

PDE-based geophysical modelling using finite elements: examples from 3D resistivity and 2D magnetotellurics

R Schaa, L Gross and J du Plessis

School of Earth Sciences, The University of Queensland, St Lucia QLD 4072, Australia

E-mail: r.schaa@uq.edu.au, l.gross@uq.edu.au and j.plessis@uq.edu.au

Received 23 June 2015, revised 10 November 2015

Accepted for publication 10 December 2015

Published 30 March 2016



Abstract

We present a general finite-element solver, *escript*, tailored to solve geophysical forward and inverse modeling problems in terms of partial differential equations (PDEs) with suitable boundary conditions. *Escript*'s abstract interface allows geoscientists to focus on solving the actual problem without being experts in numerical modeling. General-purpose finite element solvers have found wide use especially in engineering fields and find increasing application in the geophysical disciplines as these offer a single interface to tackle different geophysical problems. These solvers are useful for data interpretation and for research, but can also be a useful tool in educational settings. This paper serves as an introduction into PDE-based modeling with *escript* where we demonstrate in detail how *escript* is used to solve two different forward modeling problems from applied geophysics (3D DC resistivity and 2D magnetotellurics). Based on these two different cases, other geophysical modeling work can easily be realized. The *escript* package is implemented as a *Python* library and allows the solution of coupled, linear or non-linear, time-dependent PDEs. Parallel execution for both shared and distributed memory architectures is supported and can be used without modifications to the scripts.

Keywords: finite element method, parallel computing, electrical resistivity, magnetotellurics

(Some figures may appear in colour only in the online journal)

1. Introduction

Numerical methods are routinely employed to calculate the geophysical response of models with known physical properties. Particularly 2D and 3D models in general require numerical solvers, except for simple geometries which exhibit a certain symmetry such as spheres and cylinders. Commonly tailor-made numerical modeling software is used within each specific geophysical discipline and consequently there exists a large collection of proprietary and open-source geophysical modeling software. In this paper we present a general finite-element solver, called *escript*, which is capable of solving a wide range of different geophysical modeling problems if they can be written in terms of partial differential equations (PDEs) with suitable boundary and initial conditions. Specifically we employ *escript* for modeling problems in applied geophysics,

here demonstrated with examples of 3D direct current (DC) resistivity and 2D magnetotelluric forward modeling. Note that *escript* is also capable of PDE-constrained inversion for geophysical applications, however not demonstrated here. This paper was motivated by the work on multi-geophysical coupling in a hydrogeophysical context involving DC Resistivity and magnetotelluric measurements, and is the first to show in mathematical detail how *escript* is used to solve PDE-based geophysical problems. The paper serves as an introduction into PDE-based modeling with *escript* for geophysical applications. Based on the descriptions here, other more complicated multiphysics problems can be relatively easily solved.

The *escript* package is an extension of *Python*, and can be imported just like any other package in *Python*. By using an object oriented programming style it provides a flexible and easy-to-use programming environment for numerical

simulations, whilst allowing for fast solutions of large models by performing time-intensive calculations in C++ and C. The code has been efficiently parallelised using MPI (mpi-forum.org), OpenMP (openmp.org) and hybrid modes; scripts can run on desktop computers as well as highly parallel supercomputers without changes. PDEs are solved on structured or unstructured 2D or 3D meshes, which facilitates modelling of complex and geologically realistic structures if required. *Escript*'s abstract interface allows geoscientists to focus on solving the actual problem without being experts in numerical modelling; *escript* is actively developed at the Centre for Geoscience Computing, University of Queensland and is released under the Open Software Licence v3.0. Up-to-date information, documentation and support are found on <https://launchpad.net/escript-finley>.

General-purpose finite element solvers have found wide use especially in engineering fields using well-known commercial software packages such as Comsol, Abaqus or Ansys among others; these packages have also been applied for geophysical modelling (e.g. Dyksterhuis *et al* 2005, Zhang 2010, Butler and Sinha 2012, Ikard *et al* 2012). Open-source projects which are used in the geophysical community include FEniCS, OpenFoam, Code_Aster and Moose, among others. FEniCS started out as a project to automate the solution of mathematical models and takes a modular approach for solving PDEs. In a geophysical context, examples of its use are in turbulent flow and pumping test modelling (e.g. Mortensen *et al* 2011, Bakhos *et al* 2014). More examples can be found in Logg *et al* (2012). OpenFoam offers a wide range of modules in the context of fluid flows involving chemical reactions (e.g. Berselli *et al* 2015). Code_Aster focuses on structural mechanics problems (e.g. Rohmer 2014), and was also used for coupled hydro-mechanical modelling (Seyedi *et al* 2009). Moose is a relatively new finite element framework developed by the Idaho National Laboratory. It started as a project to tackle nuclear physics problems but has found applications in geophysical areas as well (e.g. Guo *et al* 2013). Most of the mentioned packages are GUI driven, which offers ease-of-access, however repeatability is arguably better served via a programming interface such as FEniCS or *escript*.

The *escript* solver has been used in a geophysical context for joint inversion of magnetic and gravity data (Gross and Kemp 2013), for erosion in porous media (Muhlhaus *et al* 2014), for reactive transport modelling in porous media (Poulet *et al* 2012), for shear band formation (Finzi *et al* 2013), and the growth of lava domes (Gross *et al* 2007). In this paper we demonstrate the application of *escript* in applied geophysics to solve the 3D DC resistivity forward problem and the 2D magnetotelluric forward problem. Both methods evaluate the geoelectric subsurface resistivity structure however with different approaches. In DC resistivity, current is injected into the ground via electrodes, which sets up an electric potential as a function of the subsurface resistivity distribution. The MT method utilises the naturally occurring fluctuations of the geo-electromagnetic fields as an inductive source to image the subsurface resistivity.

Given a known 3D resistivity distribution, the DC resistivity forward modelling solves a Poisson-type PDE for the

electric potential. This involves Dirac delta functions as source terms and mixed boundary conditions. One of the main difficulties in calculating the DC resistivity response (i.e. solving for the electric potential) arises from the source term containing Dirac delta functions, which leads to singularities in the solution. Several different numerical approaches have been followed to solve the resistivity equation. For example Ma (2002) employed a boundary element method, a special case of the integral equation method, for DC resistivity modelling of 3D inhomogeneous bodies buried in a layered earth. The boundary element method was also employed by Blome *et al* (2009) for singularity removal to calculate the primary potential. Most numerical schemes employ finite-difference (FDM) or finite-element (FEM) methods, which are suitable for modelling complex 3D geometries (Li and Spitzer 2002). Dey and Morrison (1979) developed the first 3D finite-difference algorithm based on self-adjoint difference equations, which were obtained by integrating the continuity equation over elemental volumes. They furthermore incorporated a mixed boundary condition based on the asymptotic behaviour of the potential field in a homogeneous medium, which had the advantage of reducing the spatial extent of the domain and hence computational costs.

The finite-difference scheme from Lowry *et al* (1989) mathematically removed the Dirac delta singularities from the modelling process and reintroduced them later. This approach is also known as the secondary potential approach. In this approach the sought-after total potential is split into a primary potential, containing the Dirac sources, and a non-singular secondary potential caused by resistivity contrasts. The primary potential can be solved via analytical solutions, applicable for flat topography geometries.

Pain *et al* (2002) developed parallelised 2D/3D finite-element code, where the source-terms at nodes are set to rescaled basis functions or are approximated by Gaussian functions if the sources are not coinciding with mesh-nodes. Li and Spitzer (2002) used the singularity removal technique for a finite-difference and finite-element scheme. They used a horizontally layered earth or a vertical contact for the primary potential. The results they obtained were very similar for both schemes. Rücker *et al* (2006) employed quadratic-shape finite elements together with the secondary potential approach using a homogeneous half-space. The resistivity was set to the resistivity values at the location of the source-electrodes. Incorporation of topography into the primary potential was numerically evaluated via mesh-refinement at the location of the source-electrodes.

Penz *et al* (2013) employed a finite-difference scheme together with a modified secondary potential approach. They also used an analytic solution for the primary potential for models with non-flat topography. To account for the topography they modified the boundary conditions for the computation of the secondary potential. Ren and Tang (2014) used this technique for an adaptive finite-element solver, which automatically refined the mesh around the current electrodes based on error-estimates.

We implemented the secondary field approach in *escript* using unstructured meshes. Both, the primary and secondary

potentials are calculated numerically for flat and non-flat topography. In order to calculate the primary potential with adequate accuracy, the unstructured mesh was refined around the locations of the source electrodes.

The second example solves the 2D magnetotelluric (MT) forward problem. The electromagnetic source fields originate in high altitudes and are therefore commonly considered to be plane wave fields with sounding periods ranging from $\sim 10^{-3}$ seconds to $\sim 10^5$ seconds, where longer periods translate to an increasing depth of investigation. The MT forward problem corresponds to a diffusion equation and is usually solved in the frequency domain. The plane wave source is incorporated via boundary conditions. This differs markedly from the DC resistivity problem. 2D models are characterised by an electric strike direction along which the electrical resistivity is constant. This leads to a decomposition of the electromagnetic fields into two independent modes: electric field intensity \mathbf{E} parallel to strike, denoted as transverse electric (TE) mode or as \mathbf{E} -polarisation; and magnetic induction \mathbf{B} parallel to strike, which is denoted as transverse magnetic (TM) mode, or \mathbf{B} -polarisation (Chave and Jones 2012, pp.37). Because the magnetotelluric problem is solved in the frequency domain, it also involves complex arithmetic. In *escript*, complex quantities have to be decomposed into real and imaginary parts.

One of the earliest finite-element programs for 2D MT is presented in Kisak and Silvester (1975), which describes the procedures for the total field parallel to strike for \mathbf{E} - and \mathbf{B} -polarisation. Wannamaker et al (1987) describe a 2D MT modelling algorithm based on a secondary field approach, similar to the resistivity procedure described earlier. The primary or background field is typically formulated as the response of a 1D homogeneous or layered half-space. Various different numerical implementations of the magnetotelluric method, such as the integral equation and finite-difference methods, are reviewed by Zhdanov et al (1997). Franke-Börner (2003) implemented a 2D finite-element program in Matlab using the total field procedure for \mathbf{E} - and \mathbf{B} -polarisation. The paper by Key and Weiss (2006) describes an adaptive finite-element method using unstructured grids. In the adaptive finite-element method, a solution is calculated iteratively using successively refined grids until converging to the desired tolerance. Rung-Arunwan and Siripunvaraporn (2010) describe a parallel direct solver algorithm to solve the 2D MT problem based on a domain-decomposition approach. In Key and Oval (2011) a parallel goal-oriented adaptive finite-element method is presented for 2.5D controlled-source electromagnetics and 2D magnetotelluric.

