

A model reduction approach to numerical inversion for a parabolic partial differential equation

Liliana Borcea¹, Vladimir Druskin²,
Alexander V Mamonov³ and Mikhail Zaslavsky²

¹Department of Mathematics, University of Michigan, 2074 East Hall, 530 Church Street, Ann Arbor, MI 48109-1043, USA

²Schlumberger-Doll Research Center, 1 Hampshire St., Cambridge, MA 02139-1578, USA

³Schlumberger, 3750 Briarpark Dr., Houston, TX 77042, USA

E-mail: borcea@umich.edu, druskin1@slb.com, amamonov@slb.com and mzaslavsky@slb.com

Received 9 June 2014, revised 26 September 2014

Accepted for publication 8 October 2014

Published 19 November 2014

Abstract

We propose a novel numerical inversion algorithm for the coefficients of parabolic partial differential equations, based on model reduction. The study is motivated by the application of controlled source electromagnetic exploration, where the unknown is the subsurface electrical resistivity and the data are time resolved surface measurements of the magnetic field. The algorithm presented in this paper considers inversion in one and two dimensions. The reduced model is obtained with rational interpolation in the frequency (Laplace) domain and a rational Krylov subspace projection method. It amounts to a nonlinear mapping from the function space of the unknown resistivity to the small dimensional space of the parameters of the reduced model. We use this mapping as a nonlinear preconditioner for the Gauss–Newton iterative solution of the inverse problem. The advantage of the inversion algorithm is twofold. First, the nonlinear preconditioner resolves most of the nonlinearity of the problem. Thus the iterations are less likely to get stuck in local minima and the convergence is fast. Second, the inversion is computationally efficient because it avoids repeated accurate simulations of the time-domain response. We study the stability of the inversion algorithm for various rational Krylov subspaces, and assess its performance with numerical experiments.

Keywords: inverse problem, parabolic equation, model reduction, rational Krylov subspace projection, CSEM

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

1. Introduction

Inverse problems for parabolic partial differential equations arise in applications such as groundwater flow, solute transport and controlled source electromagnetic oil and gas exploration. We consider the latter problem, where the unknown $r(\mathbf{x})$ is the electrical resistivity, the coefficient in the diffusion Maxwell system satisfied by the magnetic field $\mathbf{H}(t, \mathbf{x})$

$$-\nabla \times [r(\mathbf{x}) \nabla \times \mathbf{H}(t, \mathbf{x})] = \frac{\partial \mathbf{H}(t, \mathbf{x})}{\partial t}, \quad (1.1)$$

for time $t > 0$ and \mathbf{x} in some spatial domain. The data from which $r(\mathbf{x})$ is to be determined are the time resolved measurements of $\mathbf{H}(t, \mathbf{x})$ at receivers located on the boundary of the domain.

Determining $r(\mathbf{x})$ from the boundary measurements is challenging especially because the problem is ill-posed and thus sensitive to noise. A typical numerical approach is to minimize a functional given by the least squares data misfit and a regularization term, using Gauss–Newton or nonlinear conjugate gradient methods [22, 23, 25]. There are two main drawbacks. First, the functional to be minimized is not convex and the optimization algorithms can get stuck in local minima. The lack of convexity can be overcome to some extent by adding more regularization at the cost of artifacts in the solution. Nevertheless, convergence may be very slow [23]. Second, evaluations of the functional and its derivatives are computationally expensive, because they involve multiple numerical solutions of the forward problem. In applications the computational domains may be large with meshes refined near sources, receivers and regions of strong heterogeneity. This results in a large number of unknowns in the forward problem, and time stepping with such large systems is expensive over long time intervals.

We propose a numerical inversion approach that arises when considering the inverse problem in the model reduction framework. We consider one and two-dimensional (1 and 2D) media, and denote the spatial variable by $\mathbf{x} = (x, z)$, with $z \in \mathbb{R}$ and $x \in \Omega$, a simply connected domain in \mathbb{R}^n with boundary \mathcal{B} , for $n = 1, 2$. The setting is illustrated in figure 1.

The resistivity is $r = r(x)$, and assuming that

$$\mathbf{H} = u(t, x) \mathbf{e}_z, \quad (1.2)$$

we obtain from (1.1) the parabolic equation

$$\nabla \cdot [r(x) \nabla u(t, x)] = \frac{\partial u(t, x)}{\partial t}, \quad (1.3)$$

for $t > 0$ and $x \in \Omega$. The boundary \mathcal{B} consists of an accessible part \mathcal{B}_A , which supports the receivers at which we make the measurements and the initial excitation

$$u(0, x) = u_o(x), \quad \text{supp}\{u_o\} \subset \mathcal{B}_A, \quad (1.4)$$

and an inaccessible part $\mathcal{B}_I = \mathcal{B} \setminus \mathcal{B}_A$, where we set

$$u(t, x) = 0, \quad x \in \mathcal{B}_I. \quad (1.5)$$

The boundary condition at \mathcal{B}_A is

$$\mathbf{n}(x) \cdot \nabla u(t, x) = 0, \quad x \in \mathcal{B}_A, \quad (1.6)$$

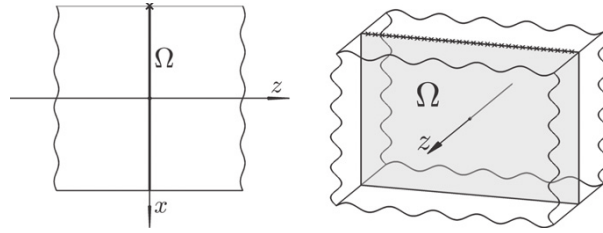


Figure 1. Examples of spatial domains for equation (1.1) in \mathbb{R}^2 (left) and \mathbb{R}^3 (right). The medium is constant in the direction z , which is transversal to Ω . The accessible boundary $\mathcal{B}_A \subset \mathcal{B} = \partial\Omega$ is marked with \times .

where \mathbf{n} is the outer normal. We choose the boundary conditions (1.5)–(1.6) to simplify the presentation. Other (possibly inhomogeneous) conditions can be taken into account and amount to minor modifications of the reduced models described in this paper.

The sources and receivers, are located on the accessible boundary \mathcal{B}_A , and provide knowledge of the operator

$$\mathcal{M}(u_o) = u(t, x) \Big|_{x \in \mathcal{B}_A}, \quad t > 0, \quad (1.7)$$

for every u_o such that $\text{supp}\{u_o\} \subset \mathcal{B}_A$. Here $u(t, x)$ is the solution of (1.3) with the initial condition (1.4). The inverse problem is to determine the resistivity $r(x)$ for $x \in \Omega$ from \mathcal{M} . Note that the operator \mathcal{M} is highly nonlinear in r , but it is linear in u_o . This implies that in one dimension where Ω is an interval, say $\Omega = (0, 1)$, and the accessible boundary is a single point $\mathcal{B}_A = \{x = 0\}$, \mathcal{M} is completely defined by $u_o(x) = \delta(x)$. All the information about $r(x)$ is contained in a single function of time

$$y(t) = \mathcal{M}(\delta(x)) = u(t, 0), \quad t > 0. \quad (1.8)$$

To ease the presentation, we begin by describing in detail the model reduction inversion method for the 1D case. Then we show how to extend it to the 2D case. The reduced model is obtained with a rational Krylov subspace projection method. Such methods have been applied to forward problems for parabolic equations in [7, 10, 13], and to inversion in one dimension in [11] and multiple dimensions in [20]. The reduced models in [11] are \mathcal{H}_2 rational interpolants of the transfer function defined by the Laplace transform of $y(t)$ from (1.8). In this paper we build on the results in [11] to study in more detail and improve the inversion method in one dimension, and to extend it to two dimensions.

The reduced order models allow us to replace the solution of the full scale parabolic problem by its low order projection, thus resolving the high computational cost inversion challenge mentioned above. Conventionally, the reduced order model (the rational approximant of the transfer function) is parametrized in terms of its poles and residues. We parametrize it instead in terms of the coefficients of its continued fraction expansion. The rational approximant can be viewed as the transfer function of a finite difference discretization of (1.3) on a special grid with rather few points, known in the literature as the optimal or spectrally matched grid, or as a finite-difference Gaussian rule [8]. The continued fraction coefficients are the entries in the finite difference operator, which are related to the discrete resistivities.

To mitigate the other inversion challenge mentioned above, we introduce a nonlinear mapping \mathcal{R} from the function space of the unknown resistivity r to the low-dimensional space of the discrete resistivities. This map appears to resolve the nonlinearity of the problem and

we use it in a preconditioned Gauss–Newton iteration that is less likely to get stuck in local minima and converges quickly, even when the initial guess is far from the true resistivity.

To precondition the problem, we map the measured data to the discrete resistivities. The inverse problem is ill-posed, so we limit the number of discrete resistivities (i.e. the size of the reduced order model) computed from the data. This number depends on the noise level and it is typically much less than the dimension of models used in conventional algorithms, where it is determined by the accuracy of the forward problem solution. This represents another significant advantage of our approach.

The paper is organized as follows: we begin in section 2 with a detailed description of our method in one dimension. The inversion in 2D media is described in section 3. Numerical results in one and two dimensions are presented in section 4. We conclude with a summary in section 5. The technical details of the computation of the nonlinear preconditioner \mathcal{R} and its Jacobian are given in the appendix.

2. Model order reduction for inversion in 1D media

In this section we study in detail the use of reduced order models for inversion in one dimension. Many of the ideas presented here are used in section 3 for the two-dimensional inversion. We begin in section 2.1 by introducing a semi-discrete analogue of the continuum inverse problem, where the differential operator in x in equation (1.3) is replaced by a matrix. This is done in any numerical inversion, and we use it from the start to adhere to the conventional setting of model order reduction, which is rooted in linear algebra. The projection-based reduced order models are described in section 2.2. They can be parametrized in terms of the coefficients of a certain reduced order finite difference scheme, used in section 2.3 to define a pair of nonlinear mappings. They define the objective function for the nonlinearly preconditioned optimization problem, as explained in section 2.4. To give some intuition to the preconditioning effect, we relate the mappings to so-called optimal grids in section 2.5. The stability of the inversion is addressed in section 2.6, and our regularization scheme is given in section 2.7. The detailed description of the 1D inversion algorithm is in section 2.8.

2.1. Semi-discrete inverse problem

In one dimension the domain is an interval, which we scale to $\Omega = (0, 1)$, with accessible and inaccessible boundaries $\mathcal{B}_A = \{0\}$ and $\mathcal{B}_I = \{1\}$ respectively. To adhere to the conventional setting of model order reduction, we consider the semi-discretized equation (1.3)

$$\frac{\partial \mathbf{u}(t)}{\partial t} = A(\mathbf{r})\mathbf{u}(t), \quad (2.1)$$

where

$$A(\mathbf{r}) = -D^T \text{diag}(\mathbf{r})D, \quad (2.2)$$

is a symmetric and negative definite matrix, the discretization of $\partial_x[r(x)\partial_x]$. The vector $\mathbf{r} \in \mathbb{R}_+^N$ contains the discrete values of $r(x)$ and the matrix $D \in \mathbb{R}^{N \times N}$ arises in the finite difference discretization of the derivative in x , for boundary conditions (1.6)–(1.5). The discretization is on a very fine uniform grid with N points in the interval $[0, 1]$, and spacing $h = 1/(N + 1)$. Note that our results do not depend on the dimension N , and all the derivations can be carried out for a continuum differential operator. We let A be a matrix to avoid unnecessary technicalities. The vector $\mathbf{u}(t) \in \mathbb{R}^N$ is the discretization of $u(t, x)$, and the

initial condition is

$$\mathbf{u}(0) = \frac{1}{h} \mathbf{e}_1, \quad (2.3)$$

the approximation of a point source excitation at B_A . The time-domain response of the semi-discrete dynamical system (2.1) is given by

$$y(t; \mathbf{r}) = \mathbf{e}_1^T \mathbf{u}(t), \quad (2.4)$$

where $\mathbf{e}_1 = (1, 0, 0, \dots, 0)^T$. This is a direct analogue of (1.8), i.e. it corresponds to measuring the solution of (2.1) at the left end point $B_A = \{0\}$ of the interval Ω . We emphasize in the notation that the response depends on the vector \mathbf{r} of discrete resistivities. We use (2.4) to define the *forward map*

$$\mathcal{F}: \mathbb{R}_+^N \rightarrow C(0, +\infty) \quad (2.5)$$

that takes the vector $\mathbf{r} \in \mathbb{R}_+^N$ to the time domain response

$$\mathcal{F}(\mathbf{r}) = y(\cdot; \mathbf{r}). \quad (2.6)$$

The measured time domain data is denoted by

$$d(t) = y(t; \mathbf{r}^{\text{true}}) + \mathcal{N}(t), \quad t > 0, \quad (2.7)$$

where \mathbf{r}^{true} is the true (unknown) resistivity vector and $\mathcal{N}(t)$ is the contribution of the measurements noise and the discretization errors. The inverse problem is: given data $d(t)$ for $t \in [0, \infty)$, recover the resistivity vector.

2.2. Projection-based model order reduction

In order to apply the theory of model order reduction we treat (2.1)–(2.4) as a dynamical system with the time domain response

$$y(t; \mathbf{r}) = \mathbf{e}_1^T e^{A(\mathbf{r})t} \frac{\mathbf{e}_1}{h} = \mathbf{b}^T e^{A(\mathbf{r})t} \mathbf{b} \quad (2.8)$$

written in a symmetrized form using the source/measurement vector

$$\mathbf{b} = \mathbf{e}_1 / \sqrt{h}. \quad (2.9)$$

The transfer function of the dynamical system is the Laplace transform of (2.8)

$$Y(s; \mathbf{r}) = \int_0^{+\infty} y(t; \mathbf{r}) e^{-st} dt = \mathbf{b}^T (sI - A(\mathbf{r}))^{-1} \mathbf{b}, \quad s > 0. \quad (2.10)$$

Since $A(\mathbf{r})$ is negative definite, all the poles of the transfer function are negative and the dynamical system is stable.

In model order reduction we obtain a reduced model (A_m, \mathbf{b}_m) so that its transfer function

$$Y_m(s) = \mathbf{b}_m^T (sI_m - A_m)^{-1} \mathbf{b}_m \quad (2.11)$$

is a good approximation of $Y(s; \mathbf{r})$ as a function of s in some norm. Here I_m is the $m \times m$ identity matrix, $A_m \in \mathbb{R}^{m \times m}$, $\mathbf{b}_m \in \mathbb{R}^m$ and $m \ll N$. Note that while the matrix $A(\mathbf{r})$ given by (2.2) is sparse, the reduced order matrix A_m is typically dense. Thus, it has no straightforward interpretation as a discretization of the differential operator $\partial_x[r(x)\partial_x]$ on some coarse grid with m points.

