) and 3-D forward modelling (Streich 2009; Um *et al.* 2010; Schwarzbach *et al.* 2011; da Silva *et al.* 2012), and are also being considered for 3-D EM inversion (Oldenburg *et al.* 2008; Yang & Oldenburg 2012; Schwarzbach & Haber 2012).

Most existing inversion schemes have been optimized to work efficiently with iterative solvers. When using direct solvers, optimization techniques have to be reconsidered. Because every model update entails new expensive matrix factorizations, it is advantageous to minimize the number of model updates. Therefore, methods with high convergence rates, such as Newton or Gauss–Newton schemes (Newman & Hoversten 2000; Haber *et al.* 2000), are preferable, even if they require additional work within each iteration to obtain the model update. Furthermore, because multiple forward solutions can be computed cheaply using direct solvers, explicit computation of the Jacobian matrix becomes affordable. The full Jacobian can then be used for improving the minimization procedure by preconditioning the system of model update equations and establishing a suitable starting value for the regularization parameter.

Although using direct solvers for computing forward solutions is becoming feasible, the system of model update equations inevitably has to be solved iteratively. For Gauss–Newton minimization, a system of normal equations is commonly formed and solved using conjugate-gradient type solvers (Egbert 2012). However, the system of normal equations may be severely ill-conditioned and thus difficult to solve iteratively. A least-squares formulation that is equivalent to the normal equations may be numerically more favourable (Björck 1996). The computational effort for solving the least-squares system using the LSQR algorithm (Paige & Saunders 1982) is nearly identical to that required for solving the normal equation system. Nonetheless, the least-squares formulation has not been used much for EM inversion (Ajo-Franklin *et al.* 2007). In striving to optimize image quality and convergence properties of the inversion, we compare the two approaches. Furthermore, the conjugate-gradient or LSQR solvers impose implicit regularization depending on the number of iterations used. With the full Jacobian available, varying the number of iterations has little impact on overall runtimes, hence the influence of this implicit regularization can be investigated.

In this contribution, we first review 3-D forward modelling based on a direct solver, and then present the newly developed distributed inversion scheme that makes use of direct forward solutions. We compare the normal equation and least-squares formulations, and include three different stabilizing functionals of practical interest. After evaluating the time and memory complexity of explicit Jacobian computation, we analyse the influence of regularization on the spectra of the Jacobian and Hessian matrices utilized in the two minimization schemes. From this analysis, we derive a starting guess for the regularization parameter, which can easily be computed because the Jacobian is available. We describe the fully distributed parallel implementation of the inversion scheme. Finally, we present inversion results for land CSEM configurations that demonstrate the excellent convergence behaviour of our algorithm, spectral properties of the Jacobian and Hessian matrices, and the scalability of our inversion approach.

2 THE FORWARD ALGORITHM

To calculate model responses, we solve the vector Helmholtz equation for the secondary electric field $\mathbf{E}^{\mathrm{S}}$

$$\nabla \times \nabla \times \mathbf{E}^S + j\omega\mu_0\sigma^*\mathbf{E}^S = -j\omega\mu_0(\sigma^* - \sigma^{P*})\mathbf{E}^P, \tag{1}$$

where ω denotes angular frequency, the complex conductivity $\sigma^* = \sigma + j\omega\varepsilon$ includes conductivity σ and permittivity ε, μ_0 denotes free-space magnetic permeability and σ^{P*} is the conductivity of a layered background model. The primary electric field $\mathbf{E}^P$ is computed for conductivity σ^{P*} using quasi-analytic expressions for 1D media (Løseth & Ursin 2007; Streich & Becken 2011). Perfect electric conductor boundary conditions are imposed.

We apply a staggered finite-difference discretization as described in Streich (2009), resulting in a system of linear equations

$$\mathbf{A}\mathbf{E}^S = \mathbf{b} \tag{2}$$

of size $N = 3N_xN_yN_z$, with N_α being the number of cells in each spatial direction. The system matrix $\mathbf{A}$ is complex, sparse and symmetric positive definite. This allows us to apply an $\mathbf{LDL}^H$ decomposition (Amestoy *et al.* 2006), which has a memory complexity of $\approx O(N^{1.5})$ (George *et al.* 1994), followed by forward and backward substitutions to solve the system 2.

In the limit $\omega \to 0$, the geometric term $\nabla \times \nabla \times \mathbf{E}^S$ in eq. (1) dominates over the conductivity-dependent term $j\omega\mu_0\sigma^*\mathbf{E}^S$, especially in low-conductivity regions such as air, where $\sigma \to 0$, making the solution inaccurate (e.g. Smith 1996). To stabilize the system at low frequencies when using a direct solver, we explicitly enforce a divergence condition (Schwarzbach 2009; Streich *et al.* 2010), resulting in an augmented system of size $4N_xN_yN_z$.

3 INVERSION ALGORITHM

3.1 Formulation of the problem

We formulate the discrete non-linear inverse problem as a minimization of the objective function

$$\phi(\mathbf{m}) = \phi_d(\mathbf{m}) + \beta\phi_m(\mathbf{m}), \tag{3}$$

where ϕ_d and ϕ_m are data- and model-dependent terms, respectively. We choose to minimize the geometric distance between predicted and observed data, thus the data-dependent term is

$$\phi_{\mathrm{d}}(\mathbf{m}) = \frac{1}{2}[\mathbf{f}(\mathbf{m}) - \mathbf{d}^{\mathrm{obs}}]^H\mathbf{W}_{\mathrm{d}}^T\mathbf{W}_{\mathrm{d}}[\mathbf{f}(\mathbf{m}) - \mathbf{d}^{\mathrm{obs}}], \tag{4}$$

where superscript H denotes the Hermitian transpose. The forward operator $\mathbf{f}(\mathbf{m})$ provides predicted data, given the vector of real model parameters $\mathbf{m}$ of size N_m. The vector $\mathbf{d}^{\mathrm{obs}}$ contains N_{d} complex values of observed data, which may be electric and magnetic field values specified at measurement locations for electric or magnetic dipole sources or other, more complex source configurations (Streich *et al.* 2011). Without loss of generality, we assume that every recorded field component represents a separate receiver. For N_{f} frequencies, N_{s} sources and N_{r} receivers, we obtain $N_{\mathrm{d}} = N_{\mathrm{f}}N_{\mathrm{s}}N_{\mathrm{r}}$. The diagonal matrix $\mathbf{W}_{\mathrm{d}}$ of size $N_{\mathrm{d}} \times N_{\mathrm{d}}$ introduces data weights as

$$\mathbf{W}_{\mathrm{d}} = \mathrm{diag}\left(\frac{1}{|d_i^{\mathrm{obs}}|r_i + \eta}\right); \quad i = 1, \ldots, N_{\mathrm{d}}, \tag{5}$$

where r_i is the estimated relative error of the i^{th} datum. A small constant η, corresponding to the noise floor of the data, is added to prevent the inversion from overemphasising low-amplitude data (Pidlisecky *et al.* 2007). The problem can then be thought of as a minimization of the weighted distance between predicted and observed data

$$\min_{\mathbf{m}} \phi_d(\mathbf{m}) = \frac{1}{2}||\mathbf{f}(\mathbf{m}) - \mathbf{d}^{\text{obs}}||^2_{\mathbf{W_d}^T\mathbf{W_d}}. \tag{6}$$

The second term $\phi_m(\mathbf{m})$ in eq. (3) is a regularization functional (Tikhonov & Arsenin 1977; Constable *et al.* 1987; Zhdanov 2002) that contains *a priori* assumptions about the model (Lelièvre & Oldenburg 2009) and stabilizes the ill-posed inverse problem. The scalar regularization parameter β controls the influence of the regularization term relative to the data-dependent term. We compare the performance of three different functionals for our models of interest. First, we use a Tikhonov stabilizing functional that seeks a model with minimum L_2-norm deviation from a reference model,

$$\phi_m^{mn}(\mathbf{m}) = \frac{1}{2}||\mathbf{m} - \mathbf{m}^{\text{ref}}||_2^2 = \min. \tag{7}$$

Secondly, we test a maximum smoothness condition that minimizes

$$\phi_m^{sm}(\mathbf{m}) = \frac{1}{2}||\nabla^2(\mathbf{m} - \mathbf{m}^{\text{ref}})||_2^2 = \min, \tag{8}$$

where ∇^2 is a discretization of the Laplacian. We use a finite-difference approximation of the Laplacian in 3-D, resulting in a sparse matrix with a seven-point stencil,

$$\mathbf{L} = a_x\mathbf{L}_x + a_y\mathbf{L}_y + a_z\mathbf{L}_z \in \mathbb{R}^{N_m \times N_m}. \tag{9}$$

Separate scalar smoothing parameters a_x, a_y, a_z are applied to the x, y and z spatial components of the discrete Laplacian (Li & Oldenburg 1996). The smoothing operator $\mathbf{L}$ is independent of the model parameters and constant throughout the inversion. *A priori* information about known interfaces can be included into $\mathbf{L}$ by reducing the smoothing parameters for the corresponding locations (Newman & Hoversten 2000).

The third regularizer is a minimum support stabilizing functional (Portniaguine & Zhdanov 1999; Ajo-Franklin *et al.* 2007; Carbajal *et al.* 2012) that attempts to minimize the size of the anomalous regions in which the model parameters deviate from the reference model,

$$\phi_m^{ms}(\mathbf{m}) = \frac{1}{2}\sum_{j=1}^{N_m} \frac{(m_j - m_j^{\text{ref}})^2}{(m_j - m_j^{\text{ref}})^2 + e^2}. \tag{10}$$

The constant e stabilizes the functional for $(m_j - m_j^{ref}) \to 0$ and controls the compactness of the solution.

3.2 Minimization of the objective function

Various classes of gradient-based minimization techniques are available, differing generally in how much first- and second-order derivative information of the objective function is considered. Nonlinear conjugate gradient (NLCG) methods (Rodi & Mackie 2001; Newman & Boggs 2004; Commer & Newman 2008) that exclusively consider first-derivative information, or limited-memory quasi-Newton approaches (Plessix & Mulder 2008) that include sparse approximate second-derivative information, exhibit linear convergence (Nocedal & Wright 1999). Therefore, despite being well suited if the forward problem is solved iteratively on low-memory machines, these techniques are impractical in combination with direct solvers, because they typically require many inversion iterations. The number of iterations is proportional to the number of model updates and thus to the number of $\mathbf{LDL}^H$ decompositions of the system matrix in eq. (2) that need to be carried out. Accordingly, combining such inversion approaches with direct forward solvers would be prohibitively expensive.

Methods with quadratic convergence, such as Newton techniques, are theoretically most favourable, because they should only require few iterations. Nevertheless, full Newton methods are difficult to apply in practice due to the immense computational effort required for computing the second derivative of the objective function, and due to unfavourable numerical properties of the Hessian matrix (Pratt *et al.* 1998; Fernández Martínez *et al.* 2012). We, therefore, use a Gauss–Newton technique, which is a modification of the full Newton approach that exhibits convergence rates lower than quadratic, but significantly higher than linear (Nocedal & Wright 1999; Habashy & Abubakar 2004) if the starting model is chosen appropriately. Newton-based methods also allow us to change the regularization parameter at each inversion iteration. We can thus conduct parameter search (Egbert 2012) or apply strategies such as cooling approaches (Newman & Alumbaugh 1997). This would not easily be possible with NLCG or quasi-Newton methods, because varying the regularization parameter would compromise the orthogonality of search directions (Egbert 2012).

