Faculdade de Engenharia da Universidade do Porto

Departamento de Engenharia Electrotécnica e de Computadores

Dissertação de Mestrado

Mestrado em Engenharia Biomédica

Tracking the NMB level via a switching system mass control strategy

Miguel Ramos Teixeira

Julho de 2012

Faculdade de Engenharia da Universidade do Porto

Departamento de Engenharia Electrotécnica e de Computadores

Dissertação de Mestrado

Tracking the NMB level via a switching system mass control strategy

Autor: **Miguel Ramos Teixeira (MEB09018)**

Orientadora: **Prof. Paula Rocha**
Co-Orientadora: **Prof. Teresa Mendonça**

Dissertação submetida à Faculdade de Engenharia da Universidade do Porto para obtenção do grau de Mestre em Engenharia Biomédica

Julho de 2012

To my parents

"The possibility to realize a dream
is what makes life interesting"
Paulo Coelho

Acknowledgments

In this final way to obtain my master degree it is more than simple words of thanks that I owe to many people, people whose presence in my life helped me to achieve this goal. I hope that these words could be enough to translate all my gratitude.

First of all, I would like to express my thankfulness to my supervisors, Professor Paula Rocha (Faculdade de Engenharia da Universidade do Porto) and Professor Teresa Mendonça (Faculdade de Ciências da Universidade do Porto) for their guidance, support, availability and friendship during this past year and half. I also thank their revisions, comments and suggestions regarding the improvement of this thesis.

To Dr. Rui Rabiço (anesthesiologist at Unidade Local de Saúde de Matosinhos - Hospital Pedro Hispano) I am thankful for the possibility that he provided me to follow closely all the work related to general anesthesia, and the chance to perform the work developed in my MSc thesis.

Also I want to thank all the teachers involved in my master course for all the knowledge provided during these last two years

To the institutions Faculdade de Engenharia da Universidade do Porto and Faculdade de Ciências da Universidade do Porto I am thankfull for the infrastructure of laboratories, study rooms, library and database provided.

I owe a special gratitude to my parents for the good home environment, essential for the stability that I needed to carry out my work, for their love, support and for their sacrifices in order to give me the opportunity to continue my studies. To them I will always thank for all I become.

I am grateful to my brother for the continuous presence in my life, being a great companion and a useful lever to achieve my goals.

I am thankful to my friends, specially to Pedro Ferreira, Carlos Faria and Tiago Esteves for their fellowship, support, knowledge and for the entertainment they provided me.

Last but not least, I would like to thank to Ana Bela, for being a complementation of myself, for being the last voice I heard during these last months when writing my MSc thesis, being always a great support and encouragement.

Abstract

Over the past 35 years the development of strategies that allow to automatically control the patient neuromuscular blockade (NMB) has met some important advances. This development allowed the evolution from a simple on-off controller to intelligent control schemes based on a variety of theories, such as the adaptive and model-based controllers.

Despite the vast work previously done in the area of NMB control, some problems still exist, like a very high degree of uncertainty in the system dynamics due to the intra- and intervariability in the patients and nonlinearities, and it is still desirable to achieve a better anesthetic solution. Therefore further study around this theme in order to overcome the drawbacks of the manual control as well as the limitations of the previous NMB control strategies will have a high impact in the anesthetic community.

This work presents the implementation of a model-based switching control strategy to drive the NMB level of patients undergoing general anesthesia to a predefined reference. *SISO* (*Single-Input-Single-Output*) *Wiener* models with a compartmental part, describing the pharmacokinetics and pharmacodynamics linear effect together with a static nonlinearity are used to model the NMB response of two muscle relaxants, *atracurium* and *rocuronium*. The switching controller is designed based on a bank of total system mass control laws, where each of such laws is tuned for an individual model from a bank of models developed to represent the behavior of the whole population. Moreover a scheme to improve the reference tracking quality based on the analysis of the patient's steady-state response is presented.

Keywords: Automatic NMB Control, Compartmental Models, Switching Control, Total System Mass Control Law, Reference Tracking Improvement.

Resumo

Durante os últimos 35 anos o desenvolvimento de controladores que permitem o controlo do nível de bloqueio neuromuscular (NMB) de um paciente tem conhecido importantes avanços. Graças a estes desenvolvimentos foi possível evoluir desde simples controladores on-off até controladores inteligentes baseados numa larga gama de teorias, tais como os controladores adaptativos e os controladores baseados em modelos.

Apesar do enorme volume de trabalho que se tem vindo a desenvolver na área do controlo do NMB, ainda se constata a presença de alguns problemas tais como o elevado grau de incerteza na dinâmica dos sistemas devido à inter- e intravariabilidade dos pacientes e a não-linearidades. Embora tenha sido proposto até hoje um elevado número de controladores, ainda é desejável alcançar-se uma melhor solução. Deste modo uma contínua aposta em estudos nesta área, de forma a ultrapassar as desvantagens do controlo manual bem como as limitações das estratégias de controlo automáticas já desenvolvidas, tem certamente um enorme valor para a comunidade anestésica.