Depending on whether the total field approach or the secondary field approach is implemented, different boundary conditions apply. The secondary field approach generally employs homogeneous Dirichlet conditions, i.e. the values for the fields at the boundaries are set to zero; this requires that the domain boundaries are far away from the anomalous region. The secondary field approach explicitly involves source terms in the PDE, which are usually based on 1D solutions and the resistivity difference between host and anomalous region (Chave and Jones 2012, p.305). Boundary conditions for the

total field formulation can also be based on 1D layered earth solutions and are implemented as non-homogeneous Dirichlet conditions; alternatively homogeneous Neumann conditions can be implemented if the sides of the domain are vertical (see Chave and Jones 2012, p.307). We implemented the total field approach in *escript* for TE and TM mode employing inhomogeneous Dirichlet boundary conditions, based on 1D layered earth solutions.

2. Escript modelling

A linear second order boundary-value-problem (BVP) for the unknown scalar function u is processed in *escript* in the form of a templated PDE with coefficients A, B, C, D, X and Y . The coefficients comprise the physical properties and are generally dependent on their locations and on time/frequency. In order to solve a PDE the user has to write a script in *Python* which defines the values for the PDE coefficients typically as constants or as location dependent functions within the PDE domain. It is also possible to define the PDE coefficients depending on the solution of another PDE. This way non-linear PDEs, time dependent PDEs and coupled PDEs can be solved. Using tensorial notation and the Einstein summation convention, the templated PDE for a single-valued solution u is of the following form (Gross et al 2007):

$$-(A_{jl}u_{,l})_{,j} - (B_j u)_{,j} + C u_{,l} + D u = -X_{l,j} + Y + \sum_s y_s^d \delta_{\mathbf{x}_s} \quad \text{in } \Omega \quad (1)$$

The notation $u_{,l}$ denotes the partial derivative with respect to the l th spatial coordinate; \mathbf{x}_s are locations of a point source (or sink) within the domain with production rates y_s^d ; $\delta_{\mathbf{x}_s}$ denotes the Dirac delta function with center \mathbf{x}_s . Summation is performed over the spatial indices j and l . For example the first term in above template conforms to the following classical notation, with $\mathbf{A} \equiv (A_{jl})$ a second-rank tensor:

$$(A_{jl}u_{,l})_{,j} \Leftrightarrow \sum_{j,l} \frac{\partial}{\partial x_j} \left(A_{jl} \frac{\partial}{\partial x_l} u \right) \Leftrightarrow \nabla \cdot (\mathbf{A} \cdot \nabla u)$$

Boundary (and initial) conditions are required in order to obtain a solution in some domain Ω . The boundary of Ω is denoted $\Gamma = \Gamma^D \cup \Gamma^N$, and $\Gamma^D \cap \Gamma^N = \emptyset$, where Γ^D specifies the parts of the boundary where essential (Dirichlet) boundary conditions apply and along which u has a prescribed value; Γ^N denotes the boundary where natural (Neumann) boundary conditions apply and along which values of the derivative of u are specified. The following natural boundary conditions are considered on the domain boundary Γ^N :

$$n_j(A_{jl}u_{,l} + B_j u) + du = n_j X_j + y \quad \text{on } \Gamma^N \quad (2)$$

where n_j denotes a component of the outer unit normal vector on Γ^N and the coefficients A, B and X are the same as defined in the PDE, equation (1). The scalar coefficients d and y need to be defined on the entire boundary Γ of the domain but are used on Γ^N only. On Γ^D essential (Dirichlet) boundary conditions prescribe the value of the solution of the form

$$u = r \quad \text{on } \Gamma^D \quad (3)$$

where r is a function defining the values at these locations. Conventionally, the essential boundary conditions on any location of the domain boundary overwrite any natural boundary conditions that may have been set on this location via equation (2). Note that both types of boundary conditions can be combined and can be homogeneous or non-homogeneous. It is pointed out that *escript* also supports a template for systems of PDEs. This formulation is not presented here, but is used in the example for solving the magnetotelluric PDE; the *escript* users guide provides more details (Gross et al 2015).

In order to solve a PDE in *escript* for the unknown scalar dependent variable u , the governing equations have to be formulated in weak form. As opposed to the classical strong formulation of the BVP, which states the conditions at every point over a domain that a solution must satisfy, the weak form states the conditions that the solution must satisfy in a weighted average sense. The weak form of the BVP is defined as an inner product of u with a set of suitable weighting function, w , which vanish on Γ_D , i.e. $w = 0$ on Γ_D . Carrying out the integration of the *escript* template over the domain Ω , integrating by parts and applying Gauss' theorem, gives,

$$\begin{aligned} & \int_{\Omega} d\Omega \ w_j (A_{jl} u_{,l} + B_j u) + \int_{\Omega} d\Omega \ w (C_l u_{,l} + Du) \\ & - \int_{\Gamma} d\Gamma \ w n_j (A_{jl} u_{,l} + B_j u) = \int_{\Omega} d\Omega \ (w_j X_j + w Y) \\ & + \sum_s y_s^{\text{dirac}} w(\mathbf{x}_s) - \int_{\Gamma} d\Gamma \ w n_j X_j \end{aligned} \quad (4)$$

Applying the natural boundary conditions from equation (2) and using the fact that $w = 0$ on Γ_D gives the weak form of the BVP, subject to essential constraints, equation (3):

$$\begin{aligned} & \int_{\Omega} d\Omega \ w_j (A_{jl} u_{,l} + B_j u) + \int_{\Omega} d\Omega \ w (C_l u_{,l} + Du) + \int_{\Gamma} d\Gamma \ w du \\ & = \int_{\Omega} d\Omega \ (w_j X_j + w Y) + \sum_s y_s^{\text{dirac}} w(\mathbf{x}_s) + \int_{\Gamma} d\Gamma \ w y \end{aligned} \quad (5)$$

The weak form PDE, equation (5), is numerically solved using the Ritz–Galerkin method, which converts the continuous weak form PDE into a discrete formulation. The solution u is approximated by a function \tilde{u}^h which is a linear combination in some basis functions ϕ_n^h ($n = 1, \dots, N$). The construction of the basis functions is based on a subdivision of the domain into smaller pieces of maximum size h , so-called elements, sharing vertices \mathbf{x}_n^h ($n = 1, \dots, N$), called nodes. The structure describing the location and the connectivity of the nodes to form the elements is called a mesh. For the sake of simplicity it is assumed that all locations \mathbf{x}_s of point sources are also mesh nodes. The basis function ϕ_n^h corresponding to node \mathbf{x}_n^h is constructed as a linear polynomial for each element, which is continuous across element boundaries and has value zero at all other nodes, i.e. $\phi_m^h(\mathbf{x}_n^h) = \delta_{mn}$ for $m, n = 1, \dots, N$, with Kronecker delta δ_{mn} , see for instance (Zienkiewicz et al 2013) for details. With this construction the solution approximation is set to

$$\tilde{u}^h = \sum_{m=1}^N u_m^h \phi_m^h \quad (6)$$

where by construction $\tilde{u}_m^h = u^h(\mathbf{x}_m^h)$ is the value of \tilde{u}^h at node \mathbf{x}_m^h . Substituting equation (6) into the weak form, equation (5), and choosing the basis functions ϕ_n^h for the function w leads to a system of linear equations

$$\mathbf{K}^h \mathbf{u}^h = \mathbf{F}^h \quad (7)$$

where $\mathbf{u}^h = (u_m^h)_{m=1 \dots N}$ is the vector of basis function coefficients, $\mathbf{K}^h = (K_{nm}^h)_{m,n=1 \dots N}$ is the stiffness matrix and $\mathbf{F}^h = (F_n^h)_{n=1 \dots N}$ is the right-hand-side vector. For a node \mathbf{x}_n^h which is not located within Γ_D one obtains for the m th element Ω_m :

$$\begin{aligned} K_{nm}^h = & \int_{\Omega_m} d\Omega \ A_{jl} \phi_{n,j}^h \phi_{m,l}^h + \int_{\Omega_m} d\Omega \ B_j \phi_{n,j}^h \phi_m^h \\ & + \int_{\Omega_m} d\Omega \ C_l \phi_n^h \phi_{m,l}^h + \int_{\Omega_m} d\Omega \ D \phi_n^h \phi_m^h + \int_{\Gamma_m} d\Gamma \ d\phi_n^h \phi_m^h \end{aligned} \quad (8)$$

for the entries in the stiffness matrix and

$$F_n^h = \int_{\Omega_m} d\Omega \ X_j \phi_{n,j}^h + \int_{\Omega_m} d\Omega \ Y \phi_n^h + y_n^{\text{dirac}} + \int_{\Gamma_m} d\Gamma \ y \phi_n^h \quad (9)$$

for the right-hand-side vector entries. To consider the essential boundary conditions, equation (3), one sets

$$K_{nm}^h = \delta_{nm} \quad \text{and} \quad F_n^h = r(\mathbf{x}_n^h) \quad (10)$$

for those rows n where \mathbf{x}_n^h are located on Γ_D . By construction of the basis function the stiffness matrix is sparse, i.e. the number of non-zero entries in the matrix is very small compared to the total size of the stiffness matrix. For these type of matrices iterative methods are most suitable (Weiss 1996) but *escript* also provides the option to use direct solvers to solve the linear system, equation (7), e.g. with UMFPAK (Davis 2004). In the discretisation approach discussed here it is assumed that values for B and C are sufficiently small so no additional stabilisation of the finite element matrix is required. For these cases *escript* provides a special solver using algebraic flux control, see Gross et al (2013).