Applying a second-order Taylor expansion to eq. (3) and dropping second-order terms results in the system of normal equations

$$\left[Re\{\bar{\mathbf{J}}^H\bar{\mathbf{J}}\} + \beta\nabla_m^2\phi_m(\mathbf{m})\right]\delta\mathbf{m} = -\left[Re\{\bar{\mathbf{J}}^H\mathbf{W}_d[\mathbf{f}(\mathbf{m}) - \mathbf{d}^{\text{obs}}]\} + \beta\nabla_m\phi_m(\mathbf{m})\right]. \tag{11}$$

Here, $\delta\mathbf{m}$ is a model update vector and $\bar{\mathbf{J}} = \mathbf{W}_{\text{d}}\mathbf{J} \in \mathbb{C}^{N_{\text{d}} \times N_m}$ is the weighted sensitivity matrix that represents the partial derivatives of the data with respect to the model parameters. The matrix on the left-hand side is the regularized approximate Hessian. The right-hand side is the negative gradient representing the descent direction. The first and second derivatives of the stabilizing functional $\phi_m(\mathbf{m})$ with respect to the model parameters are, for the minimum norm functional (eq. 7),

$$\nabla_m\phi_m^{mn}(\mathbf{m}) = (\mathbf{m} - \mathbf{m}^{\text{ref}}), \tag{12a}$$

$$\nabla_m^2\phi_m^{mn}(\mathbf{m}) = \mathbf{I}. \tag{12b}$$

For the maximum-smoothness condition (eq. 8) we obtain

$$\nabla_m\phi_m^{sm}(\mathbf{m}) = \mathbf{L}^T\mathbf{L}(\mathbf{m} - \mathbf{m}^{\text{ref}}), \tag{13a}$$

$$\nabla_m^2\phi_m^{sm}(\mathbf{m}) = \mathbf{L}^T\mathbf{L}. \tag{13b}$$

Whereas the minimum-norm and smoothing regularizers can be expressed as matrix-vector multiplications of a constant model weighting matrix $\mathbf{W}_m$ with the model vector, this is not the case for the minimum support functional (eq. 10). Instead, we need to recompute the entire weighting matrix at the nth iteration using the latest available inversion model $\mathbf{m}^{n-1}$ (Farquharson & Oldenburg 1998). We can write the first and second derivatives of eq. (10) as

$$\nabla_m\phi_m^{ms}(\mathbf{m}) = \mathbf{P}(\mathbf{m} - \mathbf{m}^{\text{ref}}), \tag{14a}$$

$$\nabla_m^2\phi_m^{ms}(\mathbf{m}) = \mathbf{P}, \tag{14b}$$

where the elements of the diagonal matrix $\mathbf{P}$ are given by

$$P_{ii} = \frac{e^2}{(|m_i^{n-1} - m_i^{\text{ref}}|^2 + e^2)^2}. \tag{15}$$

We solve the system of normal equations (eq. 11) using a preconditioned conjugate-gradient (CG) algorithm (Björck 1996). To precondition the system, we use Jacobi diagonal scaling (Newman & Boggs 2004), that is, we left-multiply eq. (11) by the inverse of

$$\mathbf{M} = \mathrm{diag}\left[Re\{\bar{\mathbf{J}}^H\bar{\mathbf{J}}\} + \beta\nabla^2\phi_m(\mathbf{m})\right]. \tag{16}$$

This preconditioner is most efficient if the Hessian matrix is diagonally dominant. Although this condition does not hold in many practical cases (Pratt *et al.* 1998), the preconditioner distributes the model update more uniformly in comparison to updates obtained without preconditioning; it prevents large updates near the sources and receivers (Ravaut *et al.* 2004).

It has been shown (e.g. Björck 1996; Aster *et al.* 2004; Hansen 2010) that eq. (11) is the system of normal equations for the linear least-squares problem

$$\begin{pmatrix} \hat{\mathbf{J}} \\ \sqrt{\beta}\mathbf{W}_m \end{pmatrix}\delta\mathbf{m} = -\begin{pmatrix} \mathbf{W}_d(\mathbf{f}(\mathbf{m}) - \hat{\mathbf{d}}^{\mathrm{obs}}) \\ \sqrt{\beta}\mathbf{W}_m(\mathbf{m} - \mathbf{m}^{\mathrm{ref}}) \end{pmatrix}, \tag{17}$$

where $\mathbf{W}_m$ is a model weighting matrix. For the minimum norm stabilizer, $\mathbf{W}_m = \mathbf{I}$ (eq. 12b), for the smoothing stabilizer $\mathbf{W}_m = \mathbf{L}$ (eq. 13b) and for the minimum support regularization, $\mathbf{W}_m = \sqrt{\mathbf{P}}$ (eq. 14b). Alternative to solving the system of normal equations, a model update can be found by solving the overdetermined system of eq. (17) using the LSQR algorithm (Ajo-Franklin *et al.* 2007; Prieux *et al.* 2012).

In eq. (17) the complex-valued Jacobian matrix and observed data vector are decomposed into (Egbert & Kelbert 2012)

$$\hat{\mathbf{J}} = \begin{pmatrix} Re\{\bar{\mathbf{J}}\} \\ Im\{\bar{\mathbf{J}}\} \end{pmatrix} \in \mathbb{R}^{2N_d\times N_m}; \quad \hat{\mathbf{d}}^{obs} = \begin{pmatrix} Re\{\mathbf{d}^{obs}\} \\ Im\{\mathbf{d}^{obs}\} \end{pmatrix} \in \mathbb{R}^{2N_d}. \tag{18}$$

The equivalence of this formulation immediately follows from

$$Re\left\{\bar{\mathbf{J}}^H\bar{\mathbf{J}}\right\} = \hat{\mathbf{J}}^T\hat{\mathbf{J}}. \tag{19}$$

From the model update $\delta\mathbf{m}^n$ obtained from system 11 or 17 at the nth iteration, a new model is derived as

$$\mathbf{m}^{n+1} = \mathbf{m}^n + \alpha\delta\mathbf{m}^n, \tag{20}$$

where the step length α controls the magnitude of the model update. To find a suitable step length, we initially test $\alpha = 1$. If this does not reduce the objective functional ϕ, we build a local quadratic approximation $\tilde{\phi}$ of the objective function from the value $\phi(\mathbf{m}^n)$ and the gradient $\nabla_m\phi(\mathbf{m}^n)$, and determine α such that $\tilde{\phi}(\mathbf{m}^n + \alpha\delta\mathbf{m}^n)$ is minimized. If the quadratic approximation fails to reduce the objective function sufficiently, we test a cubic approximation (Nocedal & Wright 1999). To allow the inversion to leave local minima, we permit an increase in the objective function if a suitable $\alpha > 0.1$ cannot be found.

3.3 Sensitivity matrix

Sensitivities relate changes in the model to corresponding changes in predicted data. In the inversion algorithm this relation is the basis for updating the model to fit the observed data. For a model vector $\mathbf{m}$ and data vector $\mathbf{d}$ we write the sensitivity matrix as

$$J_{i,j} = \frac{\partial d_i}{\partial m_j}; \quad i = 1, 2, \ldots, N_d; \quad j = 1, 2, \ldots, N_m. \tag{21}$$

We express the Jacobian in block matrix form as

$$\mathbf{J} = \begin{pmatrix} \mathbf{J}_1 \\ \vdots \\ \mathbf{J}_k \\ \vdots \\ \mathbf{J}_{N_{\mathrm{f}}N_{\mathrm{s}}} \end{pmatrix}, \tag{22}$$

with each block k representing the rows in the Jacobian for all receivers with respect to one frequency and source. To compute the Jacobian efficiently, we define for each block an operator $\mathbf{Q} \in \mathbb{R}^{N_r\times 3N_m}$ that interpolates the field values from the gridpoints to the receiver locations (Newman & Hoversten 2000; Egbert & Kelbert 2012),

$$\mathbf{f}_k(\mathbf{m}) = \mathbf{Q}\mathbf{E}^S, \tag{23}$$

where vector $\mathbf{E}^S$ is the solution of eq. (2) for one frequency and source. For the magnetic field components, $\mathbf{Q}$ includes the discretization of the $\nabla\times$ operator. Taking the derivative of eq. (23) with respect to the model parameters, we obtain the kth block of the Jacobian,

$$\mathbf{J}_k = \frac{\partial\mathbf{f}_k(\mathbf{m})}{\partial m} = \mathbf{Q}\mathbf{A}^{-1}\mathbf{G}, \tag{24}$$

where $\mathbf{G} \in \mathbb{C}^{3N_m\times N_m}$ is given by

$$\mathbf{G} = \left[\left(\frac{\partial\mathbf{b}}{\partial m_1} - \frac{\partial\mathbf{A}}{\partial m_1}\mathbf{E}^S\right), \left(\frac{\partial\mathbf{b}}{\partial m_2} - \frac{\partial\mathbf{A}}{\partial m_2}\mathbf{E}^S\right), \ldots, \left(\frac{\partial\mathbf{b}}{\partial m_{N_m}} - \frac{\partial\mathbf{A}}{\partial m_{N_m}}\mathbf{E}^S\right)\right]. \tag{25}$$

This formulation would require N_m forward solutions to compute the Jacobian for one frequency and source. Transposing eq. (24), because $\mathbf{A}$ is symmetric, we obtain

$$\mathbf{J}_k^T = \mathbf{G}^T\mathbf{A}^{-1}\mathbf{Q}^T, \tag{26}$$

and recognize that $\mathbf{J}_k^T$ can be obtained by solving only N_r forward problems, using the rows of the interpolation operator $\mathbf{Q}$ as the source terms.

To solve eqs (11) or (17) iteratively using Krylov subspace methods, different strategies for Jacobian computation may be adopted. In previous CSEM 3-D inversions, explicit computation of $\mathbf{J}$ is avoided (Newman & Boggs 2004; Plessix & Mulder 2008; Egbert 2012). Instead, the problem is reformulated such that forward solutions do not produce individual Jacobian entries, but Jacobian-vector products. A single forward solution is then sufficient to compute the contributions from all receivers for a single source and frequency (Egbert & Kelbert 2012). However, following this approach, we would have to compute new Jacobian-vector products at every iteration of the CG or LSQR algorithm used for solving eqs (11) or (17). Each CG or LSQR iteration requires evaluations of two Jacobian-vector products per frequency and source. Thus, if N_{iter} CG or LSQR iterations generate a sufficiently accurate model update, the number of forward solutions for one inversion iteration is $O(2N_{\mathrm{iter}}N_{\mathrm{f}}N_{\mathrm{s}})$. For anisotropic conductivity, the Jacobian contains derivatives with respect to each conductivity component, such that this number of forward solutions needs to be multiplied by the number of independent conductivity components (Newman *et al.* 2010). In inversions we have run, typical numbers for N_{iter} range between several tens to several hundreds.

Table 1. Complexity of the different formulations to calculate model update at each inversion iteration.

	Memory complexity	Time complexity
Formulation without storing Jacobian explicitly	$O[N_f\,(3N)^{1.5}]$	$O(2N_{\text{iter}}\,N_f\,N_s)$
Calculation of the full Jacobian	$N_f\,N_s\,N_r\,N_m + O[(3N)^{1.5}]$	$O(N_f\,N_s\,N_r)$

Alternatively, the Jacobian may be computed explicitly. In this case, $O(N_\text{f}N_\text{s}N_\text{r})$ forward solutions need to be evaluated (independent of model anisotropy) before starting the CG or LSQR iterations. Thus, for typical CSEM surveys, the numbers of forward calculations required by either strategy are of similar order. For land-based surveys with limited numbers of receivers, explicit Jacobian computation may well require fewer forward computations than the implicit approach. Marine surveys typically include more source and receiver locations than land surveys. Accordingly, it is likely that $N_\text{r} > 2N_\text{iter}$, such that the explicit approach requires more forward solutions than the implicit one. Nevertheless, if the forward solutions are computed using a direct solver, both approaches require N_f system matrix factorizations. Because the cost per solution is low once the factorization is completed, the additional computational effort for explicit Jacobian computation is limited.