Este trabalho apresenta a implementação de uma estratégia de controlo por comutação baseada em modelos de forma a controlar a resposta de NMB de um paciente sujeito a cirurgia geral para um valor de referência desejado. Modelos de *Wiener* com uma entrada e uma saída, constituidos por um sistema compartimental, que descreve o efeito farmacocinético e farmacodinâmico linear, em série com uma não linearidade estática, são utilizados para modelar a resposta do NMB à administração de dois relaxantes musculares, o *atracurium* e o *rocuronium*. O controlador por comutação é construído com base num banco de leis de controlo de massa total, estando cada uma dessas leis sintonizada para um modelo de um banco de modelos desenvolvido de forma a representar o comportamento da população. Além disso, é apresentada uma estratégia de melhoramento do seguimento de referência, baseada na análise da resposta NMB do paciente em estado estacionário.

Palavras-Chave: Controlo Automático do NMB, Modelos Compartimentais, Controlo por Comutação, Lei de Controlo de Massa, Melhoramento do Seguimento de Referência.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Motivation

Health related research has accomplished major advances in the last decades. In this period the world has witnessed an exponential development in this area and the tendency is to continue at a faster scale.

Health care is one of the basic foundations for a good quality of life. So any improvement in this area is a great social accomplishment. In this context, surgical procedures have a crucial importance as a way of treating and correcting a very large range of health problems.

Anesthesia is a key factor to the success of a surgery since it provides good conditions for surgical activity. Therefore, there is a considerable interest in research in this area, leading to procedures that can assist and simplify the anesthesiologists work as well as for the surgeons.

General anesthesia is achieved by the administration of three types of drugs, a hypnotic, an analgesic, and a muscle relaxant. Usually, these drugs are manually administered through boluses given by the anesthesiologists during the surgery. Automatic methods for drug administration can make this procedure smoother for the patient and lead to less waste of drug.

The present work was carried out within the research project GALENO - Modeling and control for personalized drug administration (PTDC/SAU-BEB/103667/2008), which aims at designing personalized drug administration system using modeling, estimation, control and advisory methods.

1.2 Aims

The automatic control of the neuromuscular blockade (NMB) level (one of the components of the general anesthesia) can be achieved with a proper positive control law, designed with basis on compartmental systems models that describe the behavior of the muscle relaxants inside the body, [Almeida (2010)]. This not only drives the total drug mass in a patient to a desired reference but also drives the NMB level to a specific target.

The present work is aimed at:

- Presenting a total system mass control law in order to perform the automatic control of the NMB level in patients with a suitably identified model.
- Developing of a switching strategy to control the NMB level in patients whose model parameters are unknown.

1.3 Previous Work and State of the Art

Over the past 35 years the development of the tunning of controllers for systems characterized by the presence of non-linearities and large uncertainties has met some important advances. A wide range of controllers from a simple on-off controllers [Vries et al. (1986), Wait et al. (1987)] or PID controllers [Lago et al. (1998)] until intelligent control schemes based on a variety of theories, like the adaptive, model-based, fuzzy and robust controllers has been achieved amoung the last years.

Cass et al. (1976) reports one of the first cases of the use of a PID controller (Proportional–Integral–Derivative controller) which the most commonly used is the feedback controller. This case consisted in the administration of four non-depolarizing muscle relaxants, *d-tubocarine*, *gallamine*, *alcuronim* and *pancuronium* to sheep with a pre-defined target of 40% and with an infusion limited to a maximum of one hour. This led to the conclusion that an automatic control scheme in order to control the NMB level is achievable in clinical practice. Other authors such Asbury and Linkens (1986), O'Hara et al. (1991), and Lago et al. (1998), have also developed work in the area of the PID controllers.

The adaptive model-based closed-loop systems bring a big advance in the control of neuromuscular blockade with a wide range of advantages. The development of these models was possible due to the publication of pharmacokinetic/pharmacodynamic (PK/PD) models, which are assumed to accurately describe the actual behaviour of NMB agents inside the body. Once these models were presented, a variety of feedback control schemes designed to lead the NMB to a desired target have appeared. During the past years authors like Bradlow et al. (1986), Jaklitsch and Westenskow (1987), Uys et al. (1988), Lendl et al. (1999) and many others have

proposed different feedback schemes in order to present a suitable strategy to control the NMB level.

Regarding fuzzy logic model based methods, there are many authors that explored this approach as a simple mean to build a nonlinear controller. Here, Mason et al. (1994), Mason et al. (1996), Mason et al. (1999), Ross et al. (1997), Edwards et al. (1998), and Shieh et al. (2000) are some of the many authors that developed this method. Fuzzy logic control consists from a structural point of view, in a set of rules built via guidelines normally provided by an experienced operator well aware of the proccess under study. Consequently, the controller looks at the process as a black box and does not need an accurate model description.

