DEVELOPMENT OF AN INTEGRATED CONTROL SYSTEM FOR ANAESTHESIA AUTOMATION

J. M. Lemos * **N. Roma** * **T. Mendonça** ** **L. Sousa** * **B. Costa** *
C. S. Nunes † **S. Esteves** *** **P. Amorim** *** **L. Antunes** ****

* *INESC-ID/IST, R. Alves Redol 9, 1000-029 Lisboa, Portugal*
** *FCUP, DMA, R. do Campo Alegre 687, 4169-007 Porto, Portugal*
*** *HGSA, Largo Prof. Abel Salazar, 4099-001 Porto, Portugal*
**** *CECAV-UTAD, Apartado 1013, 5001-801 Vila Real, Portugal*
† *Division of Engineering, King's College London, U. K.*

Abstract: This paper describes the architecture of a system that integrates several variables for automation of anaesthesia. The system is structured in two main units, one mainly concerned with feedback control and the other with supervision functions. For each block, the problems and the corresponding design methods are discussed.

Keywords: Control Applications, Biomedical Engineering, Anaesthesia, Embedded systems.

1. INTRODUCTION.

Progress in sensor and actuator technology, as well as in modeling of physiological systems is opening more and more wide fields of application to automatic control. In particular in the field of anaesthesia, major efforts have been made in the recent past in order to develop new tools to monitor the brain activity for evaluating the level of consciousness. The introduction of the Bispectral index from the electroencephalogram (BIS) or the entropy measure (EM) for measuring the degree of unconsciousness (Depth of Anaesthesia - DoA) are examples of such improvements(Martorano *et al.*, 2006). The level of neuromuscular blockade (NMB) is another important variable that can be measured (Kalli, 2002) and, in both cases (DoA and NMB) the dynamics may be described by underlying pharmaco-kinetic/pharmaco-dynamic (PK/PD) models (Bailey *et al.*, 2005).

The existence of adequate monitors (sensors) for these variables and actuators, together with an increasingly deeper understanding of the underlying phenomena dynamics, allows the automation of anaesthesia.

Both the level of unconsciousness (Struys *et al.*, 2001; Absalom *et al.*, 2001; Gentilini *et al.*, 2001; Mahfouf *et al.*, 2003) and the neuromuscular blockade level (Jaklitsch *et al.*, 1987; Kansanaho *et al.*, 1996; Mendonça *et al.*, 1998; Mason *et al.*, 1999; Lemos *et al.*, 2005(a)) have been, and are currently being, the subject of theoretical and practical clinical studies concerning the application of feedback control. The recognition that anesthesia variables interact and depend of a complex system lead to the consideration of controllers were these variables are integrated (Araki *et al.*, 2005).

According to the above motivations, the objective of project IDEA [3] is the development of an autonomous integrated system for the automation of anaesthesia that incorporates advanced control algorithms able to tackle the specific challenges of anaesthesia.

The control unit, designed according to the methodology of embedded systems (Crespo *et al.*, 2007), so as to be able to be translated to a clinical device, con-

---

[1] Email of the corresponding (first) author: *jlml@inesc.pt*

[2] This work has been done under the framework of project IDEA – Integrated Design for Automation of Anaesthesia, contract PTDC/EEA-ACR/69288/2006.

[3] For further information on the project IDEA see the Web page http://ramses.inesc.pt/IDEA

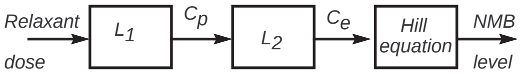

Fig. 1. Block diagram of the NMB dynamics.

sists of a multiprocessor unit that can be connected to standard anaesthesia sensors and actuators (perfusion syringes) and comprises two blocks.

One of the blocks performs the functions associated to control and comprises control algorithms embedded in a Digital Signal Processor that decide the dose of drugs to administer such that the physiological variables of the patient to control are close to the desired target values. The variables to control are the neuromuscular blockade (NMB) and the depth of anaesthesia (DoA) levels.

