



Recycling Krylov subspaces for efficient large-scale electrical impedance tomography

Luís Augusto Motta Mello^{a,*}, Eric de Sturler^b, Glaucio H. Paulino^c, Emílio Carlos Nelli Silva^a

^a Department of Mechatronics and Mechanical Systems Engineering, School of Engineering, University of São Paulo, Brazil

^b Department of Mathematics, 460 McBryde Hall, Virginia Tech, Blacksburg, VA 24061–0123, USA

^c Department of Civil and Environmental Engineering, Newmark Laboratory, University of Illinois at Urbana-Champaign, USA

ARTICLE INFO

Article history:

Received 4 September 2009

Received in revised form 7 May 2010

Accepted 8 June 2010

Available online 18 June 2010

Keywords:

Iterative methods

Preconditioning

Krylov subspace recycling

Three-dimensional electrical impedance tomography

Sequential linear programming

ABSTRACT

Electrical impedance tomography (EIT) captures images of internal features of a body. Electrodes are attached to the boundary of the body, low intensity alternating currents are applied, and the resulting electric potentials are measured. Then, based on the measurements, an estimation algorithm obtains the three-dimensional internal admittivity distribution that corresponds to the image. One of the main goals of medical EIT is to achieve high resolution and an accurate result at low computational cost. However, when the finite element method (FEM) is employed and the corresponding mesh is refined to increase resolution and accuracy, the computational cost increases substantially, especially in the estimation of absolute admittivity distributions. Therefore, we consider in this work a fast iterative solver for the forward problem, which was previously reported in the context of structural optimization. We propose several improvements to this solver to increase its performance in the EIT context. The solver is based on the recycling of approximate invariant subspaces, and it is applied to reduce the EIT computation time for a constant and high resolution finite element mesh. In addition, we consider a powerful preconditioner and provide a detailed pseudocode for the improved iterative solver. The numerical results show the effectiveness of our approach: the proposed algorithm is faster than the preconditioned conjugate gradient (CG) algorithm. The results also show that even on a standard PC without parallelization, a high mesh resolution (more than 150,000 degrees of freedom) can be used for image estimation at a relatively low computational cost.

© 2010 Elsevier B.V. All rights reserved.

1. Introduction

Electrical impedance tomography (EIT) finds the admittivity (conductivity and permittivity) distribution in a given model of a body that corresponds to the boundary measurements of currents and potentials on electrodes attached to that body [1]. The model of the body is based on an elliptic partial differential equation obtained from Maxwell's equations and the admittivity distribution represents the solution of a non-linear and ill-posed inverse problem. Several combinations of current-carrying electrodes can be chosen and, therefore, many induced electric potential values may be available for the admittivity estimation.

The EIT applications are the spatial conductivity estimation of carbon nanotube composite thin films for sensing purposes [2], the detection of faults in coatings of endoprostheses [3] and ground water resources [4], mammography [5], the monitoring of lung aeration imposed by mechanical ventilation [1,6,7] (see Fig. 1), etc. In the context of the last application, which is the main

interest of our group, the technique is harmless to the patient and the hardware device is portable with relatively low cost; in addition, the reconstruction of absolute admittivity values has shown its relevance because these values help distinguish certain lung pathologies [1].

Several algorithms have been proposed to solve the non-linear inverse problem for the absolute admittivity values. They are usually based on iterative methods, such as Gauss–Newton [2,9–13], which require the solution of the forward problem, i.e., the computation of the electric potential for a known admittivity distribution and prescribed boundary conditions [14]. The solution of the forward problem involves the solution of a linear system of equations, often obtained through the finite element method (FEM) [1,2,9–16] (the method of fundamental solutions [17] and the boundary element method [18] have also been reported). Since several combinations of current-carrying electrodes are considered in the image estimation, the linear systems have different right-hand sides. Iterative methods are usually reported rather than direct methods [1,11,13,15,16], because of low storage requirements and fast computations, even for multiple right-hand sides (see [19]).

* Corresponding author.

E-mail address: luis.mello@poli.usp.br (L.A.M. Mello).

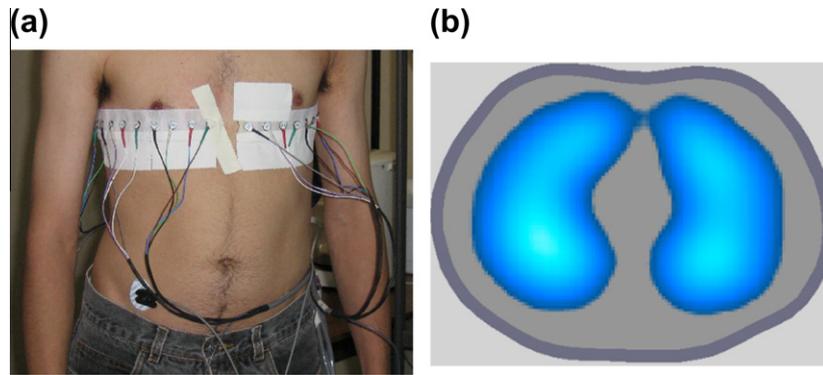


Fig. 1. EIT applied to monitor the lung. (a) Experimental apparatus, including the electrodes around the volunteer's chest. (b) Ventilation map [8], a difference image obtained based on the impedivity distribution (lighter colors within the lungs indicate higher variations).

In fact, iterative methods [19,20] for solving the forward problem offer several advantages over direct methods [21]. First, the systems do not need to be solved accurately at the start of the iterative image estimation process. Second, iterative solvers are easier to parallelize than direct solvers. Third, one can use solutions from previous systems as starting guesses, which significantly speeds up the convergence. Fourth, for a sequence of linear systems that change slowly, the total runtime can be reduced by recycling subspaces of search spaces corresponding to earlier linear systems [22,23].

One of the main goals of EIT is to achieve high resolution and accuracy at low computational cost (for instance, to monitor lung aeration, a resolution of less than 0.01 m for an adult thorax is desirable). However, as finer finite element meshes are considered, the computation time increases as well, especially in the case of three-dimensional (3D) meshes and the estimation of absolute admittivity distributions. In fact, the computation time can be prohibitive in practical 3D cases.

Some authors have studied the reduction of runtimes in EIT. In [18], for instance, neural network approximation models are used in order to significantly speed up the image estimation process. On the other hand, in [15], a black-box algebraic multigrid preconditioner is tested, showing reduced time cost of solving the forward problem.

In this work, we consider the fast iterative solver proposed in [21] with additional improvements in conjunction with a powerful preconditioner [21,24] to reduce the EIT computation time for a constant and high resolution finite element mesh. In addition, we provide a detailed pseudocode for the improved iterative solver. The considered iterative method is based on the concept of Krylov subspace recycling, which is applied to diminish the time and number of iterations of each FEM solution.

We employ sequential linear programming (SLP) [25] to solve the inverse problem (a similar method has shown good results in the EIT context [1,16]), and apply a simple regularization method that proved effective in [1]. However, the approaches proposed in this paper are general and do not assume a specific image estimation or regularization method. The SLP algorithm finds the admittivity distribution that minimizes a square error function for measured and numerically computed potential values (see Section 3).

Krylov subspace recycling speeds up the convergence of each linear system by exploiting the fact that in the iterative optimization process (or SLP iterative process) the changes in the linear system matrices are small. In addition, this technique can be used to speed up the computation of the gradient of the error function during one SLP iteration, which is based on the solution of several linear systems with the same matrix and many right-hand sides. The

preconditioner, on the other hand, reduces the condition numbers of the linear system matrices, giving rise to faster convergence and shorten solution times in spite of the overhead from its computation and application.

We consider the image reconstruction of a cylinder with high resistivity within a conductive body. We also simulate the data, which means that a numerical phantom [26] provides the measured potentials. Results show that the image estimation using the proposed iterative solver is approximately 35% faster than the estimation using the corresponding conventional Krylov subspace solver, and 24% faster than with the well-known (preconditioned) Conjugate Gradient (CG) algorithm [20], for a finite element mesh with 849,995 elements and 152,226 nodes. The results also show that the solver can handle high mesh resolutions at relatively small computational cost (for the same finite element mesh, the whole estimation process took approximately 6 h and 25 min), even on a standard PC without parallelization.

This paper is organized as follows. In Section 2, the FEM-based computational model is presented. In Section 3, the solution of the inverse problem is discussed. In Section 4, Krylov subspace recycling is explained. In Section 5, we describe the preconditioner. In Section 6, implementation details are provided. Results are presented in Section 7, and concluding remarks are offered in Section 8. In the Appendix A, we list the pseudocode for the improved iterative solver.

