Christian Dietz*, Jonas H. Pfeiffer, Max E. Dingler and Tim C. Lüth

Neuro control – using neuro monitoring data to control networked active instruments

Abstract: Intraoperative neurophysiological monitoring (IONM) can be used to monitor the neural integrity and help the surgeon to reduce iatrogenic patient damage. We present a networked digital assistance system based on real-time IONM data. The system automatically reduces the power of active medical ablation instruments when they endanger neural structures. We integrated a nerve monitor into a real-time network of medical devices. In doing so, neurophysiological signals are available to all connected medical devices in real-time. With the IONM data a so-called function module Neuro Control calculates and controls the maximum allowed power for active instruments like an ultrasonic dissector.

Keywords: Neuro Control, neuromonitoring, OR.NET, SRTB, real-time network, medical devices, interoperability.

https://doi.org/10.1515/cdbme-2017-0094

1 Introduction

Surgical interventions on the human head (e.g. ear-nose-throat, neuro or craniofacial and maxillofacial surgery) can be very challenging. The surgeon is frequently confronted with the task to remove bone structures and other tissue while he has to take care of delicate areas and structures inside the complex anatomy. Iatrogenic damages to nervous tissue such as the facial nerve can have severe and irreparable impact [1].

Computer assistance systems are able to support the staff with additional information in real time. A nerve monitor for instance is capable of monitoring the activity of specific nerves intraoperatively. With the use of a stimulation probe it enables the localization of a nerve and to a certain degree also the identification of the distance between nerve and probe [2],[3]. Nerve identification is only available as long as the stimulation with the probe is ongoing, so that the surgeon has to remember the nerve's position. However, researchers have recently shown that it is possible to integrate the stimulation probe into the tip of an ultrasonic dissector (USD) without significant loss in nerve response data [4].

The calculated distance information is currently neither stored nor used for other purposes. One such purpose could be a safety shutdown of active resection devices. At present, signals generated by a nerve monitor are not used to control any active surgical instrument, even though the idea for such assistance systems came up years ago [5]–[7]. One reason for that might be the lack of interoperability of medical devices, a well known issue that has been topic of different research activities in the recent past. Successful standards for cross-vendor data communication in the operating room, such as DICOM or HL7, mainly focus on the exchange of image and patient data. Time-critical communication, that is necessary for the control of active devices, is hardly addressed by standards, yet [8].

The German research project OR.NET developed an open hybrid network architecture, with a dedicated part for real-time communication. The non-real-time data is exchanged via a Service-Oriented Architecture based on a web-service implementation while the so-called Surgical Real-Time Bus (SRTB) enables communication with pre-defined timing guarantees and allows for distributed real-time device control [9]–[11]. SRTB is based on the Ethernet Powerlink (EPL) protocol and differentiates between three types of network participants: SRTB master, connectors and function modules. The SRTB master orchestrates the real time network. Connectors translate between vendor specific protocols and SRTB making it possible to attach legacy devices to the SRTB network. Function modules consume data from the network and generate control streams. Function modules are normally used to implement assisting functionality such as the shutdown of surgical instruments in case that the gathered data suggests danger to the patient.

***Corresponding author: Christian Dietz:** Institute of Micro Technology and Medical Device Technology, Technical University of Munich (TUM), Boltzmannstraße 15, 85748 Garching, Germany, e-mail: christian.dietz@tum.de
Jonas H. Pfeiffer, Max E. Dingler, Tim C. Lüth: Institute of Micro Technology and Medical Device Technology, Technical University of Munich (TUM), Boltzmannstraße 15, 85748 Garching, Germany, e-mail: jonas.pfeiffer@tum.de, max.dingler@tum.de, tim.lueth@tum.de

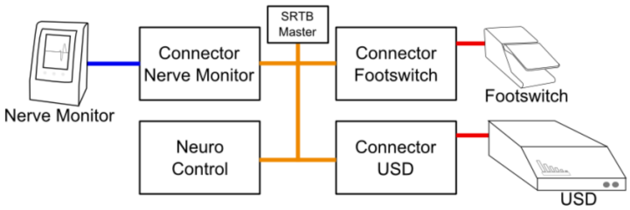

Figure 1: Overall network structure.

