International Journal of Bioelectromagnetism
Vol. 20, No. 1, 2018

Real-time MEG data-processing unit for online medical imaging and brain-computer interface: a model-based approach

Tao Chen[1,2], Sergey Suslov[1], Michael Schiek[1], Jürgen Dammers[2], N. Jon Shah[2,3,4], Stefan van Waasen[1,5]

[1]Central Institute of Electronic Systems (ZEA-2), Forschungszentrum Jülich GmbH, 52425 Jülich, Germany
[2]Institute of Neuroscience and Medicine (INM-4), Forschungszentrum Jülich GmbH, 52425 Jülich, Germany
[3]Department of Neurology, Faculty of Medicine, RWTH Aachen University, 52074 Aachen, Germany
[4]JARA–BRAIN–Translational Medicine, RWTH Aachen University, 52074 Aachen, Germany
[5]Communication Systems, Faculty of Engineering, University of Duisburg-Essen, 47057 Duisburg, Germany
Contact: {t.chen, s.suslov, m.schiek, j.dammers, n.j.shah, s.van.waasen }@fz-juelich.de

Abstract — **In magnetoencephalography (MEG) data processing, biological artifacts in particular overtop the signal of interest by orders of magnitude and must be removed from the measured signals to avoid reconstruction errors. However, many real-time brain computer interface (BCI) approaches neglect real-time artifact removal as it is computationally demanding.**

We are working on the project of developing a real-time signal processing system for magnetoencephalography (MEG-RT 2.0) to be used as an add-on to an existing MEG system. The aim is to provide real-time current source reconstruction and based on this to enable neuro feedback or BCI applications. We introduce a requirement-driven model-based development methodology (RDD & MBD) for real-time computation systems based on vendor neutral specifications. It handles the ever-rising complexity of computation and control systems by raising the abstraction level in the design. The approach was applied to the design of a system-on-chip (SoC), capable of performing real-time artifact rejection, based on the recently presented method 'Ocular and Cardiac Artifact rejection for Real-Time Analysis in MEG' (OCARTA). In this paper, we demonstrated the principle and power of RDD&MBD by focusing on the dimension reduction step in OCARTA.

Keywords - MBSE; SysML; heterogeneous systems; real-time systems; MEG; PCA

Introduction

MEG is a non-invasive neuroimaging technique that directly measures the electrophysiological processes of the living human brain. Using MEG, the spatio-temporal dynamics can be investigated at very high temporal and good to moderate spatial resolution, depending on the depth of the active sources [1]. The magnitude of the magnetic field components is in the range of a few tens to hundreds of femto Tesla while some noise and biological artifacts, such as ocular or cardiac activity, can be tens to a hundred times larger [1]. These artifacts must be removed from the measured signals as they can lead to source reconstruction errors [2]. However, many real-time BCI approaches neglect real-time artifact removal as it is computationally demanding.

In MEG studies, real-time MEG data analysis has recently become a topic of high interest and, in order to be viable, it requires automatic real-time artifact removal [2] [3]. One of the first real-time MEG source analyses was demonstrated in [4], but only included cardiac artifact rejection. In other studies, real-time MEG analysis is limited to certain frequency bands [2], e.g., μ (9-12 Hz) [5], μ (9-15 Hz) and β (18-30 Hz) [6] or α (8-13 Hz) [3].

The paper addresses the development of a real-time signal processing system to be used as an add-on to an existing MEG system. The aim of the project is to provide real-time current source reconstruction and thus to enable neuro feedback or BCI applications. To deal with the complexity of the system, the RDD & MBD method for real-time computation systems is introduced. The methodology handles the ever-rising complexity of computation and control systems by raising the abstraction level in the design. The approach allows us to change from tactical to strategical development decisions, as the cost of principal mistakes increases with the complexity of a design [7]. The practice of model-based system engineering (MBSE) rests on three pillars: a modeling language, a modeling tool, and a modeling method [8].

