

3-D inversion of airborne electromagnetic data parallelized and accelerated by local mesh and adaptive soundings

Dikun Yang, Douglas W. Oldenburg and Eldad Haber

Geophysical Inversion Facility, Department of Earth, Ocean and Atmospheric Sciences, University of British Columbia, Vancouver, British Columbia, Canada. E-mail: yangdikun@gmail.com

Accepted 2013 November 12. Received 2013 October 2; in original form 2013 May 3

SUMMARY

Airborne electromagnetic (AEM) methods are highly efficient tools for assessing the Earth's conductivity structures in a large area at low cost. However, the configuration of AEM measurements, which typically have widely distributed transmitter-receiver pairs, makes the rigorous modelling and interpretation extremely time-consuming in 3-D. Excessive overcomputing can occur when working on a large mesh covering the entire survey area and inverting all soundings in the data set. We propose two improvements. The first is to use a locally optimized mesh for each AEM sounding for the forward modelling and calculation of sensitivity. This dedicated local mesh is small with fine cells near the sounding location and coarse cells far away in accordance with EM diffusion and the geometric decay of the signals. Once the forward problem is solved on the local meshes, the sensitivity for the inversion on the global mesh is available through quick interpolation. Using local meshes for AEM forward modelling avoids unnecessary computing on fine cells on a global mesh that are far away from the sounding location. Since local meshes are highly independent, the forward modelling can be efficiently parallelized over an array of processors. The second improvement is random and dynamic down-sampling of the soundings. Each inversion iteration only uses a random subset of the soundings, and the subset is reselected for every iteration. The number of soundings in the random subset, determined by an adaptive algorithm, is tied to the degree of model regularization. This minimizes the overcomputing caused by working with redundant soundings. Our methods are compared against conventional methods and tested with a synthetic example. We also invert a field data set that was previously considered to be too large to be practically inverted in 3-D. These examples show that our methodology can dramatically reduce the processing time of 3-D inversion to a practical level without losing resolution. Any existing modelling technique can be included into our framework of mesh decoupling and adaptive sampling to accelerate large-scale 3-D EM inversions.

Key words: Numerical solutions; Inverse theory; Electrical properties; Electromagnetic theory.

1 INTRODUCTION

A typical airborne electromagnetic (AEM) system consists of a transmitter, a loop of wire carrying a time varying current, and a receiver that measures the magnetic field. The transmitter and the receiver are attached to the aircraft and move together with a constant separation. Many soundings (data from a particular transmitter-receiver pair) are taken as the aircraft moves along a flight line. The configuration of distributed sources and receivers makes AEM highly efficient in assessing the Earth's conductivity over large areas. Millions of line kilometres of data are collected every year for applications in mining, geotechnical, hydrological and environmental problems. However, the benefits of fast acqui-

sition over large areas raises serious challenges for 3-D modelling and inversion. Because of the large area of survey, the governing Maxwell's equations must be discretized on a very large mesh and the solution of the resultant matrix system is computationally challenging. Moreover this large computational problem needs to be solved for each new transmitter location. Thus hundreds of thousands of Maxwell's solutions need to be computed to solve one complete forward problem. The inverse problem requires many of these complete forward modellings to find a solution. Primarily due to the unaffordable computational cost, AEM data have been routinely interpreted by the methods that assume a simplified model, for example, apparent conductivity (Fraser 1978; Palacky & West 1991; Palacky 1993), time constant (Palacky & West 1973; Macnae

1998), conductivity depth transform/imaging (Wolfgram & Karlik 1995; Eaton 1998; Macnae 1998; Fullagar & Reid 2001; Macnae *et al.* 2010), 1-D layered Earth inversion (Farquharson & Oldenburg 1993; Lane *et al.* 2000; Wolfgram *et al.* 2003; Sattel 2005; Brodie & Sambridge 2006; Vallee & Smith 2009; Fullagar *et al.* 2010) and plate modelling (Keating & Crossley 1990; Raiche 2004). While these methods provide critical first-order information about the conductivity, they can fail to yield reasonable models when their basic assumptions are violated. For instance, in a previous paper Yang & Oldenburg (2012a) we showed an example where a 1-D inversion of airborne time-domain EM data generated a model that contradicted known geology. This motivates the goal of being able to routinely invert all airborne data in 3-D.

Rigorous EM modelling has physical complexities in time and space domains. The time complexity requires accurate modelling of Maxwell's equations on timescales ranging from microseconds to hundreds of seconds depending on the system. The space complexity comes from the large number of cells needed to discretize the Maxwell system in 3-D space with distributed sources. Over the last decade, much effort has been made to accelerate 3-D inversion by reducing the time complexity and/or space complexity. On the time complexity side, Zaslavsky *et al.* (2011, 2013) have shown that optimal Laplace frequencies can be chosen using rational Krylov subspace model reduction. On the space complexity side, Cox *et al.* (2010) and Wilson *et al.* (2010) used an integral equation method and a moving footprint to parallelize the computation of the sparsified sensitivity matrix for speed-up. In our work we use a finite volume method (Bossavit 1998; Haber & Ascher 2001) and direct time stepping to discretize Maxwell's equations. This paper is a progression of advances using that methodology. The first paper (Haber *et al.* 2007) develops the computational strategy for inverting time domain data from a single (or very few) transmitters. Addressing the challenge of multiple sources, Oldenburg *et al.* (2013) used a direct solver that allowed relatively large problems (hundred transmitters and a half-million cells) to be solved on a cluster in a timescale from a few days to weeks. Unfortunately, although the Maxwell matrix can be factored with algorithms, such as MUMPS (Amestoy *et al.* 2006), the amount of memory and time needed eventually becomes prohibitive due to its poor scalability. To handle these larger problems, Yang & Oldenburg (2012b) used both an adaptive mesh and a tiling method that decomposes the whole survey into many tiles and merged the inversion results. These advances brought down the processing time of 3-D AEM inversion by several orders of magnitude, but the algorithms were still very time-consuming when there was a large number of tiles. Outside of the airborne EM community, some inspiring works have been done in marine controlled-source EM (CSEM), in which the data are acquired by a towed system similar to airborne EM. Commer & Newman (2006a,b) first decoupled the mesh of forward simulation from the mesh for inversion so that the simulation mesh, used for lower frequencies or later times, could be much coarser. Then when inverting large-scale marine CSEM data, the frequency-dependent simulation meshes were further optimized based on the geometry of the transmitter and the receiver so the simulation meshes were smaller than the inversion mesh; the large number of sources (and data) were tackled by massive parallelization involving 32 768 processors (Commer & Newman 2008; Commer *et al.* 2008).

In this paper, we concentrate upon airborne EM data and recognize the essential need to reduce the space complexity. We improve our algorithm through two approaches. First, in our previous work we used a single mesh that was large enough and finely discretized everywhere to serve all the soundings. However, such a mesh is not

economical for forward modelling of any specific sounding since a diffusive EM problem at very fine scale is solved in the whole domain. To circumvent this, we allow each sounding to have its own mesh. In a parallel computing environment, forward simulations on local meshes can be solved concurrently, making the decomposition of the large problem into many local mesh problems a very promising way of speeding-up the computations. The second advance deals with reducing the computation by limiting the number of soundings. In our formulation, early iterations of inversion are dominated by the model regularization, so only a few soundings are required to estimate the large-scale structure; when the data misfit becomes more important at later iterations, more soundings can be added to delineate finer scale structure. This sets the basic idea of our strategy of adaptive soundings. The procedures developed here are equally applicable to frequency and time domain systems. In this paper, we will illustrate them with time-domain data. It is also important to note our approaches are fully inclusive frameworks that should work seamlessly with any existing EM modelling technique.

The paper is structured as follows. We first briefly review the forward modelling and inversion methodology in Oldenburg *et al.* (2013) and illustrate where the improvements can be achieved. We then focus upon the implementation of local meshes and show how the sensitivity-vector product, an essential step in inversion, is implemented in parallel. We discuss the performance of parallelization in the local mesh method. We next introduce our methodology for adaptive sounding selection. The final algorithm is demonstrated on a synthetic example that was challenging to work with using our previous codes. Finally, our method is applied to a field data set from Mt Milligan, which is a porphyry deposit in British Columbia, Canada. The entire data set of 7.3 km² containing 14 362 soundings is inverted at 50 m resolution on 24 processors within 4.3 hr.

2 FORWARD MODELLING AND INVERSION ALGORITHMS

2.1 Forward modelling: finite volume method

In the following, we briefly summarize the essential computational framework for solving our time-domain EM problem (Haber *et al.* 2007; Oldenburg *et al.* 2013). Maxwell's equations with the quasi-static approximation are

$$\nabla \times \mathbf{E} + \mu \frac{\partial \mathbf{H}}{\partial t} = 0, \quad (1a)$$

$$\nabla \times \mathbf{H} - \sigma \mathbf{E} = \mathbf{J}_s(t), \quad (1b)$$

where \mathbf{E} is the electric field, \mathbf{H} is the magnetic field, μ is the magnetic permeability assumed to be equal to the vacuum value $\mu_0 = 4\pi \times 10^{-7} \text{ H m}^{-1}$, σ is conductivity in S m^{-1} and $\mathbf{J}_s(t)$ is current density from the source. Approximating the time derivative with finite differences using the backward Euler method, eq. (1) becomes

$$\nabla \times \mathbf{E}^{i+1} + \mu_0 \frac{\mathbf{H}^{i+1} - \mathbf{H}^i}{\delta t} = 0, \quad (2a)$$

$$\nabla \times \mathbf{H}^{i+1} - \sigma \mathbf{E}^{i+1} = \mathbf{J}_s^{i+1}, \quad (2b)$$

where superscript i denotes the field or flux at time i . By eliminating the electric field we obtain a second-order system for the magnetic field

$$\nabla \times \sigma^{-1} \nabla \times \mathbf{H}^{i+1} + \gamma \mathbf{H}^{i+1} = \nabla \times \sigma^{-1} \mathbf{J}_s^{i+1} + \gamma \mathbf{H}^i, \quad (3)$$

where $\gamma = \mu_0/\delta t$. Upon discretization with finite volumes on a staggered grid, eq. (3) can be written in terms of discrete operators and expressed as a linear system of equations

$$\mathbf{A}(\sigma, \delta t)h^{i+1} = \text{rhs}^{i+1}, \tag{4}$$

where h represent the three-component \mathbf{H} fields in a vector, and the right-hand side (rhs) depends on the source at time step $i + 1$ and h' . For the details of the discretization and the composition of eq. (4), the readers are referred to Oldenburg *et al.* (2013) and Appendix A. For a single forward modelling of a few sources, eq. (4) can be inexpensively solved by iterative solvers such as Krylov subspace methods. However, when solving the inverse problem, which involves many \mathbf{A}^{-1} operations, a direct solver is preferable. The formulation in eq. (3) guarantees a symmetric and positive definite Maxwell matrix \mathbf{A} , so a Cholesky decomposition can be applied

$$\mathbf{A} = \mathbf{L}\mathbf{L}^T. \tag{5}$$

Once \mathbf{A} is factorized and stored, the operator \mathbf{A}^{-1} is available for solving any arbitrary right-hand side. Then the field \mathbf{H} at a particular time can be quickly solved by forward and backward substitutions.

Eq. (4) must be solved for every time step. Since the conductivity model is fixed, a new factorization in eq. (5) is only required if δt changes. To accommodate the diffusive nature of the field the complete time interval for a sounding is divided into multiple segments, each with its own δt . A separate factorization is required for each of these. Therefore, a complete time-domain modelling consists of several factorizations and solving multiple time steps using a constant δt within each segment. While many factors contribute to the time and memory costs of 3-D AEM modelling, the number of cells in mesh and the number of sources (soundings) are the most crucial elements.

