



ONLINE DISTRIBUTED SOURCE LOCALIZATION FROM EEG/MEG DATA

Christof Pieloth ¹⁾, Thomas R. Knösche ²⁾, Burkhard Maess ²⁾, Mirco Fuchs ^{1,2)}

¹⁾ Laboratory for Biosignal Processing, Leipzig University of Applied Sciences, Germany,
www.labp.htwk-leipzig.de

²⁾ Max Planck Institute for Human Cognitive and Brain Sciences, Leipzig, Germany,
www.cbs.mpg.de

Abstract: Electroencephalography (EEG) and Magnetoencephalography (MEG) provide insight into neuronal processes in the brain in a real-time scale. This renders these modalities particularly interesting for online analysis methods, e.g. to visualize brain activity in real-time. Brain activity can be modeled in terms of a source distribution found by solving the bioelectromagnetic inverse problem, e.g. using linear source reconstruction methods. Such methods are particularly suitable to be used on modern highly parallel processing systems, such as widely available graphic processing units (GPUs). We present a system that, according to its modular scheme, can be configured in a very flexible way using graphical building blocks. Different preprocessing algorithms together with a linear source reconstruction method can be used for online analysis. The algorithms use both CPU and GPU resources. We tested our system in a simulation and in a realistic experiment. *Copyright © Research Institute for Intelligent Computer Systems, 2014. All rights reserved.*

Keywords: EEG, MEG, Source Imaging, GPGPU.

1. INTRODUCTION

The functioning of the brain is linked to biochemical and biophysical processes between interacting neurons and neuronal populations. This interaction is strongly related to current flows. Synchronous activity in a neuronal population, i.e. a large number of neurons concurrently respond to an excitatory input, produces currents which are strong enough to be detected. This can be done either on the head surface by measuring potentials between attached electrodes using Electroencephalography (EEG), or by measuring the magnetic flux density using Magnetoencephalography (MEG). In contrast to imaging techniques, e.g. functional magnetic resonance imaging (fMRI) or positron emission tomography (PET), which are essentially related to metabolism, EEG/MEG provides insight into brain processes in a real-time scale. To exploit the advantage of high time resolution, techniques for online processing of such signals are particularly interesting.

An important discipline in EEG/MEG analysis is source localization, i.e. inferring the current density which might have produced the measured data. This, however, is an ill-posed problem. Different approaches to solve this so called bioelectromagnetic inverse problem have been

presented in literature, see [1] for an overview of this topic. A very common method is distributed source reconstruction, where a spatial current distribution is estimated from the data. In this approach, a large number of equivalent current dipoles (ECD), each of which has a fixed orientation and location, densely cover the source space. Usually, this is the cortical sheet which is approximated using a triangulated surface. Given a model of the volume conductor, which describes the conductivity profile inside the head, a so called leadfield vector can be calculated for each source, which relates its activity to a spatial pattern at sensor level. Based on this linear forward model, this approach provides the possibility for linear inverse solutions such as the popular minimum norm solution [2, 3], which actually minimizes the L2-norm of weighted current strengths.

Linear problems such as the mentioned EEG/MEG source localization technique can easily be parallelized, which renders them particularly suitable for online processing using modern high performance computing hardware such as general purpose graphic processing units (GPGPUs). Today, such hardware can easily be utilized using NVIDIA's CUDA technology which, among others, provides a comfortable way to use graphic processing units for computationally demanding problems. However, the

practical realization of a system capable for the online reconstruction of brain activity from electromagnetic signals requires much more than only the parallelization of some algorithm. First, a system concept including an efficient data structure used to share information between processing steps is strictly necessary. Second, the processing pipeline needs to be flexible enough to be adopted and parameterized during measurement. Third, additional algorithms need to be provided to sufficiently prepare EEG/MEG data in real-time for source reconstruction. Fourth, other information, e.g. at least a general forward model, needs to be provided in advance to do source reconstruction.

To exploit the fact that EEG/MEG data provides insight into fast neuronal processes, we developed a software framework which principally allows online processing and visualization of brain activity in a real-time scale. According to our best knowledge, this framework is conceptually different from other solutions which provide online processing capabilities, e.g. [4, 5, 6, 7], because we aim to provide source localization using a high density source space (more than 250.000 ECDs) from high channel EEG/MEG data. Without loss of generality, we focused on the analysis of evoked brain responses in the time domain, i.e. activity which occurs due to sensory, auditory or visual stimulation. Based on simulations we have shown previously that our solution holds real-time requirement which are important for online processing and visualization [8].