SysML (System Modeling Language) is an extension of UML (Unified Modeling Language) and was created specifically for the systems engineering domain [8]. Such language enables movement through different abstraction levels and effective communication of the essential aspects of a large-scale and complex system in the following details: structure, behavior, requirements, and parametrics (mathematical models) [8].

SysML is a vendor neutral modeling language enabled in a number of system-level modeling tools. Based on the project requirements and cost studies, Sparx Enterprise Architect [9] was selected for our project.

A system design methodology is like a road map that helps transforming the fragmented modeling language syntax into a model of the target system. The RDD & MBD approach, as the further development of comprehensive methodology for functional design in digital systems presented in [7], has been elaborated in the Central Institute of Electronic Systems (ZEA-2) to address the problems of heterogeneous computation and control system design using SysML [10]. This methodology is exploited in the design of MEG-RT 2.0. Applying system modeling to our design allows us a comprehensive and thorough evaluation of actual design requirements, as well as enabling the forward planning of possible technical options to find an optimal solution and allows for a reuse of the main development steps when a system adaption or refinement is required.

We applied the RDD & MBD approach to the design of a system-on-chip (SoC), capable of performing real-time artifact rejection, based on the recently presented method OCARTA [2]. The number of channels used in a whole-head MEG system lead to an overdetermined mathematical problem during artifact rejection [2]. In this paper, we elaborated on the RDD & MBD approach with dimension reduction using principal component analysis (PCA) as an example. Based on the performed system modeling and analysis, the dimensionality reduction of MEG data is developed and implemented on a single FPGA board (Xilinx Virtex 6). This work is part of the development of an MEG-2.0 Real-Time add-on device to a 248 channel whole-head magnetometer MEG system (MAGNES_3600WH) from 4D-Neuroimaging, located at the Research Centre in Jülich, Germany. The design is parameterized and generally applicable to other MEG systems.

The paper is structured as follows: Section II describes the implemented system model. The experiment results and analysis are discussed in Section III. Finally, the conclusion and an outlook on our future work are given in section IV.

Model-driven system design

RDD & MBD is a SysML-based system engineering approach whose workflow is plotted in Fig. 1[10].

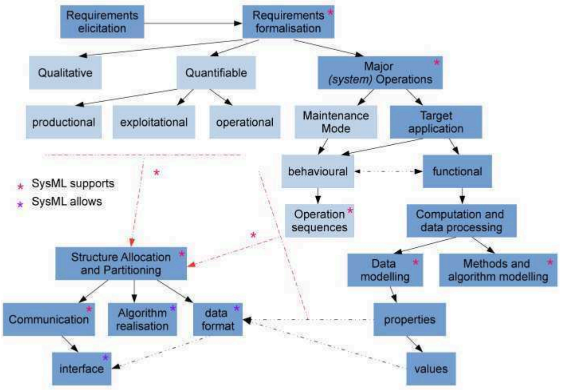

Figure 1: Model-based system engineering for real-time systems (MBSE-RT) workflow

It starts off by specifying general system requirements and constraints of stakeholders and formalizing them into three categories, i.e. *Qualitative*, *Quantifiable* and *Major operations*, which are further refined into sub-categories. The *Qualitative* category mainly includes immeasurable requirements such as safety, continuous observability, ease of operation, etc. The *Quantifiable* category captures measurable parameters and constraints. With the refinement of system models, some qualitative requirements may become quantifiable. The *Major Operations* category depicts the system functionality, including maintenance modes and functional targets of the application. After the system's technical requirements are specified, the second step of the methodology – functional and structural analysis is performed. The corresponding system dynamic (behavior) and block structure are modeled in parallel, including *Data modeling*, i.e. data structure, data dependency, data exchange flows, and *Methods and algorithm modeling*. Quantifiable requirements are then imposed in the form of mathematical constraints to the relevant elements of structural models to properly allocate them to relevant computation structures with predictable implementation parameters such as performance, resource budget and even power consumption. Thus, we partition the functions and allocate them to corresponding computation hardware. The procedure may be repeated for each sub-system to perform refinement of the system on different abstraction levels [10].