Projection-based methods search for reduced models of the form

$$A_m = V^T A V, \quad \mathbf{b}_m = V^T \mathbf{b}, \quad V^T V = I_m, \quad (2.12)$$

where the columns of $V \in \mathbb{R}^{N \times m}$ form an orthonormal basis of an m -dimensional subspace of \mathbb{R}^N on which the system is projected. The choice of V depends on the matching conditions for Y_m and Y . They prescribe the sense in which Y_m approximates Y . Here we consider moment matching at interpolation nodes $s_j \in [0, +\infty)$ that may be distinct or coinciding,

$$\left. \frac{\partial^k Y_m}{\partial s^k} \right|_{s=s_j} = \left. \frac{\partial^k Y}{\partial s^k} \right|_{s=s_j}, \quad k = 0, \dots, 2M_j - 1, \quad j = 1, \dots, l. \quad (2.13)$$

The multiplicity of a node s_j is denoted by M_j , so at non-coinciding nodes $M_j = 1$. The reduced order transfer function Y_m matches Y and its derivatives up to the order $2M_j - 1$, and the size m of the reduced model is given by $m = \sum_{j=1}^l M_j$.

Note from (2.11) that $Y_m(s)$ is a rational function of s , with partial fraction representation

$$Y_m(s) = \sum_{j=1}^m \frac{c_j}{s + \theta_j}, \quad c_j > 0, \quad \theta_j > 0. \quad (2.14)$$

Its poles $-\theta_j$ are the eigenvalues of A_m and the residues c_j are defined in terms of the normalized eigenvectors \mathbf{z}_j ,

$$c_j = (\mathbf{b}_m^T \mathbf{z}_j)^2, \quad A_m \mathbf{z}_j + \theta_j \mathbf{z}_j = 0, \quad \|\mathbf{z}_j\| = 1, \quad j = 1, \dots, m. \quad (2.15)$$

Thus, (2.13) is a rational interpolation problem. It is known [16] to be equivalent to the projection (2.12) when the columns of V form an orthogonal basis of the rational Krylov subspace

$$\mathcal{K}_m(s) = \text{span} \left\{ (s_j I - A)^{-k} \mathbf{b} \mid j = 1, \dots, l; \quad k = 1, \dots, M_j \right\}. \quad (2.16)$$

The interpolation nodes, obviously, should be chosen in the resolvent set of $A(\mathbf{r})$. Moreover, since in reality we need to solve a limiting continuum problem, the nodes should be in the closure of the intersection of the resolvent sets of any sequence of finite-difference operators that converge to the continuum problem. This set includes $\mathbb{C} \setminus (-\infty, 0)$ for problems on bounded domains. In our computations the interpolation nodes lie on the positive real axis, since they correspond to working with the Laplace transform of the time domain response.

Ideally, in the solution of the inverse problem we would like to minimize the time (or frequency) domain data misfit in a quadratic norm weighted in accordance with the statistical distribution of the measurement error. When considering the reduced order model, it is natural to choose the interpolation nodes that give the most accurate approximation in that norm. Such interpolation is known in control theory as \mathcal{H}_2 (Hardy space) optimal, and in many cases the optimal interpolation nodes can be found numerically [17]. Moreover, it was shown in [11], that the solution of the inverse problem using reduced order models with such interpolation nodes also minimizes the misfit functional (in the absence of measurement errors). When such nodes are not available, we can select some reasonable interpolation nodes chosen based on *a priori* error estimates, for example, the so-called Zolotarev points or their approximations obtained with the help of potential theory [10, 24]. In most cases such choices lead to interpolation nodes that are distributed geometrically.

2.3. Finite difference parametrization of reduced order models

As we mentioned above, even though the reduced order model (A_m, \mathbf{b}_m) comes from a finite difference operator A , it does not retain its structure. In particular, A_m is a dense matrix. The model can be uniquely parametrized by $2m$ numbers, for example the poles $-\theta_j$ and residues c_j in the representation (2.14). Here we show how to reparametrize it so that the resulting $2m$ parameters have a meaning of finite difference coefficients.

A classical result of Stieltjes says that any rational function of the form (2.14) with negative poles and positive residues admits a representation as a Stieltjes continued fraction with positive coefficients

$$Y_m(s) = \frac{1}{\hat{\kappa}_1 s + \frac{1}{\kappa_1 + \frac{1}{\hat{\kappa}_1 s + \frac{1}{\ddots + \frac{1}{\hat{\kappa}_m s + \frac{1}{\kappa_m}}}}}}. \quad (2.17)$$

Obviously, this is true in our case, since $-\theta_j$ are the Ritz values of a negative definite operator A and the residues are given as squares in (2.15).

Furthermore, it is known from [8] that (2.17) is a boundary response $w_1(s)$ (Neumann-to-Dirichlet map) of a second order finite difference scheme with three point stencil

$$\frac{1}{\hat{\kappa}_j} \left(\frac{w_{j+1} - w_j}{\kappa_j} - \frac{w_j - w_{j-1}}{\kappa_{j-1}} \right) - s w_j = 0, \quad j = 2, \dots, m, \quad (2.18)$$

and boundary conditions

$$\frac{1}{\hat{\kappa}_1} \left(\frac{w_2 - w_1}{\kappa_1} \right) - s w_1 + \frac{1}{\hat{\kappa}_1} = 0, \quad w_{m+1} = 0. \quad (2.19)$$

These equations closely resemble the Laplace transform of (2.1), except that the discretization is at m nodes, which is much smaller than the dimension N of the fine grid.

It is convenient to work henceforth with the logarithms $\log \kappa_j$ and $\log \hat{\kappa}_j$ of the finite difference coefficients. We now introduce the two mappings \mathcal{Q} and \mathcal{R} that play the crucial role in our inversion method. We refer to the first mapping $\mathcal{Q}: C(0, +\infty) \rightarrow \mathbb{R}^{2m}$ as *data fitting*. It takes the time-dependent data $d(t)$ to the $2m$ logarithms of reduced order model parameters

$$\mathcal{Q}(d(\cdot)) = \left\{ \left(\log \kappa_j, \log \hat{\kappa}_j \right) \right\}_{j=1}^m \quad (2.20)$$

via the following chain of mappings

$$\begin{aligned} \mathcal{Q}: d(t) &\xrightarrow{(a)} Y(s) \xrightarrow{(b)} Y_m(s) \xrightarrow{(c)} \left\{ (c_j, \theta_j) \right\}_{j=1}^m \\ &\xrightarrow{(d)} \left\{ (\kappa_j, \hat{\kappa}_j) \right\}_{j=1}^m \xrightarrow{(e)} \left\{ (\log \kappa_j, \log \hat{\kappa}_j) \right\}_{j=1}^m. \end{aligned} \quad (2.21)$$

Here step (a) is a Laplace transform of the measured data $d(t)$, step (b) requires solving the rational interpolation problem (2.13), which is converted to a partial fraction form in step (c), which in turn is transformed to a Stieltjes continued fraction form in step (d) with a variant of Lanczos iteration, as explained in the appendix.

Note step (b) is the only ill-conditioned computation in the chain. The ill-posedness is inherited from the instability of the parabolic inverse problem and we mitigate it by limiting m . The instability may be understood intuitively by noting that step (b) is related to analytic continuation. We give more details in section (2.6). In practice we choose m so that the resulting κ_j and $\hat{\kappa}_j$ are positive. This determines the maximum number of degrees of freedom that we can extract from the data at the present noise level.

The mapping $\mathcal{R}: \mathbb{R}_+^N \rightarrow \mathbb{R}^{2m}$ is the *nonlinear preconditioner*. It takes the vector of discrete resistivities $\mathbf{r} \in \mathbb{R}_+^N$ to the same output as \mathcal{Q} . In simplest terms, it is a composition of \mathcal{Q} and the forward map (2.6) $\mathcal{R} = \mathcal{Q} \circ \mathcal{F}$ given by

$$\mathcal{R}(\mathbf{r}) = \mathcal{Q}(y(\cdot; \mathbf{r})). \quad (2.22)$$

Unlike the data fitting, the computation of \mathcal{R} can be done in a stable manner using a chain of mappings

$$\begin{aligned} \mathcal{R}: \mathbf{r} &\xrightarrow{(a)} A(\mathbf{r}) \xrightarrow{(b)} V \xrightarrow{(c)} A_m \xrightarrow{(d)} \left\{ (c_j, \theta_j) \right\}_{j=1}^m \\ &\xrightarrow{(e)} \left\{ (\kappa_j, \hat{\kappa}_j) \right\}_{j=1}^m \xrightarrow{(f)} \left\{ (\log \kappa_j, \log \hat{\kappa}_j) \right\}_{j=1}^m. \end{aligned} \quad (2.23)$$

Here step (a) is just the definition of A , in (b) we compute the orthonormal basis V for the rational Krylov subspace (2.16) on which A is projected in step (c) to obtain A_m and \mathbf{b}_m . Then, the poles and residues (2.15) are computed. The last two steps are the same as in the computation of \mathcal{Q} .

2.4. Nonlinearly preconditioned optimization

Now that we defined the data fitting \mathcal{Q} and the nonlinear preconditioner \mathcal{R} mappings, we can formulate our method for solving the semi-discrete inverse problem as an optimization. Given the data $d(t)$ for $t \in [0, +\infty)$ (recall (2.7)), we estimate the true resistivity \mathbf{r}^{true} by \mathbf{r}^* , the solution of the nonlinear optimization problem

$$\mathbf{r}^* = \arg \min_{\mathbf{r} \in \mathbb{R}_+^N} \frac{1}{2} \|\mathcal{Q}(d(\cdot)) - \mathcal{R}(\mathbf{r})\|_2^2. \quad (2.24)$$

This is different than the typical optimization-based inversion, which minimizes the L_2 norm of the (possibly weighted) misfit between the measured data $d(\cdot)$ and the model $\mathcal{F}(\mathbf{r})$. Such an approach is known to have many issues. In particular, the functional is often non-convex with many local minima, which presents a challenge for derivative-based methods (steepest descent, nonlinear conjugate gradients, Gauss–Newton). The convergence is often slow and some form of regularization is required. In our approach we aim to convexify the objective functional by constructing the nonlinear preconditioner \mathcal{R} .

To explain the map \mathcal{R} , let us give a physical interpretation of the reduced model parameters $\{(\kappa_j, \hat{\kappa}_j)\}_{j=1}^m$ by introducing the change of coordinates $x \rightsquigarrow \xi$, so that $r^{1/2} \partial_x = \partial_\xi$. The equation for $U(s, \xi)$, the Laplace transform of $u(t, x(\xi))$, is

$$r^{-1/2} \partial_\xi (r^{1/2} \partial_\xi U) - sU = 0 \quad (2.25)$$

and (2.18) is its discretization on a staggered grid. The coefficients $\hat{\kappa}_j$ and κ_j are the increments in the discretization of the differential operators $r^{-1/2} \partial_\xi$ and $r^{1/2} \partial_\xi$, so they are proportional to the local values of $r^{1/2}$ and $r^{-1/2}$, respectively. Since $\mathcal{R} = \mathcal{Q} \circ \mathcal{F}$, an ideal choice of \mathcal{Q} would be an approximate inverse of \mathcal{F} for resistivity functions that can be approximated well on the discretization grid. It would interpolate in some manner the grid

values of $r(x)$, which are defined by $\{\hat{\kappa}_j^2, \kappa_j^{-2}\}_{j=1}^m$, up to some scaling factors that define the grid spacing.

Not all grids give good results, as explained in more detail in the next section. We can factor out the unknown grid spacings by working with the logarithm of the coefficients $\{\hat{\kappa}_j^2, \kappa_j^{-2}\}_{j=1}^m$ instead of the resistivity r . Although it is possible to calculate a good grid, we do not require it in our inversion method. However, the grids can be useful for judging the quality of different matching conditions and for visualizing the behavior of \mathcal{R} as shown in section 2.5.

The ill-posedness of the inverse problem is localized in the computation of the data fitting term $Q(d(\cdot))$ that is computed once. The instability of this computation can be controlled by reducing the size m of the reduced order model projection subspace. A good strategy to follow in practice is to choose the largest m such that all the coefficients $\kappa_j, \hat{\kappa}_j, j = 1, \dots, m$ computed by Q are positive.

Conventional optimization-based methods regularize the inversion by adding a penalty term to the objective function. Our approach is different. We acknowledge that there is a resolution versus stability trade-off by reducing the size of the reduced model, and view regularization only as a means of adding prior information about the unknown resistivity. If such information is available, we can incorporate it at each iteration of the Gauss–Newton method via a correction in the null space of the Jacobian $D\mathcal{R}$. The regularization scheme is discussed in detail in section 2.7.

Inversion via model order reduction is a recent idea that was introduced in [11, 12, 20]. In particular, the approach in [11] uses maps Q_{c_o} and \mathcal{R}_{c_o} to the spectral parameters of the reduced order model θ_j and $c_j, j = 1, \dots, m$. Unlike the continued fraction coefficients κ_j and $\hat{\kappa}_j$, the poles $-\theta_j$ and residues c_j do not have a physical meaning of resistivity and the mapping \mathcal{R}_{c_o} does not behave like an approximate identity, as \mathcal{R} does. Our definition of the mapping \mathcal{R} is based on the ideas from [5]. It allows us to improve the results of [11]. The improvement becomes especially pronounced in cases of high resistivity contrast, as shown in the numerical comparison in section 4.

The idea of convexification of the nonlinear inverse problem has been pursued before for hyperbolic equations in [3, 21] and global convergence results were obtained in [2]. However, it is not clear if these approaches apply to parabolic equations. Although both hyperbolic and parabolic equations can be transformed to the frequency domain via Fourier and Laplace transforms, their data is mapped to different parts of the complex plane where the spectral parameter lies. It is known that the transformations from the hyperbolic to the parabolic data are unstable [19], so one cannot directly apply the methods from [2, 3, 21] to the parabolic problem. The model reduction approach in this paper gives a specially designed discrete problem of small size which can be inverted stably.