For execution on a parallel compute cluster the rows of the stiffness matrix \mathbf{K}^h are distributed across the processors. The distribution of the rows correspond to a distribution of mesh nodes and the attached elements can be seen as a subdivision of the domain into smaller subdomains that are assigned to processors for parallel calculation. Efficiency of this domain decomposition approach is controlled by the costs for exchanging nodal FEM data across sub-domain boundaries and an equally distributed computational load across processors (Saad 1996). For structured meshes this can easily be implemented but for unstructured meshes a compromise of these two objectives can be achieved only. In *escript* the ParMetis library is used for mesh decomposition (Schloegel et al 2002). It is pointed out that scripts for *escript* do not refer to the actual mesh or its distribution across a set of processors. As a consequence scripts can be run with different mesh types, e.g. structured or unstructured, and across different processor configurations without modifications to the script.

3. Examples

3.1. DC resistivity in 3D

The electric potential over a given three-dimensional domain Ω is obtained as the solution of a Poisson-type PDE, which is derived in weak form and numerically solved using *escript*. Following (Pridmore *et al* 1981), the partial differential equation derives from the steady-state equation of continuity of current density, $\nabla \cdot \mathbf{J}^c = 0$, where \mathbf{J}^c is the conduction current density in the subsurface. Ohm's law can be substituted for the current density, $\mathbf{J}^c = \sigma \mathbf{E}$, with σ the electrical conductivity and \mathbf{E} the electric field intensity. Considering artificial sources of current density, \mathbf{J}^s , and writing the electric field as the gradient of a scalar potential, u , leads to the Poisson equation for the electric potential:

$$\nabla \cdot (\sigma \nabla u) = \nabla \cdot \mathbf{J}^s \quad (11)$$

For a single point source at location \mathbf{x}_0 and a measurement location \mathbf{x} the non-stationary source current density, \mathbf{J}^s , follows the continuity equation:

$$\nabla \cdot \mathbf{J}^s = -\frac{\partial q_c}{\partial t} \delta(\mathbf{x} - \mathbf{x}_0) = -I \delta(\mathbf{x} - \mathbf{x}_0) \quad (12)$$

where q_c is the electrical charge density and I the source current strength. For N point sources at locations \mathbf{x}_i with current strengths I_i , the potential field, $u(\mathbf{x})$, is the solution of:

$$-\nabla \cdot (\sigma \nabla u) = \sum_{s=1}^{N_s} I_i \delta_{\mathbf{x}_s} \equiv Q \quad (13)$$

Note that for the case of a pair of charging electrodes at locations \mathbf{x}_i^+ and \mathbf{x}_i^- with current strengths $I_i^+ = -I_i^-$ of opposite polarity, one sets for the source term Q

$$Q = \sum_{s=1}^{N_s} (I_i^+ \delta_{\mathbf{x}_s^+} + I_i^- \delta_{\mathbf{x}_s^-}) \quad (14)$$

The Dirac delta functions in the source term lead to singularities and can be tackled by refining the mesh near the current source points and by splitting up the potential into a primary potential, u^p , and a secondary potential, u^s , for a background or primary conductivity σ^p and an anomalous or secondary conductivity $\sigma^s = \sigma - \sigma^p$, respectively (Lowry *et al* 1989):

$$u = u^p + u^s \quad (15)$$

The primary potential can be calculated analytically for flat-earth models as the response for a uniform half-space with conductivity σ^p . For topographic models u^p is calculated numerically, applying a local mesh refinement at the source locations (see Rücker *et al* 2006). Here, we calculate the primary potential for flat and non-flat topographies numerically. The conductivity σ^p is chosen as the conductivity near the source points (Zhao and Yedlin 1996). The primary potential u^p satisfies the PDE equation (13):

$$-\nabla \cdot (\sigma^p \nabla u^p) = Q \quad (16)$$

The PDE for the secondary potential follows from substituting equation (15) into equation (13) and use equation (16), yielding:

$$\nabla \cdot (\sigma \nabla u^s) = \nabla \cdot ((\sigma^p - \sigma) \nabla u^p) \quad (17)$$

To complete the problem, boundary conditions have to be applied for the partial differential equations of the primary and secondary potential. The boundary conditions are introduced below in combination with the weak form. The weak form is derived from the weighted residual method and is simply obtained by multiplying the PDEs, equations (16) and (17), with suitable weight functions, w , and integrating over the domain Ω ; subsequent use of integration by parts and Gauss' theorem provides the weak form PDE, see also figure 2. The weak form PDE for the primary potential evaluates to:

$$-\int_{\Gamma} d\Gamma \sigma^p \omega \mathbf{n} \cdot \nabla u^p + \int_{\Omega} d\Omega \sigma^p \nabla \omega \cdot \nabla u^p = \int_{\Omega} d\Omega \omega Q \quad (18)$$

and for the secondary potential one gets:

$$\begin{aligned} & -\int_{\Gamma} d\Gamma \sigma \omega \mathbf{n} \cdot \nabla u^s + \int_{\Omega} d\Omega \sigma \nabla \omega \cdot \nabla u^s \\ & = \int_{\Gamma} d\Gamma (\sigma^p - \sigma) \omega \mathbf{n} \cdot \nabla u^p + \int_{\Omega} d\Omega (\sigma^p - \sigma) \nabla \omega \cdot \nabla u^p \end{aligned} \quad (19)$$

Homogeneous natural and essential boundary conditions for the primary and secondary potentials can now be applied. At the air-earth boundary, Γ^N , the electric flux $\mathbf{n} \cdot \nabla u$ is zero, since air is virtually non-conductive, i.e. Lowry *et al* (1989):

$$\left. \begin{aligned} \sigma \mathbf{n} \cdot \nabla u^s &= 0 \\ \sigma^p \mathbf{n} \cdot \nabla u^p &= 0 \\ (\sigma^p - \sigma) \mathbf{n} \cdot \nabla u^p &= 0 \end{aligned} \right\} \text{ on } \Gamma^N \quad (20)$$

For all other boundaries, Γ^D , it is assumed that the domain is adequately large for the potentials to have dissipated to zero, i.e.:

$$u = 0 \text{ on } \Gamma^D \quad (21)$$

Therefore the integrals to be solved reduce to the integrals over the domain Ω ; for the primary and the secondary potential, in *escript*'s formulation, one gets:

$$\text{(primary)} \quad \int_{\Omega} d\Omega \sigma^p \omega_j u_j^p = \int_{\Omega} d\Omega \omega Q \quad (22)$$

$$\text{(secondary)} \quad \int_{\Omega} d\Omega \sigma \omega_j u_j^s = \int_{\Omega} d\Omega (\sigma^p - \sigma) \omega_j u_j^p \quad (23)$$

The integrals are solved within the *escript* framework for all admissible (i.e. adequately smooth) weight functions ω . The next step is therefore to match the PDE coefficients, see equation (5), of the *escript* PDE template to the corresponding quantities as defined in the weak form formulations for the primary and secondary potentials. The boundary integrals in the template formulation evaluate to zero due to the homogeneous boundary conditions on Γ^N , i.e. the *escript* PDE coefficients d and y are identical zero. Via comparison, the non-zero PDE coefficients for the primary potential are:

$$A_{jl}^p = \sigma^p \delta_{jl}, \quad y_s^d \delta_{\mathbf{x}_s} = Q \quad (\text{for } s \text{ electrodes}) \quad (24)$$

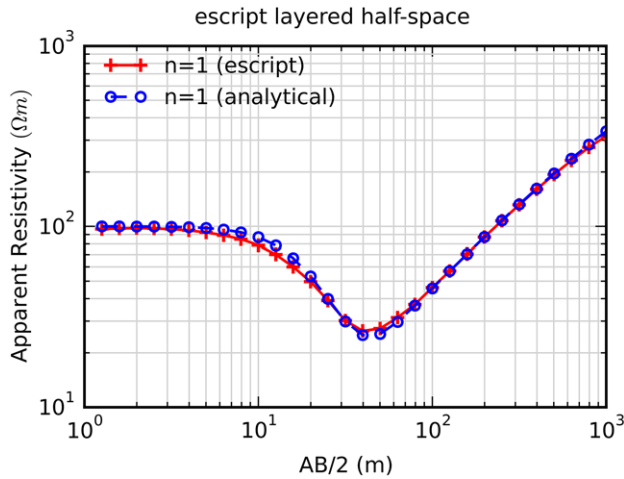


Figure 1. Schlumberger sounding over a three layered half-space with resistivities 100 Ωm, 10 Ωm and 1000 Ωm, respectively. Layer thickness is 10 m for top and middle layer. Soundings were computed at a total of 61 positions from −1000 m to 1000 m; plot points are at the centre of the electrode array at AB/2. The analytical response is based on digital filters (Johansen and Sørensen 1979). The 3D *escript* domain on the left and right extents to ±5000 m and ±2000 m to the south and north; vertical extent is 4000 m. The *escript* curve closely matches the analytical curve. The change in resistivity at the interface of the middle layer causes some deviations noticeable at the 10 m mark.

where $y_s^d \delta_{\mathbf{x}_s}$ are the Dirac point source terms defined in equation (1) and where δ_{jl} is Kronecker's delta function, so that the general tensor expression $A_{jl} = \sigma$ for all $j = l$. This can be seen by writing out the terms in equations (22), viz.,

$$\begin{aligned} \sigma^p \omega_j u_j^p &= \sigma^p \partial_x \omega \partial_x u^p + \sigma^p \partial_y \omega \partial_y u^p + \sigma^p \partial_z \omega \partial_z u^p \\ &= w_j \sigma^p \delta_{jl} u_j^p \end{aligned} \quad (25)$$

For the secondary potential one obtains:

$$A_{jl}^s = \sigma \delta_{jl}, \quad X_j = (\sigma^p - \sigma) u_j^p \quad (26)$$

the source term for the secondary potential, X_j , involves the solution for the primary potential, thus a coupled PDE system is solved together with essential boundary conditions enforced as per equation (21). The *escript* formulation of the DC resistivity problem is thus:

$$\int_{\Omega} d\Omega \quad w_j A_{jl}^p u_{l,l}^p = \int_{\Omega} d\Omega \quad w y_s^d \delta_{\mathbf{x}_s} \quad (27)$$

$$\int_{\Omega} d\Omega \quad w_j A_{jl}^s u_{l,l}^s = \int_{\Omega} d\Omega \quad (\sigma^p - \sigma) \omega_j u_j^p \quad (28)$$

The secondary potential approach was implemented in *escript* for a layered half-space, a half-sphere model and a prism model. A scripted example for the prism model can be found in appendix.