2.2 Inverse problem: Gauss–Newton method

The inverse problem of finding a model $m = \ln \sigma$ that reproduces the observed data is posed as an optimization problem

$$\begin{aligned} &\underset{m}{\text{minimize}} && \phi_m(m) \\ &\text{subject to} && \phi_d(m) < \epsilon, \end{aligned} \tag{6}$$

where ϕ_m is a model norm that measures the complexity of model, ϕ_d measures how well a model predicts the observed data, and ϵ is a small tolerance. The problem in eq. (6) can be solved as an unconstrained optimization problem by introducing a trade-off parameter β ,

$$\underset{m, \beta}{\text{minimize}} \quad \phi_m(m) + \beta\phi_d(m). \tag{7}$$

Here we define ϕ_d and ϕ_m as

$$\phi_d = \frac{1}{2} \sum_{i=1}^N \left(\frac{F_i(m) - d_i^{\text{obs}}}{\epsilon_i} \right)^2, \tag{8}$$

$$\phi_m = \frac{1}{2} \alpha_s \int_{\Omega} (m - m_0)^2 dv + \frac{1}{2} \sum_{i=x,y,z} \alpha_i \int_{\Omega} \left\{ \frac{\partial(m - m_0)}{\partial_i} \right\}^2 dv. \tag{9}$$

In the data misfit term eq. (8), N is the total number of data, F is the forward modelling from model to data, d^{obs} is the observed data and ϵ is the estimated uncertainty. In the model norm term eq. (9), Ω is

the modelling domain, m_0 is a reference model, and $\alpha_s, \alpha_x, \alpha_y, \alpha_z$ are scalar weighting parameters adjusting the relative importance of different components in the model norm.

Expressing eqs (8) and (9) in discrete form, we have eq. (7) written as

$$\phi = \frac{1}{2} \|\mathbf{W}_d[F(m) - d^{\text{obs}}]\|_2^2 + \frac{1}{2} \|\mathbf{W}_m(m - m_0)\|_2^2, \tag{10}$$

where \mathbf{W}_d is a diagonal weighting matrix containing the information about the data uncertainty, $F(m)$ and d^{obs} are vectors and \mathbf{W}_m is an assembly of one diagonal matrix and three directional first-order differential matrices. Differentiating eq. (10) with respect to m and setting the gradient to zero yield a system

$$g(m) = F'(m)^T \mathbf{W}_d^T \mathbf{W}_d [F(m) - d] + \beta \mathbf{W}_m^T \mathbf{W}_m (m - m_0) = 0, \tag{11}$$

where $F'(m)$ is the derivative of forward modelling operator. Eq. (11) is non-linear and is solved iteratively using a Gauss–Newton approach. At each iteration we solve

$$[\mathbf{J}(m^k)^T \mathbf{W}_d^T \mathbf{W}_d \mathbf{J}(m^k) + \beta \mathbf{W}_m^T \mathbf{W}_m] \delta m^{k+1} = -g(m^k), \tag{12}$$

where $\mathbf{J}(m^k)$ is the sensitivity or Jacobian matrix corresponding to the model m^k . The updated model $m^{k+1} = m^k + \alpha \delta m^{k+1}$ where α is a step length parameter obtained by a line search. Eq. (12) is commonly solved iteratively by conjugate gradient (CG) solvers that only require \mathbf{J} and \mathbf{J}^T times a vector. We note the sensitivity matrix \mathbf{J} can be computed either explicitly or implicitly. In the implicit method, \mathbf{J} can be decomposed as

$$\mathbf{J} = \mathbf{Q}\mathbf{B}^{-1}\mathbf{G}, \tag{13}$$

where \mathbf{G} is a sparse known matrix, \mathbf{B}^{-1} collectively represents all \mathbf{A}^{-1} at different times and \mathbf{Q} is a sparse projection matrix that interpolates the computed fields on the staggered mesh to the observation locations. The derivation of eq. (13) is detailed in Appendix B. Since the Maxwell matrices \mathbf{A} are factorized and the operation \mathbf{B}^{-1} is available, $\mathbf{J}v$ or $\mathbf{J}^T v$ can be rapidly computed through forward and backward substitution. The implicit method has long been preferred because the operation of $\mathbf{J}v$ or $\mathbf{J}^T v$ is readily available after the forward solutions are obtained and one does not need to work with dense matrix \mathbf{J} . However, implicit sensitivity requires a substantial amount of memory for the storage of \mathbf{A}^{-1} and the fields in time-domain problem. An explicit \mathbf{J} can be formed using eq. (13) and solving for $\mathbf{J}^T r_i$ for the i th datum where r_i is the zero vector with a unit value in the i th entry. Once \mathbf{J} is computed and stored, $\mathbf{J}v$ or $\mathbf{J}^T v$ does not requires \mathbf{A}^{-1} and the factorized matrices and the fields can be erased from the memory. Table 1 compares the implicit and explicit methods in terms of memory and time. Neither one is always advantageous since the computational costs depend upon the number of data and the number of CG iterations, but general statements can be made: (1) use an implicit method if there are many data (time channels), if there are not too many CG iterations and if memory is not a concern and (2) use an explicit method if there are not too many time channels, if many CG iterations are expected and if memory is an issue. We choose the explicit method for this particular problem. The explicit method also make the calculation of the pre-conditioner easier when solving eq. (12).

The minimizer of eq. (7) obtained by solving eq. (12) may not satisfy the constraint in eq. (6) if β is too large and the data are under-fit. We use a cooling strategy to find the largest β that offers acceptable ϕ_d . That is, the inversion starts with a large β implying the simplicity of the model is more important than the reduction of data misfit and the data are only roughly fit; then β is gradually

Table 1. Comparison between implicit and explicit methods of computing sensitivity. N_d is the number of data (time channels in airborne TEM). N_{iter} is the number of CG iterations when solving eq. (12).

Method	Memory	Time (computing \mathbf{J})	Time (solving eq. 12)
Implicit	\mathbf{A}^{-1} and fields (more)	None (faster)	$2N_{iter}$ operations of \mathbf{A}^{-1} (slower)
Explicit	\mathbf{J} matrix (less)	N_d operations of \mathbf{A}^{-T} (slower)	Simple matrix-vector product (faster)

decreased until the model contains enough structure to reproduce the observed data. The cooling strategy is a process of transferring information from the data to the model and plays an important role in our adaptive soundings method that cuts off unnecessary modellings described later in this paper.

3 LOCAL MESH METHOD

The advantage of using a direct solver, as illustrated in Oldenburg *et al.* (2013), is that the forward problem with many transmitters and the sensitivity times a vector can efficiently be solved once the matrix decomposition has been carried out. However, the time to perform this factorization and the storage needed to store the factored matrix, increases with the size of the problem. Moreover, since the factored matrix is stored on many computers, the amount of intercomputer communication time increases. Eventually, the progress of the matrix inversion stagnates and the benefits of decomposition are severely marginalized. We demonstrate the demand of memory and time by a sequence of one-sounding ATEM forward modellings using meshes with variable number of cells (Fig. 1). The poor scalability makes the forward solution computationally expensive when solving realistically sized problems.

As seen in eq. (12), however, this slow-down could be avoided if the forward modelling, and equivalently sensitivity evaluation, can be carried out faster. To achieve this, and to make better use of a parallel computing architecture, we design a local mesh which has

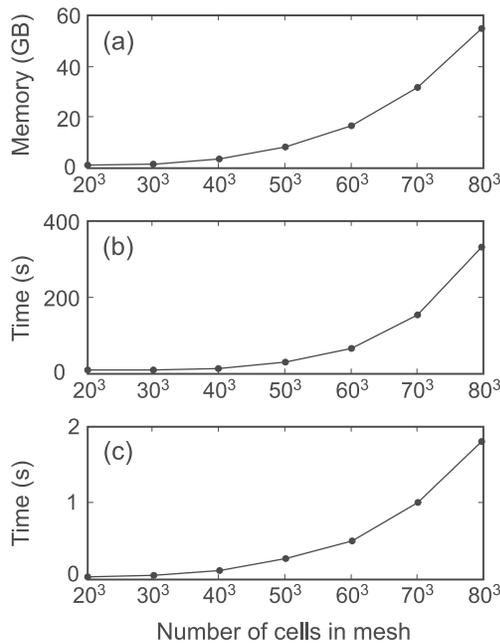


Figure 1. Computational costs are poorly scaled as the number of cell in mesh increases. Forward modelling experiments are measured by (a) memory required by storing one factorized matrix; (b) time required to carry out one Maxwell matrix factorization and (c) time required to solve for the fields at one time step.

finer cells near the sounding location and only covers the sensitivity volume of a single sounding. Because these meshes are small, they are easily factorized and hence forward modelling is readily carried out. Sensitivities are also computed on the local mesh and interpolated onto the global mesh for the purpose of inversion. Since the modelling of soundings are independent, and there are typically hundreds or thousands of soundings in a survey, great time-saving for inversion could be obtained by using massive parallelization, and computation time will scale with the size of survey and the number of processors.

3.1 Design of local mesh

Two basic rules of mesh design are that the cells must be small enough to evaluate the fields for the earliest time channels and that the mesh extends sufficiently in volume so that fields have adequately decayed for the latest time channels. Cell sizes must also be small enough to handle topography and highly variable conductivity. The use of an individual local mesh for each sounding is motivated by the fact that the induced EM fields are diffusive. Fig. 2 shows the magnitude of the total magnetic field $|\mathbf{H}|$ on a cross section at four different times after the turn-off of transmitter current. Two characteristics of the EM field that are important to local mesh design are observed: (1) the field has short-wavelength variation close to the sounding location and is smooth away from the source and (2) most of the energy concentrates in a region around the source and the strength of the field decays exponentially away from the source.

The first characteristic implies that a local mesh can have small cells near the sounding location and larger cells that gradually expand outwards and downwards away from the source without degrading the modelling accuracy. This prevents overcomputing of smooth fields and smooth sensitivities on fine cells that are far way from the transmitter. A similar mesh coarsening scheme was previously implemented in single-well simulation (Davydycheva *et al.* 2003). The second characteristic means that a local mesh can be safely truncated at a certain distance, beyond which the fields decay to a level of strength that is too small to be important. This avoids computation on remote cells that have no effect on the data and sensitivity. Fig. 3 illustrates the concept using a 2-D mesh. In the conventional approach, the global mesh is designed to be accurate for all transmitters (see Fig. 3c). The local meshes, of which there are five in this case, are optimized for each sounding (Fig. 3b). Modelling on the global mesh requires factorization of the Maxwell matrix, which is expensive in time and storage, and subsequent solution of the fields over the global mesh for every transmitter. Modelling on the local meshes requires factorization on five small meshes, which is a relatively inexpensive operation, and solution of a smaller matrix system for every source. For large problems, modelling on the mesh in Fig. 3(b) is more efficient, both in terms of CPU time and memory demand, than that on the mesh in Fig. 3(a). The efficiency of using local mesh is exacerbated if many soundings are modelled over a large area. While conventional methods model each sounding on a large global mesh (Fig. 3c), which could be

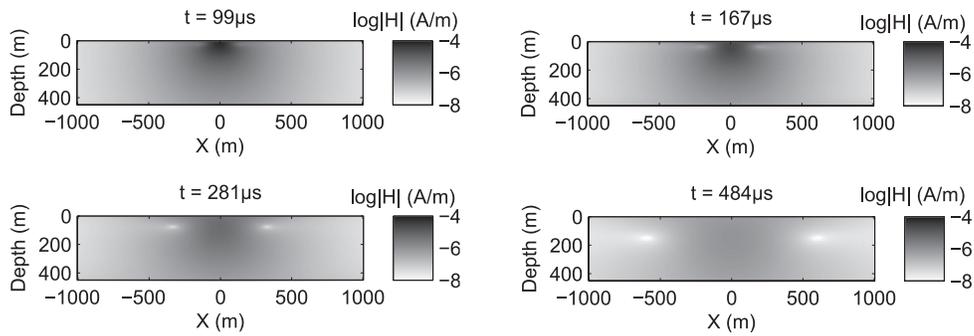


Figure 2. Amplitude of the total magnetic field excited by a magnetic dipole source at $X = 0$ over a uniform half-space model.