Here, we introduce important extensions of the software which are essentially needed for practical applications. This includes a physical link to a real measurement device and a solution for the preparation of a forward model on demand. These extension allowed us to explore our framework under practical considerations using a simple experimental paradigm.

The paper is structured as follows. First, we will discuss some issues related to the online analysis of EEG/MEG signals. Second, the general concept of our framework will be introduced. Moreover, we will give a short explanation on each of the implemented modules, i.e. algorithms. Third, we describe how the system's online capabilities were tested. Fourth, we will demonstrate a practical application of the software in a realistic environment. Finally, we will discuss and conclude our results.

2. CHALLENGES FOR ONLINE SOURCE RECONSTRUCTION

In contrast to offline processing, online localization of brain activity depends on two

important prerequisites. First, EEG/MEG data needs to be preprocessed in real-time. Second, a forward model is already required during the measurement whose calculation, however, partially depends on information only available after measurement setup (see below).

Signal preprocessing requires computational performance which competes with source localization resources. However, the quality of source localization highly depends on well preprocessed data, in which noise and artifacts are widely reduced. Noise can be distinguished in technical noise of the measurement system and neuronal noise from other brain regions, e.g. task-unrelated brain activity. To account for noise, filter techniques and signal averaging can be applied. The latter is only possible for phase related activity, such as evoked responses. Artifacts are either caused by strong electromagnetic activity emerging from external brain regions, e.g. eye blinks, heartbeat and even nearby passing trains, or result from defective sensors. Artifacts need to be detected and removed from the data. Sometimes, a correction rather than a rejection is possible. Due to the different types and characteristics of artifacts this is particularly challenging. Some artifacts can easily be detected and removed online, e.g. using a threshold based approach. For others, a universal and reliable automated detection is computationally intensive and, therefore, difficult to be realized in online systems [9, 10].

The calculation of a forward model to be used in an online framework is difficult in several aspects. First, the calculation requires a head model and the exact sensor positions. The former includes a description of the source space and of the volume conductor. The use of an individual head model would be optimal, but involves the recording of other modalities, e.g. MRI data, before the EEG/MEG measurement. Often, this is not available. The use of a standard head model is a feasible alternative. Second, the calculation of the forward model, i.e. in our case the leadfield matrix¹, is computationally expensive and takes a long time, in particular when a source space with high spatial resolution is used. But since it depends on the sensor positions, it cannot be prepared in advance. The solution to this problem is different for EEG and MEG.

For EEG, the sensor positions depend on the placement of electrodes on the head. Therefore, it is

¹The Leadfield matrix describes the forward solution, i.e. the impact from each source on each sensor. It is computed by physical modeling of the electromagnetic fields in the head tissue.

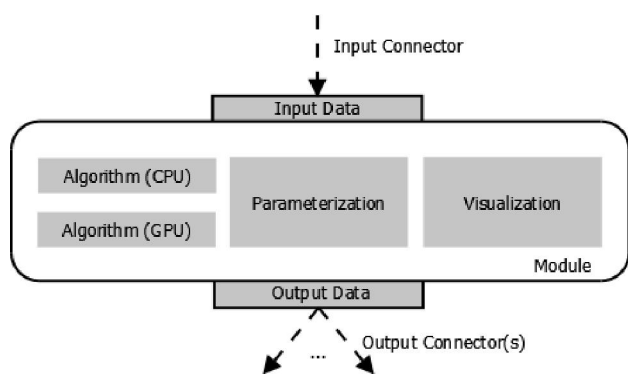


Fig. 1 - Abstract structure of a module in OpenWalnut. Gray shaded blocks represent functions that are provided by each module.

possible to place a dense grid of virtual sensors on the head surface and use them to calculate the leadfield matrix. When the true sensor positions are known, their leadfield can be derived from the virtual sensors, e.g. based on interpolation[11].

For MEG, the sensor positions depend on the relative position of the head in the measurement device. Thus, the forward model can be calculated under the assumption that the head is centered in the device. This can actually be achieved by applying head movement detection and correction algorithms to the recorded MEG data.

