SpezialForschungsBereich F 32

Karl–Franzens Universität Graz
Technische Universität Graz
Medizinische Universität Graz

Towards Highspeed Diffuse Optical Tomography on Graphics Hardware

M. Freiberger H. Egger M. Liebmann

H. Scharfetter

SFB-Report No. 2010–018 April 2010

A–8010 GRAZ, HEINRICHSTRASSE 36, AUSTRIA

Supported by the
Austrian Science Fund (FWF)

Der Wissenschaftsfonds.

SFB sponsors:

- **Austrian Science Fund** (FWF)
- **University of Graz**
- **Graz University of Technology**
- **Medical University of Graz**
- **Government of Styria**
- **City of Graz**

Towards Highspeed Diffuse Optical Tomography on Graphics Hardware

Manuel Freiberger[1], Herbert Egger[2], Manfred Liebmann[2], Hermann Scharfetter[1]
[1]Graz University of Technology, Institute of Medical Engineering, Graz, Austria
[2]University of Graz, Institute for Mathematics and Scientific Computing, Graz, Austria

Abstract

Diffuse optical tomography is a functional imaging technique based on propagation and absorption of light in biological tissues. We investigate the fast solution of the governing partial differential equations and the nonlinear inverse reconstruction problem on graphics hardware. In particular, we discuss the discretization by finite element methods, the efficient numerical solution by block conjugate gradient methods and multigrid preconditioners. As our numerical tests on a simulation phantom demonstrate, a drastic speed-up of reconstruciton times can be achieved by utilizing the compute power of modern graphics hardware instead of convential computers. With typical solution times in the order of two seconds, our implementation allows in principle for online reconstructions.

1 Introduction

Diffuse optical tomography (DOT) is a non-invasive imaging modality that seeks to reconstruct optical parameters of an object like a small animal or the human head from boundary measurements of light, which is injected at the surface and back-scattered inside the sample. As the optical properties are often related to the biological environment, the method is capable of functional imaging rather than just providing information about the anatomy. One of the main applications is near-infrared spectroscopy in the human brain. There, the absorption coefficient of the tissue, which in turn is related to oxygenation and provides a measure for brain activity.

In medical applications, fast image reconstruction is a key issue for a successful use of the method, but on the other hand, there is also demand for high quality images, which for optical diffusion tomography requires the solution of complicated, non-linear reconstruction problems.

Over the last couple of years, graphics processing units (GPU) have been established as high performance compute kernels, which allow (at least in theory) to achieve hundred fold operation counts compareed to standard computers. A proper implementation of computationally expensive algorithms on graphics hardware can therfore result in a significant speed-up of computation times, see, e.g., [1, 2, 3].

In the current work, we discuss the efficient reconstruction of the absorption coefficient from DOT data. The governing partial differential equations are discretized by finite elements, and the resulting systems are solved by block-conjugate gradient methods and multigrid preconditioners. Moreover, iterative regularization methods are utilized for the solution of the inverse reconstruction problem. Except of some initialization steps, the full reconstruciton is performed on the graphics hardware.

2 Methods

2.1 Forward model

A widely used model for light propagation in highly scattering media, like biologicial tissue, is the diffusion approximation:

$$-\nabla \cdot (\kappa \nabla \varphi) + \mu_a \varphi = q, \qquad \text{in } \Omega, \tag{1}$$

$$\varrho \varphi + \kappa \frac{\partial \varphi}{\partial n} = 0, \qquad \text{on } \partial\Omega, \tag{2}$$

where $\kappa = 1/(3(\mu_s' + \mu_a))$, μ_a and μ_s' are the diffusion, the absoprtion and the reduced scattering coefficients of the tissue, φ is the photon density, q is a source term and ϱ is the reflection coefficient at the boundary. Ω is the computation domain, $\partial\Omega$ its boundary and n the outward-pointing unit normal of the surface. For a detailed derivation of the equations, we refer to [4].

A measurement of the d-th detector consists of photons leaving the domain Ω at the detector location, i.e.

$$m_d = -\int_{\partial\Omega} w\kappa \frac{\partial \phi}{\partial n} ds \overset{(2)}{=} \int_{\partial\Omega} w\varrho\varphi ds \tag{3}$$

where the function w defines the location and the physical properties of the detector. In a typical DOT experiment, several sources are placed on the surface of the object which are switched on sequentially while all photodetectors measure at the same time. Thus, a full set of measurements consists of $N_D \times N_S$ measurements of the form (3), where N_S is the number of sources and N_D the number of detectors.

