

Using ParExpPDE for the numerical solution of bioelectric field problems

Christoph Freundl*

Friedrich-Alexander-Universität Erlangen-Nürnberg
Christoph.Freundl@informatik.uni-erlangen.de

Harald Köstler*

Friedrich-Alexander-Universität Erlangen-Nürnberg
Harald.Koestler@informatik.uni-erlangen.de

Abstract

The library *Parallel Expression Templates for Partial Differential Equations (ParExpPDE)* is designed to be a convenient and efficient tool for writing numerical solvers for problems that involve partial differential equations. ParExpPDE achieves this goal by using an advanced C++ programming technique, so-called *Expression Templates*. This technique allows to produce highly efficient code when it comes to arithmetic operations on large matrices and vectors. The high efficiency is obtained by evaluating all arithmetic expressions at compile time. Furthermore, Expression Templates are very convenient to use, for a user of the library can stay close to the mathematical language in the implementation.

The ParExpPDE library has been under development for two years and has recently reached a state that makes the library eligible for the solution of real applications. For this reason, we investigate in this paper the implementation of numerical solvers for bioelectric field problems, more precisely the reconstruction of electrical sources within the human brain from EEG data. A part of the reconstruction, which eventually provides a solution for this inverse problem, consists of solving several forward problems. We present some implementation details of a multigrid algorithm for the solution of the forward problem along with some numerical results.

1 Introduction

The reconstruction of the bioelectric field in the human brain from non-invasive measurements like electroencephalographic recordings (EEG) has the potential to become a powerful tool in neurology ([Sch98]). From a mathematical point of view, the reconstruction can be considered as an inverse problem, which can be solved by repeated numerical simulations of the potential distribution, that we call forward problem, for assumed dipolar current sources in the brain. An accurate reconstruction of the electrical brain activity involves the consideration of a realistic head model, which is ideally obtained from registered Computer Tomography (CT) and Magnetic Resonance (MR) images. Furthermore,

*Institut für Informatik, Lehrstuhl für Systemsimulation, D-91058 Erlangen, Germany

with the advent of Diffusion Tensor Magnetic Resonance Imaging (DT-MRI), an estimation of the anisotropic conductivities within the brain has been made available, which allows a further, patient specific, refinement of the realistic head model ([Van01],[MV03],[WKAR02],[Max04],[Moh04]).

The sources to be localised during the inverse problem and to be modeled in the forward problem are electrolytic currents within the dendrites of the large pyramidal cells of activated neurons in the cortex sheet of the human brain. The resulting primary current is generally formulated as a mathematical dipole

$$\vec{j}^p(\vec{x}) = \vec{M}\delta(\vec{x} - \vec{x}_0) \quad (1)$$

with the moment \vec{M} and the position \vec{x}_0 .

The total current distribution is modeled as

$$\vec{j} = \vec{j}^p + \sigma \vec{E} \quad , \quad (2)$$

where σ is a conductivity tensor. The fields are quasistatic and therefore the electric field \vec{E} can be expressed as the negative gradient of a scalar potential u (cf. [No196])

$$\vec{j} = \vec{j}^p + \sigma \nabla u \quad . \quad (3)$$

Because the divergence of \vec{j} must be zero, we arrive at the quasistatic Maxwell's equations of electrodynamics in the domain $\Omega \subset \mathbb{R}^3$ with Neumann boundary conditions

$$\begin{aligned} \nabla \cdot (\sigma \nabla u) &= \nabla \cdot \vec{j}^p & \text{in } \Omega \\ \sigma \frac{\partial u}{\partial \vec{n}}|_{\Gamma} &= 0 & \text{on } \Gamma \quad . \end{aligned} \quad (4)$$

The direct application of variational and FE techniques to equation (4) yields with meshsize h , the number of unknowns $N_h \in \mathcal{O}(h^{-3})$ as h tends to zero and for an FE-basis $\mathbb{V}_h = \text{span}\{\psi_i\}_{i=1}^{N_h}$ a system of linear equations

$$K_h u_h = J_h \quad , \quad (5)$$

where $K_h \in \mathbb{R}^{N_h \times N_h}$ is the stiffness matrix with entries

$$K_h^{[i,j]} = \int_{\Omega} \nabla \psi_j \sigma \nabla \psi_i d\Omega \quad . \quad (6)$$

The right hand side $J_h \in \mathbb{R}^{N_h}$ depends on the chosen dipole model and will be discussed in the following section.