3. SOFTWARE CONCEPT AND ARCHITECTURE

The basic idea of our concept is to split the signal processing chain into separate functional units, i.e. modules that can be put together. Each module is realized in terms of a prototype that becomes part of a processing chain. The module parameters can be changed during measurement. This principle allows to set up and reuse modules easily, even several times during the same measurement. The module's intended functionality is reflected by a certain algorithm. Based on NVIDIA's CUDA technology, each module can separately use the graphic processing unit for high performance tasks. This allows an easy extension by new and efficient algorithms, which can either be executed on the CPU or on the GPU. Moreover, each module holds a separate visualization instance.

The possibility for a flexible organization of the signal processing chain requires a hybrid data structure consisting of static and dynamic areas which is shared among all modules at run time. This structure provides access to all data that might be used by any module. For example, the static part of this structure contains measurement parameters, e.g. EEG sensor positions, but also the shape of the source space. The dynamic part actually contains raw or transformed (filtered) sensor data, localized source time courses, or might be used differently.

The modular concept allows signal branching, i.e. to set up signal chains using a tree structure. Such a configuration could be used to test different parameterizations of a certain algorithm or to realize different source localizations algorithms concurrently.

The implementation of our concept is based on *OpenWalnut*, a software for multi-modal brain visualization [12, 13]. This platform also follows a strict modular concept. The modules have so called input and output connectors to interact with other modules. These connectors can have any data type, only outputs and inputs that are directly connected need to share the same data structure. Whenever a module updates data at its output connector, connected modules are scheduled to allow seamless processing. Besides this useful architecture, *OpenWalnut* provides an intuitive mechanism to select modules and put them together by means of graphical building blocks.

According to our concept and the architecture of *OpenWalnut*, the structure of a module is summarized in Fig. 1. It is worth to note that each module is responsible for the visualization of its results. On the one hand, this provides a very high flexibility and allows a relatively easy independent implementation because of non-existing dependencies or information from other modules. On the other hand, already a few sophisticated active visualizations instances can become computationally intractable and, besides that, drastically increase the complexity of the user front-end. Therefore visualization can be deactivated manually.

A brief overview of our data structure is depicted in Fig. 2. The main class for all modules is *EMMeasurement*, which allows to maintain any information which could be needed for EEG/MEG

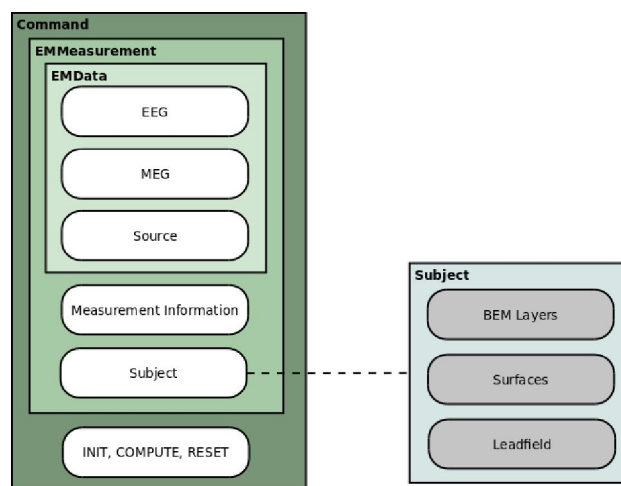


Fig. 2—Shared command and data structure. Modules can be connected based on the EMMeasurement class, which contains any sub-data that may be needed. EMData holds dynamic data, all other parts contain static information, e.g. individual surfaces.

online analysis. Within this class, other structures are nested, e.g. *EMData*, *Measurement Information* and *Subject*. While the *EMData* holds dynamic data, i.e. it contains raw and processed data, the other parts contain static data. Given this structure, the modules interact on the basis of commands. The *INIT* command forces a module to initialize all its required data structures. The *RESET* command forces a module to reset all data structures. *EMData* is computed by the module's algorithm on receiving the *COMPUTE* command. Note, the modular concept implies that data is processed in a block-wise manner.