Via (1)–(3), the matrix of measurement M depends on the optical parameters. For ease of presentation, we assume here that all parameters except the absorption coefficient

are known a-priori. Hence, this dependence can be expressed as $M = F(\mu_a)$, where F denotes the *forward operator*, i.e., the mathematical model for the experiment.

2.2 Inverse problem

The inverse problem of optical tomography is to reconstruct the absorption coefficient μ_a from (possibly perturbed) measurements M^δ, i.e., to solve $F(\mu_a) = M^\delta$. Since the inverse problem is ill-posed, we utilize Tikhonov regularization for its stable solution, i.e., we aim at minimizing the functional $J(\mu_a) := \frac{1}{2}\|F(\mu_a) - M^\delta\|^2 + \alpha\|\mu_a - \mu_a^0\|^2$, where $\alpha > 0$ is a regularization parameter, and μ_a^0 is some a-priori guess for the solution. The minimization task can be realized via Gauß-Newton schemes leading to the iterations of the form

$$\begin{aligned}\left(F'^*(\mu_a^k)F'(\mu_a^k) + \alpha I\right)\Delta\mu_a^k \\ = F'^*(\mu_a^k)\left(M^\delta - F(\mu_a)\right) + \alpha\left(\mu_a^0 - \mu_a^k\right).\end{aligned} \tag{4}$$

Here F' is the derivative of the forward operator F, F'^* is the adjoint derivative, and $\Delta\mu_a^k = \mu_a^{k+1} - \mu_a^k$ is the update for the unknown absorption coefficient in the k-th iteration.

2.3 Discretization

After discretization by finite elements, (1)–(2) can be written as a linear system of the form

$$S\,V := [K + G(\mu_a) + R]\,V = Q. \tag{5}$$

The matrices K. $G(\mu_a)$ and R result from integration with the finite element basis functions χ_i, i.e.

$$[K]_{ij} = \int_\Omega \kappa\nabla\chi_i\nabla\chi_j dx, \quad [G]_{ij} = \int_\Omega \mu_a\chi_i\chi_j dx$$
$$[R]_{ij} = \int_{\partial\Omega} \varrho\chi_i\chi_j dx, \quad [Q]_{is} = \int_\Omega \chi_i q_s dx.$$

The s-th column of Q corresponds to the s-th light source q_s, and the s-th column of V is the corresponding photon density field. The measurement matrix is then defined by $M = DV$, where each row of D corresponds to one of the detectors.

The code for solving the inverse problem utilizing the Gauß-Newton method (4) is sketched in Algorithm 1.

The linear system (5) is positive definite and symmetric, and is solved with a *block conjugate gradient method*, together with a *multigrid preconditioner*. Since in our numerical tests, blocking the conjugate geradient method had a higher impact on performance than the acceleration by multigrid, we describe the former method in Algorithm 2, and refer to [5, 6] for details on multigrid methods.

Algorithm 1 Gauß-Newton reconstruction algorithm

1: Load mesh and measurements M^δ
2: Initialize μ_a^0, Q, D, ϵ
3: **repeat**
4: $S = \text{assembleSystemMatrices}(\mu_a^k)$
5: Solve $SV = Q$ for V
6: Compute $M = DV$
7: Solve $S^H W = D(M^\delta - M)$ for W
8: $J = \text{assembleSensitivityMatrix}(\mu_a^k, V, W)$
9: Solve $(J^T J + \alpha^k I)\Delta\mu_a = J^T(M^\delta - M) + \alpha I(\mu_a^0 - \mu_a^k)$ for $\Delta\mu_a$
10: Update $\mu_a^{k+1} = \mu_a^k + \Delta\mu_a$
11: **until** $\|M^\delta - M\| < \epsilon$

Algorithm 2 Preconditioned Block Conjugate Gradient (PBCG) algorithm for solving $AX = B$

1: $D = B$; $W = \text{pre}(D)$; $S = W$; $\rho = (W : D)$; $\rho_0 = \rho$; $X = 0$;
2: **for** $i = 1, 2, \ldots$ **do**
3: $AS = A \cdot S$;
4: $\alpha = \rho/(AS : S)$;
5: $X = X + \alpha \cdot S$;
6: $D = D - \alpha \cdot AS$;
7: $W = \text{pre}(D)$;
8: $\rho^* = \rho$; $\rho = (W : D)$;
9: **if** $\rho <$ tol, **break**;
10: $\beta = \rho/\rho^*$;
11: $S = W + \beta \cdot S$;
12: **end for**

$W : D$ is the Euclidian scalar product of two matrices which is defined as $\sum_{i,j} W_{i,j} D_{i,j}$.
$\text{pre}(D)$ denotes the application of the preconditioner to a matrix D.