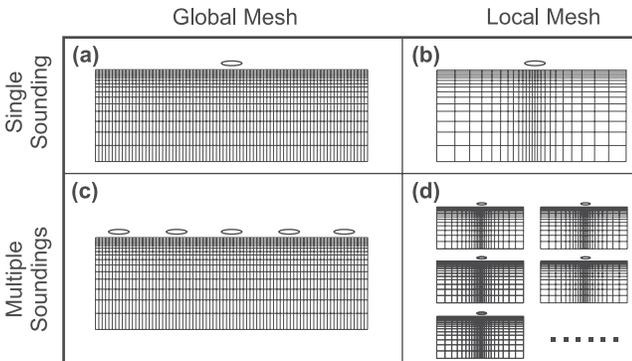


Figure 3. Modelling using global mesh and local meshes. Modelling on local mesh that is specifically designed for EM diffusion at each sounding is more efficient and massively parallelizable.

slow or even impractical due to lack of computing resources, the local mesh approach can decompose the problem into many local meshes and solve them concurrently using massive parallel computing techniques (Fig. 3d). In this way, the size of survey is not a prohibitive concern for forward modelling and the computation cost scales linearly with the number of soundings and with the area of airborne survey.

A local mesh is designed to have a core whose cell sizes are equal to those of the surface cells in the global mesh. Cell sizes outside the core region increase by a constant expansion rate downwards and outwards. A large expansion rate results in a coarser mesh and thus speeds up the computation at the cost of accuracy, whereas a small rate produces a finer mesh with better modelling accuracy but requires increased computation time. Generally a sufficient number of cells per diffusion distance (conductivity-dependent) are needed to capture the fields of EM diffusion. We have found an expansion rate of 1.2–1.4 yields reasonable results and balances cost and accuracy in most cases of mining exploration, although a rate up to 2 has been found acceptable in a uniform half-space and flat surface model. In order to determine the size of the modelling domain, we begin with a small distance of truncation to the outer boundary, carry out the forward modelling, and then keep expanding the boundary until the difference between two successive forward modellings falls beneath a prescribed limit. This adaptive search procedure is carried out for every sounding and for every updated model during inversion, so soundings at different locations, and at different iterations of inversion, have their own customized boundary distances. This can be important in regions of severe topography or highly variable conductivity. Once a local mesh is designed the (local) forward modelling matrix \mathbf{A} is factorized and \mathbf{L} is stored and used for data simulation and for the future use of computing sensitivity.

The use of local meshes requires that a model defined on the global mesh must be converted to a model on local meshes. Since the local meshes are non-conformal with the global inversion mesh some interpolation is needed. This is done by a material averaging technique similar to Commer & Newman (2006a). The conductivity of a cell in a local mesh is a volume-weighted averaging of all the overlapping global cells. In practice, we design the local mesh to conform with the global mesh near the sounding location so that the conversion from global to local is exact for the area where detailed structure of model matters most. Because of EM diffusion, the small-scale structures far away from the sounding location are not important.

3.2 Forward modelling on local mesh

To show that our projection method can provide reasonable accuracy for forward modelling with greatly reduced computational costs we use a synthetic model with a complex conductivity structure shown in Fig. 4(a). The original model is defined on a global mesh finely discretized to 50 m resolution for a survey area of 4×4 km. We choose the sounding at the centre of the model. The first local mesh (local mesh 1) has an expansion rate 1.2 and a boundary distance 1500 m. This mesh performs well at early times compared to the global mesh but has up to one order of magnitude error at late times due to an insufficient distance to the boundary. The local mesh is then expanded to a boundary distance of 3000 m (local mesh 2). With the additional three padding cells in every direction, local mesh 2 is able to provide forward modelled data within 5 per cent of the global mesh results at late times (Fig. 5).

It is important to note that although fine structures of the model away from the sounding location are represented by bulky cells, a local mesh is still capable of producing a good simulation at a much reduced cost. Table 2 summarizes the three forward modellings on the global and local meshes. The global mesh, being large for the entire survey, is expensive in both time and memory. Because a large Maxwell matrix needs to be factorized, 12 processors are required. The local meshes are much smaller and thus more efficient, even though only one processor is used. Multiplying the cost by the number of soundings further signifies the benefits of using local mesh approach in airborne inversion. If the entire survey contained 1080 soundings and one forward modelling contains four factorizations and 50 time steps, the global mesh would require at least 54 492 s for one complete forward modelling on 12 processors and further speed-up by adding more processors is difficult. However local mesh 2 needs 19.4 s for one sounding (local mesh) and a total of 20 952 s if all computations were carried out on a single processor. Distributing these jobs over the 12 processors reduces this to

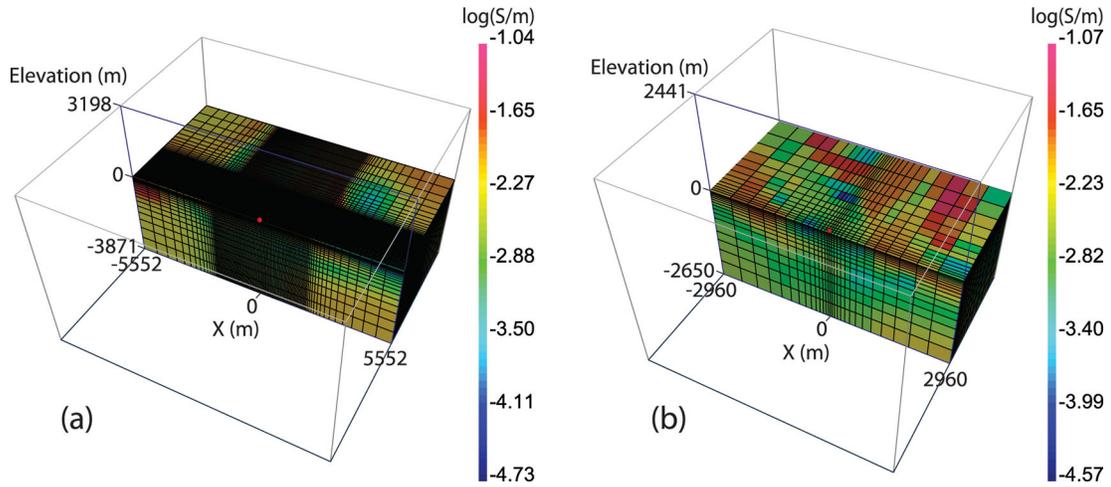


Figure 4. The synthetic model on the global mesh and a local mesh. The global mesh (a) has fine cells over a 4×4 km survey area. An airborne sounding indicated by a red mark at the centre of the model is simulated. The same sounding is modelling on a local mesh (b) with an expansion rate 1.2 and boundary distance 3000 m.

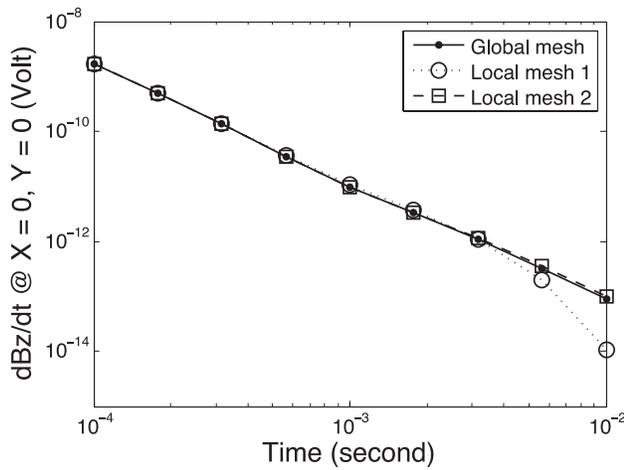


Figure 5. Forward modelled data on global mesh and the two successive local meshes in the test. Local mesh 1 and 2 have the same expansion rate 1.2 but different boundary distances 1500 and 3000 m, respectively.

1746 s. Increasing the number of processors continues to achieve this linear benefit.

3.3 Sensitivity on local mesh

Having solved the forward problem we next turn attention to the sensitivity matrix \mathbf{J} in eq. (12). The sensitivity also has fine-scale variations near the sounding location and falls off smoothly and quickly due to the geometric decay. Therefore, the same local mesh used in the forward modelling can also be used for the computation of sensitivity. Once the sensitivity on a local mesh is available, whether implicitly or explicitly, the local sensitivity can be projected onto the global mesh through interpolation. The global cells that are

outside of a local mesh are deemed not to contribute to the sounding, and zeros are assigned to the corresponding entries in the sensitivity matrix.

The calculation of sensitivity for an airborne sounding is derived in Appendix B. Using that method, we can compute the local sensitivity in a symbolic form $\mathbf{J} = \mathbf{Q}\mathbf{B}^{-1}\mathbf{G}$ (or $\mathbf{J}^T = \mathbf{G}^T\mathbf{B}^{-T}\mathbf{Q}^T$ for its transpose). In order to compute the global sensitivity, the sensitivity on a local mesh is first normalized by the cell volumes to obtain a cell-size independent sensitivity, which is considered a 3-D function in space

$$\hat{\mathbf{J}}_{li} = \mathbf{J}_{li}\mathbf{V}_{li}^{-1}, \quad (14)$$

where \mathbf{J}_{li} is the un-normalized sensitivity matrix on i th local mesh and \mathbf{V}_{li} is a diagonal matrix of cell volumes on that local mesh. The sensitivity matrix of i th sounding for the global mesh is available through

$$\mathbf{J}_{gi} = \hat{\mathbf{J}}_{li}\mathbf{R}_i\mathbf{V}_g = \mathbf{J}_{li}\mathbf{V}_{li}^{-1}\mathbf{R}_i\mathbf{V}_g, \quad (15)$$

where \mathbf{R}_i is an interpolation matrix mapping a 3-D function defined on the cell centres of a mesh to the cell centres of another mesh and \mathbf{V}_g is a diagonal matrix of cell volumes on the global mesh. Similarly, we have

$$\mathbf{J}_{gi}^T = \mathbf{V}_g\mathbf{R}_i^T\mathbf{V}_{li}^{-1}\mathbf{J}_{li}^T. \quad (16)$$

The matrices \mathbf{V}_l , \mathbf{V}_g and \mathbf{R}_i are sparse, so the projection of sensitivity from local to global is very efficient when doing the operation of sensitivity times a vector.

We use the same synthetic model in the forward modelling example (Fig. 4) to show the global sensitivity can be reasonably reconstructed using a local mesh. At the same sounding, we choose a row of the sensitivity matrix corresponding to the dB_z/dt datum at time $t = 0.001$ s and present the sensitivity as a model. The sensitivity is computed directly on the global mesh (Fig. 6a) and on local

Table 2. Forward modelling of a single sounding (source) on the global mesh and local meshes.