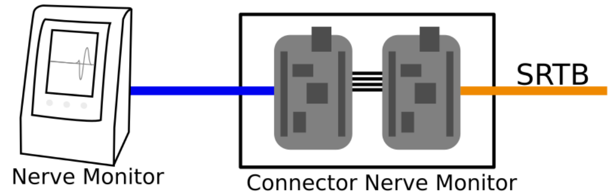

Figure 2: Structure of the connector nerve monitor consisting of two microcontroller units connected via a parallel interface.

Within this contribution, we show how a nerve monitor can be integrated into the SRTB network of medical devices to allow networked access to nerve response data. A function module Neuro Control, also connected to SRTB, receives the data and calculates the maximum allowed output power for connected devices. The calculated control signal is sent to the respective devices in real-time and their output is reduced or stopped instantly as soon as the IONM detects an externally stimulated nerve activity. By this means the system can provide a margin in patient safety.

2 Concept

Figure 1 illustrates the overall concept of our solution: The basic network consists of the SRTB master orchestrating the network, a footswitch and an ultrasonic dissector each connected to the SRTB via their respective connectors. In this state it is possible to control the USD via the footswitch as in a traditional setup. By additionally connecting a nerve monitor via the nerve monitor connector, all network participants will be able to receive the measured IONM signals. As an example module utilizing IONM data we add a function module called Neuro Control. After adding this function module to the network, the new assisting functionality becomes available and can be activated via the SRTB master's touch panel. Note that with exception of the USD instrument's tip, none of the medical devices (nerve monitor, footswitch, USD) had to be modified in order to implement the new functionality.

3 Implementation

In this paper we focus on the description of the two modules connector nerve monitor and function module Neuro Control. The nerve monitor connector (see Figure 3) integrates an inomed C2 nerve monitor into the SRTB network. As can be seen in Figure 2, the connector hosts two microcontroller units (BeagleBoneBlack, AM3356, ARMv7). One of the units connects to the nerve monitor and the other one connects to the SRTB network. Both units are connected via an internal real time capable interface. Two separate boards are used to limit the effects of possibly complex and hard to implement vendor specific nerve monitor interfaces. A faulty implementation may under no circumstances influence the real time network. This is ensured by distributing these functionalities onto two physically separated and independent microcontroller units. The vendor-specific nerve monitor interface is based on webservices and standard Ethernet/IP which by default does not meet the required deterministic real time characteristics: In the event of two concurrent data transmissions, unpredictable delays may occur. Our setup mitigates these effects by communicating over a point-to-point full-duplex connection.

The nerve-monitor-facing microcontroller runs a Linux distribution, provides a DHCP server and implements a webservice client. As soon as the nerve monitor is started, the DHCP server assigns an IP address and the webservice client tries to subscribe to the IONM data metrics. Whenever a metric value changes, the nerve monitor will notify the connector which immediately forwards the data to the SRTB-side microcontroller unit via an internal parallel interface. The second, SRTB-side microcontroller unit runs the QNX real time operating system as well as the openpowerlink EPL stack. It continuously receives IONM data via the internal interface and forwards this information to the SRTB.

Figure 3: Connector nerve monitor. The blue cable on the left connects to the nerve monitor. The black cable on the right connects to the SRTB.

Figure 4: Function module Neuro Control

The function module Neuro Control (see Figure 4) is built around a single BeagleBoneBlack board which runs QNX as well as the openpowerlink EPL stack. The module receives IONM data from the nerve monitor connector and decides whether the currently operated instrument needs to be stopped based on the IONM data. In case of the first prototype presented in this paper, the analysis of IONM data only relies on whether the nerve monitor has detected a nerve response. Therefore, the decision process is simple and fast enough to be executed between EPL cycles without further delay.