The other major block consists of a supervisor, performing the functions of a Fault Detection Monitor (FDM). This block verifies the correctness of the actions undertaken by the controller block, while being able to include the adequate high level information on the patients state to the practitioner. In particular, the FDM will detect malfunctions of sensors and actuators (e. g. blocking, occurrence of outliers or unexpected behaviour), oscillatory and other poor performance behaviour of the feedback loop.

The contribution of this paper consists in the description of the structure of the integrated anaesthesia automation control system, including a discussion of the different problems associated to it and the corresponding design methods.

the paper is organized as follows: After the introduction, anaesthesia is described as a controlled system in section 2. This is the basis for structuring the control unit. Section 3 is devoted to cost-benefit and risk analysis of anaesthesia automation. Section 4 describes the proposed system architecture from an hardware/software point of view and section 5 discusses the algorithms to use. Finally, section 6 draws conclusions.

2. ANAESTHESIA AS A CONTROLLED SYSTEM

A system's description of anaesthesia is an essential preliminary step for designing the integrated control unit. This description includes listing input, output, disturbances, the dynamic description of these variables (models) as well as their interaction.

Fig. 1 shows a model currently assumed to represent the dynamics of NMB. It consists of a linear compartment model relating the relaxant drug (*e. g. atracurium* dose with the plasma compartment concentration C_p (pharmaco-kinetic model) and the effect compartment concentration C_e (blocks L_1 and L_2) in series with a static nonlinearity described by the Hill equation. For further details see (Mendonça *et al.*, 1998; Magalhães, 2006). Fig. 2 shows a typical record of NMB after the administration of a relaxant (*atracurium* bolus.

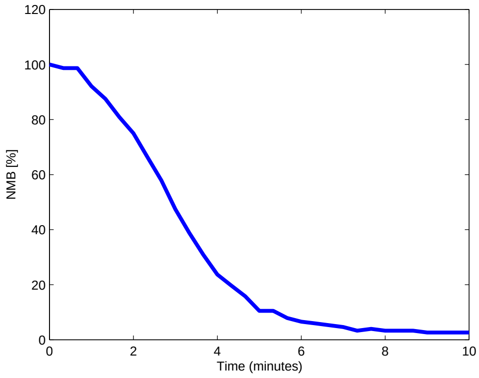

Fig. 2. Time Record of NMB induced by a bolus in a clinical case.

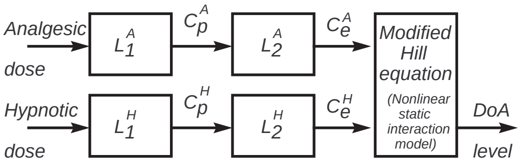

Fig. 3. Block diagram of the DoA dynamics.

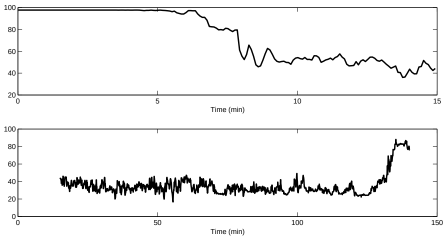

Fig. 4. Time record of DoA (BIS) in a clinical case. The record is divided in two time periods with the time scale expanded above.

Fig. 3 shows a similar model for DoA (Bailey *et al.*, 2005; Minto *et al.*, 1997). The main difference consists in the interaction of the hypnotic with the analgesic drug, whose dose can be seen as an accessible disturbance. The Hill equation is thus modified in order to reflect this fact. Fig. 4 shows a typical evolution of DoA, as measured by the BIS signal during an entire surgery.

In what concerns the structure of the control system, the main variables are thus:

Measured outputs to control:

- NMB
- DoA

Manipulated variables:

- Relaxant dose
- Hypnotic dose

Accessible disturbances:

- Analgesic dose

It should be mentioned hat there are other types of disturbances and interactions, as well. An important example is provided by blood loss that affects dynamics and causes a leak in the amount of drugs administered. Furthermore, the surgical stimulus may also change the need from drugs and, often, there are no clear clinical responses collected by the sensor.