Mesh	Size	One factorization time	One time step time	Processors	Memory for one \mathbf{A}^{-1}
Global	$108 \times 108 \times 33$	123 s	1 s	12	40.5 GB
Local 1	$22 \times 22 \times 27$	1.5 s	0.02 s	1	174 MB
Local 2	$28 \times 28 \times 30$	4.1 s	0.06 s	1	395 MB

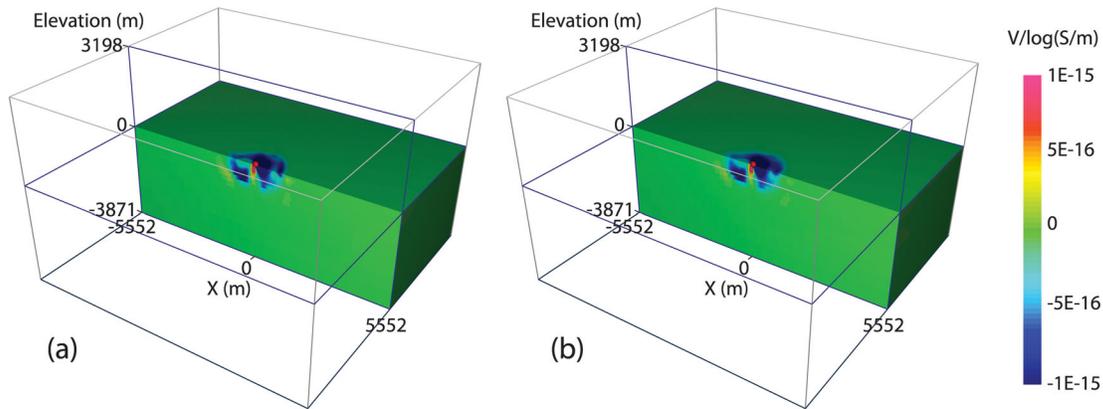


Figure 6. Sensitivities of dBz/dt datum at $t = 0.001$ s for the synthetic random model. (a) Directly computed on the global mesh and (b) computed on local mesh 2 then interpolated to the global mesh.

mesh 2 using eqs (14) and (15) (Fig. 6 b). The local mesh result matches well with the sensitivity computed on the global mesh.

Solving eq. (12) requires $\mathbf{J}_g^T \mathbf{W}_d^T \mathbf{W}_d \mathbf{J}_g$ times a vector v and $\mathbf{J}_g^T \mathbf{W}_d^T \mathbf{W}_d$ times a vector of data misfit $F(m) - d^{\text{obs}}$. Although these operations are on the global mesh, the computation can be carried out on local meshes in parallel. We denote the global sensitivity of the entire survey to be assembly of eq. (15) on many soundings (local meshes)

$$\mathbf{J}_g = \begin{pmatrix} \vdots \\ \mathbf{J}_{gi} \\ \vdots \end{pmatrix}. \quad (17)$$

Then

$$\mathbf{J}_g^T \mathbf{W}_d^T \mathbf{W}_d \mathbf{J}_g v = \sum_i \mathbf{J}_{gi}^T \mathbf{W}_{di}^T \mathbf{W}_{di} \mathbf{J}_{gi} v, \quad (18)$$

where the diagonal matrix \mathbf{W}_{di} is part of \mathbf{W}_d corresponding to the i th sounding, and

$$\mathbf{J}_g^T \mathbf{W}_d^T \mathbf{W}_d [F(m) - d^{\text{obs}}] = \sum_i \mathbf{J}_{gi}^T \mathbf{W}_{di}^T \mathbf{W}_{di} [F_i(m) - d_i^{\text{obs}}], \quad (19)$$

where $F_i(m) - d_i^{\text{obs}}$ is a vector of data misfit for the i th sounding.

The computation of eqs (18) and (19) on the local meshes allows each local mesh to behave independently in inversion. First, a global model is broadcast to many local meshes. The forward solutions are computed and stored on local meshes without need to return to the host thread. Before solving eq. (12), the right-hand side containing $\mathbf{J}_g^T \mathbf{W}_d^T \mathbf{W}_d [F(m) - d]$ can be calculated, with locally stored information, using eq. (19). This only requires inter-processor communication when summing resultant vectors from the local meshes. During each CG iteration for solving eq. (12), a vector is broadcast to the local meshes for $\mathbf{J}_{gi}^T \mathbf{W}_{di}^T \mathbf{W}_{di} \mathbf{J}_{gi}$ times the vector, then many resultant vectors are collected, summed and returned to the host thread. The calculation of pre-conditioner for eq. (12) can be carried out in the similar way. The blindness of the host thread to the forward solutions and sensitivities keeps the most computationally expensive operation at the level of individual parallel workers and allows great potential of speed-up using massive parallelization.

3.4 Parallelization

In recognition of the amount of memory and CPU time by multisource 3-D EM inversion, much work has been done in the past

decades to make use of parallel computing. This includes work stations, computer clusters and emerging GPUs. Many of the works, for example, Alumbaugh *et al.* (1996), Newman & Alumbaugh (1997), Xie *et al.* (2000) and Commer & Newman (2004), used domain decomposition at cell level or subdomain level. These approaches can scale well with the number of processors at early stages when the benefit of using many processors outplays the cost of interprocessor communication. If the number of processors increases further, the communication time can dominate and eventually make the parallelization less economical. In addition to model decomposition, Commer & Newman (2008) also used data decomposition to distribute marine transmitters over an array of clustered processors and nearly perfect scaling was reported. Fortunately, modern computer technology has made the processors powerful enough so that a small, or even mid-sized EM problem, can be solved on a single processor (or a core on a multicore processor). Since local meshes designed for airborne EM are usually small, and each local mesh is a self-contained EM problem, solving one or more local mesh problems on each individual processor requires much less interprocessor communications compared to distributing the work via a cell or subdomain level decomposition. In this section, we examine an inversion run time model for the local mesh method and discuss its scalability in massive parallelization. For simplicity, the computing environment is assumed to be a uniform array of processors, each of which has its allocated random-access memory.

Communication between workers is the primary concern when an algorithm is to be parallelized. For the local mesh method, the communication between processors occurs when the host thread broadcasts a vector of model parameters and collects computed results from the workers. Broadcasting a vector v is well known to have a highly efficient time complexity of $O(\log n)$. After the local $\mathbf{J}^T \mathbf{J} v$ operation is finished on every processor, the resultant vectors from the local meshes are summed to compute the global $\mathbf{J}^T \mathbf{J} v$ vector. Because the operation of vector summation is easily parallelizable in a manner of a binary tree, the time required to send the results back to the host thread is also $O(\log n)$. The total time of each inversion iteration, consisting of the solving time and the two-way communication time, can be estimated by

$$T = T_s + T_c = t_s \cdot \frac{N_{\text{LM}}}{N_p} + 2 t_c \log_2 N_p, \quad (20)$$

where t_s is the time for solving one local mesh problem, N_{LM} is the number of local meshes to be solved, N_p is the number of processors and t_c is the time for the communication of passing a vector between

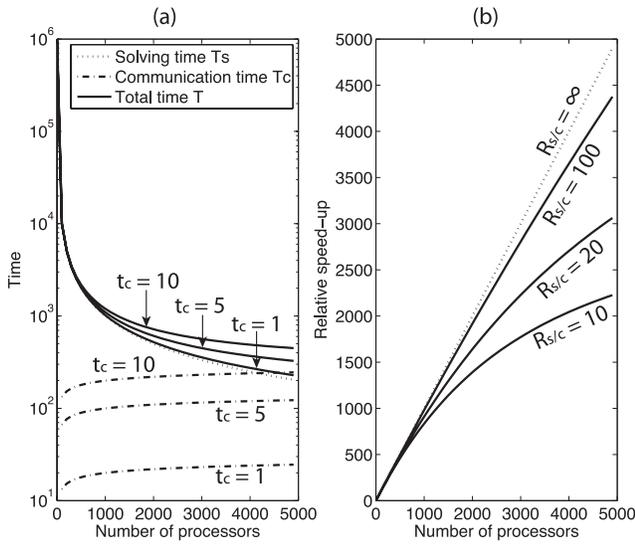


Figure 7. Massive parallelization of the local mesh method. (a) Solving time, communication time and total time as functions of the number of processors. (b) Relative speed-up as a function of the number of processors with reference to the sequential mode; the dotted line indicates an ideal linear speed-up if the communication time is zero. These plots assume the number of local meshes is 10 000 and the time of solving one local mesh problem is 100; three different communication times, 1, 5 and 10 are considered for the solving-communication time ratios of 100, 20 and 10, respectively.

two processors. As N_p increases, T_s decreases and T_c increases. So it is useful to know the optimal N_p that provides the maximum speed-up. Considering eq. (20) as a function of N_p , the total time for a given N_{LM} has a minimum at

$$N_p^* = \frac{\ln 2}{2} \frac{t_s \cdot N_{LM}}{t_c} \approx 0.35 R_{s/c} N_{LM}, \quad (21)$$

in which $R_{s/c}$ is defined as the ratio of t_s to t_c . If $N_p^* > N_{LM}$, the inversion can always gain further speed-up by adding extra proces-

sors (up to N_{LM} processors). This condition, which we refer to as continual speed-up, occurs if

$$R_{s/c} > \frac{2}{\ln 2} \approx 2.88. \quad (22)$$

In the local mesh method, the operation of $\mathbf{J}^T \mathbf{J} \mathbf{v}$ usually takes much longer time than passing a vector through the networks. Therefore, the condition of continual speed-up is satisfied in most cases.

Assuming $N_{LM} = 10\,000$, $t_s = 100$, Fig. 7(a) shows the decay of the solving time T_s in eq. (20) and logarithmic growth of the communication time T_c for three different example scenarios $t_c = 1, 5$ and 10. The minimizer of the total time function, calculated by eq. (21) for three t_c values, are all greater than N_{LM} , so continual speed-up is possible by using up to N_{LM} processors and the minimum time required for $N_p = N_{LM}$ is $t_s + 2t_c \log_2 N_{LM}$. If t_c is relatively large, the communication time begins to dominate the total time more quickly at smaller number of processors. Fig. 7(b) shows that the linearity of speed-up depends on $R_{s/c}$. In reality, t_c is a hardware-dependent parameter and may vary significantly but it is generally true that for the local mesh method $R_{s/c}$ is large enough that the end users are more bound by the constraint of computational resources than by the algorithm’s ability of taking advantage of more processors.