4. MODULES AND ALGORITHMS

Based on this concept the open source toolbox *NA-Online*² was developed. It extends OpenWalnut by the following modules: *MNE Realtime Client*, *FIFF Reader/Writer*, *EMM Simulator*, *Alignment*, *Leadfield Interpolation*, *FIR Filter*, *Epoch Separation*, *Epoch Averaging* and *Source Reconstruction*. Their functions and purposes are introduced briefly:

(1) *MNE Realtime Client*: based on MNE software [14], it provides a physical link to a NeuromagVectorView System (Elekta, Helsinki, Finland).

(2) *FIFF Reader/Writer*: Reading and writing files using the FIFF format. It can read a previously recorded measurement or save a running measurement to disk.

(3) *EMM Simulator*: this module allows streaming of a recorded measurement for simulation purposes.

(4) *(Coordinate) Alignment*: co-registration between EEG/MEG device coordinate system and head model coordinate system, which is needed to calculate a leadfield matrix (see below) and for visualization. We implemented a semi-automatic coordinate transformation. The procedure depends on the manual labeling of three fiducial points (location of nasion, left ear, and right ear) on the skin surface in the head model before the measurement, and the digitization of these points with the EEG/MEG system. The subsequent transformation is based on iterative closet point algorithm [15]. The coordinate transformation is done only once, after digitizing fiducials and electrode positions.

(5) *Leadfield Interpolation*: its purpose is to provide the EEG leadfield matrix for the digitized sensor positions before online processing. Before a measurement, a generic leadfield matrix is computed using the MNE toolbox [14] by placing virtual EEG electrodes on the skin surface in the head model.

The virtual electrode positions are derived by randomly selecting approximately 1.000 nodes from the triangulated head surface (usually consisting of more than 5.000 nodes). Note that the resulting leadfield matrix is already about 1 GB in size. During measurement setup, the true sensor positions are digitized. For each real electrode, we identify the four nearest virtual electrodes in the close vicinity. Then, linear interpolation scheme is applied. We preferred this naive linear approach over others, e.g. spline based interpolation [11], because it allows a fast and efficient computation. The leadfield information of each virtual sensor is weighted according the reciprocal value of its euclidean distance to the true electrode. Then, the leadfield at the true sensor position can be estimated by adding up the weighted leadfield vectors.

(6) *FIR Filter*: we implemented time domain filtering and provide lowpass, highpass, bandpass and bandstop characteristic. Filtering can either be done on CPU or on the GPU.

(7) *Epoch Separation*: Activity in the brain is evoked by presenting stimulus material. It is then particularly interesting to analyze this activity when a stimulus is applied. This module splits the continuous data stream according to stimulus presentation information into single epochs, i.e. data frames that range from a time point before to a point after stimulus onset. Only extracted epochs are passed to further modules.

(8) *Epoch Averaging*: Responses to an identical stimulus are expected to be phase related and very similar. This allows to apply averaging to increase the signal to noise ratio. This module implements total average and moving average. The former uses all epochs detected during a measurement to calculate an averaged evoked response. The latter only uses a certain number of the last epochs that were detected.

(9) *Source Reconstruction*: this module provides the concurrent linear reconstruction of distributed sources of all sampling points in an epoch, e.g. the average evoked response. We implemented the weighted minimum norm method [2, 3]. This algorithm requires information about the signal to noise ratio, which still has to be estimated manually. Source reconstruction is implemented for CPU and GPU. The module provides a view of the reconstructed source time courses and a 3D shape of the source space. By selecting a time point in the signal view, the corresponding spatial distribution pattern is mapped to the 3D surface. We ensured that the full resolution cortical surface can be used for the reconstruction (usually more than 200.000 sources). It is worth to note that the amount of both incoming and processed data, which has to be transferred between CPU and GPU, has a