The structure of the control system reflects these choice of variables.

3. COST-BENEFIT AND RISK ANALYSIS

The main benefits of automated anaesthesia are:

- The anaesthetist is relieved from repetitive tasks and has more capacity to concentrate on patient supervision and on the decisions requiring expert reasoning capabilities;
- Automation follows more systematic procedures, ensuring a tighter tracking of the target values of the physiological variables to regulate, together with a reduction of the quantity of drugs administered;
- In animal surgery, where for economic reasons the surgeon many times has to play the double role of anaesthetist, an important aid will be provided by the automatic system, thereby improving the quality of anaesthesia and animal welfare, while keeping the cost low.

The main risks are inherent to the connection of an automatic controller to a living body. The decisions taken by the technical device should in no case harm the human body. Wrong decisions may result from faults occurring in the sensors, the actuators, the processing unit (hardware/software) or in the communication links. They may also be a consequence of algorithms, *e. g.* a controller exiting the stability domain. Embedding fault tolerance in control systems implies some form of redundancy and or supervision (Blanke *et al.*, 2001; Izadi, 1999). These concepts have been already applied to anaesthesia (Frei, 2000), but there are still many unexplored problems. The system to develop relies on control structures that allow a natural embedding of fault tolerant features, such as switched multiple model adaptive control (SMMAc) (Lemos *et al.*, 2005(a)) and Model Predictive Control (Maciejowski *et al.*, 2001).

To increase safety, animal model may be employed. The testing and validation of the partial algorithms and the overall system for automatic anaesthesia in rats should increase safety before a major translation into humans.

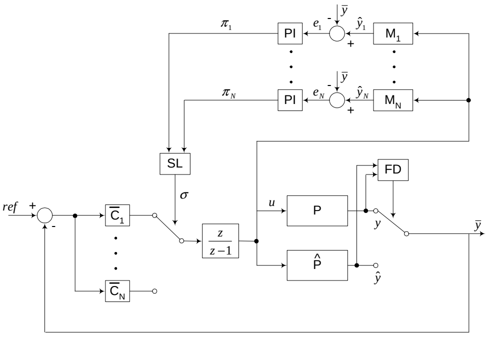

Fig. 5. Switched multiple model adaptive controller.

4. CONTROL AND SUPERVISION ALGORITHMS

The major problems to tackle by the control algorithms are how to deal with uncertainty, embedding fault tolerance and satisfying constraints. The control algorithms considered exploit the connection between Model Predictive Control (MPC) and Switched Multiple Model Adaptive Control (SMMAC) in order to take advantage of the specific features of the critical application considered, *e. g.* the fact that one is dealing with positive systems (Magalhães, 2006).

Besides its predictive features, MPC allows to easily incorporate constraints and feedforward from accessible disturbances, three key aspects for the control of anaesthesia. On the other side, the great variability inter and intra patient calls for the use of adaptive methods. SMMAC provides the desired adaptive features, including the possibility of controller reconfiguration, an important issue in fault tolerant control, as well as tackling signal artifacts (Lemos *et al.*, 2005(a)).

Fig. 5 shows the block diagram employed with SMMAC. This comprises a bank of controllers C_1 to C_N and a supervisor that switches among them. Fig. 6 shows results obtained with this algorithm, where the temporary sensor faults are removed with a Baysean filter.

Other forms of adaptation are possible. Fig. 7 shows simulation results when DoA is controlled with MUSMAR, a predictive adaptive controller that tackles uncertainty by using redundant multiple models that are combined on-line (Nunes *et al.*, 2006). Another approach consists in combining a hybrid identification method (Aloso *et al.*, 2008), used as an observer of drug plasma concentration, with an estimated cascade feedback controller. The hybrid method is a neural network based technique for on-line parameter estimation, that has proven to improve the identification and prediction of the individual patient response to drug administration in real patients.