2.5. Connection to optimal grids

We explain here that the map \mathcal{R} relates to an interpolant of the resistivity r on a special, so-called optimal grid. Although our inversion method does not involve directly such grids, it is beneficial to study them because they provide insight into the behavior of the nonlinear preconditioner \mathcal{R} . We give a brief description of the grids, and show that they depend strongly on the interpolation nodes s_j in the matching conditions (2.13). We use this dependence to conclude that not all rational approximants of the transfer function $Y(s; \mathbf{r})$ are useful in inversion, as shown in section 2.5.2. Then, we use in section 2.5.3 the optimal grids and the continued fraction coefficients κ_j and $\hat{\kappa}_j$ to visualize the action of \mathcal{R} on \mathbf{r} . This allows us to display how the nonlinear preconditioner approximates the identity.

2.5.1. Optimal grids. The optimal grids have been introduced in [8, 9, 18] to obtain exponential convergence of approximations of the Dirichlet-to-Neumann map. They were used and analyzed in the context of inverse spectral problems in [5] and in the inverse problem of electrical impedance tomography in [4, 6]. The grids are defined by the coefficients of the continued fraction representation of the rational response function corresponding to a reduced model of a medium with constant resistivity $r^{(0)}$. In principle we can define grids for other reference resistivities, not necessarily constant, but the results in [4–6] show that the grids change very little with respect to $r(x)$. This is why they are very useful in inversion.

Let then $r^{(0)}(x) \equiv 1$ so that the change of coordinates in (2.25) is trivial ($\xi = x$) and obtain from (2.18)–(2.19) that $\{(\kappa_j^{(0)}, \hat{\kappa}_j^{(0)})\}_{j=1}^m$ are the optimal grid steps in a finite difference scheme for the equation

$$\frac{\partial^2 w}{\partial x^2} - sw = 0.$$

The primary and dual grid points are

$$x_j^{(0)} = \sum_{k=1}^j \kappa_k^{(0)}, \quad \hat{x}_j^{(0)} = \sum_{k=1}^j \hat{\kappa}_k^{(0)}, \quad j = 1, \dots, m, \quad (2.26)$$

with boundary nodes $x_0^{(0)} = \hat{x}_0^{(0)} = 0$. In the numerical experiments we observe that the grid is staggered, i.e. the *primary* nodes $x_j^{(0)}$ and the *dual* nodes $\hat{x}_j^{(0)}$ obey the interlacing conditions

$$0 = \hat{x}_0^{(0)} = x_0^{(0)} < \hat{x}_1^{(0)} < x_1^{(0)} < \hat{x}_2^{(0)} < x_2^{(0)} < \dots < x_{m-1}^{(0)} < \hat{x}_m^{(0)} < x_m^{(0)} \leq 1. \quad (2.27)$$

We do not prove (2.27) here, although it is possible to do so at least in some settings.

The optimal grids are closely connected with the *sensitivity functions*, which for the semi-discrete problem are given by the rows of the Jacobian $DR \in \mathbb{R}^{2m \times N}$ defined as

$$(DR)_{j,k} = \begin{cases} \frac{\partial \log \kappa_j}{\partial r_k}, & \text{if } 1 \leq j \leq m, \\ \frac{\partial \log \hat{\kappa}_j}{\partial r_k}, & \text{if } m+1 \leq j \leq 2m, \end{cases} \quad k = 1, \dots, N.$$

The studies in [4, 6] show that the sensitivity function corresponding to κ_j is localized around the corresponding grid cell $(\hat{x}_j^{(0)}, \hat{x}_{j+1}^{(0)})$, and its maximum is near $x_j^{(0)}$. The same holds for $\hat{\kappa}_j$, after interchanging the primary and dual grid nodes. Moreover, the columns of the pseudoinverse $(DR)^\dagger$ have similar localization behavior. The Gauss–Newton update is the linear combination of the columns of $(DR)^\dagger$, and therefore optimal grids are useful for inversion. They localize well features of the resistivity that are recoverable from the measurements.

A good grid should have two properties. First, it should be refined near the point of measurement to capture correctly the loss of resolution away from \mathcal{B}_A . Second, the nodes should not all be clustered near \mathcal{B}_A because when the nodes get too close, the corresponding rows of DR become almost linearly dependent, and the Jacobian is ill-conditioned.

2.5.2. Matching conditions. We study here the grids for three choices of matching conditions (2.13). The first corresponds to $l = 1$, $s_1 = 0$ and $M_1 = m$ (simple Padé) and yields the rational Krylov subspace

$$\mathcal{K}_m(0) = \text{span} \{ A^{-1}\mathbf{b}, A^{-2}\mathbf{b}, \dots, A^{-m}\mathbf{b} \}.$$

This approximant has the best accuracy near the interpolation point ($s = 0$), and is obviously inferior for global approximation in s when compared to multipoint Padé approximants. The other two choices match $Y(s; \mathbf{r})$ and its first derivative at nodes $\tilde{\mathbf{s}} = (\tilde{s}_j)_{j=1}^m$ that are distributed geometrically

$$\tilde{s}_j = \tilde{s}_1 \left(\frac{\tilde{s}_2}{\tilde{s}_1} \right)^{j-1}, \quad j = 1, \dots, m. \quad (2.28)$$

We use henceforth the tilde to distinguish these nodes from those obtained with the change of variables

$$s_j = \frac{\tilde{s}_j}{\tilde{s}_m} \in (0, 1], \quad (2.29)$$

intended to improve the conditioning of the interpolation. The matching conditions at $\tilde{\mathbf{s}}$ yield the rational Krylov projection subspace

$$\mathcal{K}_m(\tilde{\mathbf{s}}) = \text{span} \left\{ (\tilde{s}_1 I - A)^{-1}\mathbf{b}, \dots, (\tilde{s}_m I - A)^{-1}\mathbf{b} \right\},$$

and the two choices of interpolants differ in the rate of growth of $\tilde{\mathbf{s}}$. The second interpolant uses the rate of growth $\tilde{s}_2/\tilde{s}_1 = 1 + 12/m$ in (2.28) and $\tilde{s}_1 = 2$, so that

$$\tilde{s}_j = 2 \left(1 + \frac{12}{m} \right)^{j-1}, \quad j = 1, \dots, m. \quad (2.30)$$

This choice approximates the Zolotarev nodes [18] which arise in the optimal rational approximation of the transfer function $Y(s; \mathbf{r})$ over a real positive and bounded interval of s . The third interpolant uses a faster rate of growth of $\tilde{\mathbf{s}}$ and gives worse results, as illustrated in the numerical experiment below.

We show the optimal grids for all three choices of matching conditions in figure 2 for reduced models of sizes $m = 5, 10$. We observe that the nodes of the grids corresponding to fast growing $\tilde{\mathbf{s}}$ are clustered too close to the measurement point $x = 0$. Thus, inversion results are expected to have poor resolution throughout the rest of the domain away from the origin. In addition, the clustering of the grid nodes leads to poor conditioning of the Jacobian DR . This is illustrated in figure 3, where the condition numbers are plotted against the size m of the reduced model. We observe that the condition number of the Jacobian for the reduced model with fast growing $\tilde{\mathbf{s}}$ increases exponentially. The condition numbers of the Jacobians for the other two choices of matching conditions grow very slowly.

We can explain why the case of fast growing $\tilde{\mathbf{s}}$ is undesirable in inversion by looking at the limiting case of approximation at infinity. The simple Padé approximant at infinity corresponds to the Krylov subspace

$$\mathcal{K}_m(+\infty) = \text{span} \{ \mathbf{b}, A\mathbf{b}, \dots, A^{m-1}\mathbf{b} \},$$

which is unsuitable for inversion in our setting. To see this, recall from (2.2) that $A(\mathbf{r})$ is tridiagonal and \mathbf{b} is a scalar multiple \mathbf{e}_1 . Thus, for any $j \in \mathbb{Z}_+$ only the first $j + 1$ components of the vector $A^j\mathbf{b}$ are non-zero. When $A(\mathbf{r})$ is projected on $\mathcal{K}_m(+\infty)$ in (2.12), the reduced model matrix A_m is aware only of the upper left $(m + 1) \times (m + 1)$ block of $A(\mathbf{r})$, which depends only on the first $m + 1$ entries of \mathbf{r} . This is unsuitable for inversion where we want A_m to capture the behavior of the resistivity in the whole interval, even for small m . The

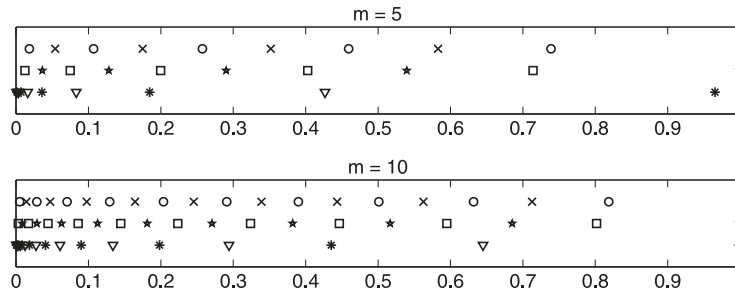


Figure 2. Primary $x_j^{(0)}$ and dual $\hat{x}_j^{(0)}$ optimal grid nodes for $j = 1, \dots, m$ ($m = 5, 10$) and different choices of matching conditions. Moment matching at $s = 0$: primary \times , dual \circ . Interpolation at geometrically distributed interpolation nodes: primary \star and dual \square for slowly growing \tilde{s} ; primary \star and dual ∇ for fast growing \tilde{s} . The number of fine grid steps in the semi-discretized model is $N = 1999$.

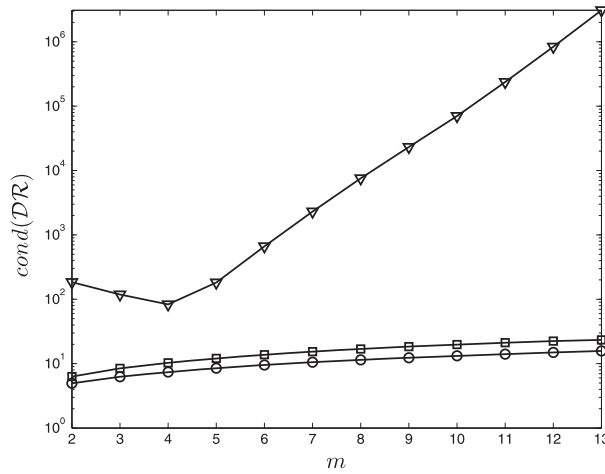


Figure 3. Dependence of the condition number of DR on the size m of the reduced model for different matching conditions. Moment matching at $s = 0$ is \circ , interpolation at geometrically distributed nodes: slowly growing \tilde{s} is \square , fast growing \tilde{s} is ∇ . The number of fine grid steps in the semi-discretized model is $N = 1999$.

corresponding optimal grid steps will simply coincide with the first m grid steps h of the fine grid discretization in (2.2).

When the interpolation nodes grow too quickly, we are near the limiting case of simple Padé approximant at infinity, and the first optimal grid steps are $\hat{x}_1^{(0)} \approx x_1^{(0)} \approx h$.

Consequently, the rows $\left(\frac{\partial \kappa_1}{\partial r_k}\right)_{k=1}^N$ and $\left(\frac{\partial \hat{\kappa}_1}{\partial r_k}\right)_{k=1}^N$ of DR are almost collinear and the Jacobian is poorly conditioned, as shown in figure 3.

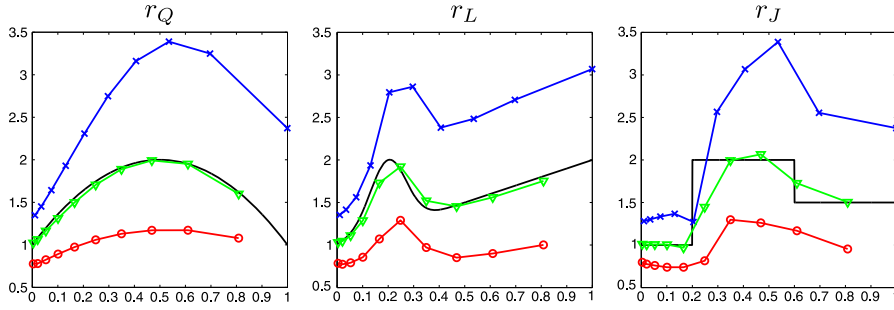


Figure 4. Action of the nonlinear preconditioner \mathcal{R} on the resistivities r_Q , r_L and r_J (solid black line) defined in (4.7) and (4.9). The ‘primary’ ratios $(x_j^{(0)}, \zeta_j)$ are blue \times , the ‘dual’ ratios $(\hat{x}_j^{(0)}, \hat{\zeta}_j)$ are red \circ for $j = 1, \dots, m$ with $m = 10$. The geometric averages $(\tilde{x}_j^{(0)}, \tilde{\zeta}_j)$ are green ∇ .

2.5.3. Action of the nonlinear preconditioner on the resistivity. The optimal grids can be used to obtain resistivity reconstructions directly, without optimization, as was done in [5] for the inverse spectral problem and in [4, 6] for electrical impedance tomography. We do not use this approach, but we show here such reconstructions to display the behavior of the nonlinear preconditioner \mathcal{R} when acting on \mathbf{r} .

Recall from equation (2.25) and the explanation in section 2.4 that $\hat{\kappa}_j$ and κ_j are proportional to the values of $r^{1/2}(x)$ and $r^{-1/2}(x)$ around the corresponding optimal grid points. If we take as the proportionality coefficients the values $\hat{\kappa}_j^{(0)}$ and $\kappa_j^{(0)}$ then we expect the ratios

$$\zeta_j = \left(\frac{\kappa_j^{(0)}}{\kappa_j} \right)^2, \quad \hat{\zeta}_j = \left(\frac{\hat{\kappa}_j}{\hat{\kappa}_j^{(0)}} \right)^2, \quad j = 1, \dots, m, \quad (2.31)$$

to behave roughly as $r(x_j^{(0)})$ and $r(\hat{x}_j^{(0)})$. In practice, a more accurate estimate of the resistivity can be obtained by taking the geometric average of ζ_j and $\hat{\zeta}_j$

$$\tilde{\zeta}_j = \sqrt{\zeta_j \hat{\zeta}_j} = \frac{\kappa_j^{(0)} \hat{\kappa}_j}{\kappa_j \hat{\kappa}_j^{(0)}}, \quad j = 1, \dots, m. \quad (2.32)$$

Since building a direct inversion algorithm is not our focus, we only show (2.32) for comparison purposes.