²https://bitbucket.org/labp/na-online_ow-toolbox

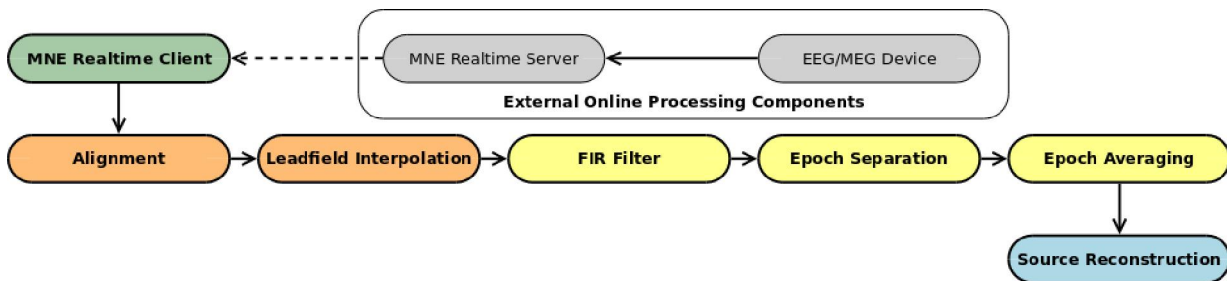


Fig. 3 – Signal processing chain used for experimental testing. The online pipeline of our software divides in data acquisition (green), which establishes a connection to an EEG/MEG device via TCP/IP, forward model preparation (orange), EEG/MEG data preprocessing (yellow) and analysis modules (blue).

tremendous impact on the total performance. If, for example, an epoch consists of 150 samples in float precision and 370 channels, the data size transferred from CPU to GPU is approximately 210 kByte. If activity is reconstructed to 250.000 sources, 143 MByte need to be transferred in the back direction.

5. SIMULATIONS AND EXPERIMENTAL TESTING

We tested our software in two steps: (1) we simulated an online measurement based on a previously recorded data set. (2) We performed real online analysis using a simple experimental paradigm. The purpose of these tests is to explore the real-time capabilities of the system and to demonstrate the signal flow. Due to the fact that a head movement correction is not yet implemented, we restricted the reconstruction to EEG data.

In the simulation, a FIFF file with 60 EEG channels (500 Hz sampling rate) and 1 trigger channel was streamed over a wireless LAN connection (54 Mbps) into our software system. The trigger channel encodes the occurrence of stimuli, here a total of 114 beep tones. The setup of the signal processing chain was according to Fig. 3 with the parameterization as follows:

- 1) *MNE Realtime Client*: block size 1s
- 2) *Alignment*: 10 ICP iterations
- 3) *Leadfield Interpolation*: 1702 virtual sensors
- 4) *FIR Filter*: bandpass (1Hz-20Hz), order 200
- 5) *Epoch Separation*: [-100ms, +200ms]
- 6) *Epoch Averaging*: total average
- 7) *Source Reconstruction*: minimum norm method

A triangulated surface approximating the cortical sheet with one ECD at each node (cortical orientation constraint) represents the source space (244.662 sources). The volume conductor is based on a 3-layer BEM model (brain, skull, and skin surface). All surfaces were segmented from the subject's individual MRI data set using FreeSurfer[16]. The positions of the virtual sensors were extracted from the head surface (skin) using the *EEG Sensor Generator*³ tool. Finally, the generic

leadfield matrix was computed using the MNE toolbox.

The general signal flow is as follows: The *MNE Realtime Client* module receives data from the streaming server, maps this into our data model (*Command*, *EMMeasurement*, *EMData*) and sets additional static data, e.g. BEM layers, source model and sensor positions. The readily prepared data block is transferred to the *Alignment* module, where the co-registration between EEG/MEG device and head model coordinate system is performed. Then, the leadfield columns at the true sensor positions are estimated in the *Leadfield Interpolation* module as described above. This results in a 60x244.662 sized leadfield matrix, which is stored in the data structure. The data blocks are passed to *FIR Filter*, where a filter routine is involved. In the subsequent *Epoch Separation* module, data segments with a length of 300ms are extracted according to stimulus information. Detected epochs are then passed to the *Epoch Averaging* module. Finally, the averaged evoked response is calculated and transferred to the *Source Reconstruction* module. Here, a source distribution is estimated at each sample point of the averaged signal (300ms), which basically involves a huge matrix-matrix

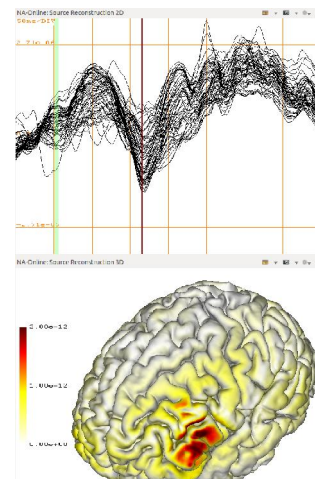


Fig. 4 - Source Reconstruction module in OpenWalnut. Top: Averaged epochs of all 60 EEG channels (overlapped). Bottom: 3D view, estimated source strength at selected time point mapped on individual cortical surface.