2. Computational model

Maxwell's equations describe the electromagnetic fields in the body [26]. Based on these equations and considering the quasi-static approximation for a linear and isotropic medium and a sufficiently small excitation frequency, a conductive medium can be assumed and the following elliptic partial differential equation is obtained [26]:

$$\nabla \cdot \sigma \nabla V = 0, \quad (1)$$

where σ and V are the real valued electric conductivity and electric potential, respectively. Taking into account Neumann and Dirichlet boundary conditions, and applying a variational approach [27] followed by the discretization of the model in finite elements (in this work, four node tetrahedral elements are employed), we can obtain:

$$\mathbf{K}(\boldsymbol{\sigma})\mathbf{V}_j = \mathbf{I}_j. \quad (2)$$

The term $\boldsymbol{\sigma} \in \mathbb{R}^{nd}$ is the vector of nodal conductivities (we assume that the conductivity of each finite element depends on nodal values, varying linearly within the element in the same way as the electric potentials [27]), nd is the number of nodes in the mesh,

$\mathbf{K}(\boldsymbol{\sigma}) \in \mathbb{R}^{nd \times nd}$ is the conductivity matrix, $\mathbf{I}_j \in \mathbb{R}^{nd}$ and $\mathbf{V}_j(\boldsymbol{\sigma}) \in \mathbb{R}^{nd}$ are the vectors of nodal electric currents and electric potentials, respectively, and the index j indicates different current-carrying electrodes (or excitation patterns, related to the Neumann boundary conditions). The matrix $\mathbf{K}(\boldsymbol{\sigma})$ does not depend on j because the same reference node [14] (corresponding to the Dirichlet boundary conditions) is employed for all excitation patterns.

Point electrodes at nodes model narrow electrodes, that is, an electrode is modeled as a boundary node of the finite element mesh. The contact interfaces between the electrodes and the body [1,14,26,28,29] are not included in this model. However, we believe these simplifications do not invalidate our results for the case of more elaborate electrode models [26,28]. The reason is that such electrode models introduce a small number of new unknowns in the forward problem [1,14,26,28,29]. In addition, the ill-conditioning that may arise due to the high resistivity values of the contact layers can probably be handled quite efficiently by diagonal scaling. A detailed discussion on diagonal scaling can be found in [24].

3. Solution of the EIT inverse problem

In this section, we succinctly describe the theory regarding the SLP algorithm employed. We also present the formulation of the conductivity estimation problem and discuss the computation of sensitivities, an important step of the SLP algorithm.

3.1. Formulation and SLP algorithm

The EIT inverse problem is usually based on the minimization of a square error function for measured and computed potential values [1,2,12,14,16], which can be conveniently given by

$$F(\boldsymbol{\sigma}) = \frac{1}{2} \sum_{j=1}^{ne} (\mathbf{A}_j \mathbf{B} \mathbf{V}_j(\boldsymbol{\sigma}) - \mathbf{V}_{0j})^T (\mathbf{A}_j \mathbf{B} \mathbf{V}_j(\boldsymbol{\sigma}) - \mathbf{V}_{0j}), \quad (3)$$

where ne is the number of different excitation patterns, $\mathbf{V}_{0j} \in \mathbb{R}^{nm}$ is the j th vector of measurements corresponding to differences of electrical potentials on adjacent narrow electrodes, nm is the number of measurements for each excitation pattern, $\mathbf{B} \in \mathbb{R}^{nt \times nd}$ selects from $\mathbf{V}_j(\boldsymbol{\sigma})$ all nt differences of potentials that correspond to adjacent point electrodes, and $\mathbf{A}_j \in \mathbb{R}^{nm \times nt}$ is a diagonal matrix that discards the values of differences on current-carrying electrodes. As suggested, each row of \mathbf{B} is given by $(0 \dots 0 \ 1 \ 0 \ \dots 0 \ -1 \ 0 \ \dots 0)$, where the positions of the ± 1 correspond to the positions of the point (or nodal) electrodes.

Based on Eqs. (2) and (3), we define the following minimization problem in order to obtain the inner distribution of conductivities of an object:

$$\min_{\boldsymbol{\sigma}} F(\boldsymbol{\sigma}) \quad \text{subject to} \quad \mathbf{K}(\boldsymbol{\sigma}) \mathbf{V}_j(\boldsymbol{\sigma}) = \mathbf{I}_j, \quad j = 1 \dots ne, \quad (4)$$

$$\sigma_B \leq \sigma_k \leq \sigma_A, \quad k = 1 \dots nd,$$

where σ_A and σ_B are the electric conductivities of the materials that compose the object (they could be the limits of a range of values in a clinical situation).

The SLP algorithm iteratively solves the constrained optimization problem in (4). At each SLP iteration, the non-linear error function (3) is linearized at the approximation (for conductivities) obtained in the previous iteration, and a linear programming (LP) algorithm [25] solves the linearized problem, obtaining a new approximation. In addition, moving limits, i.e., additional box constraints for each conductivity σ_k , are applied in the linearized problem to assure that a good approximation for the solution of the original non-linear problem is obtained. The range of values within the moving limits is reduced if the corresponding conductivity oscillates or stagnates, and it is increased otherwise. The SLP iterative process is continued until a convergence criterion is satisfied.

As LP solutions approach convergence, one can expect the conductivity values to oscillate or stagnate due to a box constraint. Thus, the reduction (in average sense) of the maximum absolute change in the nodal conductivities for two conductivity vectors corresponding to consecutive SLP iterations and, consequently, the reduction of the differences between the corresponding conductivity matrices \mathbf{K} are also expected. This feature of the SLP algorithm is exploited by using the Krylov recycling technique.

3.2. Computation of sensitivities

For brevity, we write \mathbf{V}_j for the vector $\mathbf{V}_j(\boldsymbol{\sigma})$ and \mathbf{K} for the matrix $\mathbf{K}(\boldsymbol{\sigma})$.

The gradient of F , used in the SLP iteration, is given by

$$\frac{\partial F}{\partial \sigma_k} = \sum_{j=1}^{ne} \left(\frac{\partial F}{\partial \mathbf{V}_j} \right)^T \frac{\partial \mathbf{V}_j}{\partial \sigma_k}, \quad (5)$$

where each element of $\frac{\partial F}{\partial \mathbf{V}_j}$ is a derivative of F with respect to an element of \mathbf{V}_j . By differentiating (2), one can obtain:

$$\frac{\partial \mathbf{V}_j}{\partial \sigma_k} = -\mathbf{K}^{-1} \frac{\partial \mathbf{K}}{\partial \sigma_k} \mathbf{V}_j. \quad (6)$$

On the other hand, one can write:

$$\frac{\partial F}{\partial \mathbf{V}_j} = (\mathbf{A}_j \mathbf{B} \mathbf{V}_j - \mathbf{V}_{0j})^T \mathbf{A}_j \mathbf{B}. \quad (7)$$

By substituting the last two equations into (5) the following is computed:

$$\frac{\partial F}{\partial \sigma_k} = - \sum_{j=1}^{ne} (\mathbf{A}_j \mathbf{B} \mathbf{V}_j - \mathbf{V}_{0j})^T \mathbf{A}_j \mathbf{B} \mathbf{K}^{-1} \frac{\partial \mathbf{K}}{\partial \sigma_k} \mathbf{V}_j. \quad (8)$$

Eq. (8) suggests that $ne + ne \ nd$ linear systems must be solved to compute the gradient of F (which corresponds to the so-called direct method [30, p. 264]). However, if we solve:

$$\lambda = \mathbf{K}^{-1} \mathbf{B}^T \mathbf{A}_j^T (\mathbf{A}_j \mathbf{B} \mathbf{V}_j - \mathbf{V}_{0j}), \quad (9)$$

for each j , transpose the results and substitute the transposed vectors into (8), only $2ne$ systems need to be solved. This is a significant reduction since the order of magnitude of nd is equal to 10^5 in this work and ne is often smaller than 32 in EIT. This alternative way to compute (8) is referred to as the adjoint or dummy-load method [30, p. 264]. Besides the reduction in work introduced by the adjoint method, a further reduction in computation can be obtained, as explained below.

The j th column vector of \mathbf{B}^T , i.e., $\mathbf{B}_j \in \mathbb{R}^{nd}$, is given by

$$\mathbf{B}_j = a(\mathbf{I}_j - \mathbf{I}_{j+1}), \quad j = 1 \dots ne - 1, \quad (10)$$

$$\mathbf{B}_{ne} = a(\mathbf{I}_1 - \mathbf{I}_{ne}),$$

where a is a given constant. Eq. (10) holds because $nt = ne$ and the same nodes, corresponding to the electrodes, are used to apply current and to measure potentials, as previously explained in the text. Based on Eqs. (2) and (10), one can write:

$$\mathbf{K}^{-1} \mathbf{B}_j = a \mathbf{K}^{-1} (\mathbf{I}_j - \mathbf{I}_{j+1})$$

$$= a (\mathbf{V}_j - \mathbf{V}_{j+1}), \quad j = 1 \dots ne - 1, \quad (11)$$

$$\mathbf{K}^{-1} \mathbf{B}_{ne} = a \mathbf{K}^{-1} (\mathbf{I}_1 - \mathbf{I}_{ne}) = a (\mathbf{V}_1 - \mathbf{V}_{ne}).$$

Therefore, we do not need to solve $\mathbf{K}^{-1} \mathbf{B}^T$ in (9) explicitly, but we only have to solve (2) for $j = 1 \dots ne$ at each SLP iteration in order to compute (8).¹ This represents a significant time reduction in

¹ Because the constant a in (11) can be relatively high, we must assure that a sufficiently small tolerance for the residual norm is used in the solution of the linear systems $\mathbf{K} \mathbf{V}_j = \mathbf{I}_j$ for $j = 1 \dots ne$. Otherwise, large errors in the gradient could arise and, consequently, SLP convergence problems might occur.

relation to the runtime reported in [1,16], for which (10) does not hold since compound electrodes are used [28].