3 Results

In the following, we present results of numerical tests, that demonstrate the potential speed up that can be achieved by utilizing graphics hardware instead of standard computers for solving the inverse problem of optical diffuse optical tomography. All computations are performed on an AMD Phenom 9950 desktop computer with a Nvidia GTX 280 graphics card.

The computational domain used for our simulations is a cylinder with dimensiones of a small animal, i.e., a diameter of 30 mm and a height of 60 mm. The sources and detectors are placed on three rings separated by 10 mm with 8 sources and 8 detectors on each ring resulting in a total of 24 source optodes and 24 detector optodes. In order to guarantee an accurate approximation of the photon field equations (1)–(2), we utilized a relatively fine tetrahedral mesh with 233,984 elements. The mesh generation was done in a preprocessing step using NETGEN [7].

Table 1 lists the computation time required for the individual steps of Algorithm 1, i.e., the assembly of the system matrix, the simulation of the forward problem to compute the residual between the simulation and the measurement, the solution of the adjoint problem, the assembly of the sensitivity matrix and the computation of the update of the

Table 1: Timing for different tasks during the reconstruction on a 233,984 element mesh. The last column is the ratio of CPU time to GPU time.

Task	CPU	GPU	Speed-up
	ms	ms	
Local matrix assembly			
Level 1	n.a.	0.01	n.a.
Level 2	n.a.	0.03	n.a.
Level 3	n.a.	0.08	n.a.
Level 4	n.a.	0.53	n.a.
Global matrix assembly			
Level 1	0.06	0.22	0.26
Level 2	0.44	0.25	1.73
Level 3	3.99	0.88	4.55
Level 4	76.94	5.71	13.46
Forward solution	6.98×10^3	317.26	22.00
Adjoint solution	7.48×10^3	311.11	24.06
Measurement computation	718.95	8.08	89.01
Sensitivity assembly	4.71×10^3	916.43	5.14
Gauß-Newton update [1]	167.80×10^3	678.86	247.18

[1]The CPU computation could be sped up using an optimized BLAS. See text for details.

fluorophore distribution.

The assembly of the system matrix is done in two steps on graphics hardware. First the local element matrices are computed and stored temporarily. Then the local matrices are assembled into the large global matrix. This two-step procedure is necessary because the GPU does not provide means for efficient random write access to its memory. There is no such constraint on orindary desktop computers is why the system matrix is assembled in a single step there.

The speed-up that can be realized for the individual steps, e.g., the assembly of the global system matrix on the coarsest multigrid level is relatively slow on the GPU since the graphics hardware has some latency when starting the threads which consumes more time than the actual computation. However, on the finer levels, which requires the most computation time, the graphics processor yields substantial speed-up.

The most significant speed-up can be achieved for solution of the Gauß-Newton system (4). This system is symmetric and positive definite, and can be solved the conjugate gradient method, which only requires multiplication with the Jacobian J and ist transpose J^T and the regularization matrix I. In particular, the multiplication with the Jacobian, which is a dense matrix here, can be realized with high efficiency on graphics hardware. We expect that this step could be accelerated also on the CPU by using highly optimized BLAS routines.

Compared to the plain conjugate gradient method (CG), the block CG decreases the computation time by 50 %. The use of the multigrid preconditioner with four levels, where the number of finite elements is decreased by a factor of 8 on each level, shows an additional decrease by 30 %. Thus, the preconditioned block CG algorithm is faster by a factor of around 3.2 compared to standard CG without preconditioner.

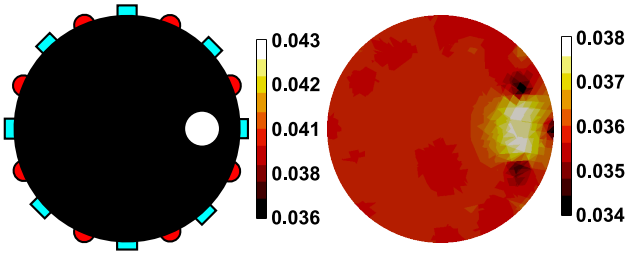

Figure 1: Transversal central slice of the 3D simulation phantom (left) and the reconstruction of the absorption coefficient performed on the GPU (right). Red balls mark the location of sources and blue boxes the one of detectors. The absorption is given in mm^{-1}.