The layered half-space response is shown in figure 1 and depicts a Schlumberger sounding over a highly resistive basement overlain by a conductive middle layer and a resistive top. Model parameters are described in the figure's caption. Note that both, the primary and secondary potentials

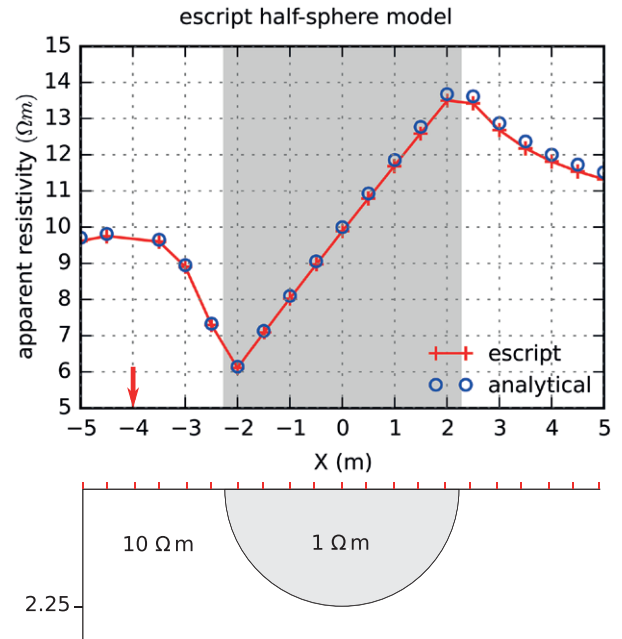


Figure 2. Half-sphere model adjacent to the half-space boundary with 21 electrodes traversing the centre of the sphere with a spacing of 0.5 m from −5 m to 5 m. The grey-shaded area indicates the extent of the half-sphere. The potentials are calculated for a pole–pole spread where the source electrode is located at coordinate $x_s = (-4; 0; 0)$, indicated by the red arrow in the profile plot. The *escript* model extents are 2 km × 2 km × 1 km for x , y and z . Primary and secondary potentials are numerically evaluated for a resistivity of 10 Ωm and 1 Ωm respectively. Analytical and numerical solution show an overall good fit.

were computed numerically with *escript*, where the primary potential was calculated with the resistivity of the top layer. Deviations between the analytical and numerical response are noticeable near the 10 m mark, coinciding with the transition of resistivities from top to middle layer.

The second example is the half-sphere model from Rücker et al (2006); model parameters and response for a pole–pole spread traversing the half-sphere are displayed in figure 2. See the caption for a description of the model parameters. The analytical response is calculated via the formulae for the electric potential of a sphere in a full space by Zhdanov and Keller (1994, p.216); the potential for a half-sphere buried in a half-space is then obtained by multiplying the full-space potential by two. The equations for the full space potential are repeated here:

$$U(r, \theta) = \frac{I \rho_p}{4\pi R} + \frac{I \rho_p}{4\pi} \begin{cases} \sum_{n=1}^{\infty} K_n \frac{a^{2n+1}}{(rd)^{n+1}} P_n(\cos \theta), & r > a \\ \sum_{n=1}^{\infty} K_n \frac{1}{d} \left(\frac{r}{d}\right)^n P_n(\cos \theta), & r \leq a \end{cases}$$

where I is the current strength, ρ_p is the resistivity of the host, P_n are Legendre polynomials with argument $\cos \theta$; a is the sphere radius and R is the distance from the source to the observation point, d the distance from the sphere centre to the source point and r the distance from the sphere centre to the observation point; θ is the angle between d and r . The

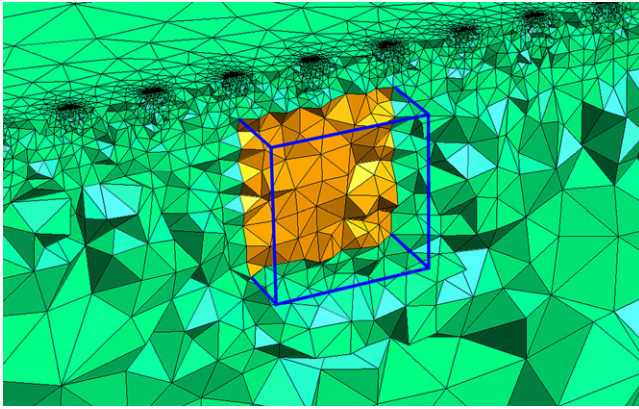


Figure 3. Finite element mesh for a buried cube with $\rho_s = 20 \Omega\text{m}$ in a half-space of $\rho_p = 100 \Omega\text{m}$. Cube size is $(2 \text{ m})^3$, with depth-to-top of 0.5 m. *Escript* model size is $2 \text{ km} \times 2 \text{ km} \times 1 \text{ km}$. The mesh was refined at the locations of the electrodes to evaluate the primary potential more accurately. The electrodes traverse a centre-line of the cube from -10 m to 10 m with a 1 m separation.

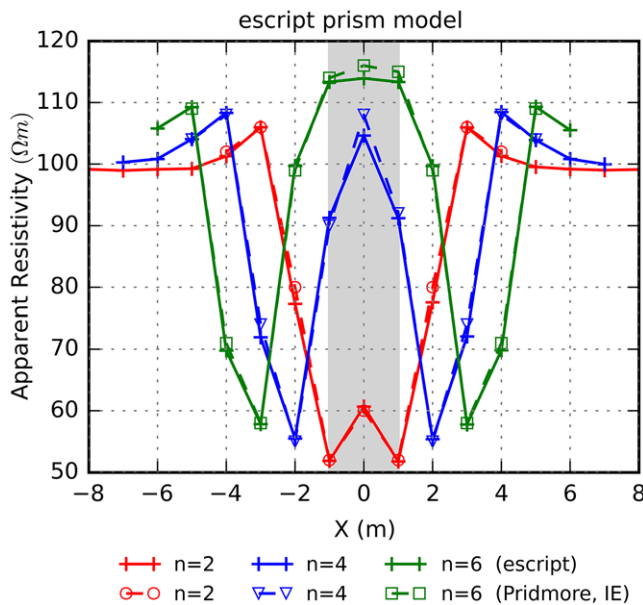


Figure 4. Apparent resistivity curves for a dipole-dipole spread for a traverse over the centre of the cubic body. The grey-shaded area indicates the extent of the cubic body. Values shown are for 3 different electrode-pair separations n . The apparent resistivity values of both schemes are very similar.

reflection coefficient K_n contains the sphere resistivity, ρ_s , and is defined as

$$K_n = \frac{n(\rho_s - \rho_1)}{(n+1)\rho_s + n\rho_1}$$

The electrical potential, based on above formulae, was calculated for 25 terms and multiplied with 2 to get the half-sphere solution, see Zhdanov and Keller (1994, p.219). The half-sphere solution matches the analytical response better on the left half of the profile which contains the source electrode; small deviations are observable on the right half.

Because the primary field was numerically evaluated, the 3D resistivity computations required a large number of nodes

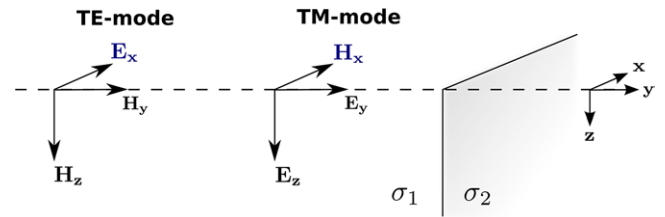


Figure 5. Simple 2D conductivity model with a vertical contact extending in the x -direction. The E_y -component is discontinuous across the vertical contact due to conservation of current. The electromagnetic field in the 2D case can be decoupled into two independent modes: (1) TE-mode, with electric fields parallel to strike and (2) TM-mode, magnetic fields parallel to strike (after Simpson and Bahr 2005, p.28).

to account for the singularities at the electrode locations. In order to calculate the response relatively accurate for the half-sphere model a total of 50 642 were employed, which is similar to the number of nodes (49 341) employed by Rücker et al (2006) for the total field computation of the half-sphere model. Incorporating analytical solvers for the primary field can markedly reduce the required number of nodes in case of flat-earth topographies.

The third example compares numerically evaluated apparent resistivities of a buried cube in a half-space. The *escript* values are compared to the integral equation (IE) values by Pridmore et al (1981, figure 5). Figure 3 shows the finite element mesh used to evaluate the primary and secondary electrical potentials with *escript*. The mesh has a total of 50 435 nodes with the same dimensions as for the half-sphere model ($2 \text{ km} \times 2 \text{ km} \times 1 \text{ km}$); depth-to-top of the $(2 \text{ m})^3$ cube is 0.5 m. The apparent resistivities are shown in figure 4 and were calculated at 21 electrodes for a dipole-dipole spread from -10 m to 10 m along a centre line over the cube with a step length of 1 m . Shown are the apparent resistivity curves for three different electrode-pair separations n . Overall, the apparent resistivity values of both schemes are close.

3.2. 2D magnetotelluric

Whereas the DC resistivity technique is driven by galvanic ground-coupling, the magnetotelluric method is driven by inductive coupling of naturally occurring electromagnetic plane-wave fields with the subsurface resistivity structures. The behaviour of magnetotelluric fields is governed by the low-frequency approximation of Maxwell's equations in a source free medium, yielding the following diffusion equations (e.g. Chave and Jones 2012, p.303):

$$\nabla \times \frac{1}{\sigma} \nabla \times \mathbf{H} + i\omega\mu\mathbf{H} = 0 \quad (29)$$

$$\nabla \times \frac{1}{\mu} \nabla \times \mathbf{E} + i\omega\sigma\mathbf{E} = 0 \quad (30)$$

where \mathbf{H} is the magnetic field, \mathbf{E} the electric field, σ is the electrical conductivity and $\mu = \mu_0$ the magnetic permeability assumed to have the value of free-space, $\mu_0 = 4\pi \cdot 10^{-7}$; ω is the angular frequency and i the imaginary unit.

For the 2D model in figure 5 there is no variation along-strike in the x -direction, implying that all derivatives with respect to x are zero. The plane-wave fields are mutually orthogonal and decouple into two independent modes with respect to the strike direction: the TE-mode (transverse electric) and the TM-mode (transverse magnetic), which are both characterised by a single field-component parallel to the strike direction, E_x and H_x , respectively. These modes are also known as **E**-polarisation and **B**-polarisation.