3.5 Synthetic inversion using local meshes

A synthetic example is designed to test the local mesh method. The true model consists of two conductive prisms, 0.1 and 0.05 S m⁻¹, buried in a 0.01 S m⁻¹ uniform half-space (Fig. 8a). A synthetic airborne TEM data set is created on a 13 × 37 data grid over a 1.2 × 3.6 km area. Seven time channels of dB_z/dt data from 10⁻⁴ to 10⁻²s are simulated at 481 sounding locations 100 m apart. The synthetic data are noise-free, but we require the inversion to fit the data with 5 per cent assigned standard deviation. The mesh used for the creation of the synthetic data, and which served as the global mesh, has 155 820 cells (53 × 98 × 30). A complete forward modelling consists of four factorizations and 48 time steps, and modelling all 481 soundings on this mesh takes about 1 hr. We

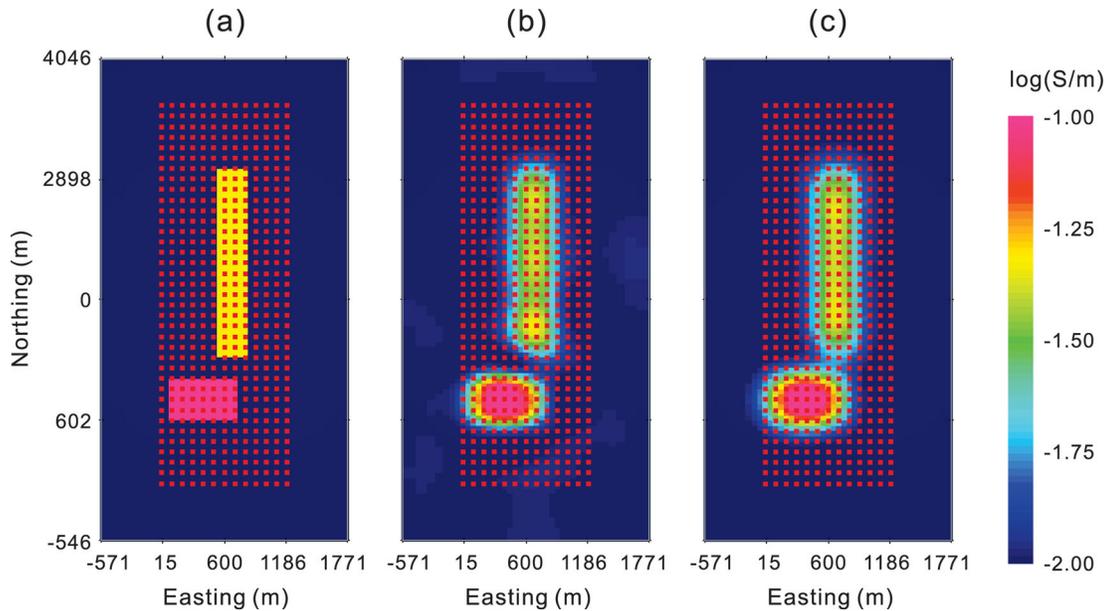


Figure 8. Synthetic inversion test: depth slice at depth of 150 m of (a) the true conductivity model, (b) the recovered model obtained by carrying out the inversion directly on the global mesh and (c) the recovered model using local mesh method. Red dots indicate the sounding locations.

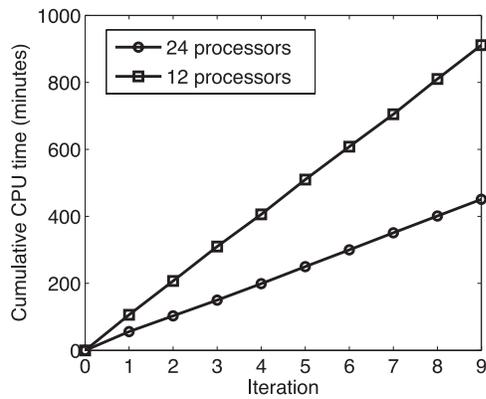


Figure 9. Cumulative time of the synthetic inversion using local mesh method on 12 and 24 processors.

first invert the synthetic data set directly on the global mesh with a 0.01 S m^{-1} half-space as the initial and reference models. The inversion, using 128 GB memory on 24 processors, takes 7 Gauss–Newton iterations and achieves the target misfit in 156 hr (see inversion model in Fig. 8b). For testing the local mesh methodology, we use the same inversion algorithm and keep the identical inversion parameters but with local meshes to compute the forward modelling and sensitivities.

The first test is carried out on 12 processors and each processor is assigned 40 local meshes. The target misfit is achieved within 15 hr after eight iterations. Fig. 8(c) shows the conductivity model recovered using the local mesh method; the two prisms are delineated with correct geometries and conductivities; this result is similar to the model recovered by the global mesh inversion (Fig. 8b). In order to test the scalability of our local mesh method, the second synthetic inversion is on 24 processors. In this case the workload on each processor is cut in half to 20 local meshes. The second inversion produces an identical inversion result, convergence curve, and data misfit, but the total CPU time is reduced to about 7.5 hr. Fig. 9 illustrates the cumulative CPU time of the inversion on 12 processors and on 24 processors. For a relatively small number of processors, the communication overhead is negligible, so a nearly linear speed-up is observed by increasing the number of processors. There is certainly great potential of further speed-up in large-scale parallelization if more processors were available to us.

4 ADAPTIVE SOUNDINGS

Although the local mesh method greatly reduces the run time of 3-D inversions, and further acceleration is achievable by adding more processors, the total number of soundings in an airborne EM survey typically numbers hundreds of thousands and this can be formidable. It is well known that airborne EM data are highly redundant, so some simple techniques are routinely used to reduce the number of soundings. It is common practice in the industry to invert a subset of the soundings based on an empirically determined coarser data grid. However, since airborne EM data are functions of space and time, we look at the selection of a subset of soundings in the data set as a problem of down-sampling continuous signals in space. In fact, a uniformly spaced grid behaves like a low-pass filter, so if the data grid is not fine enough some artifacts could be created in the model due to aliasing. For example, if the down-sampling rate is close to the spatial frequency of the bird's swing, the inversion model can be systematically biased.

Instead of using a specific grid or formula for the data reduction, we choose a down-sampling that is random in space and dynamic in the number of soundings selected at different stages of inversion. Our procedure is to number all soundings sequentially from 1 to N (total number of soundings in the survey). At each iteration of the inversion, a random sampler with uniform distribution chooses N_s (number of soundings for current iteration, $N_s < N$) integers from 1 to N . The soundings numbered with the chosen integers are the elements of a random subset of size N_s . For each iteration the inversion requests data misfits and a model update based on different, and independent, realizations of random subsets. An attractive property of random down-sampling is that the irregular grid is not frequency selective. With irregularly down-sampled soundings, the inversion produces a model consisting of two components, $m = m^{\text{reg}} + \Delta m$. The model m^{reg} is the model obtained if all the soundings are used, and the artifact Δm is a consequence of the sampling scheme. Statistically, Δm could contain contents at any spatial frequency because of the random sampler, but its magnitude is primarily controlled by N_s . It is obviously impossible to separate m^{reg} and Δm from the model recovered at a particular iteration, but if the random subset changes dynamically every iteration, if the samplings are completely uncorrelated, and if the achieved data misfit for any random subset is reduced below a prescribed tolerance, then a good estimate for m^{reg} should be obtained. If desired, final validation of the model can be tested by carrying out a single forward modelling using all of the soundings.

It can be seen that Δm must be controlled so that the recovered m is not overwhelmed by this noise. In early iterations when β is large, and large-scale features of the model are sought, a small N_s is adequate. As β decreases, and more fine-scale structure is sought, a larger N_s is needed. To find the necessary number of soundings for a particular β and for each iteration, we develop an adaptive sounding method. The method is data-driven and requires little analysis of the data prior to inversion. The adaptive sounding procedure involves two random subsets of the entire data set: an inversion subset and a test subset. A model update is calculated using the inversion subset. If this update can also sufficiently reduce the misfit for the test subset, then the model is updated and this N_s is deemed sufficient for the current β . However, if the model update proposed by the inversion subset does not reduce the misfit for the test subset, then the N_s soundings in the inversion subset is considered not to be representative. More soundings are added to proceed with the current β . With this procedure, the number of local meshes needed is minimized by not increasing N_s until necessary. Since N_s matches β closely, if the cooling of β is too fast, one has to deal with large N_s for many iterations. Therefore, we would rather choose a relatively slower cooling of β , for example, $\beta^{k+1} = 0.6 \times \beta^k$, to have more iterations with smaller N_s . Our code uses the Algorithm 1 in Appendix C.

4.1 Synthetic inversion using local mesh and adaptive soundings

The combination of using local meshes and adaptive soundings can greatly reduce the CPU time for the inversion. We illustrate this with the same synthetic data set from the two-prism model in Fig. 8 and invert with the same parameters using 24 processors. The initial and reference models are still 0.01 S m^{-1} half-space.

To implement the adaptive soundings, the initial number of soundings is set to 48. This gradually increases to 384 until the target misfit is achieved within 2.5 hr after six iterations (Fig. 10).

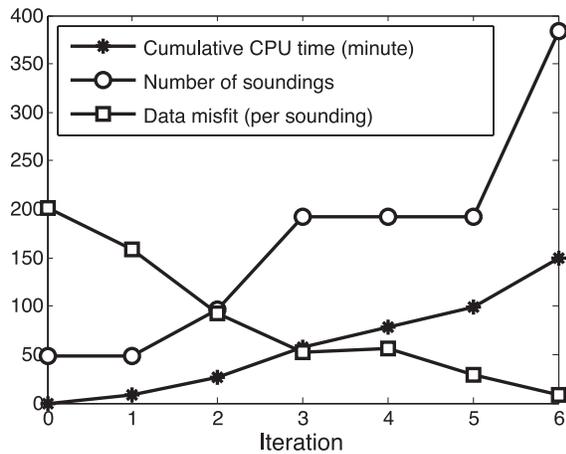


Figure 10. Summary of synthetic inversion using both local mesh and adaptive soundings

The final inversion model (Fig. 11d) is almost identical to the recovered model in the inversion which only uses local mesh method. The intermediate models created by adaptive soundings are somewhat different (Fig. 11). The first iteration only has 48 soundings, so the inversion recovers the global trend of the conductivity plus significant artifacts near surface (Fig. 11a). As the number of soundings increases, the desired targets become clearer (Figs 11b and c). By using adaptive soundings, about 60 per cent of the CPU time is saved compared to modelling all 481 soundings for every iteration and the final model in Fig. 11(d) is visually the same as the inversion model in Fig. 8 if they are rendered in the same colour scale. Although random subsets of soundings have been used, the overall coverage of data, in terms of how often a sounding is used during the inversion,

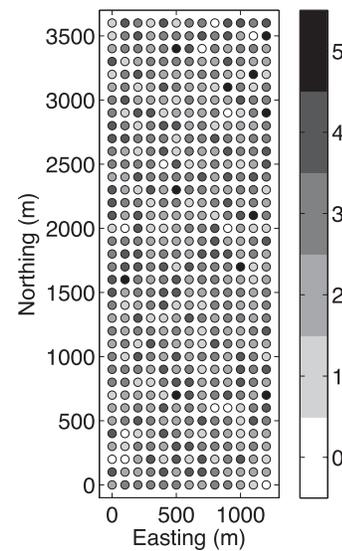


Figure 12. Counts of selection for every sounding throughout the entire inversion. Darkness indicates the times a sounding is selected and used. On average, each sounding is selected for approximately 2.4 iterations.

is adequately balanced in space (Fig. 12). At the end of the procedure there were only 10 soundings that were not used. We also note that most commercial airborne EM systems have a much denser spatial sampling rate (5–10 m) than our synthetic survey (100 m). Therefore, using adaptive soundings to find the necessary number of soundings for inversion is more economical than inverting every sounding in the overly redundant data set. We show this in the field example in the next section.