5. SYSTEM ARCHITECTURE

From an hardware/software point of view, the block diagram of the system is presented in fig. 8. It con-

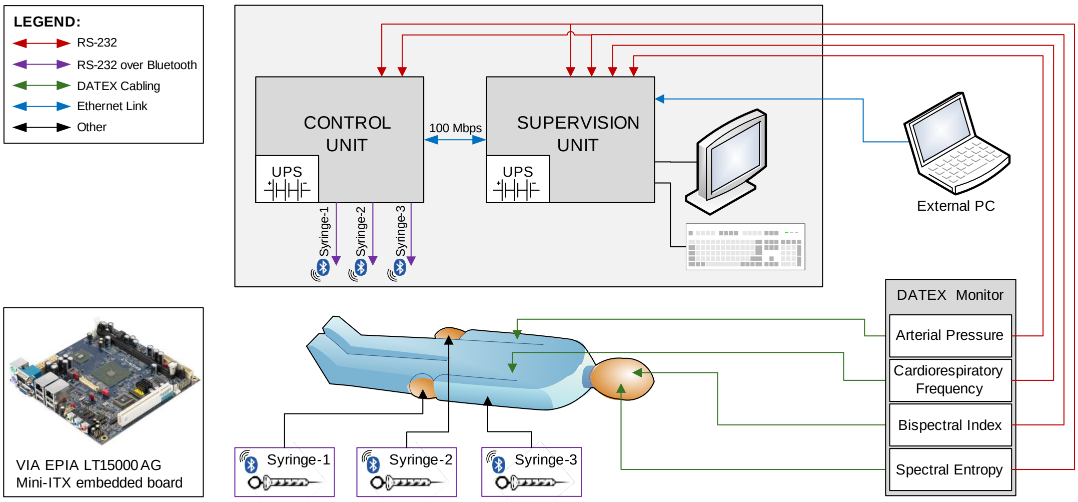

Fig. 8. Block diagram of the implemented platform.

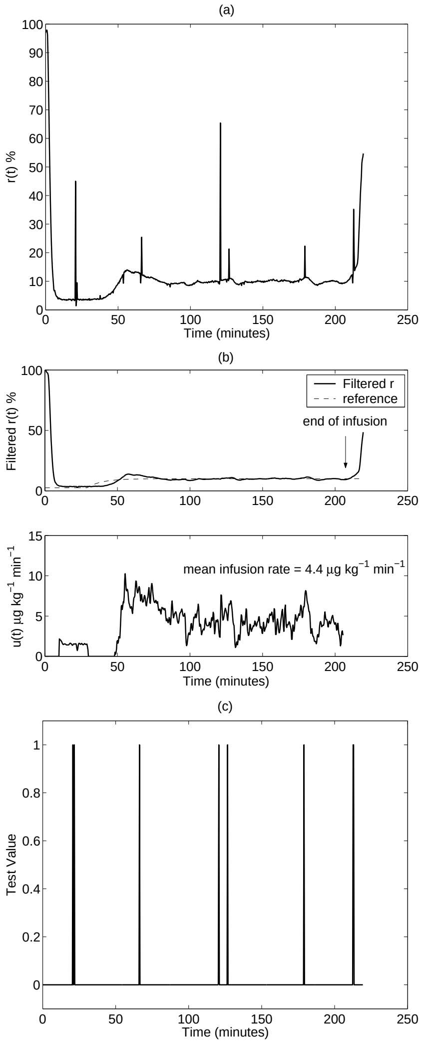

Fig. 6. Clinical results obtained with a Bayesian filter in which the signal is reconstructed using the currently active model from the bank. (a) - Sensor output measurement $r(t)$ $[\%]$, (b) - filtered $r(t)$ $[\%]$ and control action $u(t)$ $[\mu g\, kg^{-1}\, min^{-1}]$, (c) - test value.

sists of an autonomous platform with two processing units. One unit is dedicated to the implementation of the control algorithms, thus requiring the capability to perform high-precision floating-point computations. The other unit implements a supervision task of the whole procedure, gathering data and information directly from the sensors and from the control unit to detect faults. According to the gathered information, this unit can dynamically change the processing parameters for the control algorithm.