Figure 1 shows an example reconstruction of a spherical perturbation performed on the graphics card. According to [8], the background scattering was set to $\mu_s' = 0.275\,\mathrm{mm}^{-1}$ and the absoprtion to $\mu_a = 0.036\,\mathrm{mm}^{-1}$. Inside the perturbation the absorption was increased by 20 %. The reflection coefficient was chosen to be $\varrho = 0.2$ [9].

The reconstruction exhibits the typical characteristics of near-field imaging techniques like DOT: The originally sharp edges of the sphere are smeared and the amplitude of the reconstructed absorption is too small due to the increased volume of the perturbation.

4 Discussion

It was shown that the use of low cost graphics hardware can speed up time consuming tasks of the inverse problem of diffuse optical tomography by a factor of at least 20. The increased computation power can either be used to reduce the reconstruction time or to solve more complex—possibly non-linear—problems during the same time.

A combination of highly efficient methods as the multigrid preconditioner and the block conjugate gradient method together with graphics hardware is considered to be a major step towards online reconstruction. If the change in the absoprtion coefficient is small between two consecutive images, a single Gauß-Newton step will already yield good results. Right now this update can be computed in $2.3\,\mathrm{s}$ and can most likely be performed even faster if the implementation is tuned a bit.

A drawback of current graphics hardware is the limitation of floating-point accuracy to 32 bit. Preliminary tests show that this amounts to around $1\,\%$ error in the reconstruction compared to double precision floating-point results computed on the CPU. However, this limitation will eventually fall with the event of a new generation of graphics cards during this year.

The second major problem is the limited memory of graphics hardware. For example, the size of the sensitivity matrix is number of measurements ($24 \times 24 = 576$) times number of finite elements and uses $514\,\mathrm{MB}$ of graphics memory in our test setup. To mitigate this problem, a cluster of graphics cards could be used which would also increase the computation power.

In future work, we would like to investigate the solution of more complex models like fluorescence tomography, for example, which is described by two coupled partial differential equations in its simplest form. Also the incorportation of non-linear regularization strategies like total-variation regularization is of interest as these methods can enhance the quality of the reconstructed images. Finally, the algorithm has to be validated on real measurement data and implemented in an online reconstruction system.

Acknowledgement

This work was supported through project F3207-N18 granted by the Austrian Science Fund.

References

[1] D. Göddeke, C. Becker, and S. Turek, "Integrating GPUs as fast co–processors into the parallel fe package FEAST," in *19th Symposium Simulationstechnique*, ser. Frontiers in Simulation, M. Becker and H. Szczerbicka, Eds. SCS Publishing House e.V., 2006, pp. 277–282.

[2] W. Wu and P. A. Heng, "A hybrid condensed finite element model with gpu acceleration for interactive 3d soft tissue cutting: Research articles," *Comput. Animat. Virtual Worlds*, vol. 15, no. 3-4, pp. 219–227, 2004.

[3] Z. Taylor, O. Comas, M. Cheng, J. Passenger, D. Hawkes, D. Atkinson, and S. Ourselin, "On modelling of anisotropic viscoelasticity for soft tissue simulation: Numerical solution and gpu execution," *Medical Image Analysis*, vol. 13, no. 2, pp. 234 – 244, 2009.

[4] S. R. Arridge, "Optical tomography in medical imaging," *Inverse Problems*, vol. 15, pp. R41–R93, 1999.

[5] J. H. Bramble, *Multigrid Methods.*, ser. Number 294 in Pitman research notes in mathematics series. Harlow, UK: Longman Scientific & Technical, 1993.

[6] W. Hackbusch, *Multigrid methods and application*. Berlin, Heidelberg, New York: Springer Verlag, 1985.

[7] J. Schöberl, "Netgen - an advancing front 2d/3d-mesh generator based on abstract rules," *Computing and Visualization in Science*, vol. 1, no. 1, pp. 41–52, 1997.

[8] G. Alexandrakis, F. R. Rannou, and A. F. Chatziioannou, "Tomographic bioluminescence imaging by use of a combined optical-pet (opet) system: a computer simulation feasibility study," *Physics in Medicine and Biology*, vol. 50, pp. 4225–4241, 2005.

[9] M. Keijzer, W. M. Star, and P. R. M. Storchi, "Optical diffusion in layered media," *Applied Optics*, vol. 27, pp. 1820–1824, 1988.