Neither of the strategies is clearly favourable in terms of runtime, but the two approaches incur different memory costs if a direct solver is used. If explicit Jacobian computation is avoided, we would have to store N_f factorizations, requiring to allocate memory for at least $O[N_f\,(3N)^{1.5}]$ double-precision complex numbers. In contrast, the memory complexity of computing and storing the full Jacobian is $N_\text{f}N_\text{s}N_\text{r}N_m$, plus the memory required for storing one factorization. For the practical surveys we consider ($N_\text{d} < 10^5$), storing the Jacobian requires less memory than storing the factorizations for multiple frequencies. This implies that, if forward modelling using a direct solver is feasible on a given computer platform, storing the full Jacobian in memory should also be feasible. Table 1 summarizes the comparison between explicit and implicit Jacobian computation. To reduce the space required for storing the Jacobian, approaches such as those presented by Li & Oldenburg (2003); Cox *et al.* (2010); Li *et al.* (2011) may be used.

The availability of the full Jacobian is also useful for model resolution (Alumbaugh & Newman 2000; Fedi *et al.* 2005) and detailed sensitivity analyses. Analysis of linear dependencies in $\mathbf{J}$ can be used to detect redundant data (Björck 1996; Hansen 1998), and we can investigate data and model weighting schemes based on the sensitivity distribution, as was shown in Weitemeyer *et al.* (2010) and Prieux *et al.* (2012) for 2-D inversions. Furthermore, the preconditioner (eq. 16) is trivial to calculate from the full Jacobian without the necessity to form the approximate Hessian explicitly or resort to a 1D approximation, as described by Newman & Boggs (2004).

3.4 Regularization

3.4.1 Effects of regularization on the normal equations and least-squares systems

The system of normal eqations (eq. 11) is typically ill-conditioned. For 3-D inverse problems we usually have $2N_\text{d} < N_m$. Accordingly, the product $\hat{\mathbf{J}}^T\hat{\mathbf{J}}$ occurring in the left hand side of eq. (11) is a singular matrix with a null space of dimension at least $N_m - 2N_\text{d}$. We can show this using singular value decomposition of $\hat{\mathbf{J}} \in \mathbb{R}^{2N_\text{d}\times N_m}$, $2N_\text{d} < N_m$,

$$\hat{\mathbf{J}} = \mathbf{USV}^T; \quad \mathbf{U} \in \mathbb{R}^{2N_d\times 2N_d},\ \mathbf{S} \in \mathbb{R}^{2N_\text{d}\times N_m},\ \mathbf{V} \in \mathbb{R}^{N_m\times N_m}. \tag{27}$$

The singular values $s_i\,(i = 1, \ldots, 2N_\text{d})$ are placed on the main diagonal of $\mathbf{S}$ and comprise the spectrum of $\hat{\mathbf{J}}$. Matrices $\mathbf{U}$ and $\mathbf{V}$ represent left and right singular vectors, respectively, and are orthonormal (i.e. $\mathbf{U}^T\mathbf{U} = \mathbf{I}_{\mathbf{2N_d}}$ and $\mathbf{V}^T\mathbf{V} = \mathbf{I}_{\mathbf{N_m}}$; Golub & van Loan 1996). Therefore,

$$\hat{\mathbf{J}}^T\hat{\mathbf{J}} = \left(\mathbf{USV}^T\right)^T\mathbf{USV}^T = \mathbf{VS}^T\mathbf{SV}^T, \tag{28}$$

and

$$\text{diag}\left(\mathbf{S}^T\mathbf{S}\right) = \left(s_1^2 \geq s_2^2 \geq \ldots \geq s_{2N_\text{d}}^2 \geq 0_{2N_d+1} = \ldots = 0_{N_m}\right). \tag{29}$$

Because the condition number of a matrix can be expressed as the ratio of the largest to the smallest singular value, we obtain $\text{cond}(\hat{\mathbf{J}}^T\hat{\mathbf{J}}) = \infty$, and $\hat{\mathbf{J}}^T\hat{\mathbf{J}}$ is singular. The last $N_m - 2N_\text{d}$ singular vectors span the null space of the product in eq. (28).

By introducing the simplest regularization in the form of a damping factor (i.e. the minimum norm regularization described in eq. 7), we obtain

$$\hat{\mathbf{J}}^T\hat{\mathbf{J}} + \beta\mathbf{I} = \mathbf{VS}^T\mathbf{SV}^T + \beta\mathbf{VV}^T = \mathbf{V}(\mathbf{S}^T\mathbf{S} + \beta\mathbf{I})\mathbf{V}^T, \tag{30}$$

and

$$\text{cond}(\hat{\mathbf{J}}^T\hat{\mathbf{J}} + \beta\mathbf{I}) = \frac{s_1^2 + \beta}{\beta}. \tag{31}$$

Eq. (30) shows that the regularization acts as a filter suppressing the contributions of singular values smaller than β and corresponding singular vectors to the model update vector (Hansen 2010). For more complex stabilizing functionals, eq. (30) will not hold exactly. Nevertheless, similar spectral analysis can be carried out by transforming the problem into a standard form (Hansen 1998), revealing that the smoothing and minimum support regularizers (eqs 8 and 10) filter the singular values in a similar manner as shown in eq. (30).

In exact arithmetic, the solutions of the system of normal equations (eq. 11) and the linear least-squares problem (eq. 17) are identical (Paige & Saunders 1982). For finite precision computations, though, eq. (17) may be advantageous, because it does not require explicit formation of the singular matrix $\hat{\mathbf{J}}^T\hat{\mathbf{J}}$ (eq. 28). Similar to the above analysis of the normal equations, we can show the effect of regularization for the linear least-squares problem. If $\mathbf{W}_m = \mathbf{I}$, the singular values of the matrix on the left-hand side in eq. (17) are (Björck 1996)

$$\hat{s}_i = \begin{cases} \sqrt{{s_i}^2 + \beta} & \text{if} \quad i \leq 2N_\text{d} \\ \sqrt{\beta} & \text{if} \quad i > 2N_\text{d} \end{cases}; \quad i = 1, \ldots, N_m, \tag{32}$$

where s_i are the singular values of $\hat{\mathbf{J}}$. Qualitatively similar relations hold for $\mathbf{W}_m \neq \mathbf{I}$ (Hansen 1998).

In addition to the regularization in the form of a stabilizing functional, the iterative methods used for solving eqs (11) and (17) have an implicit regularization effect. The iterative process in the CG or LSQR algorithm successively builds Krylov subspaces (Egbert 2012). Regularization occurs because the Krylov subspace obtained at the kth iteration is an approximation to the subspace spanned by the first k right singular vectors. Because these singular vectors generally represent the most relevant information (Jupp & Vozoff

1975; Tompkins *et al.* 2011), the problem is additionally regularized by the iterative process, and the amount of regularization is controlled by the number of iterations N_{iter} used to obtain model updates (Hansen 1998). With the Jacobian computed explicitly, additional CG or LSQR iterations do not require additional forward solutions. Therefore, the influence of this implicit regularization becomes inexpensive to investigate.

3.4.2 Choice of the regularization parameter

Denote $\mathbf{R}$ a regularization matrix, a discrete representation of $\nabla_m^2\phi_m(\mathbf{m})$ and $\mathbf{H}^d = Re\{\tilde{\mathbf{J}}^H\tilde{\mathbf{J}}\}$ the non-regularized part of the approximate Hessian matrix. In practice, the absolute values of the elements in $\mathbf{H}^d$ and $\mathbf{R}$ may differ by many orders of magnitude. Therefore, a reasonable range of regularization parameter values is difficult to locate. The regularization parameter has to be sufficiently large to ensure stabilization of the inverse problem and consideration of the *a priori* information imposed by the regularization term, and sufficiently small to minimize bias in the solution (Hansen 2010). To balance this trade-off, we assume that if

$$||\mathbf{H}^d||_p \approx ||\beta\mathbf{R}||_p \tag{33}$$

holds for any matrix norm (e.g. $p = 1, 2, \infty$), the amplitude levels of the matrix entries and the upper bounds of their spectra are comparable. To ensure that condition 33 is fulfilled, we choose the scaling factor

$$\beta_{L_p} = \frac{||\mathbf{H}^d||_p}{||\mathbf{R}||_p}. \tag{34}$$

For $p = 2$, the L_2 norm of any matrix $\mathbf{A}$ is given by (Golub & van Loan 1996)

$$||\mathbf{A}||_2 = s_{\max}(\mathbf{A}), \tag{35}$$

where $s_{\max}$ is the largest singular value of $\mathbf{A}$. Therefore, using the scaling of eq. (34) with the L_2-norm equalizes the largest singular values of the Hessian and regularization terms.

Because we precompute the Jacobian, and $\mathbf{R}$ is either a diagonal or very sparse matrix, estimating the L_2 norms of $\mathbf{H}^d$ and $\mathbf{R}$ requires little computational effort. At every inversion iteration, we estimate the largest singular values of $\mathbf{H}^d$ and $\mathbf{R}$ by initiating a Lanczos algorithm with a random, unit-norm RHS vector. The values $s_{\max}(\mathbf{H}^d)$ and $s_{\max}(\mathbf{R})$ are approximated from matrices produced by successive Lanczos iterations (Saad 2011). We found that less than 30 iterations are required to determine the largest singular values to an accuracy of four digits.

The regularization parameter β_{L_2} computed using eq. (34) weights the data and *a priori* information equally, and thus represents a practical upper bound for β. Furthermore, its meaning is closely related to the spectral properties of the matrices used in the inversion, which makes the parameter search systematic and eliminates the need for manual problem-dependent scaling of the regularization parameter. We have also tested infinity norms (Newman & Alumbaugh 1997), which are cheaper to compute than L_2 norms. In contrast to the L_2 norm, the L_∞ norm does not have a strict relation to the spectrum of the matrix (Golub & van Loan 1996). In our tests, L_∞-norm based scaling of the regularization matrix without further manual adjustment of the regularization level resulted in somewhat lower-quality inversion images.

For the least-squares system (eq. 17), we require analogously to eq. (34)

$$\sqrt{\beta_{L_2}} = \frac{||\hat{\mathbf{J}}||_2}{||\mathbf{W}_m||_2}. \tag{36}$$

Because the Jacobian in eq. (36) is not square, a Lanczos algorithm cannot directly be applied for computing its largest singular value. However, because the singular values of the square matrix $\mathbf{H}^d$ and the Jacobian are related through eq. (29), we compute the L_2 norm of the Jacobian as

$$||\hat{\mathbf{J}}||_2 = \sqrt{||\mathbf{H}^d||_2}. \tag{37}$$

The actual regularization parameter β we use in the inversion is computed at each iteration as

$$\beta = \frac{\gamma\beta_{L_2}}{n_{\text{iter}}^c}, \tag{38}$$

where $\gamma \in [0, 1]$ is user-defined. The division by n_{iter}^c implements a cooling approach (Haber *et al.* 2000). By choosing $c > 0$, we let the regularization parameter gradually decrease during the inversion. Accordingly, singular vectors that correspond to smaller singular values gradually gain influence on the solution.