Derivation of the partial differential equations for TE- and TM-mode follows from the quasi-static diffusion equations (29) and (30), where $\nabla = (0, \partial_y, \partial_z)^T$ and the relevant parallel electric or magnetic field is denoted as $\mathbf{u} = (u_x, 0, 0)$. The curl-curl expressions in the diffusion equation for both 2D modes evaluate then to a single term:

$$(\nabla \times f \nabla \times \mathbf{u})_x = -\partial_y f \partial_y u_x - \partial_z f \partial_z u_x \quad (31)$$

where $f = f(y, z)$ is the property function which is either $1/\mu$ or $1/\sigma$ for the TE or TM mode respectively. The PDE for 2D magnetotellurics therefore simplifies to a scalar partial differential equation of the form

$$-\nabla \cdot (f \nabla u) + g u = 0 \quad (32)$$

where the coefficients and the field are defined below:

$$\text{TE:} \quad f = \frac{1}{\mu} \quad g = i\omega\sigma \quad u = E_x \quad (33)$$

$$\text{TM:} \quad f = \frac{1}{\sigma} \quad g = i\omega\mu \quad u = H_x \quad (34)$$

Equation (32) defines the modelled field as the solution of the PDE. The orthogonal auxiliary fields are subsequently calculated from the PDE solution via Ampere's law (derivatives of E_x) and Faraday's law (derivatives of H_x):

$$\text{TE:} \quad H_y = -\frac{1}{i\omega\mu} \partial_z E_x \quad H_z = \frac{1}{i\omega\mu} \partial_y E_x \quad (35)$$

$$\text{TM:} \quad E_y = \frac{1}{\sigma} \partial_z H_x \quad E_z = -\frac{1}{\sigma} \partial_y H_x \quad (36)$$

Apparent resistivity and phase are defined for both modes via the magnetotelluric impedances, i.e. the ratio of the transverse electric and magnetic fields,

$$Z_{\text{TE}} = -\frac{E_x}{H_y} \text{ and } Z_{\text{TM}} = \frac{E_y}{H_x} \quad (37)$$

and apparent resistivity and phase are given as:

$$\rho_m = \frac{1}{\omega\mu} |Z_m|^2 \text{ and } \phi_m = \tan^{-1} \left(\frac{\Im Z_m}{\Re Z_m} \right) \quad (38)$$

where the subscript m denotes either the TE- or TM-mode.

In 2D MT the modelled field quantities, E_x and H_x are both continuous across the entire domain; also all other components of the TE mode are continuous under the assumption that $\mu = \mu_0$. Only the TM-mode electrical field component E_y , equation (36), is discontinuous across the vertical contact. Telluric currents flow across structures in the TM mode and

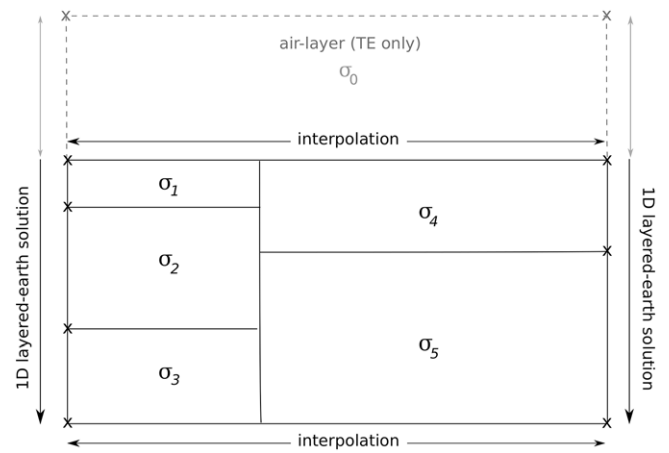


Figure 6. Schematic of the boundary conditions for 2D MT modelling.

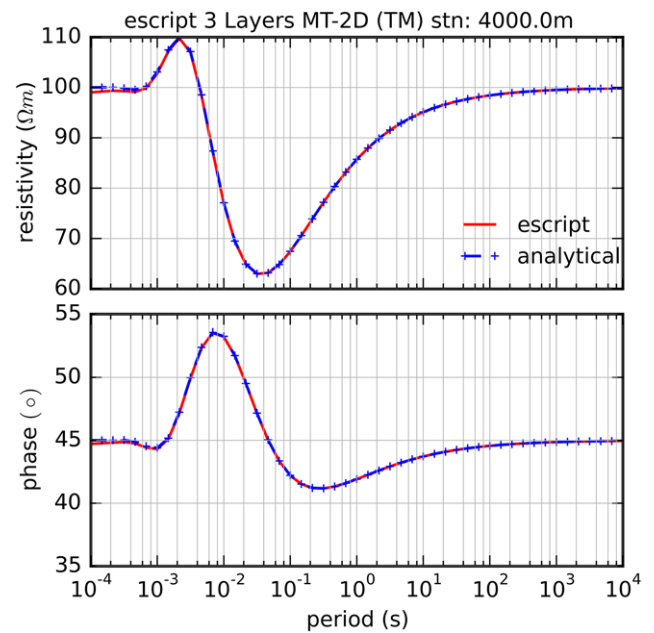


Figure 7. Layered half-space response for a three-layered model. The *escript* responses are calculated at the centre of a $8 \text{ km} \times 8 \text{ km}$ model for TM-mode excitation. Top and basement layer have a resistivity of $100 \text{ } \Omega\text{m}$, enclosing a more conductive middle layer of $20 \text{ } \Omega\text{m}$. Layer thicknesses of the top and middle layer are 300 m and 100 m , respectively. MT responses were calculated for periods from 10^{-4} s to 10^4 s with 6 points per decade. The apparent resistivity curve shows an overshoot near period $T = 10^{-3} \text{ s}$, which is a characteristic of the apparent resistivity response in stratified media (e.g. Spies and Eggers 1986). The magnetotelluric phase for early periods is 45° and indicative of a uniform half-space—at these periods the skin depth is smaller than the thickness of the top layer. With increasing periods the phase also increases which is diagnostic of substrata of increasing resistivity (Simpson and Bahr 2005, p.26).

along structures in the TE mode. The TM mode is therefore characterised by anomalies of galvanic nature due to charge build-up, whereas TE mode anomalies are of inductive nature (Berdichevsky *et al* 1998). In consequence TM mode data is more sensitive to near-surface resistive structures and TE mode data more sensitive to deeper conductive structures. The modelling domain, Ω , is set up differently for TE- and

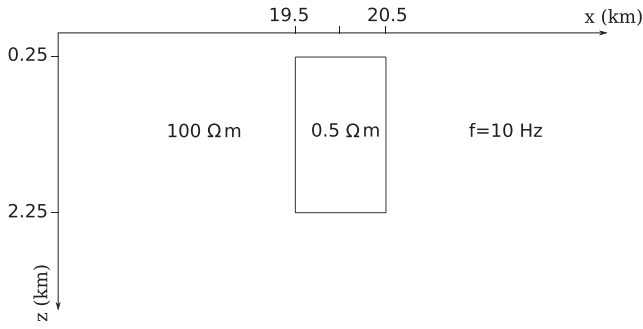


Figure 8. COMMEMI-1 test model: single prism in a homogeneous host. The *escript* domain is 20 km × 20 km, and a 20 km air-layer is added for the TE-mode calculations. The responses are calculated for a 10 Hz frequency.

TM-mode. Due to the virtually infinite impedance at the air-earth interface, there are no induced electric field components in the air-layer, as such the air-layer can be ignored for TM modelling. In case of the magnetic field components of the TE mode, an air-layer must be included to account for magnetic field perturbations from longitudinal current gradients.

For all boundaries Γ of the modelling domain, inhomogeneous boundary conditions of the Dirichlet type apply, i.e. essential boundary conditions are enforced (figure 6). On the left and right sides of the domain, the assumption is made that the conductivity distribution is a function of depth only and the electromagnetic fields are calculated analytically via 1D solutions on each side (see Wait 1953). The conditions on the top and bottom, respectively, are calculated via interpolation of the values at the left and right sides. For the TM-mode, the boundary condition is satisfied with a homogeneous Dirichlet condition at the air-earth interface with $H_x = 1 \text{ A m}^{-1}$. For the TE-mode, the electric field at the top of the air-layer is calculated via the free-space value of the electric field.

The equations all contain complex terms. For *escript* these have to be decomposed into real and imaginary parts, $u = u^r + i u^i$, where u is any of the complex field components and i is the imaginary unit. Substituting $u = u^r + i u^i$ into the PDE, equation (32), yields,

$$\nabla \cdot (f \nabla u^r + i f \nabla u^i) + g u^r + i g u^i = 0 \quad (39)$$

and substituting the terms for f and g provides the solution for TE- and TM-mode, which are written as two PDEs for real and imaginary part. For the TE-mode one obtains:

$$\left. \begin{aligned} \nabla \cdot (\mu^{-1} \nabla E_x^r) + \omega \sigma E_x^i &= 0 \\ \nabla \cdot (\mu^{-1} \nabla E_x^i) - \omega \sigma E_x^r &= 0 \end{aligned} \right\} \quad (40)$$

and for the TM-mode:

$$\left. \begin{aligned} \nabla \cdot (\sigma^{-1} \nabla H_x^r) + \omega \mu H_x^i &= 0 \\ \nabla \cdot (\sigma^{-1} \nabla H_x^i) - \omega \mu H_x^r &= 0 \end{aligned} \right\} \quad (41)$$

Thus in *escript* a coupled PDE must be solved for both modes which simultaneously solves for the real and imaginary part of the electric or magnetic field. To decompose apparent resistivity and phase into real and imaginary parts, the ratio of electric to magnetic fields are specified as (see equation (37)):