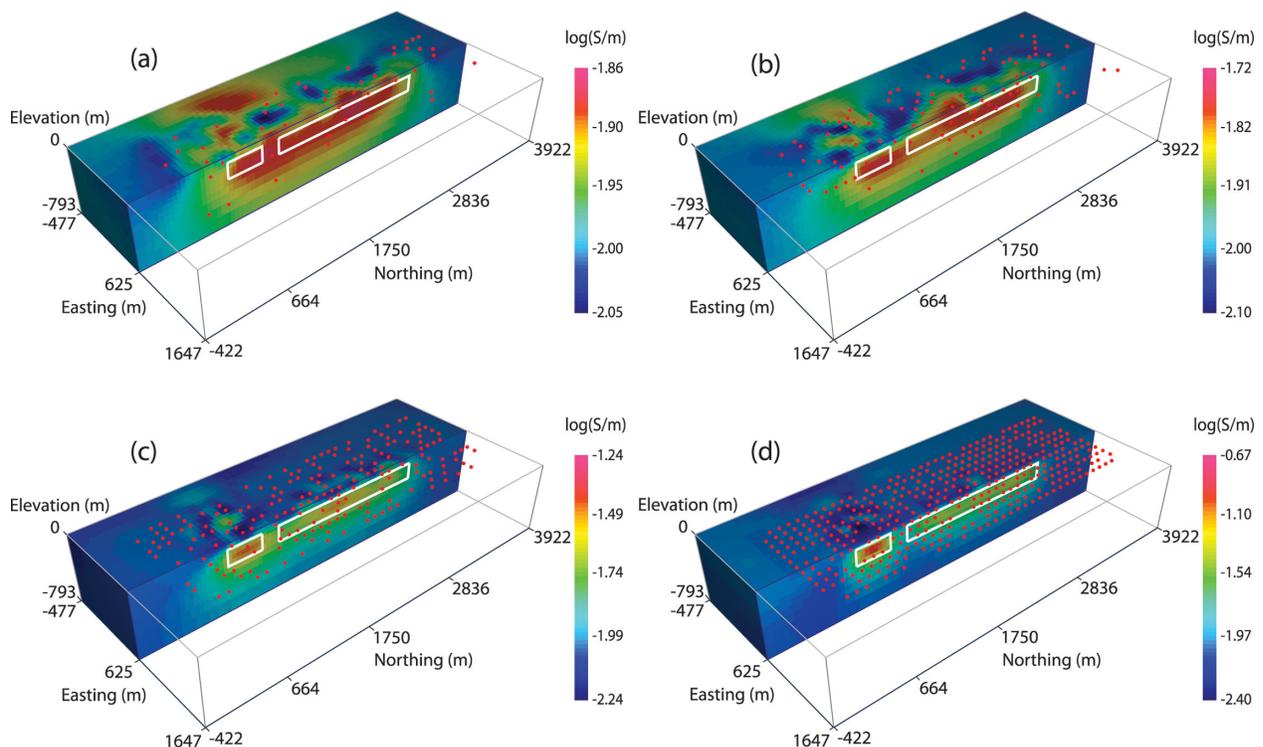


Figure 11. Conductivity models recovered at different iterations of adaptive soundings. (a) Iteration 1 with 48 soundings; (b) iteration 2 with 96 soundings; (c) iteration 4 with 192 soundings and (d) iteration 6 with 384 soundings. The red dots indicate the sounding locations. The white boxes outline the exact locations of the two prisms.

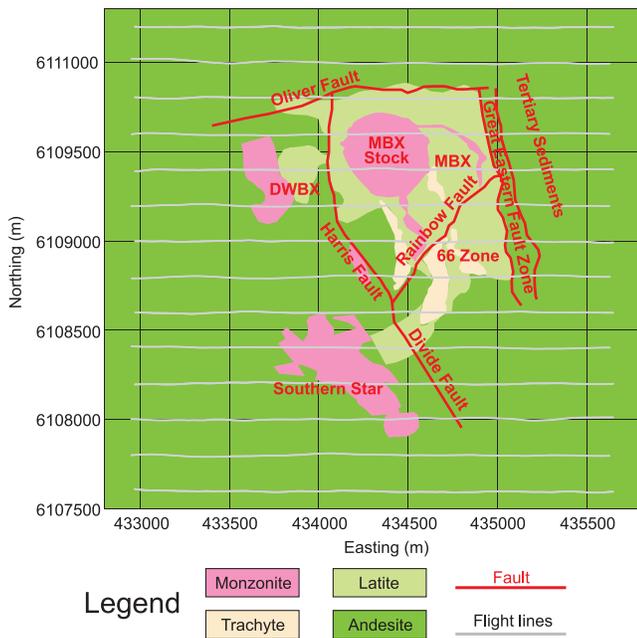


Figure 13. Geology and VTEM survey at Mt Milligan porphyry deposit in British Columbia, Canada.

5 FIELD DATA EXAMPLE

The local mesh and adaptive sounding methodology developed in this paper are intended to alleviate the challenge of interpreting the entire airborne EM data set in 3-D in real applications. We demonstrate the efficiency of our new method by inverting an airborne EM data set over Mt Milligan, British Columbia, Canada for mineral exploration. Mt Milligan is a copper and gold porphyry deposit within the Early Mesozoic Quesnel Terrane, a Late Triassic to Early Jurassic magmatic arc complex that lies along the western North American continental margin. Previous geologic and geophysical work (Oldenburg *et al.* 1997; Yang & Oldenburg 2012a) has revealed that the mineralization is associated with the monzonite stocks that intruded into the basaltic volcanoclastic rocks (Fig. 13). The airborne EM data set collected at Mt Milligan is versatile time-domain electromagnetic (VTEM) data that uses a helicopter-borne system with coincident transmitter loop and receiver coil. The survey measures the vertical component of the time derivative of the magnetic field (dB_z/dt) from 99 to 9245 μ s and contains 14 flight lines, 200 m apart and 14 362 soundings covering a total area of 7.3 km² (Fig. 13). The geophysical target is the conductive alteration that is rich in pyrite and chalcopyrite surrounding the resistive monzonite. The same data set was processed in 3-D in Yang & Oldenburg (2012a). They successfully recovered the most prominent intrusive monzonite stock MBX that was not recoverable by using 1-D inversion. However, due to the size of the problem, in that attempt only a small portion of the data set near the MBX stock was inverted by directly solving the inverse problem on a global mesh at 50 m resolution. Although a multilevel meshing strategy was developed to speed up the inversion in Yang & Oldenburg (2012a), the 3-D inversion still took about 18 hr on two cluster nodes with 24 processors. Here we implement our local mesh and adaptive sounding methods on the same computers for a fair comparison of efficiency.

In the local mesh method, the forward modelling is never carried out on the global mesh, so the number of cells in the global mesh becomes less crucial. We design a global mesh with 443 520 (88 × 84 × 60) cells to hold the entire survey area at 50 m horizontal

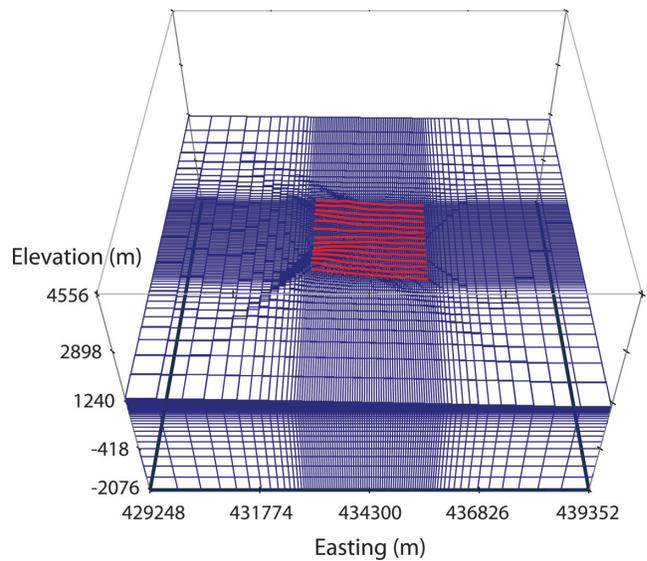


Figure 14. Global mesh for Mt Milligan VTEM 3-D inversion. The VTEM survey locations are indicated by the red lines.

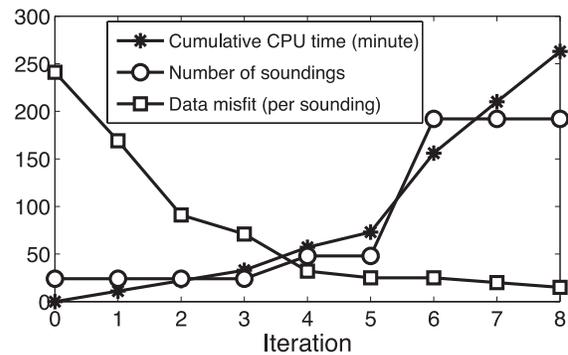


Figure 15. Summary of Mt Milligan VTEM 3-D inversion.

resolution and 20 m vertical resolution for the topography (Fig. 14). This mesh, along with the large number of soundings, is too large for the algorithm in Oldenburg *et al.* (2013) to be practically carried out. The local mesh inversion starts with a 0.002S m⁻¹ uniform initial and reference model. Fig. 15 summarizes three key parameters of the inversion: the cumulative run time, the number of soundings used at each iteration and the data misfit. For early iterations only 24 soundings were needed to build up the large-scale conductivity distribution. Smaller-scale features were built up by adding soundings. The number of soundings used in the final iteration is only 192, which implies that the necessary number of sounding for 3-D VTEM inversion at Mt Milligan is about 200, only 1.4 per cent of the total number of soundings acquired in the survey. During the entire inversion procedure, information from 744 soundings (5 per cent of the total number of soundings) was incorporated into the final model and thus 744 local meshes were needed. The total time was 4.3 hr and we anticipate more speedup if additional processors were available. A depth slice of the 3-D inversion model over the whole survey area at 950 m elevation and a cross section cutting the major stock MBX are shown in Fig. 16. Like the small-scale inversion in Yang & Oldenburg (2012a), the resistive MBX stock is clearly delineated by 3-D inversion.

Lastly, we carry out a complete forward modelling using the recovered model in Fig. 16 and all 14 362 soundings. The misfit normalized by the total number of soundings is $\phi_d/N = 19$ which compares satisfactorily with $\phi_d/N = 15$ estimated by 192 random

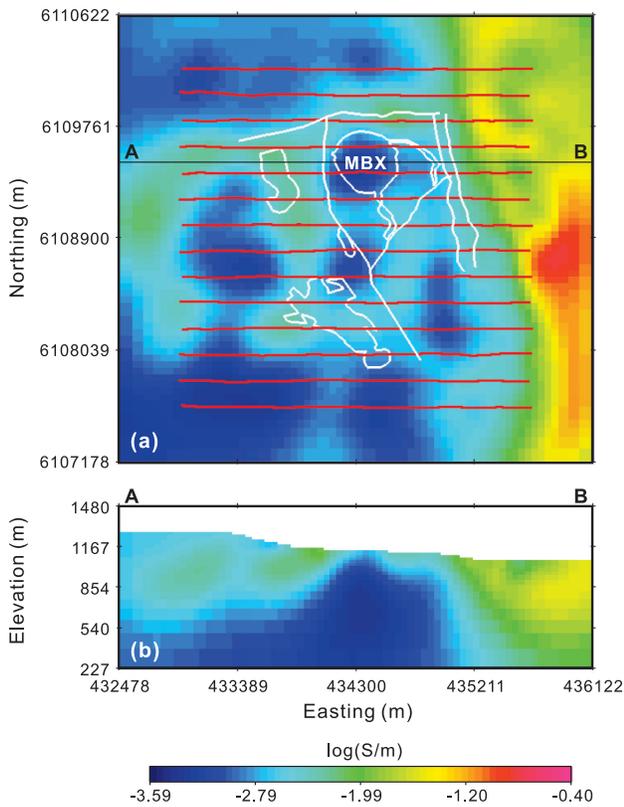


Figure 16. Conductivity model of Mt Milligan VTEM 3-D inversion: (a) a depth slice at elevation = 950 m and (b) a cross-section A–B at 6109500N. White lines outline the major monzonite stocks and the faults. VTEM flight lines are indicated by the red lines.

soundings in the last iteration and $\phi_d/N = 17$ we have achieved in Yang & Oldenburg (2012a). The time channel grids of the observed and predicted data at 680 μ s in Fig. 17 show the 3-D model recovered by using local mesh and adaptive soundings can reasonably reproduce the observed data.