3.5 Conductivity bounds

To prevent the inversion from finding physically unreasonable conductivity values, we apply a bounded logarithmic transform (Commer & Newman 2008; Abubakar *et al.* 2008; Kim & Kim 2011) to the model parameters. This also allows us to include efficiently *a priori* knowledge about the lower and upper conductivity bounds. We enforce lower and upper bounds a and b by applying the transformation

$$x_k = \frac{1}{p} ln\left(\frac{m_k - a_k}{b_k - m_k}\right); \quad a_k < m_k < b_k, \tag{39}$$

where subscript k denotes the kth model parameter and will be omitted for brevity. The positive constant p controls the steepness of the transformed space, generally becoming steeper as p increases (Kim & Kim 2011). To minimize bias in the inversion, we mostly keep a and b constant throughout the model. The original model parameters (i.e. conductivity) are related to the transformed ones by

$$m = \frac{a + b\exp(px)}{1 + \exp(px)}; \quad -\infty < x < \infty. \tag{40}$$

To carry out the inversion in the transformed space, we recast the Jacobian in terms of the transformed variables (Commer & Newman 2008) using

$$\mathbf{J}' = \frac{\partial\mathbf{d}}{\partial\mathbf{x}} = \frac{\partial\mathbf{d}}{\partial\mathbf{m}}\frac{\partial\mathbf{m}}{\partial\mathbf{x}}, \tag{41}$$

where $\mathbf{J}'$ is the transformed Jacobian and

$$\frac{\partial m}{\partial x} = \frac{p(b - m)(m - a)}{b - a}. \tag{42}$$

The model vectors $\mathbf{m}$ and $\mathbf{m}^{ref}$ are transformed using eq. (39) and substituted into eq. (11) or (17). The regularization is then directly applied to the transformed quantities.

3.6 Model updates at source locations

Separation of the EM field into primary and secondary fields allows us to remove source singularities from the numerical formulation of the forward problem (see eq. 1). Secondary sources are only present where $(\sigma^* - \sigma^{P*}) \neq 0$; at the source locations, the 3-D and background conductivities σ^* and σ^{P*} are equal. If a source extends laterally over more than one cell that needs to be updated during the inversion, we introduce singularities there, because lateral

conductivity variation cannot be captured by a 1-D background model. This case is usually avoided for marine or airborne sources that are located in regions which are fixed during the inversion. In contrast, for land surveys with kilometre-long galvanic grounded sources, this problem may seriously deteriorate inversion results and cause severe near-surface artefacts. To prevent such artefacts, we include into our conductivity models one layer with small z extent at the air-ground interface. The sources are located in this thin layer. After each model update we assign the average conductivity over this slice to the corresponding layer in the background and 3-D models. A similar approach was shown to be effective in seismic inversion (Ravaut *et al.* 2004).

4 IMPLEMENTATION

We have implemented the direct-solver-based inversion in a parallel, fully distributed fashion. Good memory scalability is achieved by having each of the N_{proc} processes used store only approximately the $1/N_{\text{proc}}$th part of any matrix or vector required by the algorithm. We use the PETSc library (Balay *et al.* 2012) for distributed linear algebra operations. For solving eq. (2), the MUMPS parallel distributed direct forward solver (Amestoy *et al.* 2006) is accessed through its PETSc interface. The Hessian matrix is not formed explicitly. Instead, the required Hessian-vector products are calculated by repeated computation of matrix-vector products.

To take further advantage of distributed platforms, we apply a domain decomposition approach for distributing the computation of the primary fields $\mathbf{E}^P$ required in eq. (1). Primary fields in multilayered 1D media may be time consuming to compute, particularly for realistic long-wire sources (Streich & Becken 2011) due to the necessity of integrating contributions along the source wire. Because parallel primary field computation requires virtually no communication, parallelizing this task is straightforward and results in significant speedup.

We have further parallelized the code over frequencies, because solutions for different frequencies can be obtained independently. When inverting multifrequency data, we may either have all processes work jointly on a single frequency at a time, or form groups of processes that work on different frequencies simultaneously. Fig. 1 demonstrates that runtimes are significantly shorter for simultaneous factorizations. This is expected, because the scalability of direct solvers is limited by their relatively low computation-to-communication ratio (Pardo *et al.* 2012). Accordingly, provided that sufficient memory is available, doing several factorizations in parallel is preferable to consecutive factorizations.

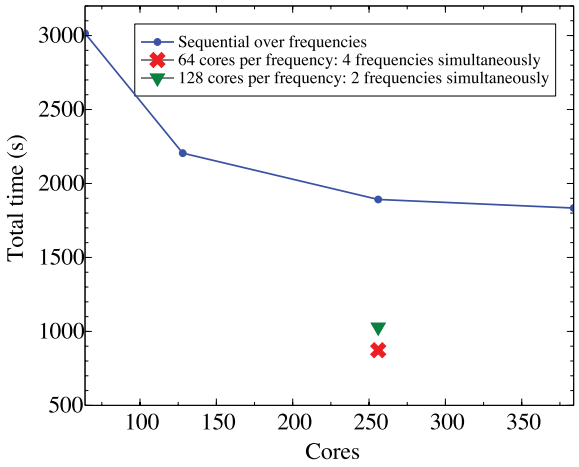

Figure 1. Scalability of the MUMPS direct solver for modelling four frequencies in sequence, and simultaneous modelling of different frequencies for a model of size 70 × 70 × 70 (i.e. the size of the system (2) is $\approx 10^6$). The total time includes the time spent on factorization and solutions for 270 RHS.

5 NUMERICAL EXPERIMENTS

We consider an 8 × 8 × 3-km model of a homogeneous half-space of 5 Ωm containing three anomalous objects (Fig. 2). A shallow 50-Ωm box with dimensions 1.2 × 1.2 × 0.08 km is embedded with its top at a depth of 0.3 km. Two larger objects of size 2.0 × 1.2 × 0.8 km are placed at a depth of 1.0–1.8 km. An air layer of conductivity 10^{-9} S m^{-1} is present at the top of the model. We generate observed data by forward modelling responses on a fine grid of 80 × 80 × 50 cells. To minimize boundary effects, we append several boundary cells at each side, growing in size at a stretching factor of 1.5, such that the grid contains 100 × 100 × 65 cells in total.

For inversion, the model is discretized into 40 × 40 × 40 cells with a uniform 200-m cell spacing in the horizontal directions and increasing cell sizes in the vertical direction. Again, boundary cells are appended. The inversion domain is limited to the subsurface region, excluding boundary cells and air, resulting in 54 400 unknown model parameters. The starting model for all examples is a homogeneous half-space. Sixteen sources and 64 receivers are deployed at the surface as shown in Fig. 2. We invert the E_x component for source-receiver distances larger than 500 m at periods of 32, 8 and 1 s, resulting in 2880 complex data values. Two percent random uniform noise is added to the data. Resistivity values throughout the model are constrained such that $0.5 < \rho_j < 1000\,\Omega$m; $j = 1, \ldots, N_m$. Within each inversion iteration, the number of CG or LSQR iterations used to solve either eq. (11) or (17) is 200 unless otherwise stated. For the regularization parameter (eq. 38), we choose $c = 1$. The mean percentage error is calculated using

$$\delta = \frac{1}{N_{\text{d}}} \sum_{i=1}^{N_{\text{d}}} \frac{|f(\mathbf{m})_i - d_i^{\text{obs}}|}{|d_i^{\text{obs}}|} \times 100. \tag{43}$$

Inversion results obtained using the approaches described by eqs (11) and (17) with smoothing regularization (eq. 8), with a scaling

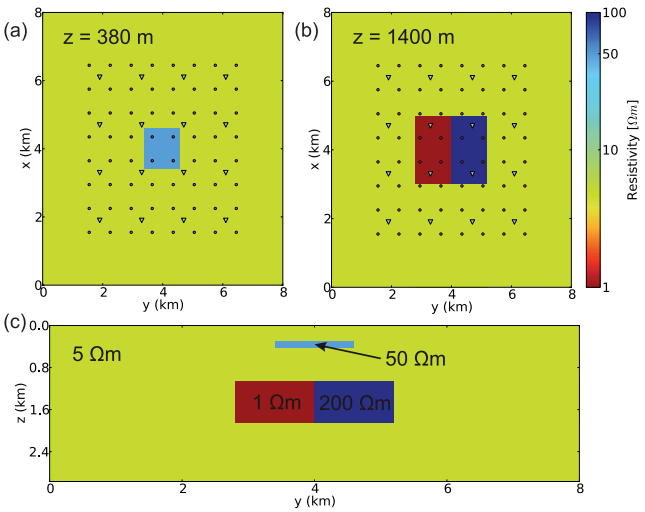

Figure 2. Plan views at (a) $z = 380$ m and (b) $z = 1400$ m, and (c) section through the centre of the model used for demonstrating the inversion. Circles indicate receivers and triangles indicate transmitters (located at the surface).

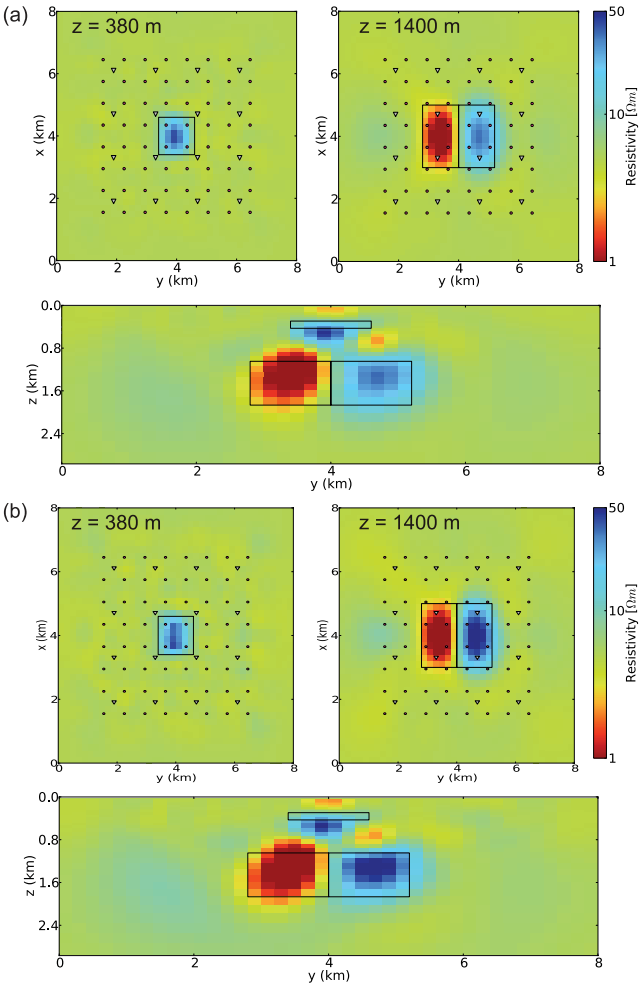

Figure 3. Inversion results obtained by solving (a) the system of normal equations (eq. 11) and (b) the linear least-squares problem (eq. 17), using smoothing regularization. Triangles and circles indicate transmitters and receivers, respectively. The true locations of the anomalous objects are outlined.