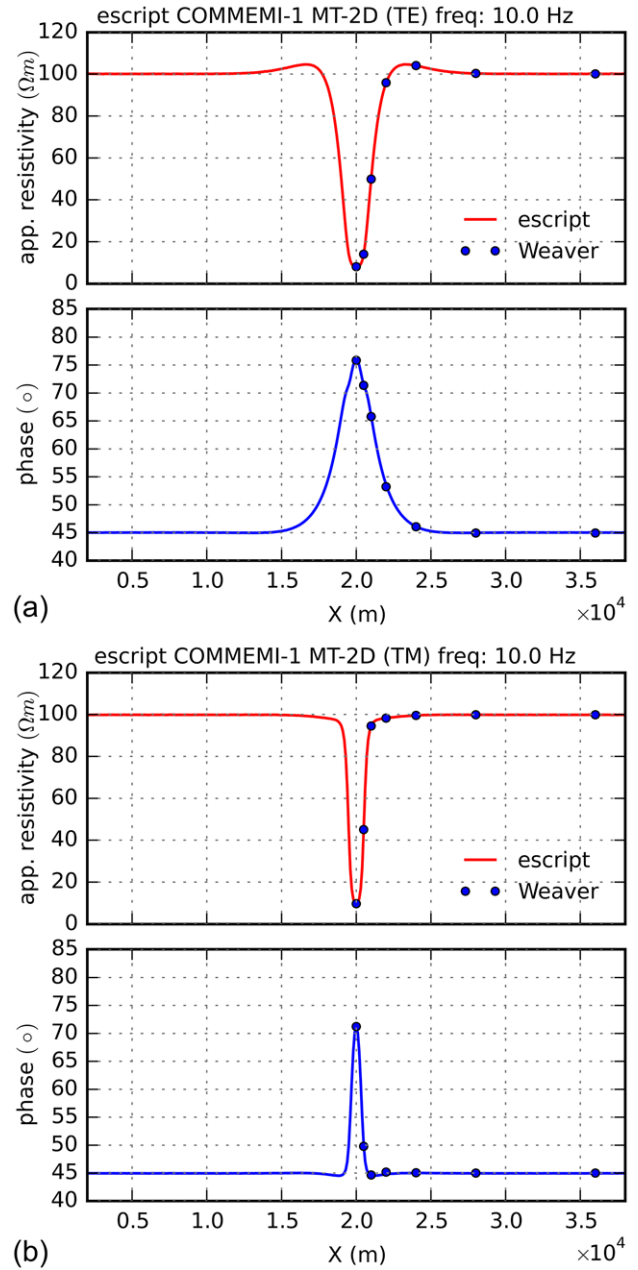


Figure 9. COMMEMI-1 model apparent resistivities and phase for TE-mode (a) and TM-mode (b). The TE-mode displays a broader anomaly, indicative of the inductive nature of the response, whereas the TM-mode curve exhibits a galvanic-type response. Comparison with apparent resistivity and phase values by Weaver are in good agreement (see table 1). (a) TE mode. (b) TM mode.

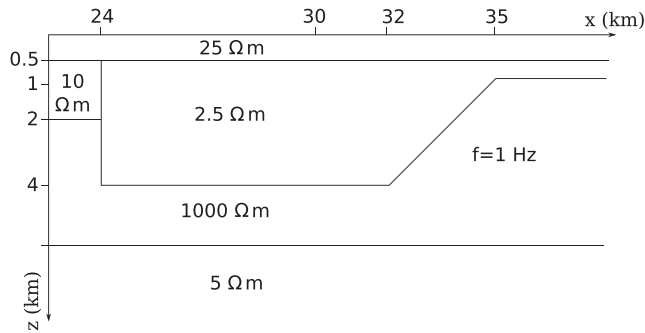
$$\frac{E}{H} = \frac{E^r + i E^i}{H^r + i H^i} = \frac{E^r H^r + E^i H^i}{(H^r)^2 + (H^i)^2} + i \frac{E^i H^r - E^r H^i}{(H^r)^2 + (H^i)^2} \quad (42)$$

where the subscripts indicating TE- or TM-mode have been omitted. Using the relationship $|z|^2 = x^2 + y^2$ for a complex number z , the apparent resistivity can now be defined via the decomposed real and imaginary terms,

$$\rho = \frac{1}{\mu \omega} \left| \frac{E}{H} \right|^2 = \frac{1}{\mu \omega} \left\{ \frac{(E^r)^2 + (E^i)^2}{(H^r)^2 + (H^i)^2} \right\} \quad (43)$$

Table 1. COMMEMI-1 *escript* apparent resistivity and phase at selected coordinates for TE- and TM-mode compared with results by Weaver.

	x (km)	20.0	20.5	21.0	22.0	24.0	28.0	36.0
ρ_a	<i>escript</i> (TE)	8.08	14.06	49.91	95.88	104.10	100.31	100.12
	Weaver (TE)	8.07	14.10	52.50	95.70	104.00	100.00	100.00
ρ_a	<i>escript</i> (TM)	9.66	45.06	94.57	98.31	99.58	99.89	99.87
	Weaver (TM)	9.86	46.40	94.80	98.30	99.70	100.00	100.00
ϕ	<i>escript</i> (TE)	76.00	71.49	65.90	53.60	46.11	44.99	45.03
	Weaver (TE)	75.83	71.38	65.77	53.24	46.07	44.94	44.94
ϕ	<i>escript</i> (TM)	71.33	49.96	44.58	44.79	45.01	44.96	44.96
	Weaver (TM)	71.24	49.80	44.65	45.17	45.06	45.00	45.0

**Figure 10.** COMMEMI-4 test model: embedded graben-like structure in layered host. The *escript* horizontal domain-extent is 60 km and 50 km in the vertical direction; a 50 km air-layer is added for the TE-mode calculations. Frequency is 1 Hz.

The phase evaluates to,

$$\phi = \tan^{-1} \left\{ \frac{E^i H^r - E^r H^i}{E^r H^r + E^i H^i} \right\} \quad (44)$$

Note that the appropriate components have to be substituted for TE and TM mode respectively.

For implementation in *escript*, the weak form is derived and matched against the *escript* template. The weak form of equations (40) and (41) are combined into a vector-valued PDE system and evaluates to (see equation (18)):

$$\int_{\Gamma} d\Gamma f \varphi \mathbf{n} \cdot \nabla u + \int_{\Omega} d\Omega f \nabla \varphi \cdot \nabla u + \int_{\Omega} d\Omega \varphi g u = 0 \quad (45)$$

where φ denotes the weighting function (to avoid confusion with the angular frequency ω), $f = \mu^{-1}$ and $g = \omega \sigma$ for TE-mode, and $f = \sigma^{-1}$ and $g = \omega \mu$ for TM-mode. The solution of the PDE, u , is a vector-valued quantity and yields $(E_x^r, E_x^i)^T$ and $(H_x^r, H_x^i)^T$, according to the TE- or TM-mode equations. In this context the functions f and g are tensors and the weighting function φ is vector-valued as well. At all boundaries the values of the fields are determined via analytical 1D solutions and the boundary integral term over Γ , which denotes the natural boundary condition, is zero.

Because equation (45) is a PDE system of two equations, it is cast to a more general weak form for PDE-systems in the templated *escript* formulation where the coefficient A is now defined as a rank-4 tensor and D a rank-2 tensor (see equation (5)),

$$\int_{\Omega} d\Omega \varphi_{i,j} A_{ijkl} u_{k,l} - \int_{\Omega} d\Omega \varphi_i D_{ik} u_k = 0 \quad (46)$$

only four tensor elements of PDE coefficients A_{ijkl} are non-zero and two elements of D_{ik} . These terms can be identified by writing out the equations (40) and (41); for example writing out equation (41) yields:

$$\partial_y(\sigma^{-1} \partial_y H_x^r) + \partial_z(\sigma^{-1} \partial_z H_x^r) + \omega \mu H_x^i = 0 \quad (47)$$

$$\partial_y(\sigma^{-1} \partial_y H_x^i) + \partial_z(\sigma^{-1} \partial_z H_x^i) - \omega \mu H_x^r = 0 \quad (48)$$

By inspection, the non-zero values of A_{ijkl} and the two non-zero elements of D_{ik} evaluate to:

$$A_{ijkl} = \sigma^{-1} \quad \text{for } A_{0000}, A_{0101}, A_{1010}, A_{1111} \quad (49)$$

$$D_{01} = \omega \mu \quad \text{and} \quad D_{10} = -\omega \mu \quad (50)$$

The 2D MT problem was implemented in *escript* based on above description. In the following the *escript* implementation is tested via comparison with an analytical solution for a layered half-space and with the numerical results of two published forward models.

Figure 7 shows the response of a three-layered half-space, with *escript* computations carried out for TM-mode excitation. Model parameters are described in the figure's caption. The analytical solution is based on Wait's recursion formula for a layered earth (Wait 1953). Small deviations occur at short periods, whereas at all other periods the curves virtually overlap.

The following two examples compare the *escript* apparent resistivity responses with published numerical solutions by the COMMEMI project (Zhdanov et al 1997). The compared models are the COMMEMI-1 and COMMEMI-4 2D models, where the apparent resistivity values for comparison are taken from those by Weaver and Brewitt-Taylor; identified as '3 Canada' in tables B.8 and B.33 in the paper by Zhdanov et al (1997, p.194 and p.219). The phase values for the COMMEMI-1 model are calculated from the field values in tables B.9 and B.10 of the same paper. Phase values for the COMMEMI-4 model are compared with phase values as calculated from field values as reported by Franke-Börner (2003, tables 5.7 and 5.8).