4 Evaluation

We evaluated our system with respect to the overall reaction time from start of simulating nerve responses to shutdown of the surgical instrument. By measuring the propagation delays between involved units separately, we were able to identify the segments of the data path where most time was spent.

The experiment setup is shown in Figure 5. It was equivalent to the system described so far (see Figure 1) with the addition of a nerve response simulation device and a time measurement testplatform. The nerve response simulation device output the pattern of a typical nerve response whenever a stimulation signal was applied. The testplatform was able to enable and disable simulated nerve responses by closing a relay that connects the stimulation probe to the nerve response simulation device. To measure the propagation delay of IONM data and the resulting control data within the network there were three testpoints TP1, TP2, TP3 located on the function module and connectors. These testpoints were connected to the testplatform and triggered a time measurement at the testplatform whenever the signal reached the respective testpoint. A fourth virtual testpoint

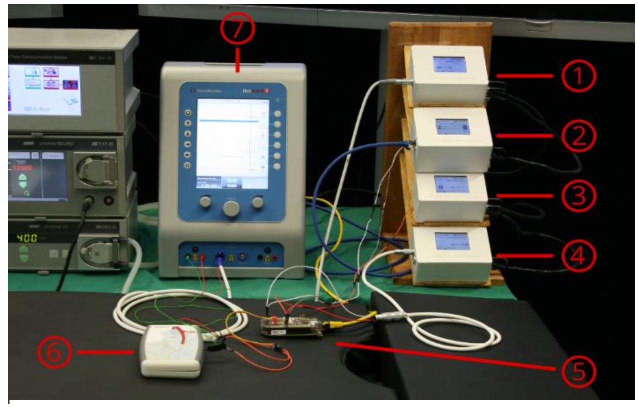

Figure 5: Experiment setup consisting of: (1) footswitch connector, (2) nerve monitor connector, (3) Neuro Control function module, (4) USD connector, (5) testplatform, (6) nerve response simulator, (7) nerve monitor

TP0 was activated by the testplatform itself when closing to relay to start simulating nerve responses. A detailed description is given in Table 1.

Table 1: Description of testpoints.

Testpoint	Location	Event which triggers time measurement
TP0	Testplatform	Start of simulated nerve responses
TP1	Connector Nerve Monitor	Information about presence of nerve response is received from the nerve monitor via OSCP-interface.
TP2	Function Module Neuro Control	Information about presence of nerve response reaches function module and shutdown of active instrument is ordered
TP3	Connector USD	USD is shut down

During the experiment, the testplatform performed the following steps automatically:

- Close relay to start simulating nerve responses
- Store timestamps at which testpoints are activated
- Wait until all testpoints have been activated and the instrument is shut down
- Open relay to disable simulating nerve responses
- Wait until the system has reached its initial state (no nerve response detected and instrument operating)

This process was repeated until 500 timestamp samples were gathered for each testpoint. After the experiment, the distribution of data propagation delays between testpoints was calculated from the testpoint activation timestamps. The results are shown in Table 2. The hardware timer used for timestamp acquisition had a resolution of 42ns. The EPL network's cycle time was set to 10ms.

Table 2: Distribution of measured data propagation delays.

