Design of a Modular Brain-Computer Interface

J. Šťastný [1], J. Doležal[1], J. Kubový[1], V. Černý[1]
[1] Biosignal processing laboratory, Department of Circuit Theory, Faculty of Electrotechnical Engineering, Czech Technical University in Prague, Technická 2, 166 27 Praha 6, Czech Republic
E-mail : stastnj1@seznam.cz, jaromir.dolezal@post.cz, jan@kubovy.cz , cernyvl3@fel.cvut.cz

Anotace:
Naše skupina se zabývá výzkumem v oblasti rozpoznávání pohybové aktivity v EEG. V současnosti pracujeme na off-line klasifikaci předem separovaných realizací pohybového EEG na základě jejich časového vývoje. Tento příspěvek představuje úvodní studii k architektuře navrhovaného BCI systému pro zpracování EEG spolu s výsledky návrhu prvních základních bloků. Systém pomáhají realizovat také studenti naší katedry, ti jsou plně integrováni do vývojového týmu.

Annotation:
We have been dealing with a research in the field of movement-related EEG recognition and currently we are working on an off-line single trial classification system able to recognize movement-related EEG on the base of its temporal development. This contribution presents a preliminary study of the proposed real-time BCI system architectural components for the EEG Processing Pipeline along with the results of the design of the first fundamental system blocks. The project also successfully involves students of our department in a real teamwork.

INTRODUCTION

Existing Brain Computer Interface (BCI) prototypes suffer from too slow communication channel between a human brain and a computer working with Information Transfer Rate (ITR) lower than 100 bits per minute. One possibility leading to higher ITR is the recognition of more distinct brain states – transferring more bits per state – *high-resolution EEG recognition*. However, existing BCI systems recognize only few very different EEG activities (left/right-hand or finger movement, mental activities, conscious EEG rhythm control, or event-related potentials, among others). Our research is targeted to the exploration of possibilities of the high-resolution movement recognition from the EEG signal.
Our previous work [1] showed that off-line single trial classification of extension and flexion movements of right index finger is possible; classification score of up to 95% was achieved. The Hidden Markov Models (HMMs) classifier used EEG temporal dynamics, rather than information stored in the difference of signal powers from different electrodes usually used in systems eg recognizing left and right hand movements, feet and tongue movements, or mental activities.

Currently our effort is directed to improve the whole algorithm first to be able to process continuous EEG stream (not only single trials) and then to work in real-time. Proposed real-time application allows to use visual feedback which is known to enforce the movement related brain patterns [2] as the movement planning, execution and observation mechanisms are interconnected. This can help to further increase classification accuracy. To reach the planned results in a feasible time we started three parallel projects:

1. research in the field of blind source separation application on EEG to further increase reached classification score with the single-trial off-line classification system.
2. research in the field of continuous EEG processing targeted to extension of current classification algorithms to a continuously working system.
3. design of a real-time EEG Processing Pipeline (EPP) software.

The third project was further divided into two phases:

1. in the first phase the target is to fully design, verify, and evaluate basic EPP and then use it for simple task of left/right hand movement recognition with visual feedback – hand moving on a computer screen.
2. in the second phase the EPP pipeline will be upgraded to implement the continuous EEG recognition algorithms we are currently working on in the frame of the second project.

This contribution presents a preliminary study of the proposed real-time BCI system architectural components for the EEG Processing Pipeline (EPP, see Fig. 1) and summarizes the first results from the phase 1 of the design.

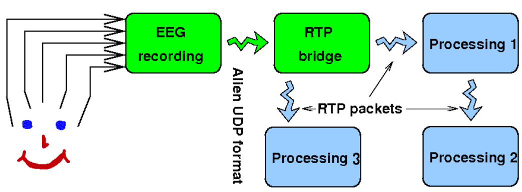

Figure 1: Modular architecture of the EEG Processing Pipeline (EPP).

EPP ARCHITECTURE

Since the EPP BCI algorithms have been still under development, we decided to build the whole EPP of loosely coupled blocks connected via network packet interfaces. This allows us to easily replace implementation of parts of the system along with future development of new signal processing algorithms.

EEG machine (manufacturer *Alien Technik*) equipped with a network interface streaming the EEG via UDP packets in a proprietary *Alien* format was used for experiments. However, we wanted to be independent on the used EEG machine, thus the first block in our EPP is a bridge converting *Alien* UDP packets into our network EEG format, see Fig. 1.