The computation process of PCA

The typical toolchain for MEG consists of data recording, filtering, artifact rejection and source localization [2]. PCA is applied in the context of artifact rejection to reduce the computational complexity of independent component analysis (ICA). PCA is one of the most commonly used techniques to reduce the dimensionality of a dataset while retaining most of the information by means of its variance [11]. PCA maps the original data into a new set of orthogonal basis vectors, where the components are stored in a decreasing order of variance [12]: The first axis (first principal component) accounts for the greatest data variance of the original data set while the last one the least data variance. The first k principle components are chosen for data processing as they explain most of the data variance (in our case more than 95%). In PCA, computation of the eigenvalues and eigenvectors of large matrices is computationally complex. For the purpose of acceleration, reconfigurable hardware solutions such as FPGA are currently good approaches to tackle the computation complexity.

A commonly used method for PCA is eigenvalue decomposition (EVD) of the data covariance matrix. It is comprised of four steps:

1), the original data set is centered by subtracting the mean vector. 2), the covariance matrix is computed. 3), EVD is applied to the covariance matrix, i.e., eigenvalues and eigenvectors of the covariance matrix are computed. This is the most computationally demanding step of performing PCA. There are various techniques to perform EVD. Considering the large dimensionality of MEG data, QR algorithm for PCA is adopted [12]. The QR algorithm is the decomposition of the covariance matrix into an orthogonal matrix and a triangular matrix. 4), finally, an orthogonal transformation is applied onto the original data.

System engineering

System engineering begins with capturing system requirements as requirements are the foundation of system design [9] [10]. All essential requirements for PCA are clearly classified by package diagrams and described by requirement diagrams. Within the requirement diagrams, the hierarchy of requirements is defined using the relationships such as *Contains* and *Derived*, etc. It is necessary to create models at different abstraction levels to unleash hidden complexity in the system. The derived system technical requirements are further detailed with child requirement diagrams, which are more exact and atomic. Apart from relationships among requirements, traceability from system requirements to behavioral and structural elements is established.

As a second step, the system is modeled with respect to block structure and behavior. There are two types of block diagrams: block definition diagram (BDD) and internal block diagram (IBD) [13]. BDD details the high-level structure of PCA. To specify the internal structure and the inner interconnections of PCA and its sub-blocks, IBDs are established [8].

Behavior models visualize, specify, construct, and document the dynamic behaviors of the system as it interacts with users and the environment. Fig. 2 shows one example of the many behavioral diagrams.

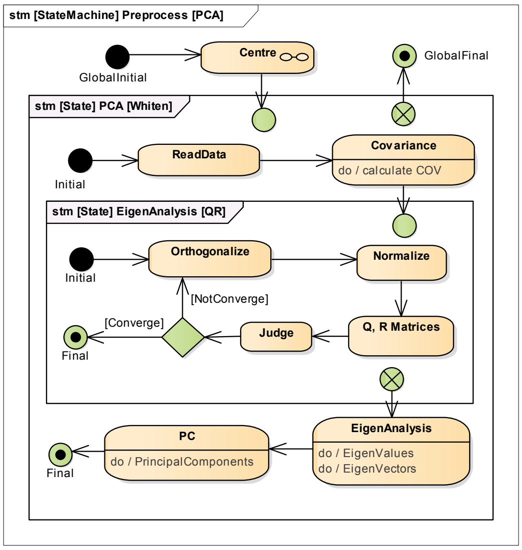

Figure 2: State machine diagram of PCA

In Fig. 2, the state machine diagram covers three decomposition levels. The top level mainly contains 2 states: *Centre* and *Whiten*. The state machine starts in a pseudostate *GlobalInitial*, then the state *Centre* centres the input data. After the data is whitened by the state *Whiten*, the computation process ends with *GlobalFinal*.