6 CONCLUSIONS

AEM data are difficult to invert in 3-D because of the large computations required to handle multiple transmitters and large meshes

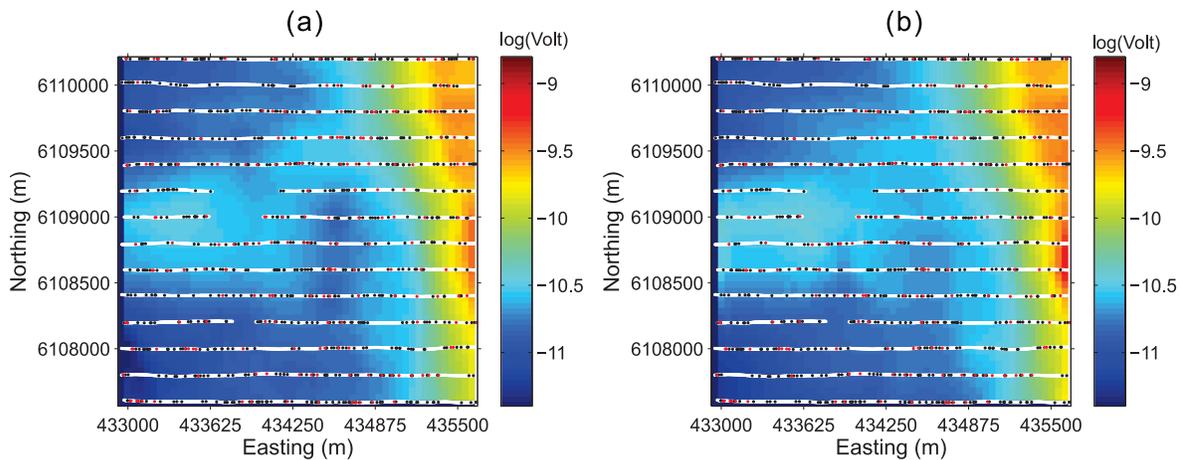


Figure 17. Data grid of time channel at 680 μ s. (a) Observed dBz/dt data; (b) Predicted dBz/dt data. The white lines indicate 14 362 soundings approximately 3 m apart to each other. The scattered black and red dots show the 744 and 192 soundings used in all iterations and the last iteration of inversion, respectively.

needed to represent the volume being modelled. We propose two methods for speeding up the operations. Local meshes are used for computing the forward responses and sensitivities, and an adaptive sounding strategy is used to reduce the number of soundings employed at each iteration.

A local mesh, which is optimally designed to handle a single sounding, can yield good accuracy with a modest number of cells. This is true even when using structured rectangular grids as done here. As such the modelling matrix and its factorization, for a single or a few soundings, can be readily stored on a single processor. An entire forward modelling or sensitivity calculation is easily distributed among an array of processors. In the parallel computing environment the inter-processor communication, which is usually a serious concern when carrying out large-scale modelling in parallel, is minimized by confining a complete EM modelling problem of local mesh in one processor. The communication only occurs when the host thread broadcasts commands to the workers and collects computed results from the workers. For this type of problem the benefit of adding more processors generally outplays the increased communication time.

The second area of progress regards the number of soundings used at each iteration in the inversion. It is well known that modelling every sounding in an airborne data set is not necessary. We propose a random and dynamic down-sampling method, which we call adaptive sounding. It is essentially a random sampler with an adaptive number of soundings selected for each iteration. Fewer soundings are selected in early iterations to build up large-scale structure and more soundings are added later as the regularization is relaxed and additional structure is needed to fit the data. Further development of the adaptive methodology is envisioned but the current strategy has worked well thus far. The procedure has an additional optional check in that the user may carry out a complete forward modelling at the end, using all the soundings, to validate the constructed model.

At an overall level, our current inversion algorithm parallels that outlined in Oldenburg *et al.* (2013) but the introduction of using/storing the direct solver on a local rather than a global mesh, and the use of adaptive sounding, has resulted in greatly improved performance. In a test using synthetic data from a two-prism model we reduced the inversion time, compared to the original method, by more than a factor of 60. That example was small and further disparity between the old and modified method increases as the size of the problem and the number of processors increase. In a field

example, which was too large to be practically solved two years ago, inversion with our new method took only 4.3 hr on 24 processors. This is a very substantial speedup and represents major step toward making routine 3-D inversion of airborne EM data a reality.

The ideas and methods presented in this paper are generic to other inverse problems that involve either a large number of localized sources or a large but decomposable source. This includes dc resistivity and other EM surveys. Our approach requires no specific method of discretization and the global and the local meshes do not need to be the same type (e.g. a global mesh could be rectangular while the local mesh is tetrahedral). We will explore these extensions in a follow-up paper.

ACKNOWLEDGEMENTS

This work was supported by NSERC CRD and IRC programs and sponsoring companies: Cameco, Teck, Newmont, Barrick, Vale, Xstrata Nickel and Anglo American. The authors thank Geoscience BC for the VTEM data set at Mt Milligan and Roman Shekhtman for his assistance of programming.

REFERENCES

- Alumbaugh, D.L., Newman, G.A., Prevost, L. & Shadid, J.N., 1996. Three-dimensional wideband electromagnetic modeling on massive parallel computers, *Radio Sci.*, **1996**(1), 1–23.
- Amestoy, P.R., Guermouche, A., L'Excellent, J.Y. & Pralet, S., 2006. Hybrid scheduling for the parallel solution of linear systems, *Parallel Comput.*, **32**(2), 136–156.
- Bossavit, A., 1998. *Computational Electromagnetism. Variational Formulation, Complementarity, Edge Elements*, Academic Press.
- Brodie, R. & Sambridge, M., 2006. A holistic approach to inversion of time-domain airborne EM, *ASEG Extended Abstracts*, **2006**(1), 1–4.
- Commer, M. & Newman, G., 2004. A parallel finite-difference approach for 3D transient electromagnetic modeling with galvanic sources, *Geophysics*, **69**(5), 1192–1202.
- Commer, M. & Newman, G.A., 2006a. An accelerated time domain finite difference simulation scheme for three-dimensional transient electromagnetic modeling using geometric multigrid concepts, *Radio Sci.*, **41**, RS3007, doi:10.1029/2005RS003413.
- Commer, M. & Newman, G.A., 2006b. Large scale 3D EM inversion using optimized simulation grids nonconformal to the model space, in *Proceedings of the 2008 SEG Annual Meeting, Technical Program Expanded Abstracts*, 2008 November 9–14, Las Vegas, Nevada, pp. 760–764.
- Commer, M. & Newman, G.A., 2008. New advances in three-dimensional controlled-source electromagnetic inversion, *Geophys. J. Int.*, **172**(2), 513–535.
- Commer, M., Newman, G.A., Carazzone, J.J., Dickens, T.A., Green, K.E., Wahrmond, L.A., Willen, D.E. & Shiu, J., 2008. Massively-parallel electrical-conductivity imaging of hydrocarbons using the IBM Blue Gene/L supercomputer, *IBM J. Res. Dev.*, **52**(1.2), 93–103.
- Cox, L.H., Wilson, G.A. & Zhdanov, M.S., 2010. 3D inversion of airborne electromagnetic data using a moving footprint, *Explor. Geophys.*, **41**(4), 250–259.
- Davydycheva, S., Druskin, V. & Habashy, T., 2003. An efficient finite-difference scheme for electromagnetic logging in 3D anisotropic inhomogeneous median, *Geophysics*, **68**(5), 1525–1536.
- Eaton, P.A., 1998. Application of an improved technique for interpreting transient electromagnetic data, *Explor. Geophys.*, **29**(2), 175–183.
- Farquharson, C.G. & Oldenburg, D.W., 1993. Inversion of time-domain electromagnetic data for a horizontally layered Earth, *Geophys. J. Int.*, **114**(3), 433–442.
- Fraser, D.C., 1978. Resistivity mapping with an airborne multi-coil electromagnetic system, *Geophysics*, **43**(1), 144–172.
- Fullagar, P.K. & Reid, J.E., 2001. Emax conductivity-depth transformation of airborne TEM data, in *ASEG Extended Abstracts*, **2001**(1), 1–4.
- Fullagar, P.K., Vrbancich, J. & Pears, G., 2010. Geologically-constrained 1D TEM inversion, *ASEG Extended Abstracts*, **2010**(1), 1–4.
- Haber, E. & Ascher, U.M., 2001. Fast finite volume simulation of 3D electromagnetic problems with highly discontinuous coefficients, *SIAM J. Scient. Comput.*, **22**(6), 1943–1961.
- Haber, E., Oldenburg, D.W. & Shekhtman, R., 2007. Inversion of time domain three-dimensional electromagnetic data, *Geophys. J. Int.*, **171**(2), 550–564.
- Keating, P.B. & Crossley, D.J., 1990. The inversion of time-domain airborne electromagnetic data using the plate model, *Geophysics*, **55**(6), 705–711.
- Lane, R., Green, A., Golding, C., Owers, M., Pik, P., Plunkett, C., Sattel, D. & Thorn, B., 2000. An example of 3D conductivity mapping using the TEMPEST airborne electromagnetic system, *Explor. Geophys.*, **31**(2), 162–172.
- Macnae, J., 1998. Fast AEM data processing and inversion, *Explor. Geophys.*, **29**(2), 163–169.
- Macnae, J., Mortimer, R. & Gilgallon, K., 2010. Deep conductor delineation through improved EMFlow data processing, *ASEG Extend. Abstr.*, **2010**(1), 1–4.
- Newman, G.A. & Alumbaugh, D.L., 1997. Three-dimensional massively parallel electromagnetic inversion—I. Theory, *Geophys. J. Int.*, **128**(2), 345–354.
- Oldenburg, D., Haber, E. & Shekhtman, R., 2013. Three dimensional inversion of multisource time domain electromagnetic data, *Geophysics*, **78**(1), E47–E57.
- Oldenburg, D.W., Li, Y. & Ellis, R.G., 1997. Inversion of geophysical data over a copper gold porphyry deposit: a case history for Mt. Milligan, *Geophysics*, **62**(5), 1419–1431.
- Palacky, G.J., 1993. Use of airborne electromagnetic methods for resource mapping, *Adv. Space Res.*, **13**(11), 5–14.
- Palacky, G.J. & West, G.F., 1973. Quantitative interpretation of INPUT AEM measurements, *Geophysics*, **38**(6), 1145–1158.
- Palacky, G.J. & West, G.F., 1991. Airborne electromagnetic methods, in *Electromagnetic Methods in Applied Geophysics*, Vol. 2: Applications, Part A and B, pp. 811–877, ed. Nabighian, M.N., Society of Exploration Geophysicists Investigations in Geophysics.
- Raiche, A., 2004. Practical 3D airborne EM inversion in complex terranes, *ASEG Extend. Abstr.*, **2004**(1), 1–4.
- Sattel, D., 2005. Inverting airborne electromagnetic (AEM) data with Zohdy's method, *Geophysics*, **70**(4), G77–G85.
- Vallee, M.A. & Smith, R.S., 2009. Inversion of airborne time-domain electromagnetic data to a 1D structure using lateral constraints, *Near Surf. Geophys.*, **7**(1), 63–71.
- Wilson, G.A., Cox, L.H. & Zhdanov, M.S., 2010. Practical 3D inversion of entire airborne electromagnetic surveys, *Preview*, **2010**(146), 29–33.
- Wolffgram, P. & Karlik, G., 1995. Conductivity-depth transform of GEOTEM data, *Explor. Geophys.*, **26**(3), 179–185.
- Wolffgram, P., Sattel, D. & Christensen, N.B., 2003. Approximate 2D inversion of AEM data, *Explor. Geophys.*, **34**(2), 29–33.
- Xie, G., Li, J., Majer, E., Zuo, D. & Oristaglio, M., 2000. 3-D electromagnetic modeling and nonlinear inversion, *Geophysics*, **65**(3), 804–822.
- Yang, D. & Oldenburg, D., 2012a. Three-dimensional inversion of airborne time-domain electromagnetic data with applications to a porphyry deposit, *Geophysics*, **77**(2), B23–B34.
- Yang, D. & Oldenburg, D., 2012b. Practical 3D inversion of large airborne time domain electromagnetic data sets, *ASEG Extend. Abstr.*, **2012**(1), 1–4.
- Yee, K., 1966. Numerical solution of initial boundary value problems involving Maxwell's equations in isotropic media, *IEEE Trans. Antennas Propag.*, **14**(3), 302–307.
- Zaslavsky, M., Druskin, V. & Knizhnerman, L., 2011. Solution of 3D time-domain electromagnetic problems using optimal subspace projection, *Geophysics*, **76**(6), F339–F351.
- Zaslavsky, M., Druskin, V., Abubakar, A., Habashy, T. & Simoncini, V., 2013. Large-scale Gauss-Newton inversion of transient controlled-source electromagnetic measurement data using the model reduction framework, *Geophysics*, **78**(4), E161–E171.