³https://bitbucket.org/labp/na-online_eeg_sensor_tools

multiplication. The calculation of the inverse operator (solution matrix based on minimum norm assumption) is done before the actual reconstruction. This is possible because it is largely independent of the data, only the signal to noise ratio has to be taken into account. The average time for this calculation was 2.2 seconds in our simulation. This means that the operator needs to be ready before a data block is subject of source localization. The source reconstruction module is invoked whenever a new epoch is detected.

Fig. 4 shows the result as it appears in OpenWalnut after all trials were averaged. The view can be rotated and different threshold values can be used to modify the color mapping. The mapping of a source distribution corresponds to a selectable time point in the signal view. The result evolves with each new epoch until the end of the measurement.

During tests we measured the processing time for each block in each module. This block processing time includes both the computational time for the algorithm and the time for visualization. The data transfer time between modules is not included in this measurement for two reasons: First, this time is difficult to estimate because the data transfer is not handled by modules but by the OpenWalnut kernel. Second, data transfer is based on the mapping and remapping of pointers and the time consumption should be negligible. It should be noted that the block size is automatically reduced at *Epoch Separation*, which is due to the selected epoch time interval.

The described signal chain was tested under two conditions: CPU-only processing and GPU-supported processing. While the former is only based on CPU execution, the latter involves GPU algorithms if modules support this. The tests were done on a high-performance workstation: Intel Xeon E5620 CPU with 2.4 GHz, 12 GB DDR3 RAM and a NVIDIA Tesla C2070 GPU.

The estimated processing times are summarized in Table 1. The times differ somewhat from earlier results [8], because, first, we now used double precision and, second, only EEG data is processed in the preprocessing stages. MEG data is excluded because source reconstruction was not yet possible at all. As can be seen, the processing time for *FIR Filter* and *Source Reconstruction* decreases if the GPU is used. *FIR Filter* scales better with increasing channel size, e.g. when processing EEG and MEG. The total processing time for the CPU-only case exceeds the block size by approximately 70 percent, while the total time for GPU-supported processing is in the range of less than 25 percent of the block size. A total processing time less than the block size is necessary in order to process incoming data in time, i.e. to provide online capabilities. Given the setup

Table 1. Averaged processing time for one block, all times in milliseconds.

| | CPU-only | GPU-supported |
|--------------------------|-----------------|---------------|
| <i>Alignment</i> | | 77.20 |
| <i>Leadfield Interp.</i> | | 3188.0 |
| FIR Filter | 13.56 | 8.34 |
| Epoch Separation | 2.16 | 2.16 |
| Epoch Averaging | 0.87 | 0.87 |
| Source Rec. | 1,742.71 | 225.84 |
| Total sum | 1,759.30 | 237.21 |

presented here, this can only be achieved using the GPU.

In addition to the above test, we practically applied the software for the online localization of evoked brain activity in the primary motor cortex, which can be triggered based on electrical stimulation. The signal processing chain was similar to the simulation setup. As before, head model information of the individual subject was available (source space with 277.370 sources). We now used the real physical link to a NeuromagVectorView MEG system (306 MEG channels, 64 EEG channels). The online analysis was performed on a notebook: Intel Core i7-3630QM with 2.4 GHz, 8 GB DDR3 RAM and a NVIDIA GeForce GTX 660M. The reconstructed brain activity could clearly be localized to areas in the primary motor cortex. This test showed, that our implementation even performs well under real world conditions and on common end-user hardware.

6. CONCLUSION

We have shown that the current system can principally be used to reconstruct and visualize evoked brain activity based on distributed source localization during EEG/MEG measurements. Particular the usage of GPUs provides promising resources that can be utilized to improve the system further and to implement additional functions and algorithms.

Still, some issues need to be tackled before the presented system can finally be used in a real online processing framework in a seamless manner.