The COMMEMI-1 test model is shown in figure 8 together with the employed model parameters. The model simulates the response of a single conductive prismatic body in a more resistive uniform host. This model tests the ability to deal with highly varying values since the resistivity contrast is high, and the prism is large and near the surface, which therefore causes a large anomalous response. The *escript* responses

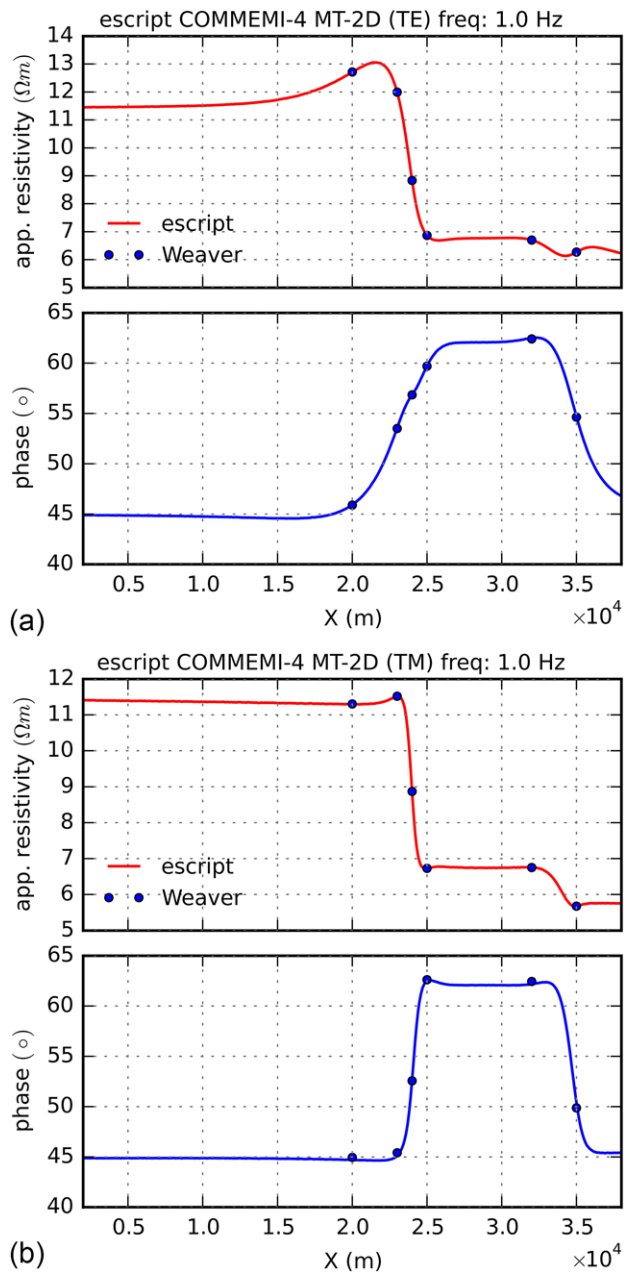


Figure 11. COMMEMI-4 model apparent resistivities and phase for TE-mode (a) and TM-mode (b). A sharp drop in apparent resistivity is observed at the edge of the graben-structure at the 24 km mark; another drop is noticeable near 32 km reflecting the 2.5 Ωm layer beneath. Comparison with apparent resistivity and phase values by Weaver are in good agreement (see table 2). (a) TE mode. (b) TM mode.

for TE- and TM-mode are displayed in figures 9(a) and (b), respectively. The apparent resistivities as well as phases are compared with the published results and are in good agreement for both modes. Numerical values are given in table 1 for comparison. The phase values for comparison were calculated via equation (44) from the reported field values in Zhdanov et al (1997).

The next example employs the COMMEMI-4 test model, which exhibits a more complex geometry as shown in figure 10; model parameters are given in the figure. The model consists of a total of five distinct regions, including

Table 2. COMMEMI-4 *escript* apparent resistivity and phase at selected coordinates for TE- and TM-mode as compared with the results by Weaver.

x (km)	20.0	23.0	24.0	25.0	32.0	35.0
ρ_a <i>escript</i> (TE)	12.70	12.00	8.84	6.87	6.70	6.27
Weaver (TE)	12.70	12.00	8.80	6.84	6.67	6.25
ρ_a <i>escript</i> (TM)	11.30	11.51	8.87	6.73	6.75	5.68
Weaver (TM)	11.40	11.50	9.03	6.78	6.80	5.71
ϕ <i>escript</i> (TE)	45.92	53.54	56.84	59.71	62.50	54.69
Weaver (TE)	45.88	53.50	56.84	59.69	62.39	54.63
ϕ <i>escript</i> (TM)	44.71	45.11	52.60	62.47	62.15	50.36
Weaver (TM)	44.94	45.40	52.57	62.61	62.43	49.87

an inclined boundary and a thin surficial layer, with different electrical resistivities varying between 2.5 Ωm and 1000 Ωm . The stratification on the left and right differs so that essential (Dirichlet) boundary conditions are calculated separately for both boundaries. The computed *escript* responses are compared in figures 11(a) and (b), and compared with Weaver's results in table 2 at selected coordinates.

In order to calculate to the desired accuracy, a total of 58044 nodes were used (TE-mode). The number of nodes can be reduced when using the secondary field approach with modified boundary conditions, instead of the total field approach employed here. The apparent resistivity and phase for both modes in figure 11 reflect the layered structure at the left side and represents the graben-structure with a steep decline in apparent resistivity near the 24 km mark. The embedded graben-like structure tapers off as a layer to the right model boundary and causes strong current-channeling effects within the 2.5 Ωm layer in both, TE- and TM-mode, as shown in Franke et al (2007). The apparent resistivities for both modes reflect the channelling effect above the 2.5 Ωm layer, but more so for the TM-mode, where a significant decrease in apparent resistivity is noticeable near the 32 km mark. The *escript* values at selected coordinates are nearly identical to those of Weaver as shown in table 2 (note that the phase values for comparison are here based on the field values as reported in Franke-Börner (2003)).

4. Summary

General-purpose finite element solvers find increasing application in the geophysical disciplines as these offer a single interface to tackle different geophysical problems. These solvers are useful for data interpretation and for research, but can also be a useful tool in educational settings. We developed the parallel FEM solver *escript* as an extension of *Python* with a focus on solving boundary value problems in geophysics. A key concept of *escript* is the abstraction from the underlying spatial discretisation method (i.e. finite elements), providing a convenient programming environment. In principle this allows implementation of other discretisation methods such as boundary or spectral elements. Because the numerical implementations are independent from data structures, simulations are easily portable across desktop computers and scalable compute clusters without modifications to the program code.

In this paper we have demonstrated the use of *escript* at two different geophysical forward modelling examples, 3D DC resistivity and 2D magnetotellurics, with the intention to accurately compute the model responses and compare with different methods. For implementation in *escript*, the weak form was derived and matched against the *escript* template. Comparisons with analytical and other numerical solutions

established confidence in the *escript* results. *Esript* is actively developed at the Centre for Geoscience Computing, University of Queensland and is released under the Open Software Licence v3.0. Up-to-date information, documentation and support are found on <https://launchpad.net/escript-finley>.

Appendix

Listing 1: sample code for 3D Resistivity modelling (single prism) with *escript*/Python

```
Description:
-----
Esript test for 3-D resistivity modelling (single prism).
Comparison with Pridmore, 1981, Fig. 5 [2].
References:
-----
[1] Gross et al., 2015, "escript-finley 4.0, User Guide", https://launchpad.net/escript-finley.
[2] Pridmore, D.F, et al., 1981, "An investigation of finite-element modeling for electrical and electromagnetic data in three dimensions", Geophysics, 7, 1009-1024
[3] C. Geuzaine and J.F. Remacle. 2009, "Gmsh: a three-dimensional finite element mesh generator with built-in pre- and post-processing facilities", International Journal for Numerical Methods in Engineering, 79, 1309-1331
Author:
-----
R. Schaa, University of Queensland, r.schaa@uq.edu.au
"""
import collections
import numpy as np
import esys.escript as escript
import esys.finley as finley
import esys.escript.pdetools as pdetools
import esys.escript.linearPDEs as pde
# ---
# Initialise required parameters and mesh.
# ---
# <Note>: tag-names & electrode coordinates must match those in the mesh.
# Mesh input-file was prepared with 'gmsh' [3] and consist of a single
# prism (sig=1/20) in an otherwise homogeneous half-space (sig=1/100).
# An inner region for mesh refinement was added around the prism.
mesh_file = "prism_v1.1.msh"
# Dipole-dipole parameters:
xe_0 = -10 # start-coordinate
xe_n = 10 # end-X-coordinate
asep = 1.0 # step size
nume = (xe_n - xe_0) / asep # number of electrodes
nmax = 6 # max Separation factor
curr = 1 # current
# Generate linear list with electrode locations:
xp = np.linspace(xe_0, xe_n, num = nume + 1)
# The Dirac points are passed to escript with an
# identifier and its location - a dictionary
# is setup to store each id and location:
dirac = collections.OrderedDict()
for i in xrange(0, len(xp)):
    dirac["e%02d"%(i)] = [xp[i], 0, 0]
# Now load the existing mesh:
diracTags = dirac.keys(); diracPoints = dirac.values()
domain = finley.ReadGmsh(mesh_file, numDim = 3,
```

```

        diracTags = diracTags, diracPoints = diracPoints)
# ---
# Setup locations for Dirichlet boundary conditions.
# ---
# Get the mesh coordinates at the nodes via 'ContinuousFunction':
x = escript.ContinuousFunction(domain).getX()
# Setup masks for the domain boundary (whereZero==1' where x==0):
mask_w = escript.whereZero(x[0] - escript.inf(x[0])) # west.
mask_e = escript.whereZero(x[0] - escript.sup(x[0])) # east.
mask_s = escript.whereZero(x[1] - escript.inf(x[1])) # south.
mask_n = escript.whereZero(x[1] - escript.sup(x[1])) # north.
mask_b = escript.whereZero(x[2] - escript.inf(x[2])) # bottom.
# Boundary mask for Dirichlet condition everywhere except the top:
qmask = mask_b + mask_w + mask_e + mask_s + mask_n
# ---
# Define the primary and secondary conductivity model.
# ---
# Primary and secondary conductivity:
p_sigma = 1/100.0; s_sigma = 1/20.0
# Define conductivities for the mesh-domains via a dictionary.
# The dictionary keys conform to the names defined in the mesh.
tag_p = "domain":p_sigma, "inner":p_sigma, "prism":p_sigma # Primary
tag_s = "domain":p_sigma, "inner":p_sigma, "prism":s_sigma # Secondary.
# Instantiate escript objects for the conductivity,
# which operate on mesh elements (i.e. 'Function'):
sig_s = escript.Scalar(0, escript.Function(domain))
sig_p = escript.Scalar(0, escript.Function(domain))
# Cycle 'tag' dictionaries and assign conductivity
# values to the escript objects via 'tagging':
for tag in tag_p:
    sig_p.setTaggedValue(tag, tag_p[tag])
for tag in tag_s:
    sig_s.setTaggedValue(tag, tag_s[tag])
# ---
# Solve the PDEs.
# ---
# Instantiate Escript PDE objects for primary and secondary potential:
pde_p = pde.LinearPDE(domain, numEquations=1)
pde_s = pde.LinearPDE(domain, numEquations=1)
# Setup A-coefficient for primary and secondary potential:
A_p = sig_p*escript.kronecker(domain)
A_s = sig_s*escript.kronecker(domain)

# Cycle n-factor values for dipole-dipole spread:
for n in xrange(1, nmax+1):
    # Initialise current/potential electrode index:
    ci = -1; pi = -1
    # Cycle electrode pairs:
    while pi < nume-1:
        # Set current/potential electrode index:
        ci = ci + 1; pi = ci + 2 + (n-1)
        # Location of nearest mesh nodes for the actual electrodes:
        loc = pdetools.Locator(escript.ContinuousFunction(domain),
                               [diracPoints[pi], diracPoints[pi+1]])
        # Electrode coordinates:
        c1 = diracPoints[ci+0]; c2 = diracPoints[ci+1]
        p1 = diracPoints[pi+0]; p2 = diracPoints[pi+1]