Measured Duration	Average	Stddev	Min	Max
TP0->TP1	805.02ms	40.67ms	662.83ms	920.38ms
TP1->TP2	54.71ms	2.77ms	49.87ms	59.87ms
TP2->TP3	51.43ms	0.19ms	51.29ms	52.75ms

The results show that the biggest time delays occurred between TP0 and TP1. This is the period of time that the nerve monitor needed to detect the presence of a nerve response and to forward this information to the nerve monitor connector. As we had no control over the nerve monitor's internal behavior we could not conduct further measurements to determine how much of this time was spent on response detection and how much was spent on signal propagation. Propagating the IONM data to the Neuro Control function module (TP1->TP2) as well as propagating the shut-down-command to the instrument connector (TP2->TP3) was in the range of 50ms to 60ms. The reason for this is because a total of 5 EPL cycles were needed to transport data from one SRTB node to another. The delay between nerve monitor connector and Neuro Control function module (TP1->TP2) had an additional jitter of one complete EPL cycle (10ms) because the arrival of new IONM data at the nerve monitor connector was not synchronized with the EPL cycles. The magnitude of the SRTB node-to-node delay largely depends on the network's cycle time which was set to a relatively high value of 10ms in our research network due to a software EPL implementation and multiple buffering switches within the network. This value can be reduced by using hardware solutions to implement EPL communication.

5 Conclusion

In this paper we presented a method to integrate a nerve monitor into the SRTB real time network to provide IONM data to other network participants. Furthermore, we developed a function module that used this data to shut down active instruments when a nerve response was detected. Evaluation showed that the real time network was able to reliably transport IONM data within predefined time intervals.

Acknowledgment: The authors gratefully thank Arne Menz for display programming and Áron Borbáth for PCB layout.

Author's Statement

Research funding: This research was supported by the Federal Ministry of Education and Research of Germany. Conflict of interest: Authors state no conflict of interest. Informed consent: Informed consent is not applicable. Ethical approval: The conducted research is not related to either human or animals use.

References

[1] Green J, Shelton C, Brackmann D. Iatrogenic facial nerve injury during otologic surgery. *Laryngoscope*. 1994.

[2] Silverstein H, Smouha E, Jones R. Routine identification of the facial nerve using electrical stimulation during otological and neurotological surgery. *Laryngoscope*. 1988;98(7):726-730.

[3] Shiban E, Krieg SM, et al. Intraoperative subcortical motor evoked potential stimulation: how close is the corticospinal tract? *J Neurosurg*. 2015;123(3):711-720.

[4] Shiban E, Krieg SM, Obermueller T, Wostrack M, Meyer B, Ringel F. Continuous subcortical motor evoked potential stimulation using the tip of an ultrasonic aspirator for the resection of motor eloquent lesions. *J Neurosurg*. 2015;123(2):301-306.

[5] Strauss M, Strauss G, Trantakis C, et al. NCF - Concept of an EMG based power control of active instruments for applications in ENT and neurosurgery. 2007.

[6] Shi J, Lueth TC. Data fusion of Electromyography signals from intraoperative nerve monitoring and position information from surgical navigation. In: *IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems*. IEEE; 2012:287-292.

[7] Shi J, Strauss G, Heininger S, Lueth TC. Surgical assistance for instruments' power control based on navigation and neuromonitoring. *Annu Int Conf IEEE Eng Med Biol Soc*. 2011;2011:2115-2118.

[8] Pfeiffer JH, Dingler ME, Dietz C, Lueth TC. Requirements and architecture design for open real-time communication in the operating room. In: *2015 IEEE International Conference on Robotics and Biomimetics (ROBIO)*. Zhuhai, China: IEEE; 2015:458-463.

[9] Lueddemann T, Pfeiffer JH, Dingler ME, Dietz C, Lueth TC. *Verbundabschlussbericht OR.NET - Sichere Dynamische Vernetzung in Operationsaal Und Klinik*. DOI: 10.2314/GBV:873806719, Munich; 2016.

[10] Kasparick M, Schlichting S, Golatowski F, Timmermann D. New IEEE 11073 standards for interoperable, networked point-of-care Medical Devices. In: *2015 37th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*. IEEE; 2015:1721-1724.

[11] Dietz C, Lueddemann T, Dingler ME, Lueth TC. Automated risk detection for medical device networks with hard real time requirements. In: *2016 IEEE/SICE International Symposium on System Integration (SII)*. Sapporo, Japan: IEEE; 2016:471-476.