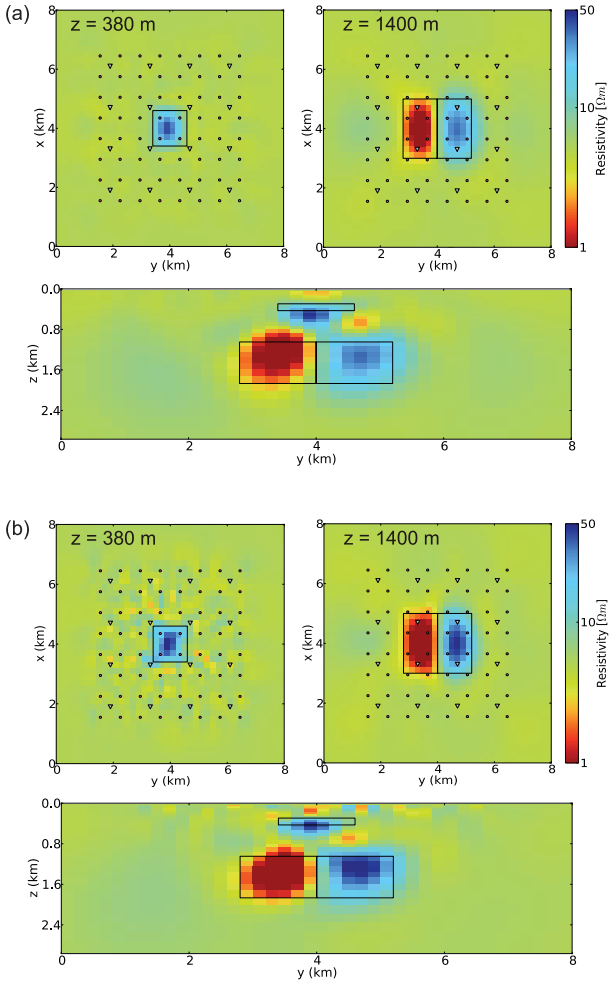

Figure 4. Inversion results obtained by solving (a) the system of normal equations (eq. 11) and (b) the linear least-squares problem (eq. 17), using minimum norm regularization. Triangles and circles indicate transmitters and receivers, respectively. The true locations of the anomalous objects are outlined.

factor $\gamma = 0.1$ (eq. 38) are shown in Fig. 3. The computations of the images displayed in Figs 3(a) and (b) only differ in the equation used for computing model updates (eq. 11 and 17, respectively), all other parameters are identical. Both inversions resolve the horizontal positions and sizes of the objects similarly well, and somewhat overestimate the depth of the thin resistive body. The LSQR minimization recovers the resistivity of the deep resistor better (Fig. 3b), yet also produces slightly stronger artefacts than the GN minimization. These artefacts likely result from overfitting data points at which noise exceeds target responses.

In Fig. 4, we display inversion results for the minimum norm stabilizer (eq. 7). Here, most rapid convergence and best resolution of the anomalous objects was achieved by using a scaling factor $\gamma = 10^{-4}$ (eq. 38). This implies that at the nth iteration we effectively permit contributions from singular values as small as $1/(10^4 n_{\text{iter}})$ times the largest singular value of the Hessian [or $1/(100 n_{\text{iter}})$ for the Jacobian]; we thus permit high-order, typically small-scale features in the images. Accordingly, the results are somewhat noisier than those obtained using a higher level of smoothing regularization (compare Figs 3 and 4). The depth of the shallow resistor is better resolved for the minimum norm regularization, whereas the images of the two large blocks are similar for both stabilizers. Again, the LSQR image resolves the resistors better than the one obtained from the normal equations, but it also is slightly more noisy. The different strengths of local oscillations in the GN and LSQR images are likely related to properties of the iterative solution algorithms; this requires further investigation.

Inversion results for the minimum support stabilizer (eq. 10 with parameter $e = 0.4$) are shown in Fig. 5. As for the minimum norm regularization, we used a scaling factor $\gamma = 10^{-4}$. Clearly, the images are focused, and the recovered objects have sharp edges that coincide well with the boundaries of the true anomalies, although the recovered objects are somewhat smaller. This behaviour is expected, because the minimum support stabilizer minimizes the volume in which conductivity deviates from that of the initial model and thus inherently prefers compact images (Minsley 2007). Notably, the resistivity of the deep resistor is significantly better recovered than for the other stabilizers, with maximum values of 243 and 251 Ωm for the normal equation and LSQR minimizations, respectively.

Convergence rates for the two minimization approaches and all stabilizing functionals are shown in Fig. 6. Remarkably, all inversions require very few iterations to achieve a target misfit of 1.5 per cent, demonstrating the excellent convergence properties of the Gauss–Newton approach. Although convergence rates are

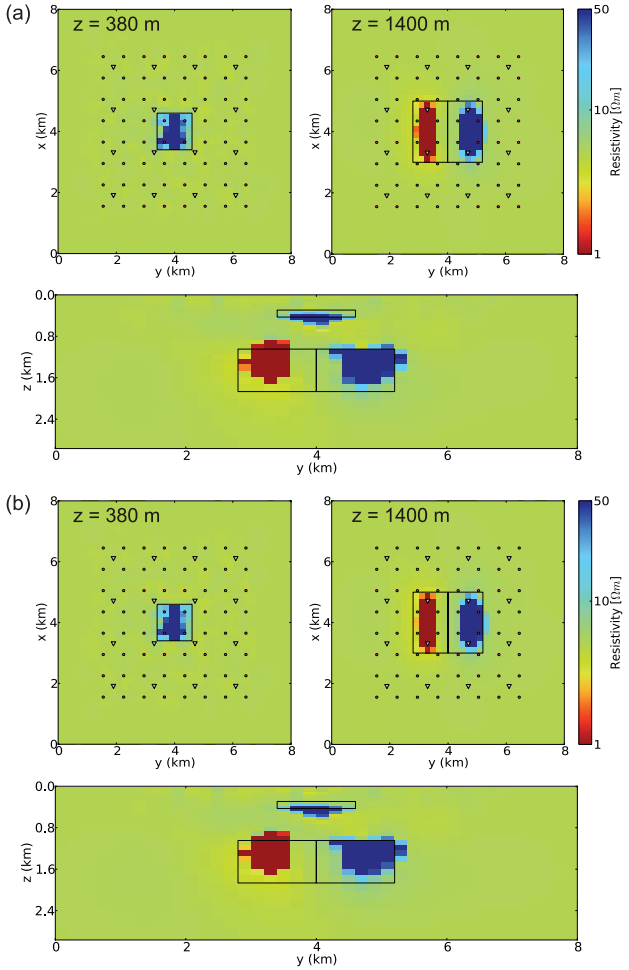

Figure 5. Inversion results obtained by solving (a) the system of normal equations (eq. 11) and (b) the linear least-squares problem (eq. 17), using minimum support regularization. Triangles and circles indicate transmitters and receivers, respectively. The true locations of the anomalous objects are outlined.

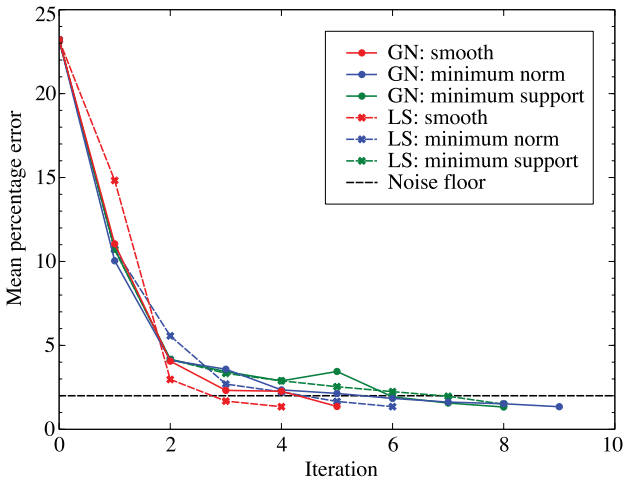

Figure 6. Mean percentage error versus iteration for the the Gauss–Newton scheme (solid lines) and linear least-squares problem (dashed lines). Different colours correspond to the three stabilizers. The black dashed line shows the noise floor.

similar for the LSQR and normal equation minimizations, the LSQR minimization reaches the target misfit a few iterations earlier for the smoothing and minimum norm stabilizers. The situation is nearly opposite for the minimum support stabilizer. This may be related to the smoothing and minimum norm stabilizers (eqs 7 and 8) being quadratic functionals and thus being particularly well-suited for combination with the LSQR solver (Björck 1996), whereas the minimum support regularizer (eq. 10) is non-quadratic.

Whereas the total misfit shown in Fig. 6 is useful for analysing overall convergence, it does not provide information on data fit for individual receivers. To facilitate analysing the data fit distribution, we display in Fig. 7 observed fields calculated for the model shown in Fig. 2, and examples of the starting and final data mismatches for normal-equation minimization with smoothing regularization. For the frequency and subset of the sources displayed, the final misfit is less than the target level at nearly all receivers, including those within 0.5 km distance from the sources that were excluded from the inversion. A histogram of the initial and final amplitude fit (Fig. 8) indicates that the data for all other sources and frequencies were fit comparably well. Final misfits for the other combinations of minimization approaches and stabilizers are virtually identical to those shown in Figs 7 and 8.

To demonstrate the effect of implicit regularization related to the number of iterations used for the approximate solution of eq. (11) or (17) at each inversion iteration, we compare in Fig. 9 inversion models obtained without explicit regularization (parameter $\beta = 0$), and with increasing numbers of CG or LSQR iterations. We used the same noise-contaminated data as for the previous tests. Clearly, the images become noisier as the number of iterations increases. This can be expected, because with increasing number of iterations, smaller singular values, which are likely related to small-scale oscillatory features in the model (Hansen 1998), increasingly influence the solution.

Indications of small-scale features should not be contained in the observed data, unless they are artificially introduced by adding noise. However, the system of normal equations also is inherently ill-conditioned (see eq. 29), such that artefacts may appear even in images of ideal data. To separately analyse the effects of ill-conditioning of the normal equation system, and of noise present in the data, we display in Figs 10(a)–(d) inversion results obtained from noise-free data. For increasing numbers of iterations used in solving the system of normal equations, these images appear increasingly noisy, yet the artefacts are not as strong as for the noisy data (compare Figs 9 a–d and 10 a–d).

For comparison, we show in Figs 10(e)–(h) inversion results for noisy data and explicit minimum-norm regularization. For this case, increasing the number of iterations beyond ~ 300 does not make the image noisier. Yet, we also do not recover any additional information once the number of iterations used for solving the system of normal equations has reached a level at which the corresponding singular values are dominated by regularization effects. As indicated by eqs (31) and (38), all singular values smaller than $s_{\max}(\hat{\mathbf{J}}^T\hat{\mathbf{J}})/(\gamma n_{\text{iter}})$ are effectively filtered at the n*th* iteration; for this example we used $\gamma = 10^{-4}$.