In figure 4 we display the ratios (2.31) plotted at the nodes of the optimal grid $(x_j^{(0)}, \zeta_j)$, $(\hat{x}_j^{(0)}, \hat{\zeta}_j)$, $j = 1, \dots, m$. We take the same resistivities r_Q , r_L and r_J that are used in the numerical experiments in section 4. They are defined in (4.8) and (4.9). We observe that the curve defined by the linear interpolation of the ‘primary’ points $(x_j^{(0)}, \zeta_j)$ overestimates the true resistivity, while the ‘dual’ curve passing through $(\hat{x}_j^{(0)}, \hat{\zeta}_j)$ underestimates it. Both curves capture the shape of the resistivity quite well, so when taking the geometric average (2.32) the reconstruction falls right on top of the true resistivity. This confirms that \mathcal{R} resolves most of the nonlinearity of the problem and thus acts on the resistivity as an approximate identity.

We can also illustrate how well \mathcal{R} resolves the nonlinearity of the problem by considering an example of high contrast resistivity. In figure 5 we plot the same quantities as

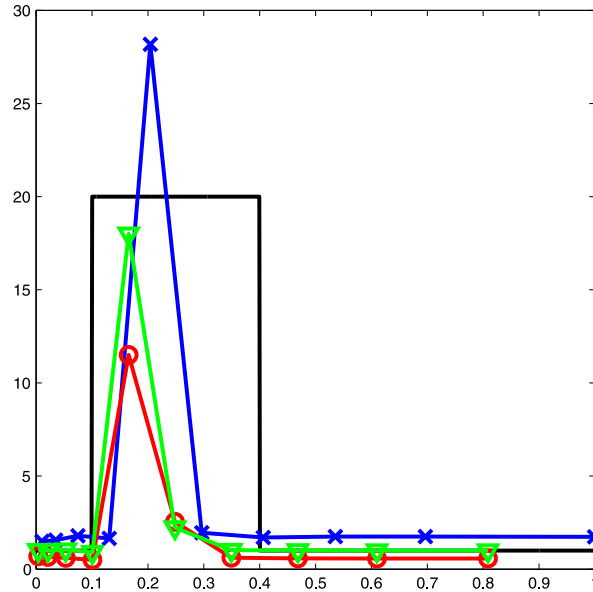


Figure 5. Action of the nonlinear preconditioner \mathcal{R} on a piecewise constant resistivity of contrast 20. Same setting as in figure 4.

in figure 4 in the case of piecewise constant resistivity of contrast 20. The contrast is captured quite well by $\tilde{\zeta}_j$, while the shape of the inclusion is shrunk. This is one of the reasons why we use \mathcal{R} as a preconditioner for optimization and not a reconstruction mapping. Optimization allows us to recover resistivity features on scales that are smaller than those captured in ζ_j and $\hat{\zeta}_j$. Moreover, since \mathcal{R} resolves most of the nonlinearity of the inverse problem, the optimization avoids the pitfalls of traditional data misfit minimization approaches, such as sensitivity to the initial guess, numerous local minima and slow convergence.

2.6. Data fitting via the rational interpolation

Unlike $\mathcal{R}(\mathbf{r}) = \mathcal{Q}(y(\cdot; \mathbf{r}))$ computed using the chain of mappings (2.23) with all stable steps, the computation of the data fitting term $\mathcal{Q}(d(\cdot))$ requires solving an oscillatory rational interpolation problem (2.13) in step (b) of (2.21) to obtain the rational interpolant $Y_m(s)$ of the transfer function $Y(s)$. This involves the solution of a linear system of equations with an ill-conditioned matrix, or computing the singular value decomposition of such matrix. We use the condition number of the matrix to assess the instability of the problem. The condition number grows exponentially with m , but the rate of growth depends on the matching conditions used in the rational interpolation. We show this for the two choices of matching conditions: interpolation of $Y(s)$ and its first derivatives at distinct nodes \mathfrak{s} distributed as in (2.30) (multipoint Padé), and matching of moments of $Y(s)$ at $s = 0$ (simple Padé). We describe first both Padé interpolation schemes and then compare their stability numerically.

Multipoint Padé: Let us rewrite the reduced order transfer function (2.11) in the form

$$Y_m(s) = \frac{f(s)}{g(s)} = \frac{f_0 + f_1 s + \dots + f_{m-1} s^{m-1}}{g_0 + g_1 s + \dots + g_m s^m}, \quad (2.33)$$

where $f(s)$ and $g(s)$ are polynomials defined up to a common factor. We use this redundancy later to choose a unique solution of an underdetermined problem. The matching conditions (2.13) of $Y(s)$ and $Y'(s)$ at the distinct interpolation nodes $0 < \tilde{s}_1 < \tilde{s}_2 < \dots < \tilde{s}_m$ are

$$\begin{cases} f(\tilde{s}_j) - Y_m(\tilde{s}_j)g(\tilde{s}_j) = 0, \\ f'(\tilde{s}_j) - Y_m'(\tilde{s}_j)g(\tilde{s}_j) - Y_m(\tilde{s}_j)g'(\tilde{s}_j) = 0, \end{cases} \quad j = 1, \dots, m. \quad (2.34)$$

Next, we recall the change of variables (2.29) and define the Vandermonde-like $m \times (m + 1)$ matrices

$$S = \begin{bmatrix} 1 & s_1 & s_1^2 & \dots & s_1^m \\ 1 & s_2 & s_2^2 & \dots & s_2^m \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & s_m & s_m^2 & \dots & s_m^m \end{bmatrix}, \quad S' = \frac{1}{\tilde{s}_m} \begin{bmatrix} 0 & 1 & 2s_1 & \dots & ms_1^{m-1} \\ 0 & 1 & 2s_2 & \dots & ms_2^{m-1} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 1 & 2s_m & \dots & ms_m^{m-1} \end{bmatrix}, \quad (2.35)$$

and the diagonal matrices

$$\mathcal{Y} = \text{diag}(Y(\tilde{s}_1), \dots, Y(\tilde{s}_m)), \quad \mathcal{Y}' = \text{diag}(Y'(\tilde{s}_1), \dots, Y'(\tilde{s}_m)). \quad (2.36)$$

This allows us to write equations (2.34) in matrix-vector form as an underdetermined problem

$$\mathcal{P}\mathbf{u} = 0, \quad \mathbf{u} \in \mathbb{R}^{2m+1}, \quad (2.37)$$

with

$$\mathcal{P} = \begin{bmatrix} S_{1:m, 1:m} & -\mathcal{Y}S \\ S'_{1:m, 1:m} & -\mathcal{Y}'S - \mathcal{Y}S' \end{bmatrix} \in \mathbb{R}^{2m \times (2m+1)}, \quad (2.38)$$

and

$$\begin{aligned} f_j &= \tilde{s}_m^{-j} u_{j+1}, \quad j = 0, \dots, m-1, \\ g_j &= \tilde{s}_m^{-j} u_{j+m+1}, \quad j = 0, \dots, m. \end{aligned} \quad (2.39)$$

The problem is underdetermined because of the redundancy in (2.33). We eliminate it by the additional condition $\|\mathbf{u}\|_2 = 1$, which makes it possible to solve (2.37) via the singular value decomposition. If we let U be the matrix of right singular vectors of \mathcal{P} , then

$$\mathbf{u} = U_{1:(2m+1), 2m+1}. \quad (2.40)$$

Once the polynomials f and g are determined from (2.39), we can compute the partial fraction expansion (2.14). The poles $-\theta_j$ are the roots of $g(s)$, and the residues c_j are given by

$$c_j = \frac{f(-\theta_j)}{g_m \prod_{\substack{k=1 \\ k \neq j}}^m (\theta_k - \theta_j)}, \quad j = 1, \dots, m,$$

assuming that θ_j are distinct. Finally, κ_j and $\hat{\kappa}_j$ are obtained from θ_j and c_j via a Lanczos iteration, as explained in the appendix.

Table 1. Condition numbers of \mathcal{P} (multipoint Padé approximant) and \mathcal{T} (simple Padé approximant). The number of fine grid steps in the semi-discretized model is $N = 299$.

m	2	3	4	5	6
$\text{cond}(\mathcal{P})$	$4.43 \cdot 10^2$	$6.73 \cdot 10^4$	$1.85 \cdot 10^7$	$6.95 \cdot 10^9$	$3.83 \cdot 10^{12}$
$\text{cond}(\mathcal{T})$	$5.28 \cdot 10^1$	$1.26 \cdot 10^5$	$1.84 \cdot 10^9$	$9.14 \cdot 10^{13}$	$2.86 \cdot 10^{16}$

Simple Padé: When $l = 1$, $s_1 = 0$ and $M_1 = m$, we have a simple Padé approximant which matches the first $2m$ moments of $y(t; \mathbf{r})$ in the time domain, because

$$\left. \frac{\partial^j Y_m}{\partial s^j} \right|_{s=0} = \left. \frac{\partial^j Y}{\partial s^j} \right|_{s=0} = (-1)^j \int_0^{+\infty} y(t; \mathbf{r}) t^j dt, \quad j = 0, 1, \dots, 2m - 1.$$

A robust algorithm for simple Padé approximation is proposed in [15]. It is also based on the singular value decomposition. If $Y(s)$ has the Taylor expansion at $s = 0$

$$Y(s) = \tau_0 + \tau_1 s + \tau_2 s^2 + \dots + \tau_{2m-1} s^{2m-1} + \dots, \quad (2.41)$$

then the algorithm in [15] performs a singular value decomposition of the Toeplitz matrix

$$\mathcal{T} = \begin{bmatrix} \tau_m & \tau_{m-1} & \dots & \tau_1 & \tau_0 \\ \tau_{m+1} & \tau_m & \dots & \tau_2 & \tau_1 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ \tau_{2m-1} & \tau_{2m-2} & \dots & \tau_m & \tau_{m-1} \end{bmatrix} \in \mathbb{R}^{m \times (m+1)}. \quad (2.42)$$

If $U \in \mathbb{R}^{(m+1) \times (m+1)}$ is the matrix of right singular vectors of \mathcal{T} , then the coefficients in (2.33) satisfy

$$\begin{pmatrix} g_0 \\ g_1 \\ \vdots \\ g_m \end{pmatrix} = U_{1:(m+1), m+1}, \quad (2.43)$$

and

$$\begin{pmatrix} f_0 \\ f_1 \\ \vdots \\ f_{m-1} \end{pmatrix} = \begin{bmatrix} \tau_0 & 0 & \dots & 0 & 0 \\ \tau_1 & \tau_0 & \dots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ \tau_{m-1} & \tau_{m-2} & \dots & \tau_0 & 0 \end{bmatrix} \begin{pmatrix} g_0 \\ g_1 \\ \vdots \\ g_m \end{pmatrix}. \quad (2.44)$$

We refer the reader to [15] for detailed explanations.

Comparison: To compare the performance of the two interpolation procedures, we give in table 1 the condition numbers of matrices \mathcal{P} and \mathcal{T} . They are computed for the reference resistivity $\mathbf{r}^{(0)}$, for several values of the size m of the reduced model. We observe that while both condition numbers grow exponentially, the growth rate is slower for the multipoint Padé approximant. Thus, we conclude that it is the best of the choices of matching conditions considered in this section. It allows a more stable computation of $Q(d(\cdot))$, it gives a good distribution of the optimal grid points and a well-conditioned Jacobian DR .

2.7. Regularization of Gauss–Newton iteration

Our method of solving the optimization problem (2.24) uses a Gauss–Newton iteration with regularization similar to that in [4]. We outline it below, and refer to the next section for the precise formulation of the inversion algorithm.

Recall that \mathcal{R} maps the vectors $\mathbf{r} \in \mathbb{R}_+^N$ of resistivity values on the fine grid to $2m$ reduced order model parameters $\{(\log \kappa_j, \log \hat{\kappa}_j)\}_{j=1}^m$. Thus, the Jacobian $D\mathcal{R}$ has dimensions $\mathbb{R}^{2m \times N}$. Since the reduced order model is much coarser than the fine grid discretization $2m \ll N$, the Jacobian $D\mathcal{R}(\mathbf{r})$ has a large null space. At each iteration the Gauss–Newton update to \mathbf{r} is in the $2m$ dimensional range of the pseudoinverse $D\mathcal{R}^\dagger(\mathbf{r})$ of the Jacobian. This leads to low resolution in practice, because m is kept small to mitigate the sensitivity of the inverse problem to noise. If we have prior information about the unknown \mathbf{r}^{true} , we can use it to improve its estimate \mathbf{r}^* .

We incorporate the prior information about the true resistivity into a penalty functional $\mathcal{L}(\mathbf{r})$. For example, $\mathcal{L}(\mathbf{r})$ may be the total variation norm of \mathbf{r} if \mathbf{r}^{true} is known to be piecewise constant, or the square of the ℓ_2 norm of \mathbf{r} or of its derivative if \mathbf{r}^{true} is expected to be smooth.

In our inversion method we separate the minimization of the norm of the residual $\mathcal{Q}(d(\cdot)) - \mathcal{R}(\mathbf{r})$ and the penalty functional $\mathcal{L}(\mathbf{r})$. At each iteration we compute the standard Gauss–Newton solution \mathbf{r} and then we add a correction to obtain a regularized iterate $\boldsymbol{\rho}$. The correction $\boldsymbol{\rho} - \mathbf{r}$ is in the null space of $D\mathcal{R}$, so that the residual remains unchanged. We define it as the minimizer of the constrained optimization problem

$$\begin{aligned} & \text{minimize } \mathcal{L}(\boldsymbol{\rho}), \\ & \text{s.t. } [D\mathcal{R}](\mathbf{r} - \boldsymbol{\rho}) = 0 \end{aligned} \quad (2.45)$$

which we can compute explicitly in the case of a weighted discrete H^1 seminorm regularization, assumed henceforth,

$$\mathcal{L}(\mathbf{r}) = \frac{1}{2} \|W^{1/2} \widetilde{D} \mathbf{r}\|_2^2. \quad (2.46)$$

Here the matrix \widetilde{D} is a truncation of D defined as

$$\widetilde{D} = D_{1:(N-1), 1:N} \in \mathbb{R}^{(N-1) \times N},$$

and $W \in \mathbb{R}^{(N-1) \times (N-1)}$ is a diagonal matrix of weights. We specify it below depending on the prior information on the true resistivity.