Telemedicine EEG data transmission over LAN is similar to a multimedia (e.g. voice over IP) network data transmission. In both cases real time performance and small latency are a must while a small packet loss is acceptable for the sake of low latency. Real time network protocol (RTP) was chosen as the intra-module communication protocol. Its advantages are datagram nature (necessary to keep low latency) and prioritization in network switches over other non-real time data flows.

Although RTP is frequently used for multimedia transmission in the field of telemedicine [3, 4], we are not aware of any application of this protocol in the field of EEG signal processing; majority of papers reporting on telemedicine EEG processing deal only with off-line transmission of EEG data.

Since the original RTP protocol does not directly support biomedical (EEG) data transfer, we developed an extension to the protocol, see [5, 6]. For the detailed description of EPP EEG RTP packet header, see [5]. The extension was created in a way not to interfere with normal RTP operation so the EEG packets can be processed by any software or hardware already supporting RTP data flows.

Java was chosen for the system implementation thanks to its multiplatformness. As the bandwidth requirements for raw EEG data transmission are only moderate, Java implementation should be fast enough. An available open-source RTP implementation [7] was used to implement the RTP stack.

DESIGN OF THE EPP

The design of the EPP closely reflects the real BCI EEG processing pipeline drawn in Figure 3. Each block in the BCI structure is represented by one module in the designed software. The modules are connected by network packet interfaces and are implemented using multithreading and parallelized way to exploit as much as possible the hardware parallelism offered by today's multicore and hyperthreading CPUs.

Currently the EEG machine interface module, RTP bridge, RTP receiver, and data flow viewer modules are implemented. Feature extraction, surface filtering, and classification modules are under development.

Surface Filtering

There are a few layers of head tissue in between the brain dipole sources and scalp electrodes: cerebrospinal fluid, head tissues, skull bone, and scalp resulting in low-frequency band pass spatial filtering of the EEG activity. Due to this the scalp potentials are blurred and any highly localized EEG activity becomes less apparent. This is in contradiction with the need of brain-computer interface experiments requiring EEG activity with as low spatial blurring as possible. The surface filter compensates this blurring to some extent. The surface filtering module implements a discrete Laplacian filter [8]. The discrete Laplacian filter was chosen as it has some remarkable advantages – it requires only a low computational power, it is easily applicable (simple linear combination of the electrode potentials), and provides rather good results with BCI experiments.

Feature Extraction

The aim of the current phase of the project is to develop a real-time operating BCI EPP recognizing left and hand movement from the EEG signal. To accomplish this task, just a simple parameterization algorithm is needed. The feature extraction module has to compute powers in predefined subbands of the EEG signal so as we may detect the μ rhythm attenuation accompanying the movement [9]. FIR filters will be used to filter out subbands of the EEG μ band and signal powers in the subbands will be used as features.

Data Flow Monitoring

Monitoring of the EEG data flows is necessary for system debugging as well as for the system operation monitoring. A dedicated visualisation module was implemented to visualise the inter-block network RTP traffic

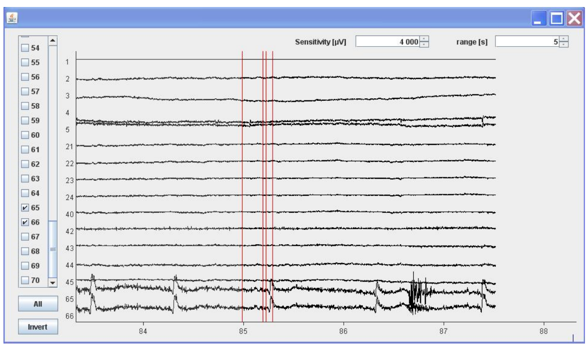

Figure 2: EPP data flow monitoring utility – screenshot during raw EEG processing. The red lines indicate missing packets (data packets which were lost during transmission).

The monitor has the following features:

- all types of dataflow (raw EEG, surface filtered EEG, extracted features) can be visualised "on the oscilloscope", see Fig. 2.
- missing packets (lost eg due to network congestion) are marked with red boxes,
- special tags (artifact, movement, etc) are indicated in the “scope” display.

Classifier

The task of left/right hand movement recognition needs only a simple classifier. The standard Perceptron is expected to provide sufficient performance in this system. The system console will be used to switch between training and testing modes of the system. The module will control the training according to a predefined session protocol.