2 Zenger mathematical dipole model

The Zenger mathematical dipole model (ZMD) uses the mathematical dipole (1) and assumes that the dipole is located inside an element and not on its surface. In that case we

can use after multiplication with a test function v and integration over the domain Ω Greens formula on the right hand side of equation (4)

$$\int_{\Omega} v \nabla \cdot \vec{j}^p d\Omega = \int_{\Gamma} v \frac{\partial \vec{j}^p}{\partial \vec{n}} d\Gamma - \int_{\Omega} (\nabla v) \vec{j}^p d\Omega . \quad (7)$$

The surface integral vanishes because of the assumption that there is no source current on the surface of the conductor

$$\frac{\partial \vec{j}^p}{\partial \vec{n}} = 0 \quad \text{on } \Gamma . \quad (8)$$

After inserting the mathematical dipole (1) and using the above FE-basis ∇_h we get the right hand side entries

$$[J_h]^i = \int_{\Omega} (\nabla \psi_i) \vec{M} \delta(\vec{x} - \vec{x}_0) d\Omega = \vec{M} \cdot \nabla \psi_i(\vec{x}_0) . \quad (9)$$

3 Implementation in ParExpPDE

3.1 Setup of the conductivity tensor and the differential operator

Differential operators within ParExpPDE ([FP03]) are discretised using Finite Elements with linear basis functions. Therefore, the conductivity has to be described in terms of its constant values within each element. Within ParExpPDE, we have to store the values in a so-called *cell variable*.

```
pxdVariable<double>* sigma = new pxdVariable<double>
( partition.template createCellVariable<double>() );
```

In case of isotropic conductivity which is constant in the whole domain, we can just assign the conductivity's value to the variable.

```
*sigma = sigma_value;
```

If the conductivity is not constant, each component of the variable has to be set individually. For head models based on CT or MR images, each voxel of the image can be mapped to an element; the corresponding component of the variable is assigned the matching conductivity value.

Once the cell variable is properly initialised, the differential operator representing “ $\nabla \sigma \nabla$ ” can be constructed:

```
pxdVariableDifferentialOperator
<pxdVariableLaplaceOperatorFunction, double> sigma_laplace( sigma );
```

3.2 Setup of the right-hand side

The dipole is modeled with the Zenger mathematical dipole model as described in section 2. The computation of the right-hand side uses the COLSAMM library by J. Härdtlein [HP05] which is also used within ParExpPDE for the implementation of differential operators.

3.3 Multigrid solver

Once the operator and the right-hand side are set up, we can apply some standard multigrid V-cycles to solve the problem. The first step in a single V-cycle is to restrict the problem down to the coarsest level.

```
for (int level = 0; level < number_levels-1; level++) {
  for (int smooth = 0; smooth < presmooth; smooth++) {
    // pre-smoothing with Gauss-Seidel
    u = u - (sigma_laplace(u) + f) /
           pxdDiag(sigma_laplace) | interior_points;
  }

  r = f + sigma_laplace(u); // calculate residual
  r = 0.0 | boundary_points; // set residual to zero on boundary
  r.doRestrict();           // restrict to coarser level

  f.levelDown();           // go down one level with right-hand
  f = r;                   // side; initialise with residual
  u.levelDown();           // go down one level with solution
  u = 0.0;                 // initialise with zero
}
```

On the coarsest level the problem can be solved exactly as its size is now usually so small that the computational effort for a direct solver is feasible. ParExpPDE makes use of the SuperLU_DIST library [LD03].

Finally, the solution has to be interpolated up to the finest level.

```
for (int level = number_levels-2; level >= 0; level--) {
  f.levelUp();             // go up one level with right-hand side
  r = u;                   // the residual on the finer level is
  r.doProlongate();        // the solution on the coarser level

  u.levelUp();             // go up one level with the solution
  u = u + r | interior_points; // apply correction

  for (int smooth = 0; smooth < postsmooth; smooth++) {
    // post-smoothing with Gauss-Seidel
    u = u - (sigma_laplace(u) + f) /
           pxdDiag(sigma_laplace) | interior_points;
  }
}
```

V-Cycle	Residual	Convergence rate
0	$1.56832 \cdot 10^{-1}$	
1	$1.82064 \cdot 10^{-3}$	0.011609
2	$7.36400 \cdot 10^{-5}$	0.040447
3	$4.35923 \cdot 10^{-6}$	0.059197
4	$2.83094 \cdot 10^{-7}$	0.064941
5	$1.92223 \cdot 10^{-8}$	0.067901
6	$1.31355 \cdot 10^{-9}$	0.068335
7	$8.99673 \cdot 10^{-11}$	0.068492
8	$6.13385 \cdot 10^{-12}$	0.068179
9	$4.17525 \cdot 10^{-13}$	0.068069
10	$2.82517 \cdot 10^{-14}$	0.067665

Table 1: Residuals and convergence rates for the model problem with $\sigma_0 = 1$ and the dipole placed at $\vec{x}_0 = (0.507812, 0.507812, 0.507812)^T$ and direction $\vec{M} = (1, 1, 1)^T$, discretised on a 64^3 grid and solved with V(2,1)-cycles.