The solutions of (2) for $j = 1 \dots ne$ are obtained by the solver described in the next section.

4. Krylov subspace recycling for symmetric matrices

The basis for our proposed, improved, algorithm is the MINRES method [31], a Krylov subspace, minimum residual, algorithm for solving linear systems of equations with symmetric (possibly indefinite) matrices; see also [19]. At each iteration, the three-term Lanczos recurrence generates a new vector, expanding the orthonormal basis of the Krylov subspace [20]. Then, an approximate solution in the Krylov subspace that minimizes the two-norm of the residual is obtained.

The recycling version of MINRES (RMINRES) approximates a low-dimensional invariant subspace associated with the smallest absolute eigenvalues while solving a linear system and uses this approximate invariant subspace in the solution of subsequent linear systems [21]. For RMINRES to be effective, subsequent matrices must have approximate invariant subspaces that are close. For nonsymmetric/non-Hermitian linear systems alternative methods with Krylov subspace recycling are available [23].

If the columns of the matrix $U \in \mathbb{R}^{nd \times r}$ provide a basis for the recycle space, where r is the dimension of the recycle space, then in the Lanczos recurrence, each new vector in the recurrence has to be orthogonalized against KU . In addition, at each iteration, we obtain an approximate solution that minimizes the residual in the new subspace spanned by the columns of U and the Lanczos vectors. As we iterate to solve the linear system, we periodically update the approximate invariant subspace for the next linear system. For the iterations between such updates (referred to as a cycle), we keep the Lanczos vectors, and at the end of the cycle we compute a new approximate invariant subspace using the harmonic Ritz vectors with respect to the space given by the direct sum of the previous approximate invariant subspace and the space spanned by the Lanczos vectors of the past cycle. The maximum length of a cycle (and hence the maximum number of Lanczos vectors kept to update an approximate invariant subspace), given by s , is provided by the user and must be tuned, as discussed in Section 7.1.2. The harmonic Ritz vectors are good choices since they yield good approximate eigenvectors of the linear system matrix [32].

The systems in (2) differ only with respect to the right-hand sides. In these cases, the RMINRES solver effectively reduces the number of iterations and the runtime of the linear solver. Furthermore, after solving ne systems, the approximate invariant subspace computed in the last cycle is recycled for the first linear system of the next SLP iteration. As the linear system matrices change slowly between SLP steps, using a recycled subspace also reduces the number of iterations and the runtime of the linear solver in these cases. In addition, since the changes in the matrix tend to diminish in the course of the optimization, the effectiveness of Krylov subspace recycling increases as the optimization converges.

The RMINRES algorithm considered in this work is discussed in detail in [21]. However, in Section 6.2, we describe several improvements to the original code that reduce the runtime of the solver significantly.

5. Preconditioning

The convergence rate of Krylov subspace methods for symmetric or Hermitian linear systems depends only on the eigenvalues and the decomposition of the right-hand side along the eigenvectors. Indeed, the condition number of the matrix, the ratio between

the absolute largest and smallest eigenvalues, governs an upper bound on the convergence rate. The larger the condition number, the slower the algorithm converges, in general.

The linear systems arising from EIT are ill-conditioned. We can significantly reduce the large number of (linear solver) iterations, due to the large condition number, by preconditioning. For the experiments in this paper we have used as preconditioner the incomplete Cholesky preconditioner with zero fill-in (IC(0)) for the matrix K [33]. Application of the IC(0) preconditioner leads to a significant reduction of iterations and computation time.

It is important to note that the preconditioner used in this work is a black-box preconditioner. This means that there are no parameters that need tuning, such as for the preconditioner proposed in [15].

6. Numerical implementation

A C/C++ code was developed in [21] in order to implement RMINRES. The solver was integrated in the open-source package Portable, Extensible Toolkit for Scientific Computation (PETSc), a suite of data structures and routines for the solution of scientific applications modeled by partial differential equations. The routines include several preconditioners and linear and non-linear equation solvers, which can be used in codes written in C. The preconditioner and the MINRES and CG solvers used in this work are implementations from PETSc. More details about PETSc can be found in [34].

6.1. Flowchart and numerical aspects

A simplified flowchart for the proposed estimation algorithm is shown in Fig. 2. In this figure, the main steps of the optimization software are given, including the solution of linear systems.

The figure also shows the recycled subspace defined by U_j , the solution V_j^* of (2) obtained by one of the iterative methods considered and used as an initial guess for the subsequent system (to reduce initial error), and the subspace defined by U_{ne} and the vector

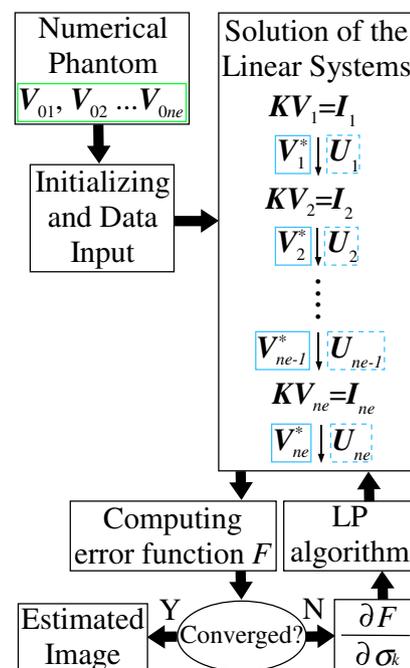


Fig. 2. Flow chart of the estimation process. If MINRES or CG is used to solve the linear systems, U_j (for $j = 1 \dots ne$) is not computed.

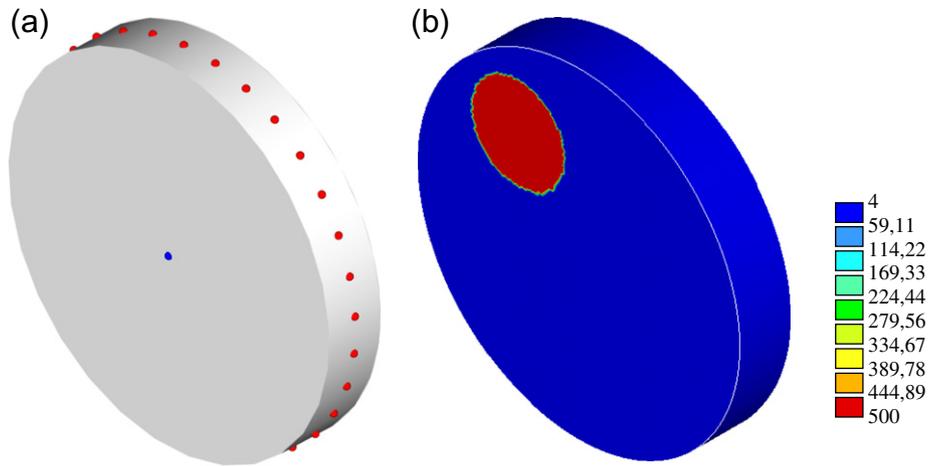


Fig. 3. (a) Schematic model, showing the point electrodes on the surface and a point on the axis of the cylinder whose potential is taken as ground; (b) resistivity distribution of the phantom. (units: Ωm).

\mathbf{V}_{ne}^* considered in the solution of $\mathbf{K}(\boldsymbol{\sigma})\mathbf{V}_1(\boldsymbol{\sigma}) = \mathbf{I}_1$ with a new set of optimization variables $\boldsymbol{\sigma}$ obtained through LP.

The LP step refers to the routine Simplex [25]. A routine from the GNU Project GNU Linear Programming Kit (GLPK) package is used in this case [35]. The GLPK package is intended for solving large-scale LP and other related problems. The routines are written in C and were compiled as a library.

The remaining routines in the image estimation process, such as those for the computation of sensitivities and for the preprocessing step, are implemented in C, and a main routine calls all the routines, including the PETSc and GLPK Simplex routines.

6.2. Improved RMINRES code

We now briefly outline several improvements to the RMINRES algorithm given in [21]. We have made the algorithm significantly more efficient in terms of floating point operations. These improvements are necessary because the matrix–vector products and the preconditioner steps for the EIT problem are much cheaper than for the structural design problems described in [21], and therefore we must reduce overhead.

The first improvement is in computing an orthonormal basis from the columns of the matrix \mathbf{KU} by a reduced QR decomposition: $\mathbf{CR}_c = (\mathbf{KU})$. To each orthogonalization of a new Lanczos vector against the matrix \mathbf{C} corresponds an update to the approximate solution of the type $\mathbf{u} = \mathbf{u} + \mathbf{K}^{-1}\mathbf{C}\mathbf{q} = \mathbf{u} + \mathbf{UR}_c^{-1}\mathbf{q}$. However, rather than updating the matrix $\mathbf{U} = \mathbf{UR}_c^{-1}$, such that $\mathbf{KU} = \mathbf{C}$ and simplifying the solution update, we use the original matrix \mathbf{U} and perform the required matrix–vector products as $\mathbf{U}(\mathbf{R}_c^{-1}\mathbf{q})$. In general, this is much more efficient. This is especially true in light of the second improvement. Since the approximate solution is not needed itself during the Lanczos iteration, we can postpone all updates with vectors of the type $\mathbf{U}\mathbf{q}$ until after the linear solve. This reduces four vector updates or `daxpy's` (multiplying a vector by a scalar and adding to another vector) for vectors of length nd to vector updates for vectors of length r (the number of columns of \mathbf{U}). We may have $nd = O(10^6)$ or $nd = O(10^5)$, whereas typically $r = 10$ or $r = 20$, and so this constitutes a significant reduction of computational work. The third improvement arises from the fact that we only need to compute a new, approximate, invariant subspace basis \mathbf{U}_j (at the end of cycle j) for the next linear system, and that the matrix \mathbf{C}_j or \mathbf{KU}_j is not needed itself for solving the current linear system. Therefore, the references to this matrix used in intermediate computations can be replaced by cheaper, alternative, recurrences.