5.1 *Hardware*

The implementation of the control system was conducted by integrating several modules that support the processing and communication parts of the platform. All these modules must comply with the electrical equipment standards and regulations applicable to the surgery room scenario.

5.1.1. *Processing units*

To comply the developed platform not only with the space restrictions that characterize the surgery room but also with the low-power requirements that are imposed to confer portability characteristics to this system, each processing unit was implemented using a VIA EPIA LT Mini-ITX embedded board (Via Technologies Inc., 2007), based on a low-power VIA C7 processor. This processor architecture runs at 1.5GHz, with 1GB DDR2 memory RAM, and has proved to be highly capable to implement both the control and the supervision tasks. An optimized Linux operating system was provided in both of these units, in order to support an easy integration and interface with the other modules of the system architecture.

5.1.2. *Communication links*

To provide a fast and reliable communication between the control and supervision units, the two processing boards were connected with a 100 Mbps ethernet

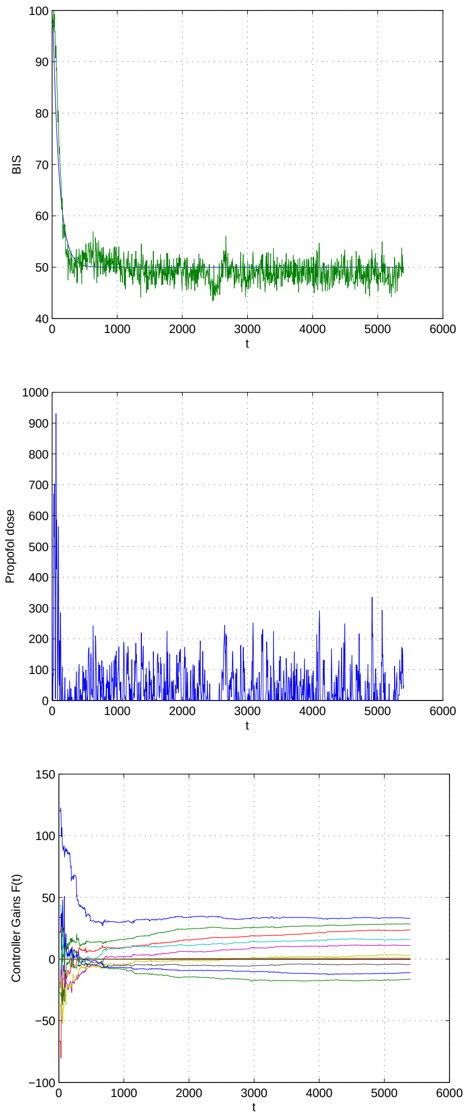

Fig. 7. Predictive Adaptive Control of DoA (simulation BIS index (above), propofol dose (manipulated variable, middled), and controller gains (bellow) when the dose of remifentanil $[ml/hr]$ is used as accessible disturbance. The remifentanil dose has a step change in the midle of the record.

link. On the other hand, serial RS-232 connections were adopted to connect each board with the existing monitoring devices. Furthermore, to reduce the cable burden at the surgery scenario and to guarantee a perfect galvanic electrical insolation, the RS-232 links between the control unit and the perfusion syringes were implemented over bluetooth wireless links. Finally, an extra ethernet connection was also provided to enable an optional communication link between the developed platform and an external PC.

5.1.3. *User interface*

The interface with the anaesthesiologist is carried out with a touch-screen LCD, where the physician gathers the several data and information concerning the control process and where he can insert most of the commands related to the anaesthesia automation mechanism. More complex procedures that require a more complex user interface may be conducted by means of a keyboard connected to the supervision unit.