Fig. 11 shows the spectrum of the weighted Jacobian matrix $\hat{\mathbf{J}}$ and the product $\hat{\mathbf{J}}^T\hat{\mathbf{J}}$ calculated for the starting model used in the inversions. Singular values from the 345*th* one are smaller than $10^{-4}s_{\max}(\hat{\mathbf{J}}^T\hat{\mathbf{J}})$, such that their contribution to the images shown in Figs 10(e)–(h) should be suppressed. This agrees well with the observation that using more than ~ 300 CG iterations does not add detail to the image. In accordance with eqs (27) and (29), the spectrum of $\hat{\mathbf{J}}^T\hat{\mathbf{J}}$ is spread over a much wider range of values than

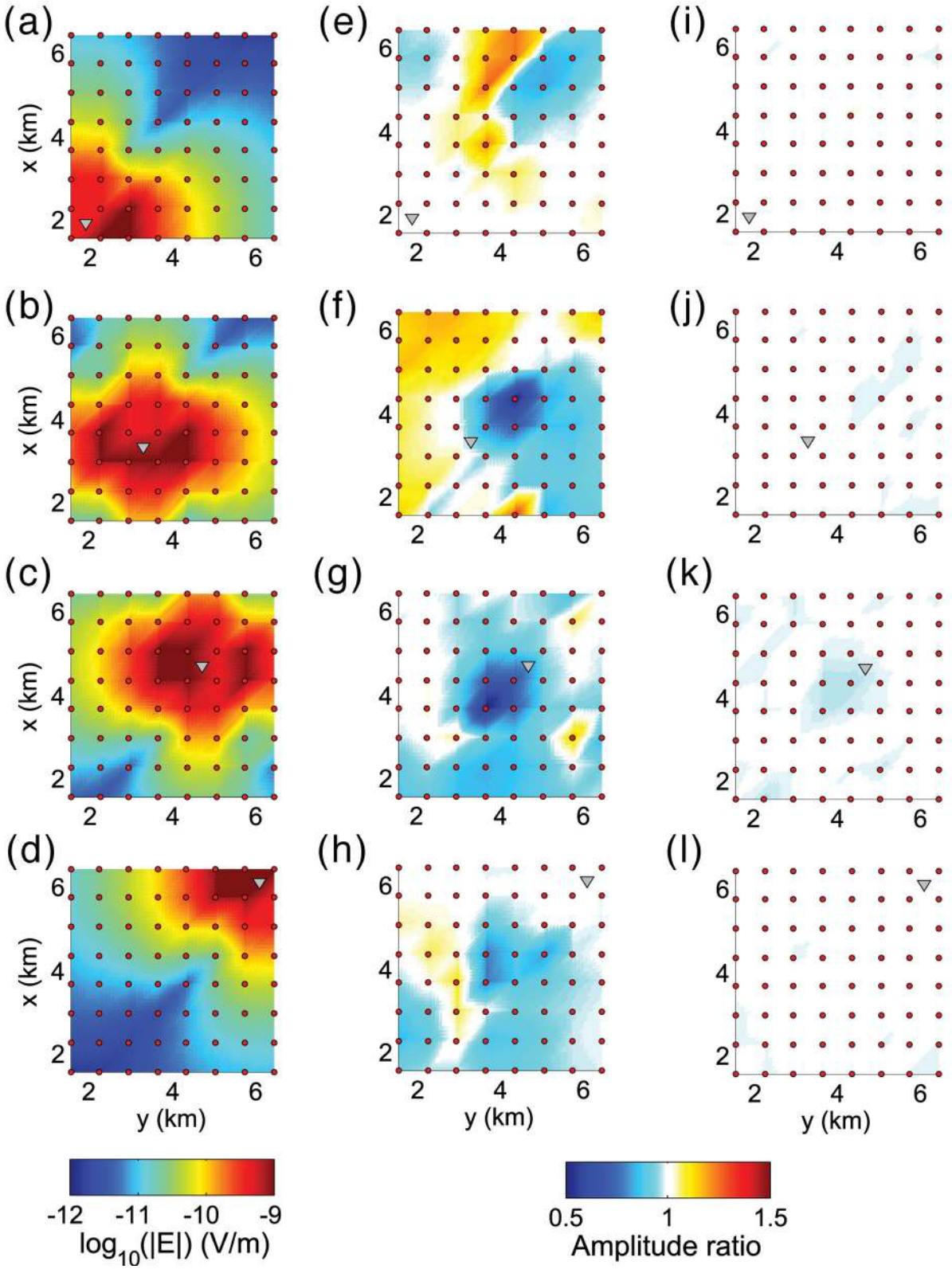

Figure 7. Amplitudes of the observed field at a period of 32 s (a–d), amplitude ratios between data for the starting model and observed data (e–h) and amplitude ratios between data for the final model obtained from the normal equations system with smoothing regularization and observed data (i–l), for four sources indicated by triangles and receivers indicated by circles.

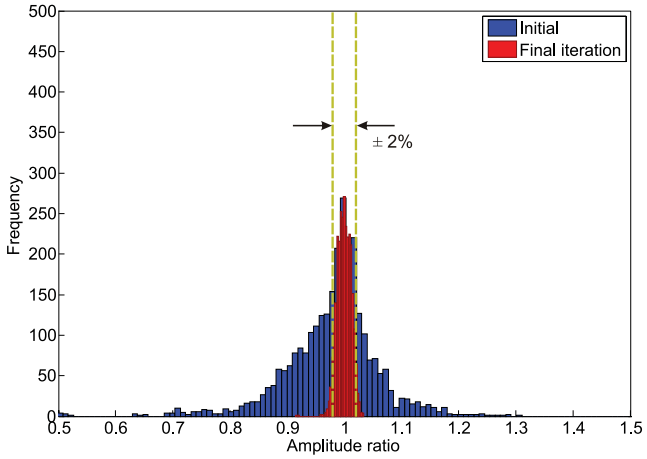

Figure 8. Histogram of the amplitude ratios of all data included in the normal-equation minimization with smoothing regularization, for the initial (blue) and final (red) iterations. Dashed lines indicate the 2 per cent interval.

that of $\hat{\mathbf{J}}$. Nevertheless, somewhat surprisingly, Figs 3–5 demonstrate that we obtain similar inversion results for both formulations, without degradation of numerical stability for the normal equation minimization that requires $\hat{\mathbf{J}}^T\hat{\mathbf{J}}$, as long as we use equivalent regularizations.

5.1 Inversion of a refined model

To demonstrate the capability of the direct-solver-based inversion algorithm to handle problems with relatively large numbers of unknowns, and its scalability when run in parallel, we present an inversion of a more finely gridded model. The model shown in Fig. 12 is discretized into 80 × 80 × 50 cells. With additional boundary cells appended at each side, this results in 96 x 96 x 65 ≈ 1.8 million unknowns in the system 2. The inversion domain includes the subsurface region without padding cells and air, and consists of 80 × 80 × 44 cells. To justify the increased number of inversion unknowns and the associated attempt at achieving higher resolution (Alumbaugh & Newman 2000; Fedi *et al.* 2005), we adjust the survey layout. We use 113 receivers for this experiment, resulting in 5184 complex data values. The larger number of receivers also increases the memory required for storing the Jacobian, thus serving to demonstrate the memory scalability of our code. The inversion results for this model agree very well with those obtained for the coarser grid (compare Figs 3 and 12). The vertical position of the shallow resistor is better resolved by the fine-grid inversion, which is probably due to the denser receiver spacing.

5.2 Performance analysis

All inversions were run on a distributed platform consisting of four interconnected nodes, each equipped with two twelve-core AMD Opteron 2.2 GHz CPUs and 64 GB of RAM. Systems of similar or larger size probably are widely available now. We used 16 and 64 processes for the coarse and fine-grid inversions, respectively. In each case, the processes were distributed uniformly across the four nodes.

Table 2 summarizes information about the time and memory requirements for inverting the coarse and fine-grid models. For both models, the total memory required is well below the amount available on our platform. As anticipated, the system matrix factorization

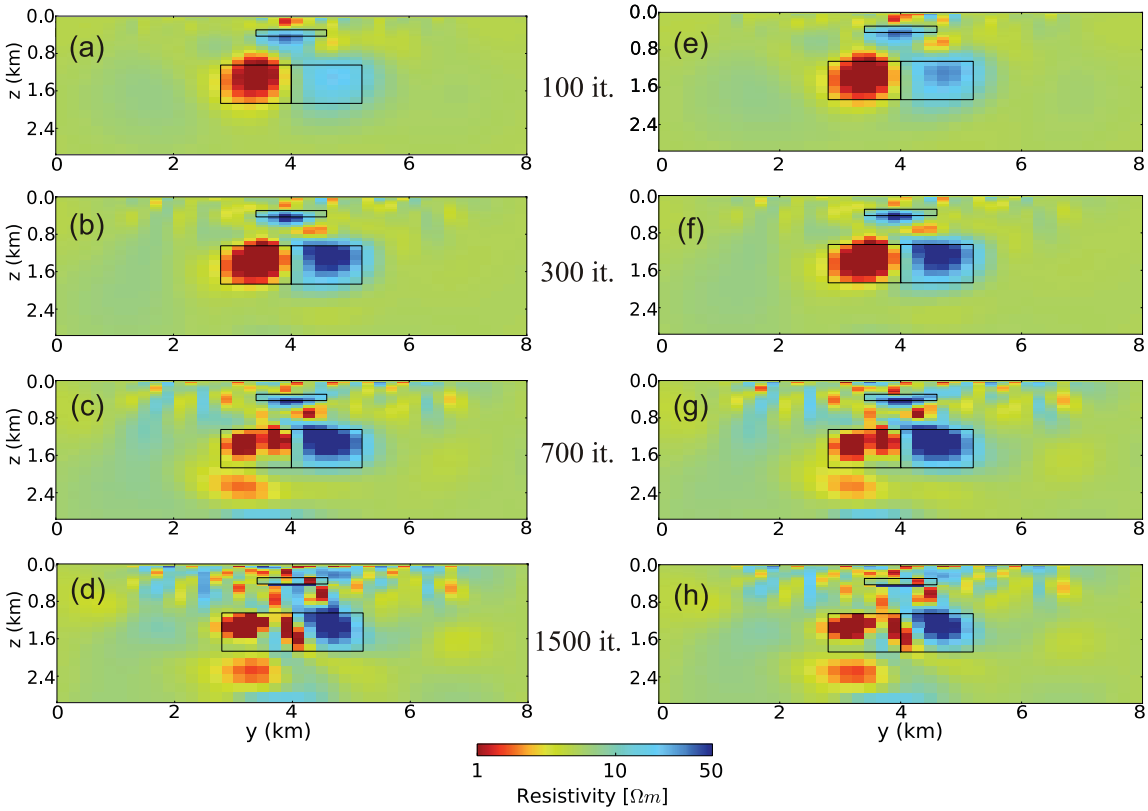

Figure 9. Results of inverting noise-contaminated data using increasing numbers of iterations when solving (a–d) eq. (11) and (e–h) eq. (17) without explicit regularization ($\beta = 0$).