The proposed improvements represent a time reduction of approximately 8% when comparing the original RMINRES code and the improved code, for the smaller mesh test problem discussed in the next section (observation or measurement noise is not considered in this particular case). We provide a detailed pseudocode for the improved RMINRES algorithm in the Appendix A.

7. Numerical results

The results are obtained on a PC with an Intel® Core™ 2 Quad Q6700 2.66 GHz processor, approximately 8 GB of RAM and the Ubuntu 9.04 (64-bit version) Linux system. The IC(0) preconditioner is used, as discussed in Section 5. Based on the work of Wang et al. [21], the performance of the improved RMINRES is studied and compared with other solvers.

On the surface of the model, thirty-two uniformly positioned nodes are regarded as point electrodes, and the electric potential in one point is taken as ground (that is, the potential in one node is equal to zero), as indicated in Fig. 3. Additionally, 32 different excitation patterns are used ($ne = 32$) and 30 electric potentials are measured for each distinct pattern ($nm = 30$), which means that 960 measurements are available. For each excitation pattern (i.e., for each value of j), one electrode is excited. The magnitude of the electric currents is equal to 0.001 A. Measurements of potential differences are taken from adjacent electrodes, as mentioned in Section 3.

The convergence criterion for the image estimation is that F in (3) is less than 4×10^{-6} . For the iterative solvers, the convergence tolerance for the residual norm is set to $rtol\|\mathbf{f}_j\|_2$ for the j th linear system, where the relative tolerance $rtol$ is equal to 10^{-10} . For all tests, the solution of the previous system is used as the initial guess of the next system, as mentioned in Section 6.1. The initial values of the optimization variables correspond to $4 \Omega\text{m}$.

We apply a simple regularization method that proved effective in [1]. It is based on an explicit spatial gradient control scheme, which avoids oscillations and thus enforces a certain amount of smoothness on the solution. For further detail, we refer to [36].

Table 1
FEM meshes MESH1 and MESH2.

	MESH1	MESH2
Number of elements	353,337	849,995
Number of nodes	64,862	152,226
Degrees of freedom	64,861	152,225

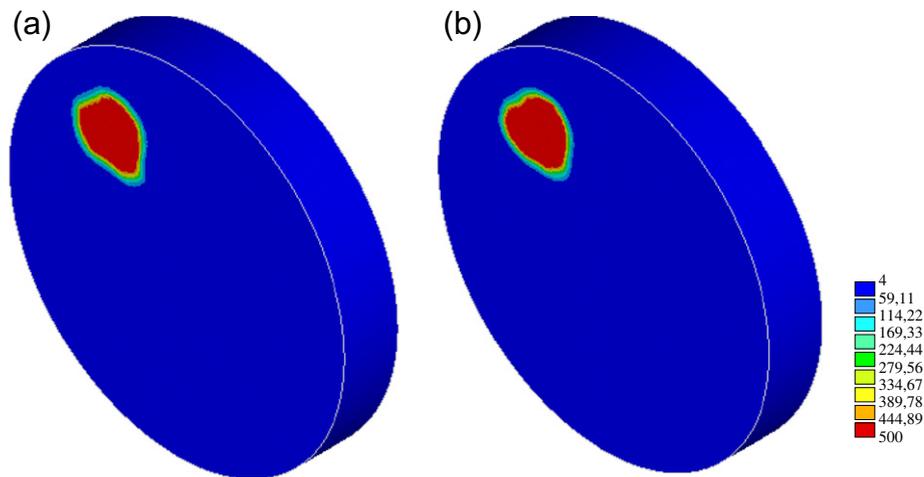


Fig. 4. Estimated images using MESH1 (a) and MESH2 (b). (units: Ωm).

To simulate the data, we consider a numerical phantom with 367,809 nodes, 2,061,992 elements, and 367,808 degrees of freedom. The number of degrees of freedom corresponds to the number of nodes minus the number of prescribed potentials, which is equal to one, here, because the electric potential in only one node is taken as ground. Zero-mean Gaussian measurement noise is added to simulated data. The standard deviation is 0.1% of the maximum measurement.

The image to be estimated is shown in Fig. 3. According to [37,38], the orders of magnitude of the resistivities are in the range found in the human thorax under normal conditions, for an excitation frequency of 125,000 Hz.

7.1. Test problems

We run two sets of test problems and use two discretizations. For the first set, we use MESH1 and for the second set MESH1 and MESH2. The number of elements, number of nodes, and the number of degrees of freedom for each mesh are shown in Table 1.

In the first set of test problems, we discuss the performance of the RMINRES algorithm. The parameters r , the dimension of the recycled subspace, and s , the maximum number of Lanczos vectors stored to periodically update the approximate invariant subspace (which is equal to the length of a cycle), are varied. In this case, RMINRES is referred to as RMINRES(s, r). In addition, we compare RMINRES with the corresponding conventional Krylov subspace method MINRES and the well-known (preconditioned) CG algorithm [20], the method of choice for symmetric and positive definite matrices that often arise in EIT. The number of iterations and time spent in solving the linear system (2) with $j = 1$ are depicted for each solver (see Section 7.1.1).

In the second set of test problems, we estimate the image using CG, MINRES and RMINRES(s, r). Then, we show the runtime of the image estimation for each method, and discuss the results.

In this paper, we do not consider the parallelization of the solution of (2) for $j = 1 \dots ne$. The maximum amount of allocated memory for the image estimation never exceeds 1400 MB and, therefore, swap memory is never required; only physical memory is used. The typical images obtained using the two meshes are shown in Fig. 4. The remaining results are discussed in the next subsections.

7.1.1. First set of test problems

The number of iterations and time spent in solving the linear system in (2) with $j = 1$ are shown in Fig. 5. First, it can be seen that

a maximum of 117 SLP steps were taken (for RMINRES(100,30), CG and MINRES), which means that 3744 (ne times 117) linear systems are solved in order to estimate the image in the worst case (if RMINRES(15,10) and RMINRES(100,10) are used, 116 SLP steps were taken; on the other hand, 115 steps were necessary using RMINRES(100,5)).

The graphs also show that recycling becomes more efficient towards the end of the optimization process. In addition, the runtimes demonstrate that RMINRES can already be more effective than MINRES and CG after a modest number of SLP steps, and that the effectiveness of RMINRES increases with r and s . However, the algorithm does not further reduce the solution time for values of r above a certain threshold, in spite of a further decrease in the number of iterations, as suggested by the results for RMINRES(100,30).

Experiments in [21] have suggested that s controls the accuracy of the approximation to the invariant subspace and, therefore, the effectiveness of recycling in reducing the number of iterations. The results for RMINRES($s, 10$) in Fig. 5(a) for SLP iterations between 22 and 106 corroborate this statement. In addition, the results in Fig. 5(b) show that reducing the number of iterations, nr , in general implies the reduction of the computation time. However, the number of Lanczos vectors computed during the iterative process is equal to the number of iterations to solve the linear system; therefore, since nr is often smaller than 100 (for $r = 10$), we can expect limited benefits for s beyond 100 in our tests (see [21]). For larger or more difficult problems, though, we can expect further improvements.

It was possible to detect values of r and s that rendered a superior performance with a few experiments. We suggest that experiments (such as the numerical tests carried out in the present subsection) should also be performed in practice before clinical implementation of the proposed solver.

7.1.2. Second set of test problems

The total runtimes of the estimation processes² for CG, MINRES and RMINRES are shown in Table 2.

The results show that RMINRES was faster than CG for several values of r and s used. Using the values that provided the best performance ($r = 10$ and $s = 100$), we see that the relative time

² Note, if the IC(0) preconditioner is not used, the image estimation for MESH1 and RMINRES(100,10) or CG takes more than 5 h. These results show that the preconditioner effectively reduces the computation runtime in spite of the overhead from the incomplete Cholesky decomposition and from preconditioning using the resulting incomplete Cholesky factors.