In the third level, eigen-analysis is the most complex operation among the 4 steps of PCA. The QR algorithm is decomposed into several states. The first 3 states *Orthogonalize*, *Normalize* and *QRMatrices* decompose the covariance matrix from the second step of PCA into a product of an orthogonal matrix and a positive upper triangular matrix [12]. Then the orthogonal matrix is right multiplied with the positive upper triangular matrix. The state *Judge* checks the equality of the bottom triangle of the new product to zero. In case of in equality, the computation is repeated; otherwise, the computation converges.

It is important to note, that "*a diagram of the model is never the model itself; it is merely one view of the model*" [8]. A number of behavioral views were created as necessary, including use case diagrams, sequence diagrams, state machine diagrams and activity diagrams, to detail the dynamic behaviors of the system [9].

In the third step of system engineering, we impose mathematical relationships (equations or inequalities) on the value properties of the design using SysML parametric models which include constraint blocks and parametric diagrams to create prognostic models of implementation. A development tool usually allows constraints to be described with programing or modeling languages, e.g. JavaScript, VBScript or Modelica, which provide the underlying mathematical foundation for further analysis or simulation.

The parametric model enables trade-off studies of different design configurations to find an optimal set of solutions. It can also be used for checking violation of the quantifiable constraints of a design, or for performing early functional prototyping (mathematical modeling). Models derived in this way can then be used as a golden reference for further functional verification。

Structure allocation and architecture

Based on the preliminary performance and resource analysis using parametric models and behavioral computation analysis, computations of the system are allocated to platform computing resources, while shared variables and exchanged data are allocated to memories and physical links.

Mathematically, the transformations of OCARTA are two matrix–matrix multiplications (or rather two matrix–vector multiplications) and computationally demanding [2]. The parallelism of FPGA deals with matrix multiplications quickly and efficiently. Xilinx Virtex 6 FPGA with 2GB DDR3 SDRAM is the selected platform for hardware implementation. Within FPGA, a precision symmetrical look-up-table (LUT) based unit was developed for potential exponential operations in ICA, which speeds up the computation and reduces error accumulation. The rest of the computation is done on a state machine driven arithmetic logic unit (ALU). The experiment duration for MEG signal measurement of a subject is around 165 seconds, which produces around 200MB of data; thus we chose the 2GB DDR3 SDRAM for data storage.

Fig. 3 shows the FPGA-based computation model and the data path of eigen-analysis. The controller consists of a scheduler and a load/writeback module. The scheduler controls the computation modules and coordinates the buffers. This load/writeback module receives control signals from the scheduler, and update data in the buffers accordingly. In the QRD module, the four computation steps of QR algorithm are performed.

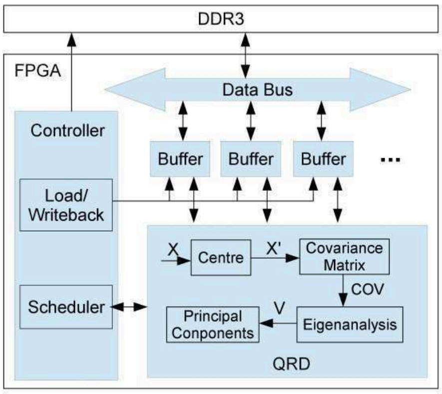

Figure 3: Data path of eigen-analysis

Results and analysis

Based on the behavioral models, HDL code can be generated automatically, which is often sufficient for direct implementation in hardware. PCA was implemented by modifying the generated Verilog HDL code. The design was verified by simulation using ISE 14.1 under the test clock frequency of 100 MHz; and in-circuit verification by means of Virtex 6 was also carried out. The tested MEG data were recorded with a 4-D-Neuroimaging whole-head magnetometer system (MAGNES-3600WHMEG). Captured waveforms indicate that the dimension of the input data is reduced from 248 to on average 25 components, explaining more than 95% of the original data variance. The design occupies 13% of total slices (56880 available) and 8% of DSP48E1s (576 available), which highly correlates with the results of the prognostic models created for system-level modeling.