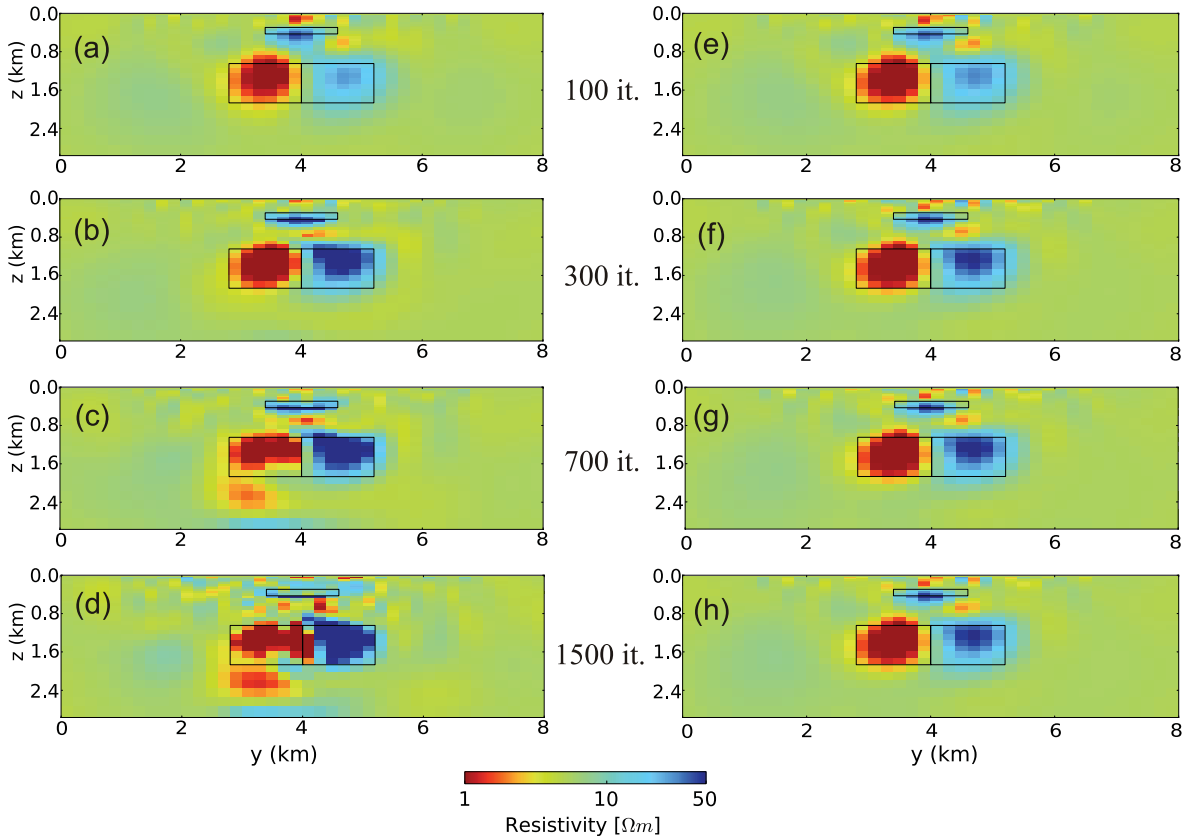

Figure 10. (a–d) Results of inverting noise-free data without explicit regularization ($\beta = 0$), using increasing numbers of iterations for solving eq. (11). (e–h) Inversion results for noisy data and minimum norm regularization with ($\gamma = 10^{-4}$ in eq. 38).

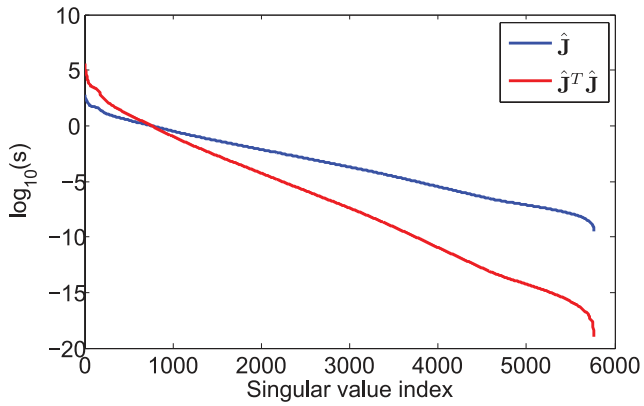

Figure 11. Singular values of the weighted Jacobian matrix $\hat{\mathbf{J}}$ and the product $\hat{\mathbf{J}}^T\hat{\mathbf{J}}$ calculated at the starting iteration. For the product $\hat{\mathbf{J}}^T\hat{\mathbf{J}}$, only the non-zero singular values are shown.

requires the largest amount of memory. To limit memory consumption, we did not run factorizations for multiple frequencies simultaneously. The Jacobian matrix and additional vectors only occupied ≈10 and 17 per cent of the total memory for the coarse and finely gridded models, respectively. The time spent on forward modelling, which includes matrix factorization and solutions of forward problems for computing the EM fields and Jacobian, comprises more than 90 per cent of the total time required for the inversion.

6 CONCLUSIONS

We have described a new implementation of a parallel 3-D inversion scheme, in which the forward modelling engine uses a direct solver. The high memory demand of the direct solver becomes manageable by utilizing modern distributed-memory platforms. To limit the computational effort necessary when using direct forward solvers, Gauss–Newton-type inversion strategies that require relatively few model updates are preferable. Using a direct forward solver, the Jacobian can be readily computed, and computing it explicitly provides strong benefits for the inversion. It makes the computation of an adequate preconditioner for the system of normal equations simple, and enables us to compute a reasonable starting regularization parameter based on the norms of the matrices involved in the inversion. In addition, explicit Jacobian computation makes the computational effort required for iterative solution of the model update equations nearly independent of the number of iterations. This has allowed us to investigate the influence of the number of iterations on the inversion results.

For computing model updates at each iteration, we have compared the conjugate-gradient solution of the system of normal equations and the LSQR solution of an equivalent non-square system of equations. For the models studied, both minimization approaches converge rapidly and provide qualitatively similar results. The LSQR approach resolves resistive bodies somewhat better, yet also produces slightly stronger artefacts. This requires further investigation; in theory, the LSQR system is better conditioned and should produce less noisy images.

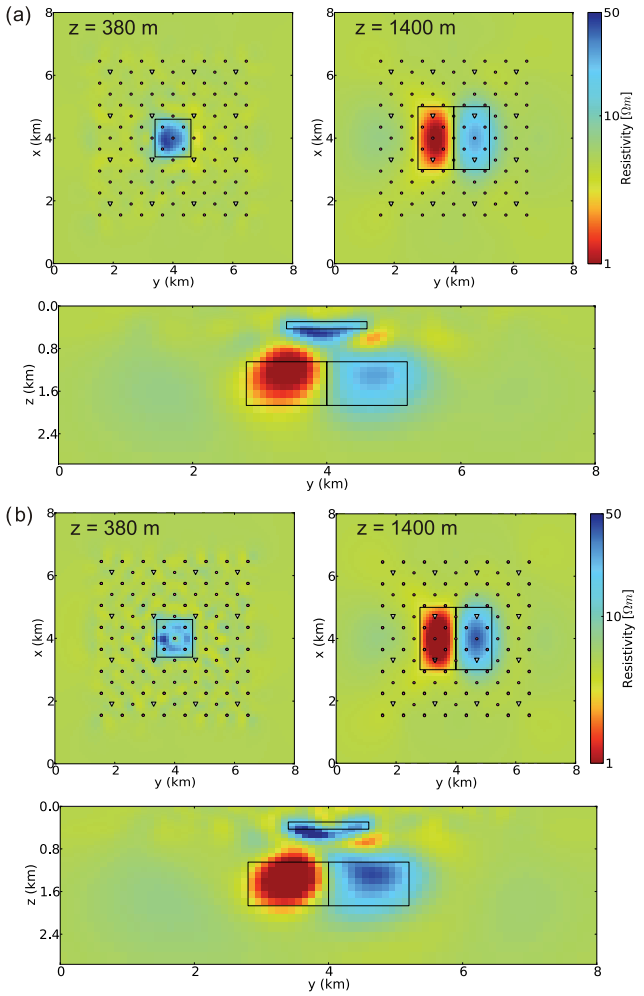

Figure 12. Inversion results obtained for refined model by solving (a) the system of normal equations (eq. 11) and (b) the linear least-squares problem (eq. 17). Smoothing regularization was used. Triangles and circles indicate transmitters and receivers, respectively. The true anomalous objects are outlined.

Table 2. Computation times and memory consumption for the case studies presented.

	Coarse grid	Fine grid
Grid size	56 × 56 × 55	96 × 96 × 65
Number of unknowns in inversion	54 400	281 600
Number of cores used	16	64
Time per factorization	00:04:10	00:15:15
Time per forward solution	00:00:01	00:00:02,5
Time per inversion iteration	01:05:00	04:20:00
Total memory usage in GB	26	152
For factorization	23.2	126.5
For Jacobian matrix	2.3	22.2

The smoothing, minimum norm and minimum support regularization operators all provide high-quality images. However, the minimum norm and minimum support operators require significantly lower regularization levels than the smoothing stabilizer. This is likely related to the distribution of singular values for the different stabilizers. For our synthetic blocky models, the minimum support stabilizer performs best in recovering absolute resistivity values of resistive objects.

In addition to explicit regularization imposed by adding regularization operators to the objective function, implicit regularization due to computing incomplete solutions of the normal equation or LSQR system serves to stabilize the inversion. Best inversion stability and highest quality of the obtained images is achieved by combining the Krylov subspace projection (i.e. iterative solution with a limited number of iterations) with explicit regularization constraints.

Although the memory scalability of direct solvers is inherently limited, our scalability tests demonstrate that the direct-solver-based scheme is able to handle models of useful sizes using moderate-size clusters that are now commonly available in many academic and industrial environments. When used in combination with rapidly convergent inversion techniques, and with secondary-field approach in the forward modelling that permits using relatively coarse grids without sacrificing accuracy, direct solver-based schemes can now be considered as a viable alternative to iterative solver-based algorithms.

ACKNOWLEDGEMENTS

We acknowledge Klaus Spitzer and his group as well as Naser Meqbel for many constructive discussions on inversion concepts. Comments by Kerry Key and an anonymous reviewer helped improve the manuscript. This work is part of the MULTI-EM project, which is funded by the German Ministry of Education and Research, Grant 03G0746A-MULTI-EM.

REFERENCES

Abubakar, A., Habashy, T.M., Druskin, V.L., Knizhnerman, L. & Alumbaugh, D., 2008. 2.5D forward and inverse modeling for interpreting low-frequency electromagnetic measurements, *Geophysics,* **73,** 165–177.

Ajo-Franklin, J., Minsley, B. & Daley, T., 2007. Applying compactness constraints to differential traveltime tomography, *Geophysics,* **4,** R67–R75.

Alumbaugh, D.L. & Newman, G.A., 2000. Image appraisal for 2-D and 3-D electromagnetic inversion, *Geophysics,* **65**(5), 1455–1467.

Amestoy, P.R., Guermouche, A., LExcellent, J.-Y. & Pralet, S., 2006. Hybrid scheduling for the parallel solution of linear systems, *Parallel Comput.,* **32**(2), 136–156.

Aster, R., Borchers, B. & Thurber, C., 2004. *Parameter Estimation and Inverse Problems,* Elsevier Academic Press, Philadelphia, USA.

Balay, S., Brown, J., Buschelman, K., Eijkhout, V., Gropp, W.D., Kaushik, D., Knepley, M.G., McInnes, L.C., Smith, B.F. & Zhang, H., 2012. PETSc users manual, Tech. Rep. ANL-95/11 – Revision 3.3, Argonne National Laboratory.

Björck, Å., 1996. *Numerical Methods for Least Squares Problems,* SIAM, Philadelphia.

Brossier, R., Etienne, V., Operto, S. & Virieux, J., 2010. Frequency-domain numerical modelling of visco-acoustic waves based on finite-difference and finite-element discontinuous Galerkin methods, in *Acoustic Waves,* pp. 125–158, ed. Dissanayake, D., Sciyo, Rijeka, Croatia.

Candansayar, M.E., 2008. Two-dimensional inversion of magnetotelluric data with consecutive use of conjugate gradient and least-squares solution with singular value decomposition algorithms, *Geophys. Prospect.,* **56,** 141–157.

Carbajal, M.R., Linde, N. & Kalscheuer, T., 2012. Focused time-lapse inversion of radio and audio magnetotelluric data, *J. appl. Geophys.,* **84,** 29–38.