```

```

# Setup primary potential with Dirac points:
Y_p = pde_p.createCoefficient("y_dirac")
Y_p = escript.Scalar(0, escript.DiracDeltaFunctions(domain))
Y_p.setTaggedValue(dircTags[ci + 0], +curr)
Y_p.setTaggedValue(dircTags[ci + 1], -curr)
# Primary potential result:
pde_p.setValue(A=A_p, y_dirac=Y_p, q=qmask, r=0)
u_p = pde_p.getSolution()

# Setup secondary potential with conductivity gradient source:
X_s = pde_s.createCoefficient("X")
X_s = escript.Scalar(0, escript.Function(domain))
X_s = (sig_p - sig_s) * escript.grad(u_p)

# Secondary potential result:
pde_s.setValue(A=A_s, X=X_s, q=qmask, r=0)
u_s = pde_s.getSolution()

# Total potential:
u_t = u_p + u_s
# Apparent resistivity:
U = loc.getValue(u_t) # Potential at given location
K = np.pi * n * (n + 1) * (n + 2) * asep # Geometric dipole-dipole factor
R = K * (U[1] - U[0]) # Apparent resistivity
# Show n-value, electrode positions and apparent resistivity:
print("%i, " + 5 * "%9.2f") % (n, c1[0], c2[0], p1[0], p2[0], R)

```

References

- Bakhos T, Cardiff M, Barrash W and Kitanidis P K 2014 Data processing for oscillatory pumping tests *J. Hydrol.* **511** 310–9
- Berdichevsky M, Dmitriev V and Pozdnjakova E 1998 On two-dimensional interpretation of magnetotelluric soundings *Geophys. J. Int.* **133** 585–606
- Berselli L C, Cerminara M and Iliescu T 2015 Disperse two-phase flows, with applications to geophysical problems *Pure Appl. Geophys.* **172** 181–96
- Blome M, Maurer H R and Schmidt K 2009 Advances in three-dimensional geoelectric forward solver techniques *Geophys. J. Int.* **176** 740–52
- Butler S and Sinha G 2012 Forward modeling of applied geophysics methods using Comsol and comparison with analytical and laboratory analog models *Comput. Geosci.* **42** 168–76
- Chave A and Jones A 2012 *The Magnetotelluric Method: Theory and Practice* (Cambridge: Cambridge University Press)
- Davis T A 2004 Algorithm 832: UMFPACK V4.3—an unsymmetric-pattern multifrontal method *ACM Trans. Math. Softw.* **30** 196–9
- Dey A and Morrison H F 1979 Resistivity modeling for arbitrarily shaped three-dimensional structures *Geophysics* **44** 753–80
- Dyksterhuis S, Albert R and Müller R 2005 Finite-element modelling of contemporary and palaeo-intraplate stress using ABAQUS *Comput. Geosci.* **31** 297–307
- Finzi Y, Mühlhaus H, Gross L and Amirbekyan A 2013 Shear band formation in numerical simulations applying a continuum damage rheology model *Pure Appl. Geophys.* **170** 13–25
- Franke A, Börner R U and Spitzer K 2007 Adaptive unstructured grid finite element simulation of two-dimensional magnetotelluric fields for arbitrary surface and seafloor topography *Geophys. J. Int.* **171** 71–86
- Franke-Börner A 2003 Zweidimensionale finite-element-modellierung niederfrequenter elektromagnetischer Felder in der Fernzone *Master's Thesis* Institut für Geophysik. Technische Universität Bergakademie, Freiberg
- Gross L, Bourguoin L, Hale A and Mühlhaus H B 2007 Interface modeling in incompressible media using level sets in Escrip *Phys. Earth Planet. Inter.* **163** 23–34
- Gross L et al 2015 escript-finley 4.0
- Gross L and Kemp C 2013 Large scale joint inversion of geophysical data using the finite element method in escript *ASEG Extended Abstracts* pp 1–4
- Gross L, Poulet T and Sheldon H A 2013 Conforming finite element methods for modeling convection in an incompressible rock matrix *Trans. Porous Media* **100** 225–46
- Guo L, Huang H, Gaston D R, Permann C J, Andrs D, Redden G D, Lu C, Fox D T and Fujita Y 2013 A parallel, fully coupled, fully implicit solution to reactive transport in porous media using the preconditioned Jacobian-free Newton-Krylov method *Adv. Water Resour.* **53** 101–8
- Ikard S, Revil A, Jardani A, Woodruff W, Parekh M and Mooney M 2012 Saline pulse test monitoring with the self-potential method to nonintrusively determine the velocity of the pore water in leaking areas of earth dams and embankments *Water Resour. Res.* **48** W04201
- Johansen H and Sørensen K 1979 Fast hankel transforms *Geophys. Prospect.* **27** 876–901
- Key K and Owall J 2011 A parallel goal-oriented adaptive finite element method for 2.5D electromagnetic modelling *Geophys. J. Int.* **186** 137–54
- Key K and Weiss C 2006 Adaptive finite-element modeling using unstructured grids: the 2D magnetotelluric example *Geophysics* **71** G291–9
- Kisak E and Silvester P 1975 A finite-element program package for magnetotelluric modelling *Comput. Phys. Commun.* **10** 421–33

- Li Y and Spitzer K 2002 Three-dimensional DC resistivity forward modelling using finite elements in comparison with finite-difference solutions *Geophys. J. Int.* **151** 924–34
- Logg A et al 2012 *Automated Solution of Differential Equations by the Finite Element Method* (Berlin: Springer)
- Lowry T, Allen M and Shive P N 1989 Singularity removal: a refinement of resistivity modeling techniques *Geophysics* **54** 766–74
- Ma Q 2002 The boundary element method for 3D dc resistivity modeling in layered earth *Geophysics* **67** 610–7
- Mortensen M, Langtangen H P and Wells G N 2011 A FEniCS-based programming framework for modeling turbulent flow by the Reynolds-averaged Navier–Stokes equations *Adv. Water Resour.* **34** 1082–101 (New computational methods and software tools)
- Muhlhaus H, Gross L and Scheuermann A 2014 Sand erosion as an internal boundary value problem *Acta Geotechnica* pp 1–10
- Pain C C, Herwanger J V, Worthington M H and de Oliveira C R 2002 Effective multidimensional resistivity inversion using finite-element techniques *Geophys. J. Int.* **151** 710–28
- Penz S, Chauris H, Donno D and Mehl C 2013 Resistivity modelling with topography *Geophys. J. Int.* **194** ggt169
- Poulet T, Gross L, Georgiev D and Cleverley J 2012 escript-RT: reactive transport simulation in Python using escript *Comput. Geosci.* **45** 168–76
- Pridmore D, Hohmann G, Ward S and Sill W 1981 An investigation of finite-element modeling for electrical and electromagnetic data in three dimensions *Geophysics* **46** 1009–24
- Ren Z and Tang J 2014 A goal-oriented adaptive finite-element approach for multi-electrode resistivity system *Geophys. J. Int.* **199** 136–45
- Rohmer J 2014 Induced seismicity of a normal blind undetected reservoir-bounding fault influenced by dissymmetric fractured damage zones *Geophys. J. Int.* **197** 636–41
- Rücker C, Günther T and Spitzer K 2006 Three-dimensional modelling and inversion of dc resistivity data incorporating topography—I. Modelling *Geophys. J. Int.* **166** 495–505
- Rung-Arunwan T and Siripunvaraporn W 2010 An efficient modified hierarchical domain decomposition for two-dimensional magnetotelluric forward modelling *Geophys. J. Int.* **183** 634–44
- Saad Y 1996 *Iterative Methods for Sparse Linear Systems* (Boston: PWS Publishing Company)
- Schloegel K, Karypis G and Kumar V 2002 Parallel static and dynamic multi-constraint graph partitioning *Concurrency Comput.: Pract. Exp.* **14** 219–40
- Seyedi D, Ducellier A, Foerster E, Guy N, Hild F and Rohmer J 2009 Coupled hydromechanical modeling to study the integrity and safety of geological storage of CO₂ *Energy Procedia* **1** 2541–8
- Simpson F and Bahr K 2005 *Practical Magnetotellurics* (Cambridge: Cambridge University Press)
- Spies B R and Eggers D E 1986 The use and misuse of apparent resistivity in electromagnetic methods *Geophysics* **51** 1462–71
- Wait J R 1953 Propagation of radio waves over a stratified ground *Geophysics* **18** 416–22
- Wannamaker P E, Stodt J A and Rijo L 1987 A stable finite element solution for two-dimensional magnetotelluric modelling *Geophys. J. Int.* **88** 277–96
- Weiss R 1996 *Parameter-Free Iterative Linear Solvers* (*Mathematical Research* vol 97) (Berlin: Akademie)
- Zhang D F 2010 Forward modeling of direct current method based on ANSYS *Proc. PIERS 2010 in Xi'an (China): The Electromagnetics Academy*
- Zhao S and Yedlin M J 1996 Some refinements on the finite-difference method for 3D dc resistivity modeling *Geophysics* **61** 1301–7
- Zhdanov M and Keller G 1994 *The Geoelectrical Methods in Geophysical Exploration (Methods in Geochemistry and Geophysics)* (Amsterdam: Elsevier)
- Zhdanov M, Varentsov I M, Weaver J, Golubev N and Krylov V 1997 Methods for modelling electromagnetic fields results from COMMEMI—the international project on the comparison of modelling methods for electromagnetic induction *J. Appl. Geophys.* **37** 133–271
- Zienkiewicz O C, Taylor R L and Zhu J Z 2013 *The Finite Element Method: Its Basis and Fundamentals* 7th edn (Amsterdam: Elsevier)