4 Results

4.1 Model problem

We consider a single dipole with moment \vec{M} placed at the point \vec{x}_0 within the unit cube. The conductivity is isotropic and constant with value σ_0 . The analytical solution for the resulting electrical field is

$$u(\vec{x}) = -\frac{\vec{M} \cdot (\vec{x} - \vec{x}_0)}{4\pi\sigma_0|\vec{x} - \vec{x}_0|^3}$$

Table 1 shows numerical results of the implementation.

4.2 Realistic head data

Some qualitative results with realistic head data by courtesy of C. Wolters, University Münster, can be seen in Figure 1. The data consists of an MR image of the brain which is segmented into different regions (skin, skull, brain matter etc.). These are mapped to different conductivity values as described in section 3.1.

References

- [FP03] C. Freundl and C. Pflaum. Parexpde web page, 2003. <http://www10.informatik.uni-erlangen.de/Research/Projects/ParEXPDE/>.
- [HP05] J. Härdtlein and C. Pflaum. Efficient and user-friendly computation of local stiffness matrices. In *Proceedings 18th Symposium on Simulation Technique, ASIM 2005*, 2005. To appear.

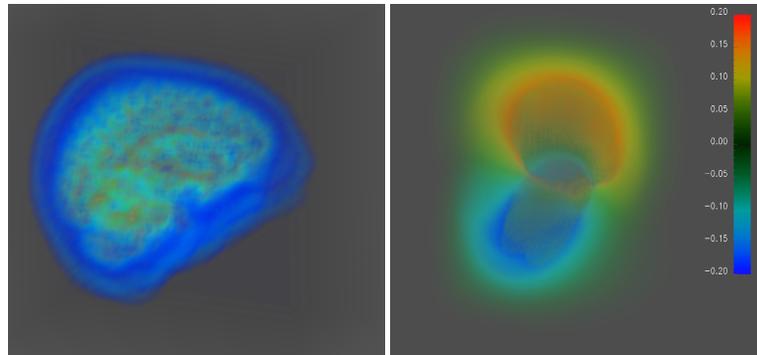


Figure 1: Solution of the bioelectric field problem for a sample brain. On the left you can see the different regions of the brain, on the right is the resulting electric field from a dipole lying in the center of the brain

- [LD03] Xiaoye S. Li and James W. Demmel. SuperLU_DIST: A scalable distributed-memory sparse direct solver for unsymmetric linear systems. *ACM Trans. Mathematical Software*, 29(2):110–140, June 2003.
- [Max04] Max Planck Institute for Human Cognitive and Brain Sciences. Neurofem web page, 2004. <http://www.neurofem.com>.
- [Moh04] M. Mohr. *Simulation of Bioelectric Fields: The Forward and Inverse Problem of Electro-encephalographic Source Analysis*. PhD thesis, Friedrich-Alexander Universität Erlangen-Nürnberg, 2004.
- [MV03] M. Mohr and B. Vanrumste. Comparing iterative solvers for linear systems associated with the finite difference discretisation of the forward problem in electroencephalographic source analysis. *Mathematical and Biological Engineering and Computing*, 41(1):75–84, 2003.
- [Nol96] W. Nolting. *Theoretische Physik 3: Elektrodynamik*. Zimmermann-Neufang Verlag, 4th edition, 1996.
- [Sch98] N. Schaul. The fundamental neural mechanisms of electroencephalography. *Electroencephalography and Clinical Neurophysiology*, 106:101–107, 1998.
- [Van01] B. Vanrumste. *EEG Dipole Source Analysis in a Realistic Head Model*. PhD thesis, Faculteit Toegepaste Wetenschappen, Universiteit Gent, 2001.
- [WKAR02] C. H. Wolters, M. Kuhn, A. Anwanger, and S. Reitzinger. A parallel algebraic multigrid solver for finite element method based source localization in the human brain. *Computing and visualization in science*, 5(3):165–177, 2002.