Commer, M. & Newman, G., 2008. New advances in three-dimensional controlled-source electromagnetic inversion, *Geophys. J. Int.,* **172,** 513–535.

Constable, S., 2010. Ten years of marine CSEM for hydrocarbon exploration, *Geophysics,* **75,** A67–A81.

Constable, S., Parke, R. & Constable, C., 1987. Occam's inversion—a practical algorithm for generating smooth models from electromagnetic sounding data, *Geophysics,* **52**(3), 289–300.

Cox, L.H., Wilson, G.A. & Zhdanov, M.S., 2010. 3D inversion of airborne electromagnetic data using a moving footprint, *Explor. Geophys.,* **41,** 250–259.

da Silva, N.V., Morgan, J.V., MacGregor, L. & Warner, M., 2012. A finite element multifrontal method for 3D CSEM modeling in the frequency domain, *Geophysics,* **77**(2), E101–E115.

Egbert, G.D., 2012. Hybrid conjugate gradient-Occam algorithms for inversion of multifrequency and multitransmitter EM data, *Geophys. J. Int.,* **190**(1), 255–266.

Egbert, G.D. & Kelbert, A., 2012. Computational recipes for electromagnetic inverse problems, *Geophys. J. Int.,* **188,** 251–267.

Ernst, O.G. & Gander, M.J., 2011. Why it is difficult to solve Helmholtz problems with classical iterative methods, *Numer. Anal. Multiscale Probl.,* **83,** 325–361.

Farquharson, C.G. & Oldenburg, D.W., 1998. Non-linear inversion using general measures of data misfit and model structure, *Geophys. J. Int.,* **134**(1), 213–227.

Fedi, M., Hansen, P.C. & Paoletti, V., 2005. Analysis of depth resolution in potential-field inversion, *Geophysics,* **70**(6), A1–A11.

Fernández Martínez, J.L., Fernández Muñiz, M.Z. & Tompkins, M.J., 2012. On the topography of the cost functional in linear and nonlinear inverse problems, *Geophysics,* **77,** W1–W15.

George, A., Liu, J. & Ng, E., 1994. *Computer Solution of Sparse Linear Systems,* Academic Press, Orlando, Florida.

Golub, G.H. & van Loan, C.F., 1996. *Matrix Computations,* 3rd edn, The Johns Hopkins University Press, Baltimore, Maryland, USA.

Gould, N.I., Scott, J.A. & Hu, Y., 2007. A numerical evaluation of sparse direct solvers for the solution of large sparse symmetric linear systems of equations, *ACM Trans. Math. Softw.,* **33**(2).

Habashy, T.M. & Abubakar, A., 2004. A general framework for constraint minimization for the inversion of electromagnetic measurements, *Prog. Electromagn. Res.,* **46,** 265–312.

Haber, E., Ascher, U. & Oldenburg, D., 2000. On optimization techniques for solving non-linear inverse problems, *Inverse Probl.,* **16,** 1263–1280.

Haber, E., Oldenburg, D.W. & Shekhtman, R., 2007. Inversion of time domain three-dimensional electromagnetic data, *Geophys. J. Int.,* **171,** 550–564.

Hansen, P.C., 1998. *Rank-Deficient and Discrete Ill-Posed Problems: Numerical Aspects of Linear Inversion,* Society for Industrial and Applied Mathematics, Philadelphia, PA, USA.

Hansen, P.C., 2010. *Discrete Inverse Problems: Insight and Algorithms,* Society for Industrial and Applied Mathematics, Philadelphia, PA, USA.

Jupp, D. L.B. & Vozoff, K., 1975. Stable iterative methods for the inversion of geophysical data, *Geophys. J. R. astr. Soc.,* **42**(3), 957–976.

Key, K. & Ovall, J., 2011. A parallel goal-oriented adaptive finite element method for 2.5-d electromagnetic modelling, *Geophys. J. Int.,* **186**(1), 137–154.

Kim, J.H. & Kim, Y., 2011. A unified transformation function for lower and upper bounding constraints on model parameters in electrical and electromagnetic inversion, *J. Geophys. Eng.,* **8,** 21–26.

Lelièvre, P.G. & Oldenburg, D.W., 2009. A comprehensive study of including structural orientation information in geophysical inversions, *Geophys. J. Int.,* **178**(2), 623–637.

Li, M., Abubakar, A., Liu, J., Pan, G. & Habashy, T.M., 2011. A compressed implicit jacobian scheme for 3D electromagnetic data inversion, *Geophysics,* **76**(3), F173–F183.

Li, Y. & Oldenburg, D.W., 1996. 3-d inversion of magnetic data, *Geophysics,* **61**(2), 394–408.

Li, Y. & Oldenburg, D.W., 2003. Fast inversion of large-scale magnetic data using wavelet transforms and a logarithmic barrier method, *Geophys. J. Int.,* **152**(3), 251–265.

Løseth, L.O. & Ursin, B., 2007. Electromagnetic fields in planarly layered anisotropic media, *Geophys. J. Int.,* **170,** 44–80.

Minsley, J.B., 2007. Modeling and inversion of self-potential data, *PhD thesis,* Massachusetts Institute of Technology, Cambridge, Massachusetts, USA.

Mulder, W., 2006. A multigrid solver for 3D electromagnetic diffusion, *Geophys. Prospect.,* **54**(5), 633–649.

Newman, G. & Boggs, P., 2004. Solution accelerators for large-scale three-dimensional electromagnetic inverse problems, *Inverse Probl.,* **20,** 151–170.

Newman, G. & Hoversten, M., 2000. Solution strategies for two- and three-dimensional electromagnetic inverse problems, *Inverse Probl.,* **16,** 1357–1375.

Newman, G.A. & Alumbaugh, D.L., 1997. Three-dimensional massively parallel electromagnetic inversion—Part I. Theory, *Geophys. J. Int.,* **128,** 345–354.

Newman, G.A., Commer, M. & Carazzone, J.J., 2010. Imaging CSEM data in the presence of electrical anisotropy, *Geophysics,* **75,** F51–F61.

Nocedal, J. & Wright, S.J., 1999. *Numerical Optimization,* Springer New York, USA.

Oldenburg, D.W., Haber, E. & Shekhtman, R., 2008. Forward modelling and inversion of multi-source TEM data, in *Proceedings of the SEG Annual Meeting,* Las Vegas, USA, pp. 559–563.

Paige, C. & Saunders, M., 1982. LSQR: An algorithm for sparse linear equations and sparse least squares, *ACM Trans. Math. Softw.,* **58,** 43–71.

Pardo, D., Paszynski, M., Collier, N., Alvarez, J., Dalcin, L. & Calo, V., 2012. A survey on direct solvers for Galerkin methods, *SeMA J.,* (57) 107–134.

Pidlisecky, A., Haber, E. & Knight, R., 2007. RESINVM3D: A 3D resistivity inversion package, *Geophysics,* **72,** H1–H10.

Plessix, R.E. & Mulder, W.A., 2008. Resistivity imaging with controlled-source electromagnetic data: depth and data weighting, *Inverse Probl.,* **24,** 1–22.

Portniaguine, O. & Zhdanov, M., 1999. Focusing geophysical inversion images, *Geophysics,* **64,** 874–887.

Pratt, R.G., Shin, C. & Hick, G.J., 1998. Gauss–Newton and full Newton methods in frequency–space seismic waveform inversion, *Geophys. J. Int.,* **133**(2), 341–362.

Prieux, V., Lambaré, G., Operto, S. & Virieux, J., 2012. Building starting models for full waveform inversion from wide-aperture data by stereotomography, *Geophys. Prospect.,* in press.

Ravaut, C., Operto, S., Improta, L., Virieux, J., Herrero, A. & Dell'Aversana, P., 2004. Multiscale imaging of complex structures from multifold wide-aperture seismic data by frequency-domain full-waveform tomography: Application to a thrust belt, *Geophys. J. Int.,* **159,** 1032–1056.

Rodi, W. & Mackie, R.L., 2001. Nonlinear conjugate gradients algorithm for 2-D magnetotelluric inversion, *Geophysics,* **66,** 174–187.

Saad, Y., 2011. *Numerical Methods for Large Eigenvalue Problems,* SIAM, Philadelphia, revised edn.

Schwarzbach, C., 2009. Stability of finite element solutions to Maxwell's equations in frequency domain, *PhD thesis,* TU Bergakademie Freiberg, Freiberg, Germany.

Schwarzbach, C. & Haber, E., 2013. Finite element based inversion for time-harmonic electromagnetic problems, *Geophys. J. Int.,* in press.

Schwarzbach, C., Börner, R.-U. & Spitzer, K., 2011. Three-dimensional adaptive higher order finite element simulation for geo-electromagnetics—a marine CSEM example, *Geophys. J. Int.,* **187**(1), 63–74.

Siemon, B., Christiansen, A.V. & Auken, E., 2009. A review of helicopter-borne electromagnetic methods for groundwater exploration, *Near Surface Geophys.,* **7**(5), 629–646.

Smith, T.J., 1996. Conservative modeling of 3-D electromagnetic fields, Part II: biconjugate gradient solution as an accelerator, *Geophysics,* **61**(5), 1319–1324.

Streich, R., 2009. 3D finite-difference frequency-domain modeling of controlled-source electromagnetic data: direct solution and optimization for high accuracy, *Geophysics,* **74,** F95–F105.

Streich, R. & Becken, M., 2011. Electromagnetic fields generated by finite-length wire sources: comparison with point dipole solutions, *Geophysical Prospecting,* **59**(2), 361–374.

Streich, R., Schwarzbach, C., Becken, M. & Spitzer, K., 2010. Controlled-source electromagnetic modelling studies: utility of auxiliary potentials for lowfrequency stabilization, in *Proceedings of the 72nd EAGE Conference, Barcelona, Extended Abstract*.

Streich, R., Becken, M., Matzander, U. & Ritter, O., 2011. Strategies for land-based controlled-source electromagnetic surveying in high-noise regions, *The Leading Edge,* **30**(10), 1174–1181.

Tikhonov, A.N. & Arsenin, V.Y., 1977. *Solutions of Ill-posed problems,* W. H.Winston and Sons, Washington, D.C., USA.

Tompkins, M.J., Fernández Martínez, J.L. & Fernández Muñiz, Z., 2011. Marine electromagnetic inverse solution appraisal and uncertainty using model-derived basis functions and sparse geometric sampling, *Geophys. Prospect.,* **59**(5), 947–965.

Um, E.S., Harris, J.M. & Alumbaugh, D.L., 2010. 3D time-domain simulation of electromagnetic diffusion phenomena: A finite-element electric-field approach, *Geophysics,* **75**(4), F115–F126.

Um, E.S., Harris, J.M. & Alumbaugh, D.L., 2012. An iterative finite element time-domain method for simulating three-dimensional electromagnetic diffusion in earth, *Geophys. J. Int.,* **190,** 871–886.

Weitemeyer, K., Gao, G., Constable, S. & Alumbaugh, D., 2010. The practical application of 2D inversion to marine controlled-source electromagnetic data, *Geophysics,* **75,** 199–211.

Yang, D. & Oldenburg, D.W., 2012. Three-dimensional inversion of airborne time-domain electromagnetic data with applications to a porphyry deposit, *Geophysics,* **77,** B23–B34.

Zhang, Z., 2003. 3D resistivity mapping of airborne EM data, *Geophysics,* **68**(6), 1896–1905.

Zhdanov, M.S., 2002. *Geophysical Inverse Theory and Regularization Problems,* Elsevier Science, Amsterdam, The Netherlands.