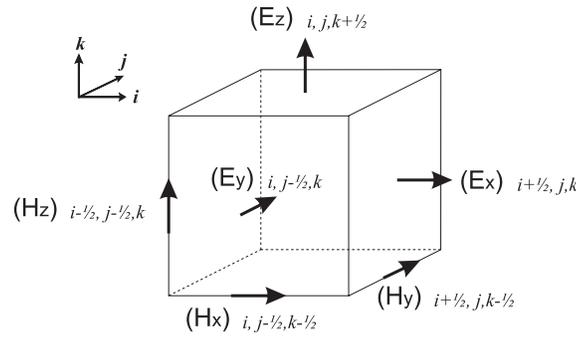


Figure A1. Staggered discretization in 3-D. \mathbf{H} is on the cell edges and \mathbf{E} on the cell faces.

APPENDIX A: 3-D FORWARD MODELLING ALGORITHM

Finite volume method integrates the differential Maxwell's equations over elemental control volumes aligned with the mesh grid. The modelling domain is discretized by a tensor mesh consisting of many rectangular cells. We use a staggered grid (Yee 1966) with \mathbf{H} fields on the edges, \mathbf{E} fields on the faces and the conductivity at the cell centres (Fig. A1).

Upon discretization, eq. (3) can be written in a discrete form

$$[\mathbf{C}^T \text{diag}(\mathbf{A}_v e^m)^{-1} \mathbf{C} + \gamma^{i+1} \mathbf{I}] h^{i+1} = \mathbf{C}^T \text{diag}(\mathbf{A}_v e^m)^{-1} \mathbf{j}_s^{i+1} + \gamma^{i+1} h^i, \quad (\text{A1})$$

where \mathbf{C} is a curl operator mapping a field from edges to faces, \mathbf{C}^T is also a curl operator but from faces to edges, \mathbf{A}_v is a harmonic averaging matrix mapping the conductivity values from cell centres to faces, e^m is a vector of conductivities for all the cell centres, $\gamma^{i+1} = \mu_0 / \delta t$ at time step $i+1$, h is a vector of \mathbf{H} fields in x , y and z directions. \mathbf{j}_s is a vector of the current density field \mathbf{J}_s due to the source. In airborne EM, the sources are closed loops much smaller than the scale of the survey. We first analytically compute the magnetic vector potential A of the loop source to ensure the source is divergence-free, then obtain the primary current density field

$$\mathbf{J}_s = \nabla \times \mu_0^{-1} \nabla \times A. \quad (\text{A2})$$

Eq. (A1) is for the \mathbf{H} field at a particular time step. Expressing the Maxwell matrix as $\mathbf{A}(m, \delta t)$ and the first term in the right-hand side as q , we have the system of equations for the entire modelling in time

$$\begin{pmatrix} \mathbf{A}(m, \delta t_1) & & & & \\ -\gamma(\delta t_2) \mathbf{I} & \mathbf{A}(m, \delta t_2) & & & \\ & & \ddots & & \\ & & & \ddots & \\ & & & & \ddots \end{pmatrix} \begin{pmatrix} h^1 \\ h^2 \\ \vdots \\ \vdots \\ \vdots \end{pmatrix} = \begin{pmatrix} q^1(m) + \gamma(\delta t_1) h^0 \\ q^2(m) \\ \vdots \\ \vdots \\ \vdots \end{pmatrix} \quad (\text{A3})$$

or in symbolic form

$$\mathbf{B}h = \text{rhs}. \quad (\text{A4})$$

If the symmetric positive definite matrices $\mathbf{A}(m, \delta t_i)$ in eq. (A3) are factorized into \mathbf{L} and \mathbf{L}^T and stored, the operator \mathbf{B}^{-1} is available by forward time stepping and \mathbf{B}^{-T} by backward time stepping. The forward modelled data is

$$F(m) = \mathbf{Q} \mathbf{B}^{-1} \text{rhs}, \quad (\text{A5})$$

where \mathbf{Q} is a sparse space-time interpolation matrix mapping the fields from cell edges to the receiver location and from time steps to the moments the data are recorded. We also note \mathbf{A} only changes if δt changes, so in practice, we usually choose a few different δt 's to match the different timescales from early to late in airborne TEM survey. In this way, only a few \mathbf{A} matrices need to be factorized.

APPENDIX B: CALCULATION OF SENSITIVITY

Differentiating both sides of eq. (A4) with respect to m yields

$$\frac{\partial h}{\partial m} = \mathbf{B}^{-1} \left(\frac{\partial \text{rhs}}{\partial m} - \frac{\partial \mathbf{B}}{\partial m} h \right) = \mathbf{B}^{-1} \mathbf{G}. \quad (\text{B1})$$

Then the sensitivity matrix is

$$\mathbf{J} = \mathbf{Q} \frac{\partial h}{\partial m} = \mathbf{Q} \mathbf{B}^{-1} \mathbf{G}, \quad (\text{B2})$$

and its transpose is

$$\mathbf{J}^T = \mathbf{G}^T \mathbf{B}^{-T} \mathbf{Q}^T, \quad (\text{B3})$$

where \mathbf{Q} is again a space-time interpolation matrix. As the operators \mathbf{B}^{-1} and \mathbf{B}^{-T} are already available in forward modelling (Appendix A), we derive other components in \mathbf{G} as

$$\frac{\partial \text{rhs}}{\partial m} = \begin{pmatrix} \frac{\partial q^1}{\partial m} \\ \frac{\partial q^2}{\partial m} \\ \vdots \\ \vdots \end{pmatrix} = \begin{pmatrix} -\mathbf{C}^T \text{diag}(j_s^1) \text{diag}[(\mathbf{A}_v e^m)^{-2}] \mathbf{A}_v \text{diag}(e^m) \\ -\mathbf{C}^T \text{diag}(j_s^2) \text{diag}[(\mathbf{A}_v e^m)^{-2}] \mathbf{A}_v \text{diag}(e^m) \\ \vdots \\ \vdots \end{pmatrix}, \quad (\text{B4})$$

and

$$\frac{\partial \mathbf{B}}{\partial m} h = \begin{pmatrix} \frac{\partial}{\partial m} [\mathbf{A}(m)h^1] \\ \frac{\partial}{\partial m} [\mathbf{A}(m)h^2] \\ \vdots \\ \vdots \end{pmatrix} = \begin{pmatrix} -\mathbf{C}^T \text{diag}(\mathbf{C}h^1) \text{diag}[(\mathbf{A}_v e^m)^{-2}] \mathbf{A}_v \text{diag}(e^m) \\ -\mathbf{C}^T \text{diag}(\mathbf{C}h^2) \text{diag}[(\mathbf{A}_v e^m)^{-2}] \mathbf{A}_v \text{diag}(e^m) \\ \vdots \\ \vdots \end{pmatrix}. \quad (\text{B5})$$

In order to save memory, we explicitly compute and store the sensitivity on the local meshes. Since the number of data is always much less than the number of model parameters, we compute \mathbf{J}^T in eq. (B3) column by column. Computation of each column of \mathbf{J}^T is equivalent to one complete forward modelling involving backward time stepping (\mathbf{B}^{-T} operation). After the sensitivity is stored, the matrices and vectors associated with \mathbf{B} , \mathbf{G} and \mathbf{Q} can all be deleted from the memory.

APPENDIX C: IMPLEMENTATION OF ADAPTIVE SOUNDINGS

Adaptive soundings require frequent solves of the inverse problem for the proposal of model update δm and the forward problems to test the model update. Therefore, in practice, we want the computation of local mesh design, matrix factorizations and forward problems done for the test subsets to be recycled for the inverse problem in the next iteration. The work flow of our inversion code using adaptive soundings is presented in Algorithm 1. Each Gauss–Newton iteration involves two distinct subsets of soundings: the inversion subset for solving δm in eq. (12) and another test subset to evaluate whether the proposed model update is acceptable. If the post-update data misfit is sufficiently reduced (controlled by a factor μ), the model is updated and the test subset, already having the forward solutions $F(m + \delta m)$ for the updated model, can be used as the new inversion subset in the next iteration after additional computations of the sensitivity $\mathbf{J}(m + \delta m)$. If the proposed δm is declined, the test subset, already having the forward solutions $F(m)$ for the original model, is appended to the existing inversion subset after additional computations of the sensitivity $\mathbf{J}(m)$ for the next Gauss–Newton step with doubled number of soundings N_s .

Algorithm 1. Inversion with adaptive soundings

Initialization:

- Operation on *inversion subset*
 - Select random N_s soundings
 - Design local meshes for m
 - Compute and store forward responses $F(m)$
 - Compute and store sensitivity $\mathbf{J}(m)$
 - Clear matrices \mathbf{B}^{-1} , \mathbf{Q} and \mathbf{G}

repeat

- Operation on *test subset*
 - Select random N_s soundings
 - Design local meshes for m
 - Compute and store forward responses $F(m)$
 - $\text{misfit}(m) = \frac{1}{N_s} \|\mathbf{W}_d(F(m) - d^{\text{obs}})\|^2$
 - Clear matrices \mathbf{B}^{-1} and \mathbf{Q}
- Operation on *inversion subset*
 - $\delta m = -(\mathbf{J}^T \mathbf{W}_d \mathbf{W}_d \mathbf{J} + \beta \mathbf{W}_m^T \mathbf{W}_m)^{-1} (\mathbf{J}^T \mathbf{W}_d^T \mathbf{W}_d [F(m) - d^{\text{obs}}] + \beta \mathbf{W}_m^T \mathbf{W}_m (m - m_0))$
- Operation on *test subset*
 - Re-design local meshes for $m + \delta m$
 - Compute and store forward responses $F(m + \delta m)$
 - $\text{misfit}(m + \delta m) = \frac{1}{N_s} \|\mathbf{W}_d(F(m + \delta m) - d^{\text{obs}})\|^2$
- if $\text{misfit}(m + \delta m) < \mu \cdot \text{misfit}(m)$ then
 - Operation on *test subset*
 - Compute and store sensitivity $\mathbf{J}(m + \delta m)$
 - Empty *inversion subset*
 - inversion subset* \leftarrow *test subset* with $F(m + \delta m)$ and $\mathbf{J}(m + \delta m)$

Algorithm 1. (Continued.)

```
Empty test subset
Reduce  $\beta$ 
 $m \leftarrow m + \delta m$ 
else
  Operation on test subset
  Compute and store sensitivity  $\mathbf{J}(m)$ 
  inversion subset  $\leftarrow$  inversion subset + test subset with  $F(m)$  and  $\mathbf{J}(m)$ 
  Empty test subset
   $N_s = N_s \times 2$ 
end if
until  $\text{misfit}(m) < \text{tol}$ 
```