With the choice of penalty in the form (2.46) the optimization problem (2.45) is quadratic with linear constraints, and thus $\boldsymbol{\rho}$ can be calculated from the first order optimality conditions given by the linear system

$$\widetilde{D}^T W \widetilde{D} \boldsymbol{\rho} + [D\mathcal{R}]^T \boldsymbol{\lambda} = 0, \quad (2.47)$$

$$[D\mathcal{R}]\boldsymbol{\rho} = [D\mathcal{R}]\mathbf{r}, \quad (2.48)$$

where $\boldsymbol{\lambda} \in \mathbb{R}^{2m}$ is a vector of Lagrange multipliers.

In the numerical results presented in section 4 we consider smooth and piecewise constant resistivities, and choose W in (2.46) as follows. For smooth resistivities we simply take $W = I$, so that (2.46) is a regular discrete H^1 seminorm. For discontinuous resistivities we would like to minimize the total variation of the resistivity. This does not allow an explicit computation of $\boldsymbol{\rho}$, so we make a compromise and use the weights introduced in [1]. The matrix W is diagonal with entries

$$w_j = \left(\left([\bar{D} \mathbf{r}]_j \right)^2 + \phi(\mathbf{r})^2 \right)^{-1}, \quad j = 1, \dots, N-1, \quad (2.49)$$

where $\phi(\mathbf{r})$ is proportional to the misfit for the current iterate

$$\phi(\mathbf{r}) = C_\phi \|Q(d(\cdot)) - \mathcal{R}(\mathbf{r})\|_2, \quad (2.50)$$

and C_ϕ is some constant, set to $1/(2m^2)$ in the numerical examples in section 4.

To ensure that $A(\mathbf{r})$ corresponds to a discretization of an elliptic operator, we need positive entries in \mathbf{r} . This can be done with a logarithmic change of coordinates, which transforms the optimization problem to an unconstrained one. However, in our numerical experiments we observed that if m is sufficiently small so that for the given data $d(t)$ all the entries of $Q(d(\cdot)) \in \mathbb{R}^{2m}$ are positive, then the Gauss–Newton updates of \mathbf{r} remain positive as well. Thus, the logarithmic change of coordinates for \mathbf{r} is not needed in our computations.

2.8. The inversion algorithm for 1D media

Here we present the summary of the inversion algorithm. The details of the computation of $\mathcal{R}(\mathbf{r})$ and its Jacobian DR can be found in the appendix.

The inputs of the inversion algorithm are the measured data $d(t)$ and a guess value of m . This m reflects the expected resolution of the reconstruction, and may need to be decreased depending on the noise level. To compute the estimate \mathbf{r}^* of \mathbf{r}^{true} , perform the following steps:

- (1) Define the interpolation nodes \tilde{s} via (2.30). Using the multipoint Padé scheme from section 2.6 compute $(\kappa_j^*, \hat{\kappa}_j^*)_{j=1}^m$ using the data $d(t)$.
- (2) If for some j either $\kappa_j^* \leq 0$ or $\hat{\kappa}_j^* \leq 0$, decrease m to $m-1$ and return to step 1. If all $(\kappa_j^*, \hat{\kappa}_j^*)_{j=1}^m$ are positive, fix m and continue to step 3.
- (3) Define the vector of logarithms $\mathbf{l}^* = (\log \kappa_1^*, \dots, \log \kappa_m^*, \log \hat{\kappa}_1^*, \dots, \log \hat{\kappa}_m^*)^T$.
- (4) Choose an initial guess $\mathbf{r}^{(1)} \in \mathcal{R}_+^N$ and the maximum number n_{GN} of Gauss–Newton iterations.
- (5) For $p = 1, \dots, n_{\text{GN}}$ perform:
 - (5.1) For the current iterate $\mathbf{r}^{(p)}$ compute the mapping

$$\mathcal{R}(\mathbf{r}^{(p)}) = \left\{ \left(\log \kappa_j^{(p)}, \log \hat{\kappa}_j^{(p)} \right) \right\}_{j=1}^m$$

and its Jacobian

$$DR^{(p)} = DR(\mathbf{r}^{(p)})$$

as explained in the appendix.

- (5.2) Define the vector of logarithms

$$\mathbf{l}^{(p)} = \left(\log \kappa_1^{(p)}, \dots, \log \kappa_m^{(p)}, \log \hat{\kappa}_1^{(p)}, \dots, \log \hat{\kappa}_m^{(p)} \right)^T.$$

- (5.3) Compute the step

$$\boldsymbol{\rho}^{(p)} = -\left(DR^{(p)} \right)^\dagger \left(\mathbf{l}^{(p)} - \mathbf{l}^* \right).$$

(5.4) Choose the step length $\alpha^{(p)}$ and compute the Gauss–Newton update

$$\mathbf{r}^{\text{GN}} = \mathbf{r}^{(p)} + \zeta^{(p)} \boldsymbol{\rho}^{(p)}.$$

(5.5) Compute the weight W using (2.49) with \mathbf{r}^{GN} or $W = I$.

(5.6) Solve for the next iterate $\mathbf{r}^{(p+1)}$ from the linear system

$$\begin{bmatrix} \widetilde{D}^T W \widetilde{D} & (D\mathcal{R}^{(p)})^T \\ D\mathcal{R}^{(p)} & 0 \end{bmatrix} \begin{bmatrix} \mathbf{r}^{(p+1)} \\ \lambda^{(p)} \end{bmatrix} = \begin{bmatrix} 0 \\ (D\mathcal{R}^{(p)}) \mathbf{r}^{\text{GN}} \end{bmatrix}. \quad (2.51)$$

(6) The estimate is

$$\mathbf{r}^* = \mathbf{r}^{(n_{\text{GN}}+1)}.$$

Let us remark that since the nonlinear preconditioner is an approximation of the identity in the sense explained in section 2.4, most of the nonlinearity of the problem is resolved by the rational interpolation in step 1. Thus, we may start with a poor initial guess in step 4 and still obtain good reconstructions. Moreover, the number n_{GN} of Gauss–Newton iterations may be kept small. In the numerical results presented in section 4 we take the number of iterations $n_{\text{GN}} = 5$ for medium contrast resistivities and $n_{\text{GN}} = 10$ for the high contrast case. In general, any of the standard stopping criteria could be used, such as the stagnation of the residual.

In order to simplify our implementation we set the step length $\alpha^{(p)} = 1$ in step 5.4. However, choosing $\alpha^{(p)}$ adaptively with a line search procedure may be beneficial, especially for high contrast resistivities.

While the Jacobian $D\mathcal{R}^{(p)}$ is well-conditioned, the system (2.51) may not be. To alleviate this problem, instead of solving (2.51) directly, we may use a truncated singular value decomposition to obtain a regularized solution. Typically it is enough to discard just one component corresponding to the smallest singular value as we do in the numerical experiments.

3. Two dimensional inversion

Unlike the 1D case, the inverse problem in two dimensions is formally overdetermined. The unknown is the resistivity $r(x)$ defined on $\Omega \subset \mathbb{R}^2$, and the data are three dimensional. One dimension corresponds to time and the other two come from the source (initial condition) and receiver locations on \mathcal{B}_A . The model reduction inversion framework described in section 2.8 applies to a formally determined problem. We extend it to two dimensions by constructing separately reduced models for certain data subsets. Each model defines a mapping \mathcal{R}_j that is similar to \mathcal{R} in one dimension, where $j = 1, \dots, N_d$ is the index of the data set. The maps \mathcal{R}_j are coupled by their dependence on the resistivity function $r(x)$, and are all taken into account in inversion as explained below.

Let $u_o^{(j)}(x)$ be the initial condition for a source that is compactly supported on a segment (interval) \mathcal{J}_j of \mathcal{B}_A . We model it for simplicity with the indicator function $\mathbb{1}_j$ of \mathcal{J}_j and write

$$u_o^{(j)}(x) = \mathbb{1}_j(x^{\parallel}) \delta(x^{\perp}). \quad (3.1)$$

Here $x = (x^{\parallel}, x^{\perp})$, with x^{\parallel} the arclength on \mathcal{B}_A and x^{\perp} the normal coordinate to the boundary, which we suppose is smooth. The semidiscretized version of equation (1.3) on a grid with N

points is

$$\frac{\partial \mathbf{u}^{(j)}(t)}{\partial t} = A(\mathbf{r})\mathbf{u}^{(j)}(t), \quad (3.2)$$

with $\mathbf{r} \in \mathbb{R}^N$ the vector of discrete samples of the resistivity. Equation (3.2) models a dynamical system with response matrix $y_{kj}(t, \mathbf{r})$ defined by the restriction of the solution

$$\mathbf{u}^{(j)}(t) = e^{A(\mathbf{r})t}\mathbf{u}_0^{(j)} \quad (3.3)$$

to the support \mathcal{J}_k of the k th receiver. We take for simplicity the same model of the sources and receivers, with support on the disjoint boundary segments \mathcal{J}_j of the accessible boundary. Thus, if we let $\mathbf{b}^{(j)} \in \mathbb{R}^N$ be the measurement vector corresponding to the j th source or receiver, we can write the time domain response as

$$y_{kj}(t; \mathbf{r}) = \mathbf{b}^{(k)T} e^{A(\mathbf{r})t} \mathbf{b}^{(j)}. \quad (3.4)$$

The diagonal of this matrix is the high dimensional extension of (2.8). The matrix valued transfer function is the Laplace transform of (3.4),

$$Y_{kj}(s; \mathbf{r}) = \int_0^{+\infty} y_{kj}(t; \mathbf{r}) e^{-st} dt = \mathbf{b}^{(k)T} (sI - A(\mathbf{r}))^{-1} \mathbf{b}^{(j)}, \quad s > 0. \quad (3.5)$$

In model order reduction we obtain a reduced model with rational transfer function $Y_{kj,m}(s; \mathbf{r})$ that approximates (3.5). The reduced model is constructed separately for each receiver–source pair (k, j) . It is defined by an $m \times m$ symmetric and negative definite matrix $A_m^{(k,j)}(\mathbf{r})$ with $m \ll N$, and measurement vectors $\mathbf{b}_m^{(k)}$ and $\mathbf{b}_m^{(j)}$. The transfer function

$$Y_{kj,m}(s; \mathbf{r}) = \mathbf{b}_m^{(k)T} (sI_m - A_m^{(k,j)})^{-1} \mathbf{b}_m^{(j)} = \sum_{l=1}^m \frac{c_l^{(k,j)}}{s + \theta_l^{(k,j)}}, \quad (3.6)$$

has poles at the eigenvalues $-\theta_l^{(k,j)}$ of $A_m^{(k,j)}$, for eigenvectors $\mathbf{z}_l^{(k,j)}$, and residues

$$c_l^{(k,j)} = \left(\mathbf{b}_m^{(k)T} \mathbf{z}_l^{(k,j)} \right) \left(\mathbf{b}_m^{(j)T} \mathbf{z}_l^{(k,j)} \right). \quad (3.7)$$

We are interested in the continued fraction representation of $Y_{kj,m}(s; \mathbf{r})$, in particular its coefficients $\left\{ \left(\hat{\kappa}_l^{(k,j)}, \kappa_l^{(k,j)} \right) \right\}_{l=1}^m$ that define the preconditioner mapping in our inversion approach. These coefficients are guaranteed to be positive as long as the residues in (3.7) satisfy $c_l^{(k,j)} \geq 0$. This is guaranteed to hold for the diagonal of (3.6), because

$$c_l^{(j,j)} = \left(\mathbf{b}_m^{(j)T} \mathbf{z}_l^{(j,j)} \right)^2. \quad (3.8)$$

Thus, we construct the reduced models for the diagonal of the matrix valued transfer function, and compute the model parameters $\left\{ \left(\hat{\kappa}_l^{(j,j)}, \kappa_l^{(j,j)} \right) \right\}_{l=1}^m$ as in one dimension. Such measurement setting has analogues in other types of inverse problems. For example, a similar setting in wave inversion is the backscattering problem, where the scattered wave field is measured in the same direction as the incoming wave.

If we have N_d boundary segments \mathcal{J}_j at \mathcal{B}_A , we define

$$\mathcal{R}_j(\mathbf{r}) = \mathcal{Q}\left(y_{jj}(\cdot; \mathbf{r})\right) = \left(\log \hat{\kappa}_l^{(j,j)}, \log \kappa_l^{(j,j)} \right)_{l=1}^m, \quad j = 1, \dots, N_d, \quad (3.9)$$

as in one dimension, using the chain of mappings (2.23). The resistivity is estimated by the solution of the optimization problem

$$\mathbf{r}^* = \arg \min_{\mathbf{r} \in \mathbb{R}_+^N} \frac{1}{2} \sum_{j=1}^{N_d} \| \mathcal{Q}(d_j(\cdot)) - \mathcal{R}_j(\mathbf{r}) \|_2^2, \quad (3.10)$$

where $d_j(t)$ is the data measured at the receiver supported on \mathcal{J}_j , for the j th source excitation. Note that the sum in (3.10) couples all the sources/receivers together in a single objective functional.

Another question that we need to address is the choice of matching conditions. For simplicity we match the moments of $Y_{jj}(s)$ at a single interpolation node \tilde{s} . This yields the matching conditions

$$\left. \frac{\partial^k Y_{jj,m}}{\partial s^k} \right|_{s=\tilde{s}} = \left. \frac{\partial^k Y_{jj}}{\partial s^k} \right|_{s=\tilde{s}}, \quad k = 0, 1, \dots, 2m - 1, \quad (3.11)$$

and rational Krylov subspaces

$$\mathcal{K}_m^{(j)}(\tilde{s}) = \text{span} \left\{ (\tilde{s}I - A)^{-1} \mathbf{b}^{(j)}, \dots, (\tilde{s}I - A)^{-m} \mathbf{b}^{(j)} \right\}, \quad j = 1, \dots, N_d. \quad (3.12)$$

Then the only parameter to determine is the node $\tilde{s} > 0$.