VERIFICATION

The designed system has been extensively tested. The following approaches have been used for system verification:

- **testing with the real EEG machine**: done irregularly as the EEG machine is in a remote location.
- **system verification with simulated EEG machine**: done regularly after each major change to the system. A set of testing scripts was prepared [6] to support regression testing of the basic functionalities. A dedicated *Alien packet generator* tool was developed so as we may test the whole EPP off-line, without access to the EEG machine. *Alien packet generator* compliance with the real device is verified with the help of utility [10].

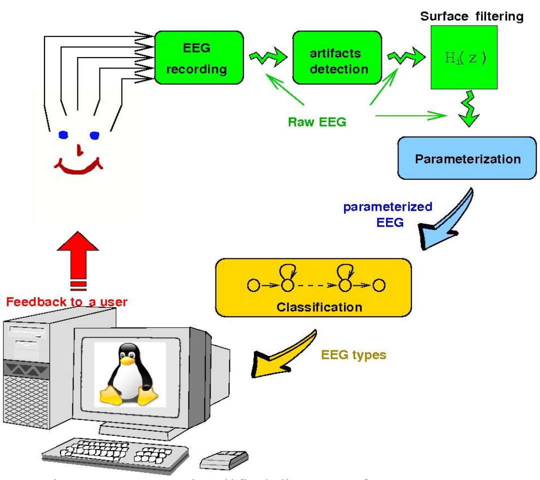

Figure 3: A simplified diagram of a BCI system.

EEG data recorded in the frame of experiment [9] are used for testing with simulated EEG machine.

SYSTEM PERFORMANCE

The biggest concern during software development was a low latency of the future EPP. The latency of the whole EPP is measured as time between the sample acquiring by the EEG machine and its processing and updating the information on the user’s monitor. The overall system latency is a sum of the following components:

- **packed generation latency** is caused by the time quantization. If one packet contains eg 20 ms of data, then the delay between the sampling of the first EEG sample in the packet and packet generation by the EEG machine is 20 ms. This delay is unavoidable and can be reduced by reducing packet size.
- **network latency** is the transport delay of the local area network. Authors of work [11] uses UDP protocol to transmit one channel EEG over 200 km network link in order to maintain dosing of anaesthetic drug. The system worked with 1 sec packet size, transport latency was of approx. 8 ms. The network latency should be minimized by using the RTP protocol which is prioritised in network equipment. Each communication link between two modules add this latency to the overall system latency.
- **packed processing latency** is the latency introduced by all the modules in the system. It can be reduced by a careful writing and profiling/optimization of our software. Each module adds its own component of this latency to the overall system latency.

For real time control, latency shall be below 60 ms [12]. We measured latency and throughput of the system with the currently designed modules. The first stable release of the system (Alien generator, RTP bridge, RTP receiver) had a latency of 17 ms which was already enough for real time control. Further profiling showed that frequent dynamic memory allocations were slowing down the whole system. After optimizing all the parts of the system we achieved nearly zero latency and negligible packet loss rate over a local area network. The already implemented part of the system is capable to work as fast as 20 times the real-time speed in simulated experiments without significant packet loss. The results indicate that – although Java is usually considered as a slow language – carefully written multi-threading applications can really meet soft real time constraints and thus there is no need to use Real Time Java extension [13] in our case.

We also evaluated the bandwidth needed to transfer the EEG signal throughout the network links. If we need to transmit 45 channel EEG at sampling rate of 256 Hz, 16 bit per sample, we need just 45×256×2B/sec = 22,5kB/sec = 180kb/sec which is only a tiny fraction of capacity offered by standard

100Mb Ethernet network link. The results indicate that no EEG data compression for transmission is needed.

TEAMWORK WITH STUDENTS

The project team is currently composed of authors of this article (of whom two are undergraduate students writing software); a fifth designer (also an undergraduate student) will join our team soon.

The whole collaborative effort has adopted some techniques from Agile project management [14]:

- we test the software frequently and integrate the newly developed modules as soon as possible; the working software is the primary measure of progress.
- the team is motivated and interested in the work as they work on a real project; not an artificial assignment (all undergraduate students are actually working on their theses in the frame of our project). This year we succeeded in obtaining a small grant from the Grant Agency of the Czech Technical University in Prague, so we can motivate students also with some small income.
- supervisors continuously monitor the technical quality of the work and in case of problems are both ready to join the development process.
- we try to have the whole team partially self-organizing; collaboration, communication, and sharing of ideas is encouraged by using of email, SVN repository for code and bugtracker to keep track of open issues. Regular meetings of the whole team are held to resolve issues.
- we take a great care of specifications as clear specifications are key for team development effort. However, the specifications are evolving in the frame of design; SVN repository is also used for document sharing.