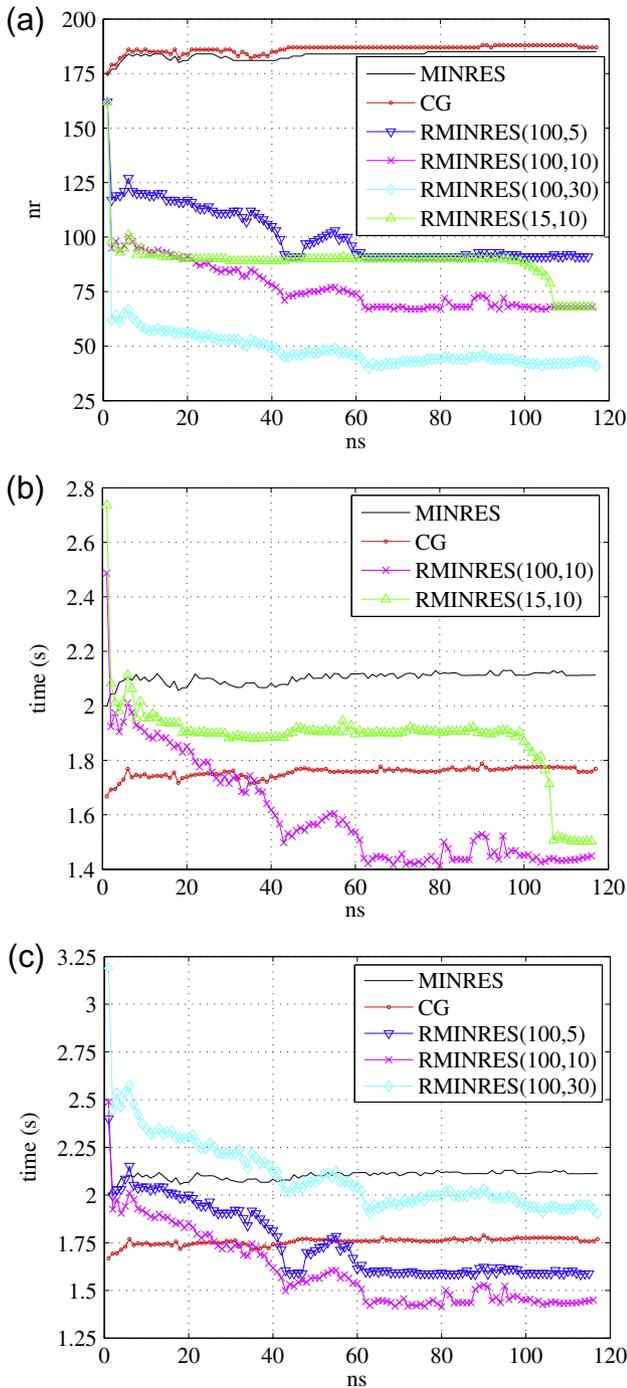


Fig. 5. (a) Number of iterations (nr) for CG, MINRES and RMINRES(s,r) for each SLP iteration (ns); (b) and (c) solution time for CG, MINRES and RMINRES(s,r) for each SLP iteration. All graphs correspond to the solution of $\mathbf{K}\mathbf{V}_1 = \mathbf{I}_1$.

Table 2
Estimation process runtime (in hours).

	MESH1	MESH2
MINRES	2.85	9.88
CG	2.47	8.42
RMINRES(15,10)	2.65	8.99
RMINRES(100,5)	2.30	7.18
RMINRES(100,10)	2.15	6.42
RMINRES(100,30)	2.62	7.23

difference increases with mesh refinement. In fact, the difference between the runtimes for CG and for RMINRES(100,10) represents approximately 13% of the runtime for CG when MESH1 is used, and 24% when MESH2 is used. Further tests must be carried out to determine if the relative solver performance increases further with mesh refinement or if this effect must be attributed to a better choice of the parameters r and s .

8. Concluding remarks and extensions

In this paper, we introduced Krylov subspace recycling in the context of EIT to reduce the computational cost used to obtain a 3D image. It was shown that for a suitable size of the recycled subspace and an appropriate length of the cycles, the RMINRES algorithm leads to a significant reduction in computation time for the solution of the forward problem. The results also show that RMINRES was faster than MINRES and CG. This suggests that RMINRES is the method of choice for large-scale EIT.

In future work, we intend to parallelize the solution of the finite element equations to speed up the optimization process even more. We also intend to consider other preconditioners, such as the algebraic multigrid preconditioner proposed in [15] or the multilevel preconditioner discussed in [24,39], and we will test an updating rule for the relative tolerance $rtol$, because the linear systems do not need to be solved accurately in the beginning of the SLP process. Finally, we will refine our model by implementing more elaborate electrode models, and we will test our algorithm using real experiments data. Further research is needed to analyze whether recycling an approximate invariant subspace corresponding to small eigenvalues is worthwhile for the estimation of contact parameters in more elaborate electrode models (we refer to [1,10] for further detail on the estimation of contact parameters).

Acknowledgments

The authors thank Shun Wang for the support provided regarding the RMINRES code. Luís Augusto Motta Mello thanks FAPESP (State of São Paulo Research Foundation) for his doctoral scholarship (Grant No. 2005/00270-1) and for the research project support (Grant No. 01/05303-4). Emílio Carlos Nelli Silva acknowledges the financial support of the National Council of Technological and Scientific Development, Grant No. 303689/2009-9. The research by Eric de Sturler was supported, in part, by the National Science Foundation under Grant Nos. DMR-0325939 and EAR-0530643. Glaucio H. Paulino acknowledges FAPESP for providing him the visiting scientist award at the University of São Paulo through Project No. 2008/51070-0.

Appendix A. Revised RMINRES code

We provide the revised RMINRES code below; however, first we briefly discuss some notational choices and other important issues. For brevity, we write $\hat{\mathbf{K}}$ for the preconditioned matrix $\mathbf{M}_1^{-1}\mathbf{K}\mathbf{M}_2^{-1}$, where $\hat{\mathbf{K}}$ is a symmetric (Hermitian) matrix. In practice, multiplication by the preconditioned matrix, $\mathbf{y} = \hat{\mathbf{K}}\mathbf{x}$, is carried out in three steps: (1) SOLVE $\mathbf{M}_2\mathbf{y}^{(1)} = \mathbf{x}$, (2) $\mathbf{y}^{(2)} = \mathbf{K}\mathbf{y}^{(1)}$, and (3) SOLVE $\mathbf{M}_1\mathbf{y} = \mathbf{y}^{(2)}$. By $[\]$ we denote an empty matrix. The scalars t_{ij} are coefficients of the tridiagonal matrix \mathbf{T} . The matrix \mathbf{G}_0 is a (2×2) Given's rotation, computed so that the second coefficient of the vector $\mathbf{G}_0[\gamma_0, t_{i+1,i}]^T$ is set to zero (see descriptions of the MINRES algorithm in [19]).

The scalar n indicates the number of degrees of freedom in the linear system, that is, number of rows of the matrix \mathbf{K} . The scalars n_Q and n_P denote the number of rows of \mathbf{Q} and \mathbf{P} , respectively. Two

subscript ranges indicate a subblock of a matrix, $\mathbf{F}^{g \times h}$ denotes the $g \times h$ matrix with ones on the diagonal and zeros elsewhere, $\mathbf{0}^{i \times j}$ denotes the $i \times j$ zero matrix, and \mathbf{I}_k denotes the $k \times k$ identity matrix.

The input variables are defined as follows. The vector \mathbf{u}_0 is an initial guess for the solution; \mathbf{U} is the recycle space, where an empty matrix indicates that no initial recycle space is provided; The matrices \mathbf{K} , \mathbf{M}_1 , and \mathbf{M}_2 are the linear system matrix and the left and right preconditioner, respectively; ϵ gives the relative convergence tolerance; \mathbf{b} is the right-hand side for the linear system; k is the dimension of the recycle space; ℓ_{\max} is the maximum number of Lanczos vectors kept for updating the approximate invariant subspace, which corresponds to the length of a cycle; and i_{\max} is the maximum number of linear solver iterations. The two output variables are the approximate solution \mathbf{u} and the new recycle space \mathbf{U}_j .

A few steps in the algorithm require some interpretation. For brevity, we write the orthogonalization of the new Lanczos vector against the matrix \mathbf{C} in two steps as $\mathbf{p} = \mathbf{C}^T \hat{\mathbf{r}}$; and $\hat{\mathbf{r}} = \hat{\mathbf{r}} - \mathbf{C}\mathbf{p}$. However, this should be implemented as a Modified Gram–Schmidt orthogonalization [40, p. 231]. All multiplications by inverse matrices, e.g., in $\mathbf{q} = \mathbf{R}_C^{-1} \mathbf{p}$, should be implemented by solving for the desired vector, typically using backward substitutions. However, for \mathbf{F}^{-1} , an LU decomposition and a forward and backward substitution is required. All QR decompositions should be so-called reduced QR decompositions [41, p. 49], which means that the decomposition $\mathbf{QR} = \mathbf{Z}^{n \times m}$ yields $\mathbf{Q}^{n \times m}$ and $\mathbf{R}^{m \times m}$. Finally, some triple matrix products, such as $\mathbf{L}_1 = \mathbf{Q}^T \mathbf{W}_2 \hat{\mathbf{P}}$, should be computed carefully to avoid unnecessary computational work.