The system design of PCA was completed successfully, which is the major objective of the study. The RDD & MBD method describes the system with great simplicity using SysML. It decreases the complexity for large system design and reduces the development time by creating a system with different abstraction levels.

Conclusion and future work

In this paper, we have presented the application of a requirement-driven model-based approach to the implementation PCA using SysML. It has been outlined how models are defined, traced and used. The approach enabled us to elaborate the hardware implementation, satisfying the demanding requirements of MEG data dimension reduction using the multi-aspect SysML model. The presented model-driven approach provides an efficient management of PCA design complexity by specifying the model at different abstraction levels. Additionally, hardware implementation of PCA based on the performed system modeling is done and preliminary experiment results are presented. The dimension of MEG data is reduced greatly. It will effectively facilitate and accelerate the implementation of the MEG real-time system enabling neuro feedback and BCI applications in an add-on system to existing MEG systems.

References

[1] Hämäläinen, Matti, et al. "Magnetoencephalography—theory, instrumentation, and applications to noninvasive studies of the working human brain." *Reviews of modern Physics* 65.2 (1993): 413.

[2] Breuer, Lukas, et al. "Ocular and cardiac artifact rejection for real-time analysis in MEG." *Journal of neuroscience methods* 233 (2014): 105-114.

[3] Sudre, Gustavo, et al. "rtMEG: a real-time software interface for magnetoencephalography." *Computational intelligence and neuroscience* 2011 (2011): 11.

[4] Rongen, H., V. Hadamschek, and M. Schiek. "Real time data acquisition and online signal processing for magnetoencephalography." *IEEE transactions on nuclear science* 53.3 (2006): 704-708.

[5] Buch, Ethan, et al. "Think to move: a neuromagnetic brain-computer interface (BCI) system for chronic stroke." *Stroke* 39.3 (2008): 910-917.

[6] Mellinger, Jürgen, et al. "An MEG-based brain–computer interface (BCI)." *Neuroimage* 36.3 (2007): 581-593.

[7] Suslov, Sergey. *Parallelisation Potential of Image Segmentation in Hierarchical Island Structures on Hardware-accelerated Platforms in Real-time Applications*. Forschungszentrum Jülich, 2013.

[8] Delligatti, Lenny. *SysML distilled: A brief guide to the systems modeling language*. Addison-Wesley, 2013.

[9] Rosenberg, Doug, and Sam Mancerella. "Embedded systems development using SysML: an illustrated example using enterprise architect." *Sparx Systems Pty Ltd and ICONIX* (2010): 4-14.

[10] Suslov, Sergey. Presentation: SysML for computing controlling system development, https://www.fz-juelich.de/SharedDocs/Downloads/ZEA/ZEA-2/DE/SysML.pdf?__blob=publicationFile, 2017.

[11] Karamizadeh, Sasan, et al. "An overview of principal component analysis." *Journal of Signal and Information Processing* 4.03 (2013): 173.

[12] Shahrouzi, S. Navid, and Darshika G. Perera. "Dynamic partial reconfigurable hardware architecture for principal component analysis on mobile and embedded devices." *EURASIP Journal on Embedded Systems* 2017.1 (2017): 25.

[13] Friedenthal, Sanford, Alan Moore, and Rick Steiner. *A practical guide to SysML: the systems modeling language*. Morgan Kaufmann, 2014.

Acknowledgements

The presented research is supported by China Scholarship Council (CSC), in cooperation with the Central Institute of Electronic Systems (ZEA-2) and the Institute of Neuroscience and Medicine (INM-4) at Forschungszentrum Jülich GmbH.