Towards Efficient Iterative Absolute EIT

Henry FJ Tregidgo[1,2], Michael G Crabb[1], William RB Lionheart[1]

[1]University of Manchester, Manchester, UK.

[2]henry.tregidgo@manchester.ac.uk

Abstract: One factor limiting the use of absolute reconstructions in 3D lung EIT is the computational cost of iterative algorithms. We show how the programming experience of the Finite Element and Research Software Engineering communities can be applied to these algorithms, resulting in a speed up of reconstructions in EIDORS 3.8 [1]. We also outline a combination of absolute and difference imaging to provide fast pseudo-absolute imaging.

1 Introduction

In situations where 3D absolute EIT could be useful the high cost of iterative inversion can be prohibitive. A large proportion of time taken is in the solution of a sparse linear system representing the forward problem [2]. We present simple software refinements to improve performance of this linear solve in Matlab. We also address refinements in construction of the Jacobian, coarse-to-fine and Laplace-prior matrices.

For purposes such as parameter fitting and control of ventilation, absolute reconstruction is only required for calibration [3]. For situations where calibration of absolute values is required at multiple time steps, we present a combination of absolute and difference imaging to provide a fast pseudo-absolute reconstruction.

2 Reconstruction Steps

Errors in floating point arithmetic can produce small asymmetries even when using seemingly symmetric constructions of the system matrix. For example the factorisation of the FEM system matrix for piecewise linear elements into mesh and conductivity dependent components [4] as performed in EIDORS. This asymmetry, shown in Figure 1, can cause checks in default sparse linear solvers to incorrectly choose slower non-symmetric algortithms. In particular the UMFPACK algorithm is chosen over CHOLMOD by MATLAB's `mldivide` function, resulting in worse performance.

We demonstrate how symmetry correction can drastically reduce the sparse linear solve time as shown in Table 1. We additionally detail other software engineering refinements to speed up construction of the Jacobian, coarse-to-fine and Laplace-prior matrices. These include vectorisation of operations, replacement of `find` operations with `sort` functions and reduction of complexity through further use of symmetry.

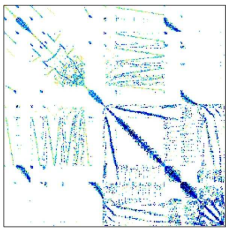

Figure 1: Sparsity pattern for the antisymmetric system matrix error component using first order elements. Entries are of order 10^{-12}.

3 Pseudo-absolute Reconstruction

We propose using additional imaging modalities to produce a segmented mesh of the thorax [5, 6] and performing a very low dimensional absolute reconstruction of a single frame on this mesh. The absolute values are then incorporated into the conductivity Jacobian for further difference imaging. This results in improved residual data-fit and only requires a small additional offline processing time. Using the 46k node mesh from Table 1 for a simulated domain with 64 electrodes and 5 level set regions required an additional 2 minutes elapsed time with the improvements from the previous section.

4 Conclusions

By ensuring conditions are met for the use of the optimal linear solvers, and reducing the dimensionality of the iterative inversion, significant speed-ups are available and pseudo-absolute reconstructions are possible.

5 Acknowledgements

The authors would like to thank A. Boyle and A. Adler for their help in confirming these efficiency savings in EIDORS.

References

[1] Adler A, Boyle A, Crabb MG, et al. In *Proc. 16th Int. Conf. Biomed. Applications of EIT*. 2015

[2] Boyle A, Borsic A, Adler A. *Physiol Meas* **33**(5):787, 2012

[3] Tregidgo HFJ, Crabb MG, Lionheart W. In *Proc. 16th Int. Conf. Biomed. Applications of EIT*, 96. 2015

[4] Vavasis SA. *SIAM J on Numerical Analysis* **33**(3):890–916, 1996

[5] Crabb M, Davidson J, Little R, et al. *Physiol Meas* **35**(5):863, 2014

[6] Grychtol B, Lionheart WR, Bodenstein M, et al. *Medical Imaging, IEEE Transactions on* **31**(9):1754–1760, 2012

Table 1: Time comparisons for symmetric and non-symmetric linear solve on two meshes of the same thorax segmentation. Timings are given in both CPU and elapsed time as measured on a 2.8GHz Intel Core i7 with 16 GB 1.6 GHz DDR3 RAM.

N. nodes	Unsymmetric (CPU time s)	Symmetric (CPU time s)	Speedup (CPU time ×)	Unsymmetric (elapsed time s)	Symmetric (elapsed time s)	Speedup (elapsed time ×)
46k	9.83	3.47	2.83	6.55	1.09	6.03
190k	281.06	91.45	3.07	146.77	28.01	5.24