Revised RMINRES

Input: \mathbf{u}_0 , \mathbf{U} (possible empty), \mathbf{K} , \mathbf{M}_1 , \mathbf{M}_2 , \mathbf{b} , ϵ , k , ℓ_{\max} , i_{\max}
Output: \mathbf{u} , \mathbf{U}_j

$\mathbf{r}_0 = \mathbf{b} - \mathbf{K}\mathbf{u}_0$; $\hat{\mathbf{r}}_0 = \mathbf{M}_1^{-1} \mathbf{r}_0$; $\hat{\mathbf{b}} = \mathbf{M}_1^{-1} \mathbf{b}$;

IF $\mathbf{U} \neq []$ THEN

$\tilde{\mathbf{C}} = \hat{\mathbf{K}}\mathbf{U}$; QR DECOMP: $\mathbf{C}\mathbf{R}_C = \tilde{\mathbf{C}}$;

$\mathbf{p} = \mathbf{C}^T \hat{\mathbf{r}}$; $\hat{\mathbf{r}} = \hat{\mathbf{r}} - \mathbf{C}\mathbf{p}$; $\mathbf{q} = \mathbf{R}_C^{-1} \mathbf{p}$;

IF $\|\hat{\mathbf{r}}\| \leq \epsilon \|\hat{\mathbf{b}}\|$ THEN

$\mathbf{u} = \mathbf{u}_0 + \mathbf{M}_2^{-1} \mathbf{U}\mathbf{q}$;

RETURN;

END

ELSE $\{\mathbf{U} = []\}$

$\mathbf{C} = []$; $\mathbf{q} = \mathbf{0}$;

END $\{\text{if } \mathbf{U} \neq []\}$

$\mathbf{V}_{1:n,1} = \hat{\mathbf{r}} / \|\hat{\mathbf{r}}\|$; $\zeta_1 = \|\hat{\mathbf{r}}\|$; $\zeta_2 = 0$;

$j = 1$; $i = 1$; $\ell = \ell_0 = \ell_1 = 1$;

WHILE $\|\hat{\mathbf{r}}\| > \epsilon \|\hat{\mathbf{b}}\|$ AND $i \leq i_{\max}$,

$\hat{\mathbf{v}} = \hat{\mathbf{K}}\mathbf{V}_{1:n,\ell_1}$; $\mathbf{d} = \mathbf{0}$;

IF $\mathbf{U} \neq []$ THEN

$\mathbf{B}_{1:k,\ell} = \mathbf{C}^T \hat{\mathbf{v}}$; $\hat{\mathbf{v}} = \hat{\mathbf{v}} - \mathbf{C}\mathbf{B}_{1:k,\ell}$; $\mathbf{d} = \mathbf{d} + \mathbf{R}_C^{-1} \mathbf{B}_{1:k,\ell}$;

END

IF $\ell_1 > 1$ THEN $\hat{\mathbf{v}} = \hat{\mathbf{v}} - t_{\ell_1-1,\ell} \mathbf{V}_{1:n,\ell_1-1}$; END

$t_{\ell_1,\ell} = \mathbf{V}_{1:n,\ell_1}^T \hat{\mathbf{v}}$; $\hat{\mathbf{v}} = \hat{\mathbf{v}} - t_{\ell_1,\ell} \mathbf{V}_{1:n,\ell_1}$;

$t_{\ell_1+1,\ell} = \|\hat{\mathbf{v}}\|$; $\mathbf{V}_{1:n,\ell_1+1} = \hat{\mathbf{v}} / t_{\ell_1+1,\ell}$; $t_{\ell_1,\ell+1} = t_{\ell_1+1,\ell}$;

IF $i > 2$ THEN

$[\gamma_2, \gamma_1]^T = \mathbf{G}_2[\mathbf{0}, t_{\ell_1-1,\ell}]^T$;

ELSEIF $i > 1$ THEN

$\gamma_1 = t_{\ell_1-1,\ell}$;

END

IF $i > 1$ THEN $[\gamma_1, \gamma_0]^T = \mathbf{G}_1[\gamma_1, t_{\ell_1,\ell}]^T$

Revised RMINRES (continued)

Revised RMINRES

ELSE $\gamma_0 = t_{\ell_1,\ell}$ END

COMPUTE $\mathbf{G}_0 : \mathbf{G}_0[\gamma_0, t_{\ell_1+1,\ell}]^T = [\hat{\star}, \mathbf{0}]^T$;

$[\zeta_1, \zeta_2]^T = \mathbf{G}_0[\zeta_1, \mathbf{0}]^T$;

$\mathbf{w} = \mathbf{V}_{1:n,\ell_1}$; $\mathbf{f} = \mathbf{d}$;

IF $i > 2$ THEN $\mathbf{w} = \mathbf{w} - \mathbf{w}_2 \gamma_2$; $\mathbf{f} = \mathbf{f} - \mathbf{f}_2 \gamma_2$; END

IF $i > 1$ THEN $\mathbf{w} = \mathbf{w} - \mathbf{w}_1 \gamma_1$; $\mathbf{f} = \mathbf{f} - \mathbf{f}_1 \gamma_1$; END

$\mathbf{w} = \mathbf{w} / \gamma_0$; $\mathbf{f} = \mathbf{f} / \gamma_0$;

$\mathbf{u} = \mathbf{u} + \mathbf{w} \zeta_1$;

IF $\mathbf{U} \neq []$ THEN $\mathbf{q} = \mathbf{q} - \mathbf{f} \zeta_1$; END

IF $\ell \geq \ell_{\max}$ OR $|\zeta_2| \leq \epsilon \|\hat{\mathbf{b}}\|$ THEN

UPDATE APPROXIMATE INVARIANT SUBSPACE

END

IF $i > 1$ THEN $\mathbf{G}_2 = \mathbf{G}_1$; $\mathbf{w}_2 = \mathbf{w}_1$; $\mathbf{f}_2 = \mathbf{f}_1$; END

$\mathbf{G}_1 = \mathbf{G}_0$; $\mathbf{w}_1 = \mathbf{w}$; $\mathbf{f}_1 = \mathbf{f}$; $\zeta_1 = \zeta_2$;

$\ell = \ell + 1$; $\ell_1 = \ell_1 + 1$;

$i = i + 1$;

END $\{\text{while } \|\hat{\mathbf{r}}\| > \epsilon \|\hat{\mathbf{b}}\| \text{ and } i \leq i_{\max}\}$

$i = i - 1$;

IF $\mathbf{U} \neq []$ THEN $\mathbf{u} = \mathbf{u} + \mathbf{U}\mathbf{q}$; END

$\mathbf{u} = \mathbf{u}_0 + \mathbf{M}_2^{-1} \mathbf{u}$;

The update of the approximate invariant subspace consists of two parts, one for the case that no (initial) recycle space is given and one for the case that \mathbf{U} is defined.

Update approximate invariant subspace

IF $\mathbf{U} = []$ THEN

UPDATE APPROXIMATE INVARIANT SUBSPACE \mathbf{U}_j

WITHOUT INITIAL RECYCLE SPACE \mathbf{U}

ELSE

UPDATE APPROXIMATE INVARIANT SUBSPACE \mathbf{U}_j

WITH RECYCLE SPACE \mathbf{U}

END

{START NEW CYCLE; KEEP LAST TWO LANCZOS VECTORS}

$j = j + 1$;

$\mathbf{V}_{1:n,1} = \mathbf{V}_{1:n,\ell_1}$; $\mathbf{V}_{1:n,2} = \mathbf{V}_{1:n,\ell_1+1}$;

$\alpha = t_{\ell_1+1,\ell}$; $\mathbf{T} = \mathbf{0}^{(\ell_{\max}+1) \times \ell_{\max}}$; $t_{1,1} = \alpha$;

$\ell_0 = 2$; $\ell = 0$; $\ell_1 = 1$;

IF $|\zeta_2| \leq \epsilon \|\hat{\mathbf{b}}\|$ THEN

IF $\mathbf{U} \neq []$ THEN $\mathbf{u} = \mathbf{u} + \mathbf{U}\mathbf{q}$; END

$\mathbf{u} = \mathbf{u}_0 + \mathbf{M}_2^{-1} \mathbf{u}$;

RETURN

END

Update approximate invariant subspace without initial recycle space

IF $j = 1$ THEN $\{\text{first cycle no } \mathbf{U}_j \text{ yet, } \ell_1 = \ell\}$

Compute harmonic Ritz vectors from $\mathbf{T}_{1:\ell+1,1:\ell}$ with k absolute smallest eigenvalues $\rightarrow \mathbf{P}^{\ell \times k}$;

QR DECOMP: $\mathbf{QR} = \mathbf{T}^{(\ell+1) \times \ell} \mathbf{P}$; $\hat{\mathbf{P}} = \mathbf{P}\mathbf{R}^{-1}$;

$\mathbf{U}_1 = \mathbf{V}_{1:n,1:\ell} \hat{\mathbf{P}}$;

$\mathbf{Q}_1 = \mathbf{Q}_{1:\ell-1,1:k}$; $\mathbf{Q}_2 = \mathbf{Q}_{\ell:\ell+1,1:k}$;

QR DECOMP: $\mathbf{Z}\hat{\mathbf{R}} = \mathbf{Q}_1$;

$n_Q = \ell + 1$; $n_P = \ell$;

Revised RMINRES (continued)