Similarly to the 1D case we use the condition number of the Jacobian \mathcal{DR} to determine the optimal choice of \tilde{s} . Here the total Jacobian $\mathcal{DR} \in \mathbb{R}^{(2mN_d) \times N}$ is a matrix of N_d individual Jacobians $\mathcal{DR}_j \in \mathbb{R}^{2m \times N}$ stacked together. It appears that for a fixed j the sensitivity functions for $\log \kappa_l^{(j,j)}$ and $\log \hat{\kappa}_l^{(j,j)}$ closely resemble the spherical waves propagating from the middle of \mathcal{J}_j . As the index l increases, the wave propagates further away from \mathcal{J}_j . This behavior is illustrated quantitatively in section 4.2 for an optimal choice of \tilde{s} . The speed of propagation of these sensitivity waves decreases as \tilde{s} grows. Obviously, to get the resolution throughout the whole domain we would like all of Ω to be covered by the sensitivity waves, which means higher propagation speed (smaller \tilde{s}) is needed. On the other hand, if the sensitivity waves propagate too far they reflect from the boundary \mathcal{B} which leads to poor conditioning of the Jacobian. The balance between these two requirements leads to an optimal choice of \tilde{s} , which we determine experimentally in the numerical example in section 4.2.

4. Numerical results

We assess the performance of the inversion algorithm with numerical experiments. To avoid committing the inverse crime we use different grids for generating the data and for the solution of the inverse problem. We describe the setup of the numerical simulations and present the inversion results in section 4.1 for 1D media, and in section 4.2 for 2D media.

4.1. Numerical experiments in one dimension

We use a fine grid with $N_f = 299$ uniformly spaced nodes to simulate the data $d(t)$, and a coarser grid with $N = 199$ uniformly spaced nodes in the inversion.

The first term $y(t; \mathbf{r}^{\text{true}})$ in (2.7) is approximated by solving the semi-discrete forward problem (2.1) with an explicit forward Euler time stepping on a finite time interval $[0, T]$, where $T = 100$. The time step is $h_T = 10^{-5}$. We denote by \mathbf{y} the vector of length $N_T = T/h_T$, with entries given by the numerical approximation of $y(t; \mathbf{r}^{\text{true}})$ at the time samples $t_j = jh_T$. Since even in the absence of noise there is a systematic error $\mathcal{N}^{(s)}(t)$ coming from the numerical approximation of the solution of (2.1), we write

Table 2. Reduced model sizes m used for various noise levels ϵ .

ϵ	$5 \cdot 10^{-2}$	$5 \cdot 10^{-3}$	10^{-4}	0 (noiseless)
m	3	4	5	6

$$\mathbf{y} = \left(y_1, \dots, y_{N_T} \right)^T, \quad y_j = y(t_j; \mathbf{r}^{\text{true}}) + \mathcal{N}^{(s)}(t_j), \quad j = 1, \dots, N_T.$$

We also define the vector $\mathcal{N}^{(n)} \in \mathbb{R}^{N_T}$ that simulates measurement noise using the multiplicative model

$$\mathcal{N}^{(n)} = \epsilon \operatorname{diag}(\chi_1, \dots, \chi_{N_T}) \mathbf{y}, \quad (4.1)$$

where ϵ is the noise level, and χ_k are independent random variables distributed normally with zero mean and unit standard deviation. The data vector $\mathbf{d} \in \mathbb{R}^{N_T}$ is

$$\mathbf{d} = \mathbf{y} + \mathcal{N}^{(n)}, \quad (4.2)$$

and we denote its components by d_j , for $j = 1, \dots, N_T$. Such noise model allows for a simple estimate for a signal-to-noise ratio

$$\frac{\|\mathbf{d}\|_2}{\|\mathcal{N}^{(n)}\|_2} \approx \frac{1}{\epsilon}. \quad (4.3)$$

The inversion algorithm described in section 2.8 determines at step 2 the size m of the reduced model for different levels of noise. The larger ϵ , the smaller m . The values of ϵ and m used to obtain the results presented here are given in table 2.

The transfer function and its derivative at the interpolation points are approximated by taking the discrete Laplace transform of the simulated data

$$Y(\tilde{s}_j) \approx h_T \sum_{k=1}^{N_T} d_k e^{-\tilde{s}_j t_k}, \quad (4.4)$$

$$Y'(\tilde{s}_j) \approx -h_T \sum_{k=1}^{N_T} d_k t_k e^{-\tilde{s}_j t_k}. \quad (4.5)$$

To quantify the error of the reconstructions \mathbf{r}^* we use the ratio of discrete ℓ_2 norms

$$\mathcal{E} = \frac{\|\mathbf{r}^* - \mathbf{r}^{\text{true}}\|_2}{\|\mathbf{r}^{\text{true}}\|_2}. \quad (4.6)$$

While this measure of error is most appropriate for smooth resistivities, it may be overly strict for the reconstructions in the discontinuous case due to the large contribution of discontinuities. However, even under such an unfavorable measure the inversion procedure demonstrates good performance.

We show first the estimates of three resistivity functions of contrast two. We consider two smooth resistivities

$$r^{\text{true}}(x) = r_Q(x) := 2 - 4 \left(x - \frac{1}{2} \right)^2, \quad (4.7)$$

$$r^{\text{true}}(x) = r_L(x) := 0.8e^{-100(x-0.2)^2} + x + 1, \quad (4.8)$$

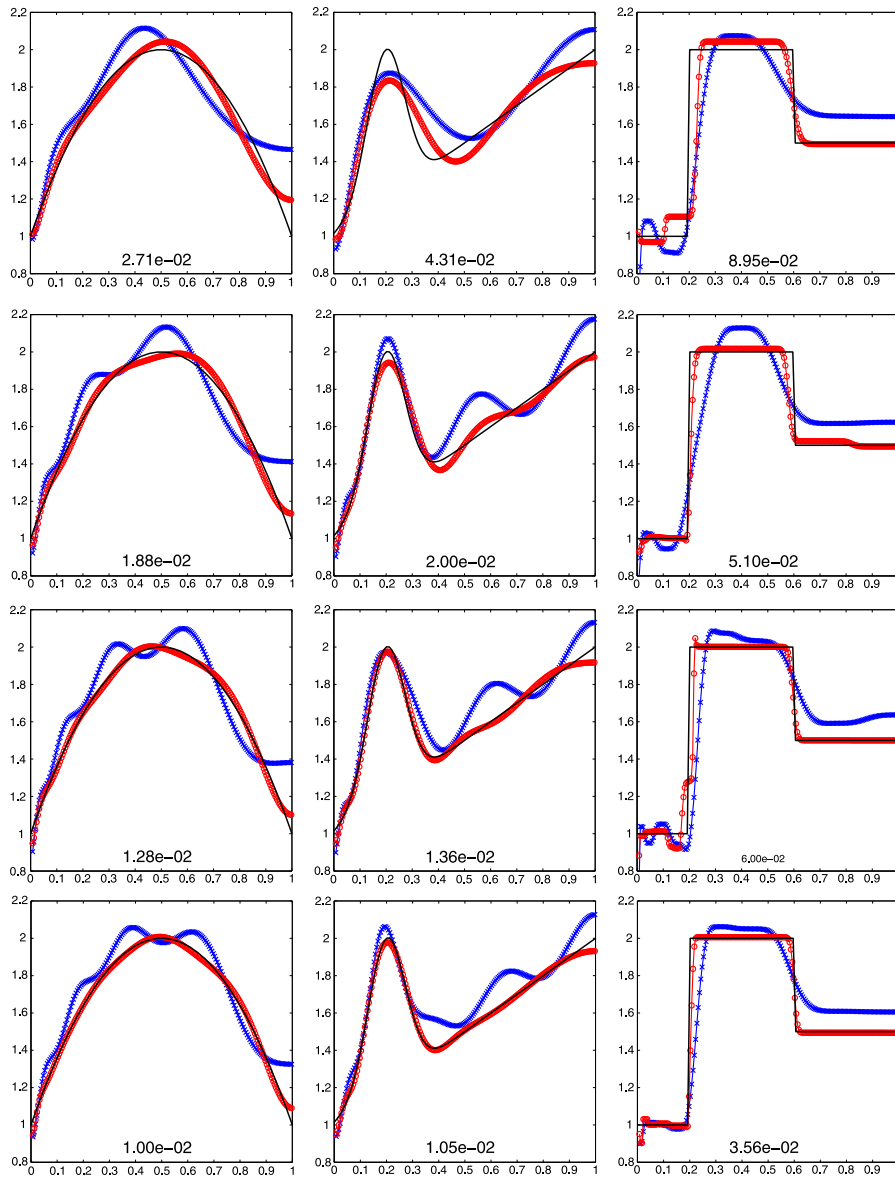


Figure 6. Reconstructions of $r(x)$ (black solid line) after one (blue \times) and five (red \circ) iterations. True coefficient by column: left r_Q , middle r_L , right r_J . Reduced model size from top row to bottom row $m = 3, 4, 5, 6$. Noise levels are from table 2. The relative error \mathcal{E} is printed at the bottom of the plots.

and the piecewise constant

$$r^{\text{true}}(x) = r_J(x) := \begin{cases} 1, & \text{for } x < 0.2, \\ 2, & \text{for } 0.2 \leq x \leq 0.6, \\ 1.5, & \text{for } x > 0.6. \end{cases} \quad (4.9)$$

The results are displayed in figure 6, for various reduced model sizes and levels of noise, as listed in table 2. Each reconstruction uses its own realization of noise. We use five Gauss–Newton iterations $n_{\text{GN}} = 5$, and we display the solution both after one iteration and after all five. The initial guess is $r^{(1)}(x) \equiv 1$. It is far from the true resistivity $r^{\text{true}}(x)$, and yet the inversion procedure converges quickly. The features of $r^{\text{true}}(x)$ are captured well after the first iteration, but there are some spurious oscillations, corresponding to the peaks of the sensitivity functions. A few more iterations of the inversion algorithm remove these oscillations and improve the quality of the estimate of the resistivity. The relative error (4.6) is indicated in each plot in figure 6. It is small, of a few percent in all cases.

We also observe in figure 6 that the inversion method regularized with the nonlinear weight (2.49) performs well for the piecewise constant resistivity r_J . Without the regularization, the estimates have Gibbs-like oscillations near the discontinuities of $r^{\text{true}}(x)$. These oscillations are suppressed by the weighted discrete H^1 regularization.

In figure 7 we are comparing our inversion procedure to an inversion approach like in [11] that fits the poles and residues $(\theta_j, c_j)_{j=1}^m$ instead of the continued fraction coefficients. The comparison is done for the case of piecewise constant resistivity of higher contrast

$$r^{\text{true}}(x) = r_H(x) := \begin{cases} 1, & \text{for } x < 0.2, \\ 5, & \text{for } 0.2 \leq x \leq 0.6, \\ 3, & \text{for } x > 0.6. \end{cases} \quad (4.10)$$

A higher contrast case is chosen since for the low contrast the difference in performance between \mathcal{R}_{ϵ_0} and \mathcal{R} is less pronounced.

As expected, the reconstructions are better when we use the mapping \mathcal{R} . In fact, the algorithm based on \mathcal{R}_{ϵ_0} diverges for $m = 4$ and $m = 6$. Thus, for $m = 4, 6$ we plot the first and second iterates only. The inversion based on the mapping \mathcal{R} converges in all three cases and maintains the relative error well below 10% for $m = 4, 5$ and around 11% for $m = 6$. The reconstruction plots in figure 7 are complimented with the plots of the relative error \mathcal{E} versus the Gauss–Newton iteration number p . Note that even when the iteration with \mathcal{R}_{ϵ_0} converges ($m = 5$) the reconstruction with \mathcal{R} has smaller error (8.5% versus 18.2%).

Aside from providing a solution of higher quality our method is also more computationally efficient since it does not require the solution of the forward problem (2.1) in time. While constructing the orthonormal basis for the Krylov subspace (2.16) requires a few linear solves with the shifted matrix A , the number m of such solves is small and thus it is cheaper than the time stepping for (2.1). For example, the explicit time stepping to generate the data \mathbf{d} for the numerical experiments above takes 275 s, whereas all the five Gauss–Newton iterations of our inversion algorithm takes less than a second on the same machine.

4.2. Numerical experiments in two dimensions

We consider a 2D example in a rectangular domain $\Omega = [0, 3] \times [0, 1]$. The fine grid to simulate the data has the dimension 120×40 nodes, while the coarse grid used in inversion is 90×30 nodes. We use $N_d = 8$ sources/receivers with disjoint supports \mathcal{J}_j uniformly distributed on the accessible boundary interval $\mathcal{B}_A = \{(x_1, x_2) \mid x_1 \in (1, 2), x_2 = 0\}$. For each

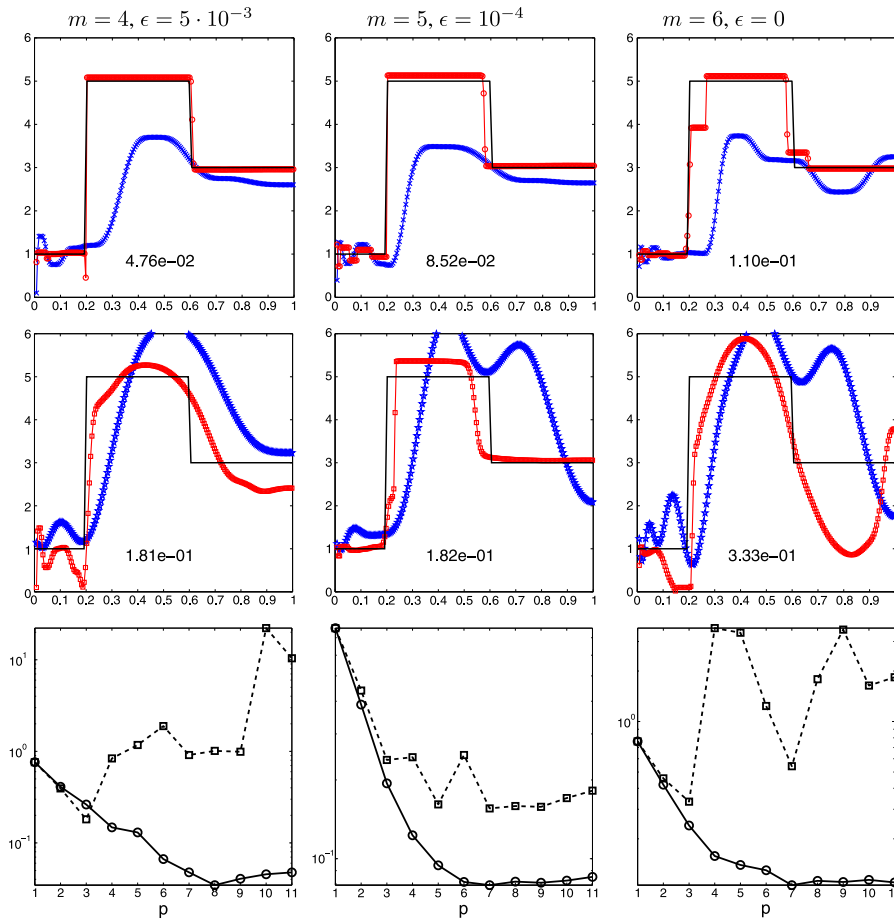


Figure 7. Comparison between two preconditioners for high contrast piecewise constant resistivity $r_H(x)$ (black solid line) after one (blue \times and \star) and n_{GN} (red \circ and \square) iterations. Top row: reconstructions using \mathcal{R} ($n_{GN} = 10$). Middle row: reconstructions using \mathcal{R}_{c_0} ($n_{GN} = 2$ for $m = 4, 6$ and $n_{GN} = 10$ for $m = 5$). Bottom row: relative error \mathcal{E} versus the iteration number p for reconstructions with \mathcal{R} (solid line with \circ) and \mathcal{R}_{c_0} (dashed line with \square). The relative error \mathcal{E} is printed at the bottom of the reconstruction plots.

diagonal entry of the measured data matrix $y_{ij}(t)$, $j = 1, \dots, N_d$ a reduced order model with $m = 5$ is constructed.