CONCLUSIONS

A foundation for a BCI modular system design was implemented. The modules communicate via a well-defined interface which is independent on the real EEG recording machine. The experimental results showed that the reached latency is feasible and Java implementation is able to handle data flow.

The proposed modular architecture allows to freely distribute parts of the BCI EPP across network; while eg the feature extraction and pattern recognition can reside on a powerful personal computer, the final presentation layer can run on any thin client (eg mobile phone, ambient light controller, doors opening, etc) substantially extending the radius of activity of possible BCI device.

The proposed distributed architecture also allows to easily exploit parallelism offered by the today's multicore systems; the EPP modules can be simply run on different CPUs. Using the standard RTP protocol ensures real time processing of EEG data in any local area network.

Last, but not least, working with the team gives students a unique experience to participate in real teamwork; all the collaborative applications and methodology are absolutely mandatory for our project as all the participants have been working from their homes and the project participants are thus extremely distributed.

The latest stable release of all the developed packages is freely available at our website [5], prospective users are kindly asked to cite this article.

ACKNOWLEDGMENT

Jakub Šťastný has been supported by the research program Transdisciplinary Research in Biomedical Engineering II No. MSM 684 0770012 of the Czech University in Prague; Jaromír Doležal by the grant GACR 102/03/H085: Biological and Speech Signal Modeling and along with the rest of the team by Grant Agency of the Czech Technical University in Prague, grant No. SGS10/178/OHK3/2T/13.

Our gratitude belongs to doc. Jan Kremláček for his huge support during EEG recording experiments at the Faculty of Medicine of the Charles University in Hradec Králové.

Last, but not least, we would like to thank Alien Technik for their support with debugging of UDP output and providing us with the *Alien UDP* monitor utility.

REFERENCES

[1] J. Doležal, J. Šťastný, P. Sovka. Recognition of Direction of Finger Movement From EEG Signal Using Markov Models. In: 3rd European Medical and Biological Conference on Biomedical Engineering. EMBEC 2005, Prague, pp. 336-341

[2] Christa Neuper. Feedback-Regulated Mental Imagery in BCI Applications: Using Non-Invasive EEG and NIRS Signals. BBCI Workshop 2009 – Advances in Neurotechnologies. Berlin, 2009

[3] Ilias Maglogiannis, Constantinos Delakouridis, Leonidas Kazatzopoulos. Enabling Collaborative Medical Diagnosis Over the Internet via Peer-to-Peer Distribution of Electronic Health Records. J Med Syst, 2006, 30, 107-116

[4] J. C. Guerri et al. A multimedia telemedicine system to assess musculoskeletal disorders. Proceedings of the 2003 International Conference on Multimedia and Expo - Volume 2. Pages: 701 - 704 Year of

Publication: 2003 ISBN:0-7803-7965-9
[5] Real Time BCI Project Page. http://amber.feld.cvut.cz/fpga/projects/real_time_bci/modular_bci.html
[6] Kubový J. Real-time EEG network transmission, Bachelor Thesis CTU FEE, Department of circuit theory, FPGA Laboratory, June 2009 (in Czech)
[7] LibRTP library. http://sourceforge.net/ projects/jlibrtp/
[8] Dennis J. McFarland, Lynn M. McCane, Stephen V. David, and Jonathan R. Wolpaw. Spatial filter selection for EEG-based communication. Electroencephalography and clinical Neurophysiology, (103):386–394, 1997.
[9] Doležal J., Šťastný J., Sovka P., Recording and recognition of movement related EEG signal. In Applied Electronics, Applied Electronics, pp 95-98, September 2009. ISBN 978-80-7043-781-0
[10] UDP receiver and personal communication with Alien Technik
[11] H. Ihmsen, K. Naguib, G. Schneider, H. Schwilden, J. Schüttler and E. Kochs. Teletherapeutic drug administration by long distance closed-loop control of propofol. British Journal of Anaesthesia, January 11, 2007, doi:10.1093/bja/ael337
[12] Adriaansen T. et al. Enhancing Human Computer Interaction in Networked Hapto-Acoustic Virtual Reality Environments on the CenTIE Network. LNCS 3038 2004; 905-912
[13] The Real Time Specification for Java. http://www.rtsj.org/
[14] Bruce P. Douglas. The basics of being Agile in a real-time embedded systems environment. embedded.com/design/opensource/224200417