Update approximate invariant subspace without initial recycle space

$\mathbf{W}_2 = \mathbf{I}^{(\ell_1+1) \times \ell}$;
 ELSE
 $\mathbf{L}_1 = \mathbf{0}^{(\ell_1+1) \times k}$;
 $(\mathbf{L}_1)_{1,1:k} = \mathbf{Q}_{n_Q-1,1:k}$; $(\mathbf{L}_1)_{2,1:k} = \mathbf{Q}_{n_Q,1:k}$;
 $\mathbf{W}_1 = \begin{pmatrix} \mathbf{I}_k & \mathbf{L}_1^T \\ \mathbf{L}_1 & \mathbf{I}_{\ell_1+1} \end{pmatrix}$;
 $\mathbf{L}_1 = \mathbf{Q}^T \mathbf{W}_2 \widehat{\mathbf{P}}$;
 $\mathbf{L}_2 = \mathbf{0}^{k \times \ell}$; $(\mathbf{L}_2)_{1:k,1} = (\mathbf{Q}_{n_Q,1:k})^T$;
 $\mathbf{L}_3 = \mathbf{V}_{1:n,1:\ell_1+1}^T \mathbf{U}_{j-1}$;
 $\mathbf{L}_4 = [\mathbf{0}^{\ell \times 1} \mathbf{I}_\ell \mathbf{0}^{\ell \times 1}]^T$;
 $\mathbf{W}_2 = \begin{pmatrix} \mathbf{L}_1 & \mathbf{L}_2 \\ \mathbf{L}_3 & \mathbf{L}_4 \end{pmatrix}$;
 $\mathbf{H} = \begin{pmatrix} \mathbf{I}_k & \mathbf{0}^{k \times \ell} \\ \mathbf{0}^{\ell_1+1 \times k} & \mathbf{T}_{1:\ell_1+1,1:\ell} \end{pmatrix}$;
 SOLVE $(\mathbf{H}^T \mathbf{W}_1 \mathbf{H}) \mathbf{x} = \lambda (\mathbf{H}^T \mathbf{W}_2) \mathbf{x}$ for k eigenvectors with smallest eigenvalues $\rightarrow \mathbf{P}$;
 $\mathbf{F} = \mathbf{I}_{k+\ell_1+1}$; $\mathbf{F}_{1:k,1:k} = \widehat{\mathbf{R}}$; $\mathbf{F}_{k+1:k+2,1:k} = \mathbf{Q}_2$;
 QR DECOMP: $\mathbf{QR} = \mathbf{FHP}$;
 $\widehat{\mathbf{P}} = \mathbf{PR}^{-1}$;
 $\mathbf{U}_j = [\mathbf{U}_{j-1} \mathbf{V}_{1:n,2:\ell_1}] \widehat{\mathbf{P}}$;
 $\mathbf{Q}_1 = \mathbf{Q}_{1:k+\ell_1-1,1:k}$;
 $\mathbf{Q}_2 = \mathbf{Q}_{k+\ell_1:k+\ell_1+1,1:k}$;
 QR DECOMP: $\mathbf{Z}\widehat{\mathbf{R}} = \mathbf{Q}_1$;
 $\mathbf{Q} = \mathbf{F}^{-1} \mathbf{Q}$;
 $n_Q = k + \ell_1 + 1$; $n_P = k + \ell$;
 END {if $j = 1$ }

Update approximate invariant subspace with recycle space

IF $j = 1$ THEN
 $\mathbf{H} = \begin{pmatrix} \mathbf{I}_k & \mathbf{B}_{1:k,1:\ell} \\ \mathbf{0}^{\ell_1+1 \times k} & \mathbf{T}_{1:\ell_1+1,1:\ell} \end{pmatrix}$;
 $\mathbf{W}_1 = \mathbf{I}_{k+\ell_1+1}$;
 $\mathbf{\Gamma}_1 = (\mathbf{C}^T \mathbf{U}) \mathbf{R}_c^{-1}$; {right order for efficiency}
 $\mathbf{L}_2 = \mathbf{0}^{k \times \ell}$;
 $\mathbf{L}_3 = (\mathbf{V}_{1:n,1:\ell_1+1}^T \mathbf{U}) \mathbf{R}_c^{-1}$;
 $\mathbf{L}_4 = \mathbf{I}^{\ell_1+1 \times \ell}$;
 $\mathbf{W}_2 = \begin{pmatrix} \mathbf{\Gamma}_1 & \mathbf{L}_2 \\ \mathbf{L}_3 & \mathbf{L}_4 \end{pmatrix}$;
 SOLVE $(\mathbf{H}^T \mathbf{W}_1 \mathbf{H}) \mathbf{x} = \lambda (\mathbf{H}^T \mathbf{W}_2) \mathbf{x}$ for k eigenvectors with absolute smallest eigenvalues $\rightarrow \mathbf{P}$;
 QR DECOMP: $\mathbf{QR} = \mathbf{HP}$; $\widehat{\mathbf{P}} = \mathbf{PR}^{-1}$;
 $\mathbf{U}_j = \mathbf{U} (\mathbf{R}_c^{-1} \widehat{\mathbf{P}}_{1:k,1:k}) + \mathbf{V}_{1:n,1:\ell_1} \widehat{\mathbf{P}}_{k+1:k+\ell_1,1:k}$;
 $\mathbf{Q}_1 = \mathbf{Q}_{1:k,1:k}$;
 $\mathbf{Q}_2 = \mathbf{Q}_{k+1:k+\ell_1-1,1:k}$; $\mathbf{Q}_3 = \mathbf{Q}_{k+\ell_1:k+\ell_1+1,1:k}$;
 QR DECOMP: $\mathbf{Z}_2 \mathbf{S}_2 = \mathbf{Q}_2$;
 $n_Q = k + \ell_1 + 1$; $n_P = k + \ell$;
 ELSE { $j > 1$ }
 IF $j = 2$ THEN
 $\Phi_2 = [\mathbf{I}_k \mathbf{0}^{k \times (n_Q-k)}] \mathbf{Q}$;
 ELSE
 $\Phi_j = [\mathbf{I}_k \Phi_{j-1} \mathbf{0}^{k \times (n_Q-2k)}] \mathbf{Q}$;
 END

Revised RMINRES (continued)

Update approximate invariant subspace with recycle space

$\mathbf{L}_1 = \mathbf{0}^{(\ell_1+1) \times k}$;
 $(\mathbf{L}_1)_{1,1:k} = \mathbf{Q}_{n_Q-1,1:k}$; $(\mathbf{L}_1)_{2,1:k} = \mathbf{Q}_{n_Q,1:k}$;
 $\mathbf{W}_1 = \begin{pmatrix} \mathbf{I}_k & \Phi_j & \mathbf{0}^{k \times \ell_1+1} \\ \Phi_j^T & \mathbf{I}_k & \mathbf{L}_1^T \\ \mathbf{0}^{\ell_1+1 \times k} & \mathbf{L}_1 & \mathbf{I}_{\ell_1+1} \end{pmatrix}$;
 $\mathbf{\Gamma}_j = [\mathbf{\Gamma}_{j-1} \mathbf{0}^{k \times (n_P-k)}] \widehat{\mathbf{P}}$;
 $\mathbf{L}_1 = \mathbf{Q}^T \mathbf{W}_2 \widehat{\mathbf{P}}$;
 $\mathbf{L}_2 = \mathbf{0}^{k \times \ell}$; $(\mathbf{L}_2)_{1:k,1} = (\mathbf{Q}_{n_Q,1:k})^T$;
 $\mathbf{L}_3 = \mathbf{V}_{1:n,1:\ell_1+1}^T \mathbf{U}_{j-1}$;
 $\mathbf{L}_4 = [\mathbf{0}^{\ell \times 1} \mathbf{I}_\ell \mathbf{0}^{\ell \times 1}]^T$;
 $\mathbf{W}_2 = \begin{pmatrix} \mathbf{\Gamma}_j & \mathbf{0}^{k \times \ell} \\ \mathbf{L}_1 & \mathbf{L}_2 \\ \mathbf{L}_3 & \mathbf{L}_4 \end{pmatrix}$;
 $\mathbf{H} = \begin{pmatrix} \mathbf{0}^{k \times k} & \mathbf{B}_{1:k,1:\ell} \\ \mathbf{I}_k & \mathbf{0}^{k \times \ell} \\ \mathbf{0}^{\ell_1+1 \times k} & \mathbf{T}_{1:\ell_1+1,1:\ell} \end{pmatrix}$;
 SOLVE $(\mathbf{H}^T \mathbf{W}_1 \mathbf{H}) \mathbf{x} = \lambda (\mathbf{H}^T \mathbf{W}_2) \mathbf{x}$ for k eigenvectors with absolute smallest eigenvalues $\rightarrow \mathbf{P}$;
 $\mathbf{F} = \mathbf{I}_{2k+\ell_1+1}$; $\mathbf{F}_{1:k,k+1:2k} = \mathbf{Q}_1$;
 $\mathbf{F}_{k+1:2k,k+1:2k} = \mathbf{S}_2$;
 $\mathbf{F}_{2k+1:2k+2,k+1:2k} = \mathbf{Q}_3$;
 QR DECOMP: $\mathbf{QR} = \mathbf{FHP}$; $\widehat{\mathbf{P}} = \mathbf{PR}^{-1}$;
 $\mathbf{U}_j = [\mathbf{U}_{j-1} \mathbf{V}_{1:n,2:\ell_1}] \widehat{\mathbf{P}}$;
 $\mathbf{Q}_1 = \mathbf{Q}_{1:k,1:k}$; $\mathbf{Q}_2 = \mathbf{Q}_{k+1:2k+\ell_1-1,1:k}$;
 $\mathbf{Q}_3 = \mathbf{Q}_{2k+\ell_1:2k+\ell_1+1,1:k}$;
 QR DECOMP: $\mathbf{Z}_2 \mathbf{S}_2 = \mathbf{Q}_2$;
 $\mathbf{Q} = \mathbf{F}^{-1} \mathbf{Q}$;
 $n_Q = 2k + \ell_1 + 1$; $n_P = k + \ell$;
 END {if $j = 1$ }