For example, we currently use a fixed value for the estimated signal-to-noise-ratio (SNR). This value is actually required to calculate the inverse operator, where it scales between the portion of data to be explained and modelling assumptions. Thus, this values should automatically be estimated. Moreover, the inverse operator should be updated whenever the SNR changes. In case of evoked responses, this actually happens with every new epoch. As shown in the simulation section, the calculation of the inverse operator exceeds the block size. One possible

solution to this issue is to calculate a set of inverse operators in advance, where each covers a certain SNR range. Thus, recalculation would be replaced by the selection of a selection mechanism. We also currently do not account for the actual noise covariance. While solutions to this issue is straightforward, it was not required to verify the online processing capabilities of our software.

Another issue is the computation of the forward model, i.e. the leadfield matrix. We believe that our solution for EEG, which is based on leadfield interpolation, is a good compromise between computational time and accuracy. Comparisons between a leadfield matrix for a given sensor setup estimated from leadfield interpolation and using a normal forward calculation revealed only slight differences between both results. Due to the head movement in MEG devices, which can be expected in most scenarios, such an interpolation approach, e.g. as proposed in [11], is not possible for MEG data. Instead, the head position in the MEG device has to be taken into account. The method presented in [17] effectively allows to correct the data for the head position based on a simplified forward model and the minimum norm approach. The corrected data can then be used for the localization with a more complex forward model, e.g. again using the individual folded surface. It seems worthwhile to employ this approach for online processing. Before this can be done, the detection of the true head position in the MEG device is necessary which, thanks to dedicated tracking coils, is possible in an effective way [18]. These methods are currently being realized.

Other important issues are the access to further EEG/MEG systems, which we plan to achieve by integrating the *FieldTrip Buffer*⁴, progress in the field of efficient artefact detection procedures, and further optimization of existing modules. For example, it is easily possible to drastically reduce the computational time for source reconstruction just by limiting the number of reconstructed time points, e.g. to only some automatically selected extrema of the signal or to a few user defined samples.

While we proofed our concept for online processing based on time domain analysis, future development will certainly include processing in the frequency domain. Possible fields of applications can be found in neurosciences, e.g. neurofeedback with the interesting possibility to measure information in the source domain.

It is worth to note that, like OpenWalnut, our toolbox is licensed under LGPL and therefore can easily be modified, improved or extended.

6. ACKNOWLEDGEMENT

This work was supported by grants of the German Federal Ministry of Education and Research (17108X10) and the European Social Fund (100098255).

7. REFERENCES

- [1] S. Baillet, J. C. Mosher and R. M. Leahy, Electromagnetic brain mapping, *IEEE Signal Processing Magazine*, Vol. 18, Issue 6, 2001, pp. 14-30.
- [2] M. S. Hämäläinen and R. J. Ilmoniemi, *Interpreting Measured Magnetic Fields of the Brain: Estimates of Current Distributions*, Helsinki University of Technology, 1984.
- [3] M. S. Hämäläinen and R. J. Ilmoniemi, Interpreting magnetic fields of the brain: minimum norm estimates, *Medical and Biological Engineering and Computing*, (32) 1 (1994), pp. 35-42.
- [4] C. Dinh, J. Rühle, S. Bollmann, J. Hauelsen and D. Güllmar, A GPU-accelerated performance optimized RAP-music algorithm for real-time source localization, *Biomedical Engineering/ Biomedizinische Technik*, DOI: 10.1515/bmt-2012-4260, September 2012.
- [5] C. Dinh, M. Luessi, L. Sun, J. Hauelsen and M. S. Hamalainen, MNE-X: MEG/EEG Real-time acquisition, real-time processing, and real-time source localization framework, *Biomedical Engineering / Biomedizinische Technik*, (58) (2013).
- [6] Y. Renard, F. Lotte, G. Gibert, E. Maby, V. Delannoy, O. Bertrand and A. L'cuyer, OpenViBE: An open-source software platform to design, test and use brain-computer interfaces in real and virtual environments, *Presence Teleoperators Virtual Environments Presence Teleoperators and Virtual Environments*, (19) 1 (2010), pp. 35-53.
- [7] Q. Noirhomme and R. I. Kitney, Single-trial EEG source reconstruction for brain-computer interface, *IEEE Transactions on Biomedical Engineering*, (55) 5 (2008), pp. 1592-1601.
- [8] C. Pieloth, J. M. Pizarro, T. Knösche, B. Maess and M. Fuchs, An online system for neuroelectromagnetic source imaging, *Proceedings of the IEEE 7th International Conference on Intelligent Data Acquisition and Advanced Computing Systems (IDAACS'2013)*, Berlin, Germany, 12-14 September 2013, pp. 270-274.
- [9] H. Nolan, R. Whelan and R. B. Reilly, FASTER: Fully automated statistical

⁴<http://fieldtrip.fcdonders.nl/development/realtime>

thresholding for EEG artifact rejection, *Journal of Neuroscience Methods*, (192) 1 (2010), pp. 152-162.