5.1.4. *Redundancy*

To maximize the system reliability, some of the most important control operations were redundantly implemented in both units. In the event of a critical failure in any of the two units, the overall control procedure may still be assured by the remaining unit. Furthermore, to guaranty an uninterruptable operation of the system in case of power failure, as well as to confer portability characteristics to the platform, each processing unit was equipped with a redundant battery pack that is able to supply the system for the amount of time required to recover the main power supply or to move the patient to another surgery facility.

5.2 *Software*

Starting from the project requirements, the model for the anesthesia software will comprise the following modules:

- The user interface module, to exchange information with the user.
- The database module, to store the configuration data and additional information which is collected during the system operation.
- The communication module, which supervise and control communications with sensors and actuators.
- The high-level supervisor module, which as the task of supervising the operation and execution of anesthesia control algorithms.
- The hardware abstraction module which represents the embedded operating system.

Due to the critical role that software plays in the anesthesia system, an analysis of hazards and risk (Rakitin, 2006) must be performed to guide the development process. This will guide the development and the inclusion of safe mechanisms to deal with the possibility of faults and to force the anesthesia system to fail safe.

As a consequence, formal methods (Jetleu *et al.*, 2006) must be employed such as state machines or Petri nets, to define the allowed states and transitions.

Software implementation is dependent of the operating system and the computer programming languages. The Linux operating system will be selected to run on the embedded boards due to its stability, support by the manufacture of the embedded boards, open-source and royalties free, however it will be strip down to the essential services. As a first approach, language C will be used for programming when execution time is a priority, but other languages such as Python can be explored to speed-up the prototyping phase such in the user interface definition.

6. CONCLUSIONS

Automation of anaesthesia requires the associated physiological variables to be seen as a system and leads to the use of advanced controllers. This, together with stringent fault tolerance constraints, suggests an embedded systems design approach. In this framework, the design of an integrated system for automation of anaesthesia is presented as an interdisciplinary biomedical engineering problem with contributions from the fields of Computer Architectures, Instrumentation, Mathematical modeling of physiological systems, Anaesthesiology and Control. Preliminary solutions to different problems involved are presented.

7. REFERENCES

Absalom, A. R.; N. Sutcliffe and G. N. Kenny. Closed-loop control of anesthesia using bispectral index: Performance assessment in patients undergoing major orthopedic surgery under combined general and regional anesthesia. *Anesthesiology*, 96(1):67-73, 2002.

Alonso, H., Mendonça, T., Rocha, P. (2008). A hybrid method for parameter estimation and its application to biomedical systems, Computer Methods and Programs in Biomedicine, 89(2), 112-122, DOI 10.1016/j.cmpb.2007.10.014

Araki, M. and E. Furutani (2005). Computer control of physiological states of patients under and after surgical operation. *Annual Reviews in Control*, 29:229-236.

Bailey, J. and W. Haddad (2005). Drug dosing in clinical pharmacology: paradigms, benefits, and challenges, *IEEE Control Syst. Mag.*, 25(2):35-51.

Blanke, M.; C. Frei, F. Kraus, R. Patton and M. Staroswiecki (2001). Fault Tolerant Control Systems, ch. 8 of *Control of Complex Systems*, K. Åstrom *et al.* (eds.), 165-190, Springer.

Crespo, A.; P. Albertos and J. Simo (2007). Embedded Control Systems: From design to implementation. *Proc. Cost Orienteds Automation*, 8, part 1, Habana, Cuba.

Frei, C. (2000). *Fault tolerant control concepts applied to anesthesia*, Ph. D. Dissertation ETH No. 13599, Swiss Federal Institute of Technology Zurich.

Gentilini, A.; M. Rossoni-Gerosa, C. W. Frei, R. Wymann, M. Morari, M. Zbinden and T. W. Schnider (2001). Modelling and closed-loop control of hypnosis by means of byspectral index (BIS) with isoflurane. *IEEE Trans. Biomed. Eng.*, 48:874-889.