References

- [1] L.A.M. Mello, C.R. De Lima, M.B.P. Amato, R.G. Lima, E.C.N. Silva, Three-dimensional electrical impedance tomography: a topology optimization approach, IEEE Trans. Biomed. Engrg. 55 (2) (2008) 531–540.
- [2] T.C. Hou, K.J. Loh, J.P. Lynch, Spatial conductivity mapping of carbon nanotube composite thin films by electrical impedance tomography for sensing applications, Nanotechnology 18 (2007) 1–9.
- [3] T. Vilhunen, L.M. Heikkinen, T. Savolainen, P.J. Vauhkonen, R. Lappalainen, J.P. Kaipio, M. Vauhkonen, Detection of faults in resistive coatings with an impedance-tomography-related approach, Meas. Sci. Technol. 13 (6) (2002) 865–872.
- [4] M. Lukaschewitsch, P. Maass, M. Pidcock, Tikhonov regularization for electrical impedance tomography on unbounded domains, Inverse Probl. 19 (3) (2003) 585–610.
- [5] M.H. Choi, T.J. Kao, D. Isaacson, G.J. Saulnier, J.C. Newell, A reconstruction algorithm for breast cancer imaging with electrical impedance tomography in mammography geometry, IEEE Trans. Biomed. Engrg. 54 (4) (2007) 700–710.
- [6] F.C. Trigo, R.G. Lima, M.B.P. Amato, Electrical impedance tomography using the extended Kalman filter, IEEE Trans. Biomed. Engrg. 51 (1) (2004) 72–81.
- [7] F.S. Moura, J.C.C. Aya, R.G. Lima, A.T. Fleury, Online transition matrix identification of the state evolution model for the extended Kalman filter in electrical impedance tomography, J. Phys. Conf. Ser. 124 (2008) 1–12.
- [8] E.L.V. Costa, C.N. Chaves, S. Gomes, M.A. Beraldo, M.S. Volpe, M.R. Tucci, I.A.L. Schettino, S.H. Bohm, C.R.R. Carvalho, H. Tanaka, R.G. Lima, M.B.P. Amato, Real-time detection of pneumothorax using electrical impedance tomography, Crit. Care Med. 36 (4) (2008) 1230–1238.
- [9] L.M. Heikkinen, T. Vilhunen, R.M. West, M. Vauhkonen, Simultaneous reconstruction of electrode contact impedances and internal electrical properties: II. Laboratory experiments, Meas. Sci. Technol. 13 (12) (2002) 1855–1861.
- [10] T. Vilhunen, J.P. Kaipio, P.J. Vauhkonen, T. Savolainen, M. Vauhkonen, Simultaneous reconstruction of electrode contact impedances and internal electrical properties: I. Theory, Meas. Sci. Technol. 13 (12) (2002) 1848–1854.

- [11] N. Polydorides, W.R.B. Lionheart, H. McCann, Krylov subspace iterative techniques: on the detection of brain activity with electrical impedance tomography, *IEEE Trans. Med. Imaging* 21 (6) (2002) 596–603.
- [12] P.J. Vauhkonen, Image Reconstruction in Three-Dimensional Electrical Impedance Tomography, Ph.D. Thesis, Department of Applied Physics, University of Kuopio, Finland, 2004.
- [13] H. Dehghani, N. Soni, R. Halter, A. Hartov, K.D. Paulsen, Excitation patterns in three-dimensional electrical impedance tomography, *Physiol. Meas.* 26 (2) (2005) S185–S197.
- [14] P.J. Vauhkonen, M. Vauhkonen, T. Savolainen, J.P. Kaipio, Three-dimensional electrical impedance tomography based on the complete electrode model, *IEEE Trans. Biomed. Engrg.* 46 (9) (1999) 1150–1160.
- [15] M. Soleimani, C.E. Powell, N. Polydorides, Improving the forward solver for the complete electrode model in eit using algebraic multigrid, *IEEE Trans. Med. Imaging* 24 (5) (2005) 577–583.
- [16] C.R. Lima, L.A.M. Mello, R.G. Lima, E.C.N. Silva, Electrical impedance tomography through constrained sequential linear programming: a topology optimization approach, *Meas. Sci. Technol.* 18 (9) (2007) 2847–2858.
- [17] N.S. Mera, L. Elliott, D.B. Ingham, Boundary identification for a 3d laplace equation using a genetic algorithm, *Comput. Meth. Appl. Mech. Engrg.* 194 (2005) 4411–4424.
- [18] N.S. Mera, L. Elliott, D.B. Ingham, The use of neural network approximation models to speed up the optimization process in electrical impedance tomography, *Comput. Meth. Appl. Mech. Engrg.* 197 (2007) 103–114.
- [19] H.A. van der Vorst, *Iterative Krylov Methods for Large Linear Systems*, Cambridge University Press, 2003.
- [20] Y. Saad, *Iterative Methods for Sparse Linear Systems*, second ed., Society for Industrial and Applied Mathematics, 2003.
- [21] S. Wang, E. de Sturler, G.H. Paulino, Large-scale topology optimization using preconditioned Krylov subspace methods with recycling, *Int. J. Numer. Meth. Engrg.* 69 (12) (2007) 2441–2468.
- [22] M.E. Kilmer, E. de Sturler, Recycling subspace information for diffuse optical tomography, *SIAM J. Sci. Comput.* 27 (6) (2006) 2140–2166.
- [23] M.L. Parks, E. de Sturler, G. Mackey, D.D. Johnson, S. Maiti, Recycling Krylov subspaces for sequences of linear systems, *SIAM J. Sci. Comput.* 28 (5) (2006) 1651–1674.
- [24] S. Wang, *Krylov Subspace Methods for Topology Optimization on Adaptive Meshes*, Ph.D. Thesis, University of Illinois at Urbana-Champaign, Urbana, Illinois, 2007.
- [25] W.H. Press, S.A. Teukolsky, W.T. Vetterling, B.P. Flannery, *Numerical Recipes in C – The Art of Scientific Computing*, Cambridge University Press, 1999.
- [26] J. Kaipio, E. Somersalo, *Statistical and Computational Inverse Problems*, Springer Science + Business Media, Inc., New York, 2005.
- [27] K.J. Bathe, *Finite Element Procedures*, Prentice-Hall, Englewood Cliffs, NJ, 1996.
- [28] P. Hua, E.J. Woo, J.G. Webster, W.J. Tompkins, Finite element modeling of electrode-skin contact impedance in electrical impedance tomography, *IEEE Trans. Biomed. Engrg.* 40 (4) (1993) 335–343.
- [29] V. Kolehmainen, M. Lassas, P. Ola, Electrical impedance tomography problem with inaccurately known boundary and contact impedances, *IEEE Trans. Med. Imaging* 27 (10) (2008) 1404–1414.
- [30] R.T. Haftka, Z. Gürdal, *Elements of Structural Optimization*, third ed., Kluwer Academic Publishers, Netherlands, 1999.
- [31] C.C. Paige, M.A. Saunders, Solution of sparse indefinite systems of linear equations, *SIAM J. Numer. Anal.* 12 (1975) 617–629.
- [32] C.C. Paige, B.N. Parlett, H.A. van der Vorst, Approximate solutions and eigenvalue bounds from Krylov subspaces, *Numer. Linear Algebr. Appl.* 2 (2) (1995) 115–133.
- [33] J. Meijerink, H.A. van der Vorst, An iterative solution method for linear equations systems of which the coefficient matrix is a symmetric M -matrix, *Math. Comput.* 31 (1977) 148–162.
- [34] S. Balay, K. Buschelman, W.D. Gropp, D. Kaushik, M.G. Knepley, L.C. McInnes, B.F. Smith, H. Zhang, *PETSc Web Page*. <<http://www.mcs.anl.gov/petsc/>>, 2001.
- [35] *GLPK Web Page*. <<http://www.gnu.org/software/glpk/>>, 2008.
- [36] J.K. Guest, J.H. Prévost, T. Belytschko, Achieving minimum length scale in topology optimization using nodal design variables and projection functions, *Int. J. Numer. Meth. Engrg.* 61 (2004) 238–254.
- [37] S. Gabriel, R.W. Lau, C. Gabriel, The dielectric properties of biological tissues: II. Measurements in the frequency range 10 Hz to 20 GHz, *Phys. Med. Biol.* 41 (1996) 2251–2269.
- [38] C. Gabriel, D. Andreuccetti, R. Fossi, C. Petrucci, Dielectric Properties of Body Tissues. <<http://niremf.ifac.cnr.it/tissprop/>>.
- [39] S. Wang, E. de Sturler, Multilevel sparse approximate inverse preconditioners for adaptive mesh refinement, *Linear Algebr. Appl.* 431 (3–4) (2009) 409–426.
- [40] G.H. Golub, C.F. Van Loan, *Matrix Computations*, third ed., John Hopkins University Press, 1996.
- [41] L.N. Trefethen, D. Bau III, *Numerical Linear Algebra*, SIAM, 1997.