As mentioned in section 3, the interpolation node \tilde{s} for the matching conditions (3.11) is chosen so that the sensitivity waves reach the boundary \mathcal{B} without reflecting from it. This is shown in figure 8 where the sensitivities for one particular source/receiver are plotted. For this Ω the interpolation node that gives the desired behavior is $\hat{s} = 60$. We cannot take a smaller \hat{s} since the sensitivity function for $\kappa_s^{(4,4)}$ already touches the boundary $x_2 = 1$. On the other

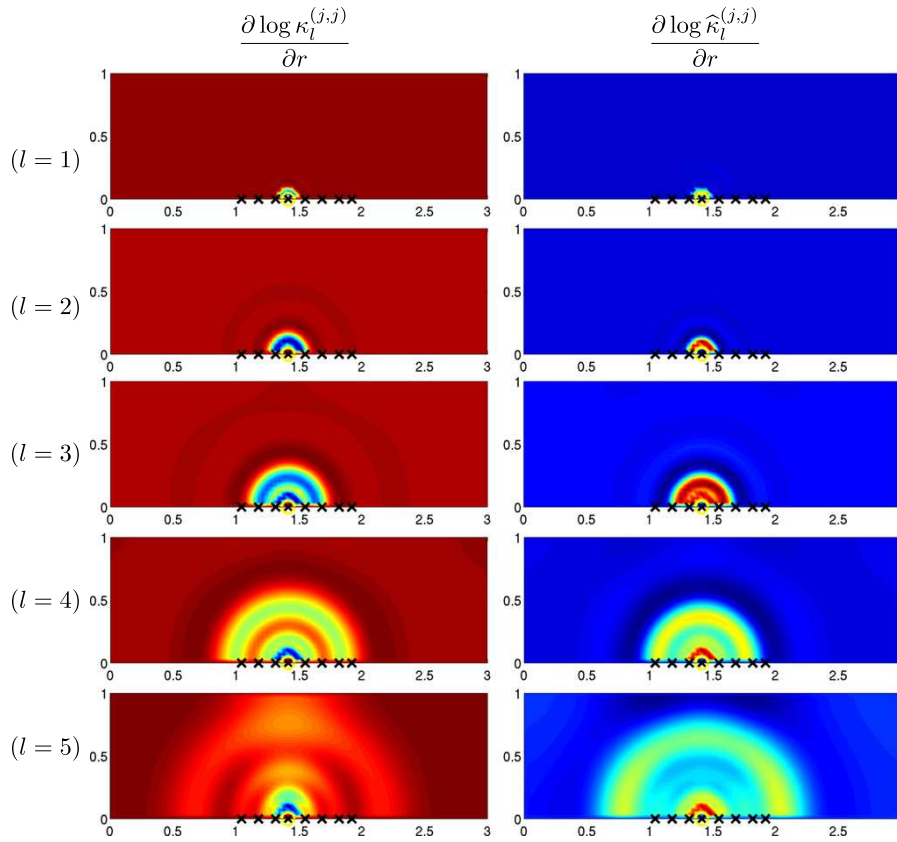


Figure 8. Sensitivity functions (rows of the Jacobian DR_j indexed by l) for the 2D uniform medium $r(x) \equiv 1$ in the rectangular domain $\Omega = [0, 3] \times [0, 1]$. Source/receiver index is $j = 4$ out of a total of $N_d = 8$ (mid-points of sources/receivers \mathcal{J}_j marked as black \times), $m = 5$.

hand, increasing $\hat{\sigma}$ will shrink the region covered by the sensitivity functions and thus will reduce the resolution away from \mathcal{B}_A .

We solve the optimization problem (3.10) for the 2D media using a regularized preconditioned Gauss–Newton inversion algorithm from section 2.8 adapted for the objective functional of (3.10). In two dimensions the preconditioner appears to be even more efficient with high quality reconstructions obtained after a single iteration. Subsequent iterations improve the reconstruction marginally, so in figure 9 we show the solutions after a single Gauss–Newton iteration starting from a uniform initial guess $r(x) \equiv 1$.

All three examples in figure 9 are piecewise constant so the inversion is regularized with a discrete H^1 seminorm ($W = I$). In the first two examples there are two rectangular inclusions in each with the contrast from $\min_{x \in \Omega} r(x) = 0.66$ to $\max_{x \in \Omega} r(x) = 1.5$ on a unit background. The inclusions touch each other at a corner and a side respectively. This demonstrates that the method handles well the sharp interfaces. In the third example there is a single tilted inclusion

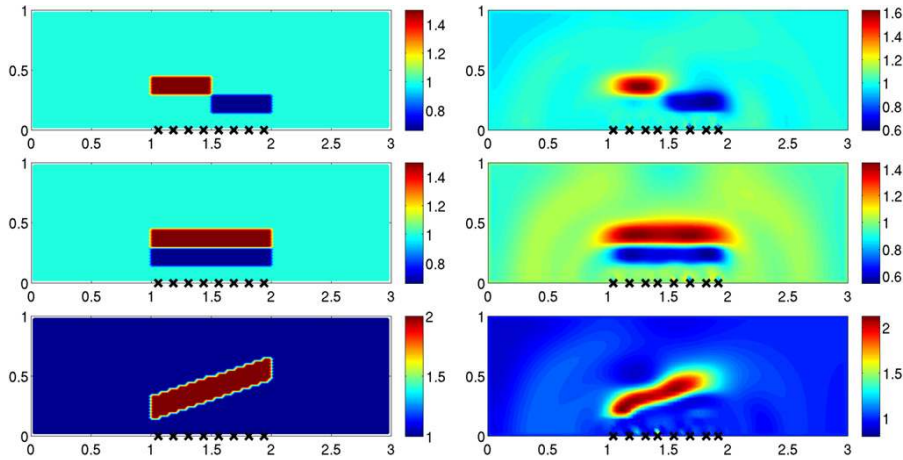


Figure 9. Reconstructions in 2D in the rectangular domain $\Omega = [0, 3] \times [0, 1]$. Left column: true coefficient $r(x)$; right column: reconstruction after a single Gauss–Newton iteration. Mid-points of sources/receivers J_j are marked with black \times , $j = 1, \dots, N_d$, $N_d = 8$.

of contrast 2 on a unit background. It is used to show the gradual loss of resolution away from B_A . All three examples are narrow aperture, meaning that the horizontal extent of the inclusions is equal to the width of B_A . Overall the reconstruction quality is good with the contrast captured fully by the first Gauss–Newton iteration. One can iterate further to improve the reconstruction, but an adaptive choice of the step length $\alpha^{(p)}$ is required for convergence.

5. Summary

We introduced a numerical inversion algorithm for linear parabolic partial differential equations. The problem arises in the application of controlled source electromagnetic inversion, where the unknown is the subsurface electrical resistivity $r(x)$ in the earth. We study the inversion method in one and 2D media, but extensions to three dimensions are possible.

To motivate the inversion algorithm we place the inverse problem in a model reduction framework. We semidiscretize in x the parabolic partial differential equation on a grid with $N \gg 1$ points, and obtain a dynamical system with transfer function $Y(s; \mathbf{r})$, the Laplace transform of the time measurements. In two dimensions the transfer function is matrix valued, and we construct reduced models separately, for each entry on its diagonal. Each model reduction construction is as in the 1D case.

The reduced models are dynamical systems of much smaller size $m \ll N$, with transfer function $Y_m(s) \approx Y(s; \mathbf{r})$. Because $Y_m(s)$ is a rational function of s , we solve a rational approximation problem. We study various such approximants to determine which are best suited for inversion. We end up with a multipoint Padé approximant $Y_m(s)$, which interpolates Y and its first derivatives at nodes distributed geometrically in \mathbb{R}^+ . The inversion algorithm is a Gauss–Newton iteration for an optimization problem preconditioned with nonlinear mappings \mathcal{Q} and \mathcal{R} . These mappings are the essential ingredients in the inversion.

Most inversion algorithms estimate $r(x)$ by minimizing over discretized resistivity vectors $\mathbf{r} \in \mathbb{R}_+^N$ the least squares misfit of the data $d(t)$ and the mathematical model $y(t; \mathbf{r})$ of the measurements. By construction, our mapping $\mathcal{R}(\mathbf{r}) = \mathcal{Q}(y(\cdot; \mathbf{r}))$ is an approximate identity when restricted to a subset of sufficiently regular resistivities \mathbf{r} . We use it as a nonlinear preconditioner in the inversion, meaning that we minimize the least squares misfit of $\mathcal{Q}(d(\cdot))$ and $\mathcal{R}(\mathbf{r})$. The advantage is the stability of the inversion and very fast convergence of the iteration.

We define the nonlinear preconditioner $\mathcal{R}(\mathbf{r})$ via an explicit chain of nonlinear mappings. Each step in the chain involves a numerically stable computation. The computation of the Jacobian $D\mathcal{R}$ follows by the chain rule and we describe it explicitly, step by step. The only unstable computation in the inversion is the data fitting calculation of $\mathcal{Q}(d(\cdot))$. The instability is inherited from that of the inverse problem and is unavoidable. We mitigate it by restricting the size m of the reduced model adaptively, depending on the noise level. The smaller m is, the lower the resolution of the estimated resistivity. This is because at each iteration the resistivity updates are in the range of $(D\mathcal{R})^\dagger$, of low dimension $2m \ll N$. We improve the results by adding corrections in the null space of $D\mathcal{R}$, so that we minimize a regularization functional that incorporates prior information about the unknown resistivity. The performance of the algorithm is assessed with numerical simulations in one and two dimensions.

Acknowledgments

The work of L Borcea was partially supported by the AFSOR Grant FA9550-12-1-0117, the ONR Grant N000141410077, and by the NSF Grant DMS-0934594. The work of A Mamonov was partially supported by the NSF grants DMS-0934594, DMS-0914465 and DMS-0914840.

Appendix. Computation of the nonlinear preconditioner and its Jacobian

The computation of the nonlinear preconditioner $\mathcal{R}(\mathbf{r}) = \mathcal{Q}(y(\cdot; \mathbf{r}))$ and its Jacobian is the most complex and time consuming computation in our inversion scheme. Nevertheless, it is much more efficient than the traditional inversion approach because it avoids the repeated computation of the time domain solution of the forward problem. We explain here the details of the computation of \mathcal{R} and $D\mathcal{R}$ via the chain of mappings (2.23). We do so only for the multipoint Padé approximant, which we showed in section 2.6 to be best suited for inversion.

(a) The matrix A is defined by (2.2) for a given \mathbf{r} . Differentiating (2.2) yields

$$\frac{\partial A}{\partial r_k} = -D^T \mathbf{e}_k \mathbf{e}_k^T D = -\mathbf{d}_k \mathbf{d}_k^T, \quad k = 1, \dots, N, \quad (\text{A.1})$$

with $\mathbf{d}_k = D_{k, 1:N}^T$. This is a rank one matrix.

(b) At this step we differentiate the orthonormal basis V of the Krylov subspace $\mathcal{K}_m(\tilde{\mathcal{S}})$. There are different ways of computing an orthonormal basis of $\mathcal{K}_m(\tilde{\mathcal{S}})$. One choice is to use a rational Lanczos algorithm [14]. While this may be a more stable way of computing V compared to other approaches, differentiation formulas are difficult to derive and implement. We consider an alternative approach, based on the differentiation of the QR decomposition.

We compute first the matrix

$$K = \left[(\tilde{s}_1 I - A)^{-1} \mathbf{b}, \dots, (\tilde{s}_m I - A)^{-1} \mathbf{b} \right] \in \mathbb{R}^{N \times m},$$

and its derivatives

$$\frac{\partial K}{\partial r_k} = - \left[(\tilde{s}_1 I - A)^{-1} \mathbf{d}_k \left[(\tilde{s}_1 I - A)^{-1} \mathbf{d}_k \right]^T \mathbf{b}, \dots, (\tilde{s}_m I - A)^{-1} \mathbf{d}_k \left[(\tilde{s}_m I - A)^{-1} \mathbf{d}_k \right]^T \mathbf{b} \right].$$

Then V can be defined via the QR decomposition of K , which we write as

$$K = VU,$$

where $V \in \mathbb{R}^{N \times m}$ is orthogonal $V^T V = I_m \in \mathbb{R}^{m \times m}$, and $U \in \mathbb{R}^{m \times m}$ is upper triangular. If we denote $L = U^T$, then (A) implies

$$K^T K = LL^T.$$

That is to say $L = U^T$ is a Cholesky factor of $K^T K$. At the same time, when we differentiate (A), we obtain

$$\frac{\partial V}{\partial r_k} = \left(\frac{\partial K}{\partial r_k} - V \frac{\partial U}{\partial r_k} \right) U^{-1}, \quad k = 1, \dots, N. \quad (\text{A.2})$$

Since we already know $\partial K / \partial r_k$, it remains to compute the derivative $\partial U / \partial r_k$ of the Cholesky factorization of $K^T K$. This is given in the following proposition, which can be proved by direct computation once we write $\delta(LL^T) = (\delta L)L^T + L(\delta L)^T$ and solve for the columns of δL , one a time, using that δL is lower triangular.