Izadi-Zamanabadi, R. (1999). *Fault-tolerant Supervisory Control - System Analysis and Logic Design*. Ph. D. thesis, Aalborg University, Department of Control Engineering, Aalborg, Denmark.

Jaklitsch, R. and D. Westenskov (1987). A model based self adjusting two-phase controller for *vecuronium*-induced muscle relaxation during anesthesia. *IEEE Trans. Biom. Eng.*, BME-34:583-594.

Jetleu, R.; S. P. Iyer and P. L. Jones (2006). A Formal Methods Approach to Medical Device Review. *IEEE Computer*, April 2006, 61-67.

I. Kalli (2002). Neuromuscular block monitoring. In: *Clinical Anesthesia Practice*, R. Kirby, N. Gravenstein, E. Lobato and J. Gravenstein, Eds. Chap. 22, 442-451. W. B. Saunders Company.

Kansanaho, M. and K. Olkkola (1996). Performance assessment of an adaptive model-based feedback controller: Comparison between *atracurium*, *mivacurium*, *rocuronium* and *vecvuronium. J. Clin. Monitor., Comput.*, 13(4):217-224.

Lemos, J. M.; H. Magalhes, T. Mendonça and R. Dionísio (2005). Control of neuromuscular blockade in the presence of sensor faults, *IEEE Trans. Biomed. Eng.*, 52(11):1902-1911.

Lemos, J. M.; T. Mendonça, H. Magalhães and R. Dionísio (2005-a). Control of neuromuscular blockade in the presence of sensor faults. *IEEE Trans. Biomedical Eng.* 52:1902-1911.

Maciejowski, J. and J. M. Lemos (2201). Predictive Methods for FCT, ch. 8 of Control of Complex Systems, K. Åstrom *et al.* (eds.), 229-240, Springer.

Magalhães, H. F. (2006), *Contributions to neuromuscuular blockade control*, Ph. D. dissertation, Faculdade de Ciências da UNiversidade do Porto, Portugal.

Mahfouf, M.; A. J. Asbury and D. A. Linkens (2003). Unconstrained and constrained generalised predictive control of depth of anaesthesia during surgery. *Control Eng. Practice*, 11:1501-1515.

Martorano, P., Falzetti, G., Pelaia, P. (2006). Bispectral index and spectral entropy in neuroanesthesia. *J. of Neurosurgical Anesthesiology*, 18(3): 205-210.

Mason, D.; J. Ross, N. Edwards, D. Linkens and C. Reilly (1999). Self-learning fuzzy control with temporal knowledge for *atracurium*-induced neuromuscular block during surghery. *Comput Biomed. Res.*, 32(3):187-197.

Mendonça, T. and P. Lago (1998). PID control strategies for the automatic control of neuromuscular blockade, *Control Eng. Pract.*, 6(10):1225-1231.

Minto, C.; T. Schnider, T. Egan, E. Youngs, H. Lemmens, P. Gambus, V. Billard, J. Hoke, K. Moore, D. Hermann, K. Muir, J. Mandema and S. Shafer (1997). Influence of age and gender on the pharmacokinetics and pharmacodynamics of remifentanil. I. Model development, *Anesthesiology*, vol. 86(1):10-23.

Nunes, C. S.; T. Mendonça, H. Magalhães, J. M. Lemos and P. Amorim (2006). Predictive Adaptive Control of the Bispectral Index of the EEG (BIS) using the intravenuous anaesthetic drug propofol. *Lecture Noters in Artificial Intelligence*, 4252:1248-1255.

Rakitin, S. R. (2006). Coping with Defective Software in Medical Devices. *IEEE Computer*, April 2006, 40-45.

Struys, M.; T. de Smet, L. Versichelen, S. Van de Vilde, R. Van der Brocke and E. Martier (2001). Comparison of closed-loop controlled administration of propofol using BIS as the controlled variable versus "standard practice" controlled administration. *Anesthesiology*, 95:6-17.

Via Technologies Inc. EPIA LT-Series Mini-ITX Mainboard - Operating Guide, version 1.01, 2007.