- [10] A. Delorme, T. Sejnowski and S. Makeig, Enhanced detection of artifacts in EEG data using higher-order statistics and independent component analysis, *NeuroImage*, (34) 4 (2007), pp. 1443-1449.
- [11] B. Yvert, A. Crouzeix-Cheylus and J. Pernier, Fast realistic modeling in bioelectromagnetism using lead-field interpolation, *Human Brain Mapping*, (14) 1 (2001), pp. 48-63.
- [12] S. Eichelbaum, M. Goldau, S. Philips, A. Reichenbach, R. Schurade and A. Wiebel, OpenWalnut: A new tool for multi-modal visualization of the human brain, *Proceedings of the EG VCBM*, 2010.
- [13] S. Eichelbaum, M. Hlawitschka, A. Wiebel and G. Scheuermann, OpenWalnut – An open-source visualization system, *Proceedings of the 6th High-End Visualization Workshop*, 2010.
- [14] A. Gramfort, M. Luessi, E. Larson, D. Engemann, D. Strohmeier, C. Brodbeck, L. Parkkonen and M. Hämäläinen, MNE software for processing MEG and EEG data, *NeuroImage*, (86) (2013), 446-460.
- [15] S. Rusinkiewicz and M. Levoy, Efficient variants of the ICP algorithm, in *Proceedings of the 3rd International Conference on 3-D Digital Imaging and Modeling*, 2001.
- [16] B. Fischl, D. H. Salat, E. Busa, M. Albert, M. Dieterich, C. Haselgrove, A. van der Kouwe, R. Killiany, D. Kennedy, S. Klaveness, A. Montillo, N. Makris, B. Rosen and A. M. Dale, Whole brain segmentation: automated labeling of neuroanatomical structures in the human brain, *Neuron*, (33) 3 (2002), pp. 341-355.
- [17] T. R. Knösche, Transformation of whole-head MEG recordings between different sensor positions, *Biomedical Engineering / Biomedizinische Technik*, (47) 3 (2002), pp. 59-62.
- [18] K. Uutela, S. Taulu and M. S. Hämäläinen, Detecting and correcting for head movements in neuromagnetic measurements, *NeuroImage*, (14) 6 (2001), pp. 1424-1431.



Christof Pieloth studied Computer Science at Leipzig University of Applied Sciences (Germany). During his master study he focused on Parallel and Distributed Systems. Christof Pieloth finished his study in 2012 with a master's degree. Since then he is working at the Laboratory for Biosignal Processing on real-time processing and analysis of brain activity.



Thomas R. Knösche studied Theoretical Electrical Engineering at Ilmenau University of Technology (Germany) from 1987 to 1992. In 1991, he finished his PhD with a thesis about the neuroelectromagnetic inverse problem at Technical University of Twente (Netherlands).

Thomas R. Knösche received his habilitation from the Ilmenau University of Technology in 2010. Since 2006 he is head of the research group "MEG and Cortical Networks" at the MPI for Human Cognitive and Brain Sciences in Leipzig.



Burkhard Maess studied Physics in Leipzig from 1982 to 1987. In 1991 he finished his PhD. Since 1992 he focused on Cognitive Neuroscience and has worked in different labs in Berlin, Jülich and Leipzig (Germany). Since 2000 he is head of the magnetoencephalography group at the

MPI for Human Cognitive and Brain Sciences in Leipzig.



Mirco Fuchs studied Electrical Engineering at the Leipzig University of Applied Sciences (HTWK Leipzig). After he received his diploma degree in 2005, he was working in several research projects. Since 2007, he focused on inverse methods for the reconstruction of neuroelectromagnetic sources. Since 2010, he is research assistant at the Laboratory for Biosignal Processing at the HTWK and guest PhD student at the MPI for Human Cognitive and Brain Sciences in Leipzig.