Proposition 1. (Differentiation of Cholesky factorization). *Let $M \in \mathbb{R}^{m \times m}$ be a matrix with Cholesky factorization $M = LL^T$. Given the perturbation δM of M , the corresponding perturbation δL of the Cholesky factor is computed by the following algorithm.*

For $k = 1, \dots, m$

$$\delta L_{kk} = \frac{1}{L_{kk}} \left(\frac{\delta M_{kk}}{2} - \sum_{j=1}^{k-1} \delta L_{kj} L_{kj} \right).$$

For $i = k + 1, \dots, m$

$$\delta L_{ik} = \frac{1}{L_{kk}} \left(\delta M_{ik} - \sum_{j=1}^k \delta L_{kj} L_{ij} - \sum_{j=1}^{k-1} \delta L_{ij} L_{kj} \right).$$

We use proposition 1 for $M = K^T K = LL^T$, with perturbation

$$\delta M = \frac{\partial M}{\partial r_k} \delta r_k, \quad \frac{\partial M}{\partial r_k} = \left(\frac{\partial K}{\partial r_k} \right)^T K + K^T \left(\frac{\partial K}{\partial r_k} \right),$$

to obtain

$$\delta L = \frac{\partial L}{\partial r_k} \delta r_k \quad \text{and} \quad \frac{\partial U}{\partial r_k} = \left(\frac{\partial L}{\partial r_k} \right)^T.$$

The computation of $\partial V / \partial r_k$ follows from (A.2).

(c) Once we have V and its derivatives we compute from (2.12)

$$\frac{\partial A_m}{\partial r_k} = -V^T \mathbf{d}_k \mathbf{d}_k^T V + \frac{\partial V^T}{\partial r_k} AV + V^T A \frac{\partial V}{\partial r_k}, \quad (\text{A.3})$$

$$\frac{\partial \mathbf{b}_m}{\partial r_k} = \frac{\partial V^T}{\partial r_k} \mathbf{b}. \quad (\text{A.4})$$

(d)–(e) There are two possible ways to go from the reduced model A_m , \mathbf{b}_m to the continued fraction coefficients κ_j , $\hat{\kappa}_j$. Both approaches use a Lanczos iteration to obtain a symmetric tridiagonal matrix, which we denote by

$$T = \begin{bmatrix} \alpha_1 & \beta_2 & & \\ \beta_2 & \alpha_2 & \ddots & \\ & \ddots & \ddots & \beta_m \\ & & \beta_m & \alpha_m \end{bmatrix}. \quad (\text{A.5})$$

Note that for an arbitrary symmetric matrix $E \in \mathbb{R}^{m \times m}$ we can compute a tridiagonal matrix T that is unitarily similar to E via a Lanczos iteration with full reorthogonalization. Since the dimension m of the problem is small, we reorthogonalize at every step to ensure maximum numerical stability. The iteration is as follows:

Initialize $\mathbf{x}_1 = \boldsymbol{\eta}$, $\mathbf{x}_0 = 0$, $\beta_1 = 0$.

For $j = 1, \dots, m - 1$

$$\alpha_j = \mathbf{x}_j^T E \mathbf{x}_j,$$

$$\tilde{\mathbf{u}}_{j+1} = E \mathbf{x}_j - \alpha_j \mathbf{x}_j - \beta_j \mathbf{x}_{j-1},$$

$$\mathbf{u}_{j+1} = (I - X_{1:N, 1:j} (X_{1:N, 1:j})^T) \tilde{\mathbf{u}}_{j+1},$$

$$\beta_{j+1} = \|\mathbf{u}_{j+1}\|,$$

$$\mathbf{x}_{j+1} = \frac{\mathbf{u}_{j+1}}{\beta_{j+1}},$$

$$\alpha_m = \mathbf{x}_m^T E \mathbf{x}_m.$$

We have two choices of the initial vector $\boldsymbol{\eta}$ and the matrix E such that $T = X^T E X$ with $X^T X = I_m$. The first takes $E = A_m$, and $\boldsymbol{\eta} = \mathbf{b}_m / \|\mathbf{b}_m\|$. It combines steps (e) and (d) and goes from the reduced model A_m , \mathbf{b}_m directly to the tridiagonal matrix T . The second approach is to compute first the eigenvalue decomposition (2.15) of A_m to get the poles $-\theta_j$ and the residues c_j , for $j = 1, \dots, m$. Then take $E = -\text{diag}(\theta_1, \dots, \theta_m)$ and

$$\boldsymbol{\eta} = (\eta_1, \dots, \eta_m)^T, \quad \eta_i = \sqrt{\frac{c_i}{\sum_{s=1}^m c_s}}, \quad i = 1, \dots, m. \quad (\text{A.6})$$

From the computed T with either of the two approaches, we obtain the coefficients of the continued fraction using the formulas from [8].

$$\hat{\kappa}_1 = \frac{1}{m}, \quad \kappa_1 = -\frac{1}{\hat{\kappa}_1 \alpha_1}, \quad (\text{A.7})$$

$$\hat{\kappa}_j = \frac{1}{\kappa_{j-1}^2 \beta_{j-1}^2 \hat{\kappa}_{j-1}}, \quad j = 2, \dots, m, \quad (\text{A.8})$$

$$\kappa_j = -\frac{1}{\alpha_j \hat{\kappa}_j + \frac{1}{\kappa_{j-1}}}, \quad j = 2, \dots, m. \quad (\text{A.9})$$

To differentiate the mappings (d) and (e), we need to differentiate the Lanczos iteration. In general, there is no explicit formula for the perturbations of the entries of T in terms of perturbations of E and $\boldsymbol{\eta}$. Thus, we would need to differentiate the Lanczos iteration directly, with an algorithm that computes the perturbations of α_j , β_j and \mathbf{x}_j iteratively, for increasing j . However, for the second approach described above, we can use the explicit perturbation formulas for the Lanczos iteration derived in [5]. To apply these formulas, we need to differentiate the steps (d) and (e) separately.

The differentiation of (d) follows directly from the differentiation of the eigendecomposition (2.15) of A_m . It is given by

$$\frac{\partial \theta_j}{\partial r_k} = -\mathbf{z}_j^T \frac{\partial A_m}{\partial r_k} \mathbf{z}_j, \quad (\text{A.10})$$

$$\frac{\partial \mathbf{z}_j}{\partial r_k} = -(A_m + \theta_j I)^\dagger \frac{\partial A_m}{\partial r_k} \mathbf{z}_j, \quad (\text{A.11})$$

$$\frac{\partial c_j}{\partial r_k} = 2 \left(\mathbf{b}_m^T \mathbf{z}_j \right) \left(\mathbf{b}^T \frac{\partial V}{\partial r_k} \mathbf{z}_j + \mathbf{b}_m^T \frac{\partial \mathbf{z}_j}{\partial r_k} \right), \quad (\text{A.12})$$

where \dagger denotes the pseudoinverse.

For the computation of the derivatives in step (e) we use the explicit formulas from [5], for $E = \boldsymbol{\theta}$ and $X = Q^T$ and $\boldsymbol{\eta}$ given by (A.6). Let us define the vectors

$$\boldsymbol{\delta}_\alpha = \begin{pmatrix} \delta\alpha_1 \\ \delta\alpha_1 + \delta\alpha_2 \\ \vdots \\ \sum_{j=1}^m \delta\alpha_j \end{pmatrix}, \quad \boldsymbol{\delta}_\beta = \begin{pmatrix} \delta\beta_1/\beta_1 \\ \delta\beta_1/\beta_1 + \delta\beta_2/\beta_2 \\ \vdots \\ \sum_{j=1}^m \delta\beta_j/\beta_j \end{pmatrix}. \quad (\text{A.13})$$

They can be expressed in terms of the perturbations of the poles $\boldsymbol{\theta} = (\theta_1, \dots, \theta_m)^T$ and the initial vector $\boldsymbol{\eta}$ as

$$\boldsymbol{\delta}_\alpha = -A_\theta \delta\boldsymbol{\theta} + A_\eta \delta\boldsymbol{\eta}, \quad \boldsymbol{\delta}_\beta = -B_\theta \delta\boldsymbol{\theta} + B_\eta \delta\boldsymbol{\eta}.$$

Here $A_\theta, A_\eta \in \mathbb{R}^{m \times m}$ are the matrices with entries

$$A_\theta^{ij} = 1 + \beta_i \sum_{\substack{p=1 \\ p \neq j}}^m \frac{1}{\theta_p - \theta_j} \left[2Q_{ip}Q_{i+1,p} - \frac{Q_{ip}}{Q_{lj}} (Q_{ip}Q_{i+1,j} - Q_{i+1,p}Q_{ij}) \right],$$

$$A_\eta^{ij} = 2\beta_i \frac{Q_{i+1,j}Q_{ij}}{Q_{lj}}, \quad A_\eta^{mj} = 0, \quad A_\theta^{mj} = 1,$$

for $i = 1, \dots, m - 1, j = 1, \dots, m$ and the entries of $B_\theta, B_\eta \in \mathbb{R}^{(m-1) \times m}$ are

$$B_{\theta}^{ij} = \sum_{\substack{p=1 \\ p \neq j}}^m \frac{1}{\theta_p - \theta_j} \left[(Q_{i+1,p})^2 - \frac{Q_{1p}}{Q_{1j}} Q_{i+1,p} Q_{i+1,j} \right],$$

$$B_{\eta}^{ij} = \frac{(Q_{i+1,j})^2}{Q_{1j}},$$

for $i = 1, \dots, m - 1$ and $j = 1, \dots, m$.

Note that the computation of $\delta \boldsymbol{\eta}$ can be done by differentiating (A.6) and using (A.12). Once the vectors $\delta \alpha$ and $\delta \beta$ are known, it is trivial to obtain the individual perturbations $\delta \alpha_j$ and $\delta \beta_j$. From those perturbations we compute the derivatives of κ_j and $\hat{\kappa}_j$ by differentiating relations (A.7)–(A.9). The computation is straightforward and we do not include it here.

(f) The differentiation at the last step in the chain of mappings (2.23) is trivial. It is just the derivative of the logarithm.

References

- [1] Abubakar A, Habashy T, Druskin V, Knizhnerman L and Alumbaugh D 2008 2.5 D forward and inverse modeling for interpreting low-frequency electromagnetic measurements *Geophysics* **73** F165–77
- [2] Beilina L and Klivanov M V 2008 A globally convergent numerical method for a coefficient inverse problem *SIAM J. Sci. Comput.* **31** 478–509
- [3] Beretta E, de Hoop M V, Qiu L and Scherzer O 2014 Inverse boundary value problem for the Helmholtz equation: multi-level approach and iterative reconstruction arXiv:1406.2391 [math.NA]
- [4] Borcea L, Druskin V and Guevara Vasquez F 2008 Electrical impedance tomography with resistor networks *Inverse Problems* **24** 035013
- [5] Borcea L, Druskin V and Knizhnerman L 2005 On the continuum limit of a discrete inverse spectral problem on optimal finite difference grids *Commun. Pure Appl. Math.* **58** 1231–79
- [6] Borcea L, Druskin V, Mamonov A and Guevara Vasquez F 2010 Pyramidal resistor networks for electrical impedance tomography with partial boundary measurements *Inverse Problems* **26** 105009
- [7] Börner R, Ernst O and Spitzer K 2008 Fast 3-D simulation of transient electromagnetic fields by model reduction in the frequency domain using Krylov subspace projection *Geophys. J. Int.* **173** 766–80
- [8] Druskin V and Knizhnerman L 2000 Gaussian spectral rules for second order finite-difference schemes *Numer. Algorithms* **25** 139–59
- [9] Druskin V and Knizhnerman L 2000 Gaussian spectral rules for the three-point second differences: part 1. A two-point positive definite problem in a semi-infinite domain *SIAM J. Numer. Anal.* **37** 403–22
- [10] Druskin V, Knizhnerman L and Zaslavsky M 2009 Solution of large scale evolutionary problems using rational Krylov subspaces with optimized shifts *SIAM J. Sci. Comput.* **31** 3760
- [11] Druskin V, Simoncini V and Zaslavsky M 2013 Solution of the time-domain inverse resistivity problem in the model reduction framework: part 1. One-dimensional problem with SISO data *SIAM J. Sci. Comput.* **35** A1621–40
- [12] Druskin V and Zaslavsky M 2007 On combining model reduction and Gauss–Newton algorithms for inverse partial differential equation problems *Inverse Problems* **23** 1599
- [13] Frommer A and Simoncini V 2008 Matrix functions *Model Order Reduction: Theory, Research Aspects and Applications* (Berlin: Springer) pp 275–303
- [14] Gallivan K, Grimme E and van Dooren P 1996 A rational Lanczos algorithm for model reduction *Numer. Algorithms* **12** 33–63
- [15] Gonnet P, Güttel S and Trefethen L N 2011 Robust Padé approximation via SVD *SIAM Rev.* **55** 101–17

- [16] Grimme E J 1997 Krylov projection methods for model reduction *PhD Thesis* Citeseer
- [17] Gugercin S, Antoulas A and Beattie C 2008 \mathcal{H}_2 Model reduction for large-scale linear dynamical systems *SIAM J. Matrix Anal. Appl.* **30** 609–38
- [18] Ingerman D, Druskin V and Knizhnerman L 2000 Optimal finite difference grids and rational approximations of the square root: part 1. Elliptic problems *Commun. Pure Appl. Math.* **53** 1039–66
- [19] Kabanikhin S I 2011 *Inverse and Ill-Posed Problems: Theory and Applications* vol 55 (Berlin: Walter de Gruyter)
- [20] Kilmer M, Gugercin S, Beattie C, Chaturantabut S, Huffman M-K and de Sturler E 2012 Challenges in modern medical image reconstruction *Proc. SIAM Conf. on Applied Linear Algebra* (Valencia, Spain)
- [21] Klivanov M V and Timonov A A 2007 Numerical studies on the globally convergent convexification algorithm in 2D *Inverse Problems* **23** 123–38
- [22] Newman G and Alumbaugh D 2000 Three-dimensional magnetotelluric inversion using nonlinear conjugate gradients *Geophys. J. Int.* **140** 410–24
- [23] Wang T, Oristaglio M, Tripp A and Hohmannl G 1994 Inversion of diffusive transient electromagnetic data by a conjugate-gradient method *Radio Sci.* **29** 1143–56
- [24] Zaslavsky M, Druskin V, Abubakar A and Habashy T 2013 Large-scale Gauss–Newton inversion of transient CSEM data using the model reduction framework *Geophysics* **78** E161–71
- [25] Zhdanov M 2002 *Geophysical Inverse Theory and Regularization Problems* vol 36 (Amsterdam: Elsevier)