The project *GALENO - Modeling and control for personalized drug administration (PTDC/SAU-BEB/103667/2008)*, a project financed by the *FCT - Fundação para a Ciência e a Tecnologia*, has the objective of designing personalized drug administration systems using modelling, estimation, control and advisory methods. Over the past years, it has seen several developments regarding the automatic control of the neuromuscular blockade with the participation of several researchers which have worked in different ways to pursue the main goal of this project. Adaptive control methods were studied in (Lemos et al. (2002a), Lemos et al. (2002b), Lemos et al. (2003), Lemos et al. (2005a), Lemos et al. (2005b), and Lemos et al. (2006)). These methods consist in using a parameter dependent controller whose parameters are updated with basis on the controller system response. PID controllers, switching strategies and observers were studied in (Magalhães et al. (2002a), Magalhães et al. (2002b), Magalhães et al. (2004), Magalhães et al. (2005), and Magalhães (2006)). Observers consist on the modelling of a real system in order to provide an estimation of its internal state. This modelling is achieved through measurements of the input and output of the real system. The NMB control with positive total system mass control laws applied to systems with uncertainties were studied in (Sousa et al. (2007), Sousa et al. (2008), and Sousa et al. (2010)). Finally, the total system mass control laws applied to positive systems were studied in (Almeida et al. (2010), Almeida et al. (2011), and Almeida (2010)). All this developed work allowed a major breakthrough on the automatic control of the NMB with a reasonable number of techniques already applied in the surgery room that have shown acceptable results.

1.4 Contributions

Several techniques are applied and tested to perform, both online and offline, the automatic control of the NMB level.

Regarding online NMB control it was addressed the total system mass control law with

controller selection based on two techniques. The first one is the switching strategy and the second one is the Extended Kalman Filter (EKF) used to perform the model identification and then tuned the corresponding total system control law.

The offline NMB control was also based on the application of the total system mass control law, but the controller selection was performed with different methods. For controller selection it was used the controller with the better reference tracking, the controller of the model with the closest parameters to the ones of the patient model, the controller of the model with the value of steady-state input (U_{SS}) required to achieve a desired reference closest to the one of the patient model, the controller of the model with the value of *Norm 2* closest to the one of the patient model, the controller of the model with the value of *Vinnicombe* metric closest to the one of the patient model, the controller of the model with the closest *Impulse Response* to the one of the patient model, and the controller of the model with the closest *Step Response* to the one of the patient model.

Comparisons between methods are presented and discussed.

Some work related to the automatic control of NMB level through total system mass control laws using a switching strategy was published in Teixeira et al. (2012).

1.5 Outline of the Dissertation

Despite all the previous work done in the area of NMB control, some problems related with the continuous infusion still exist. This is mainly due to a very high degree of uncertainty in the system dynamics caused by the intra- and intervariability in the patients, noise level, sensor faults, nonlinearities, time variations and control actions constrains. Although a large range of control strategies have been proposed and studied, a suitable and acceptable anesthetic solution has not yet been achieved. So research developments that allow to overcome the drawbacks of the manual control as well as the limitations of the past NMB control strategies have a high impact in the anesthetic community.

Chapter 2 presents an overview of anesthesia. Here the existence of a large range of anesthetic solutions is emphasized and the particular case of the general anesthesia will be further studied once it is the type of anaesthesia addressed in this work. The different general anesthesia components as well as the drugs used to achieve it will be presented. Moreover an explanation is given about the importance of anesthetic procedures during surgeries, as well as, about the induction and maintenance of anesthesia. In a second part of chapter the particular case of Neuromuscular Blockade (NMB) was studied, with a focus in the anatomy and physiology related with the NMB process, which means to explain how the muscle relaxants work over the muscle paralysis. Furthermore a brief description of the main muscle relaxants addressed in this work

is made.

In chapter 3 all the theoretical ground of the work is presented. Here two compartmental models are introduced, firstly the main *PK/PD* compartmental model is addressed and secondly a new compartmental model, known as *reduced parameter* compartmental model is presented. These models describe the relation between the quantity (or concentration) of administered drug and the quantity (or concentration) of drug which is actually effective. After that the Hill Equation adapted to each compartmental model, is introduced. This is a nonlinear equation that relates the drug effect concentration with its effect, i.e., with the NMB level of the patient. The total system mass control law is presented and studied in order to show that leading the total system mass to a reference value the NMB level is also drived to a specific desired target. Also the open-loop control technique TCI is briefly described. After that a switching strategy between total system mass control laws is introduced. The switching strategy is explained together with the switching criterion. Then some results of NMB control with switching strategy are presented, and a study comparing the switching strategy and the *Extended Kalman Filter* (EKF) strategy is performed, as well as a comparative study with several offline controllers chosen based on different metrics. Finally a strategy to improve the reference tracking quality is presented together with the results obtained after its application.

Chapter 4 is devoted to the clinical environment, i.e., to all the practical work developed on the application of the NMB control on real patients during surgery ranging from the hardware to the software used for the work, the protocol applied by the anesthesiologists as well as the outcomes obtained.

Finally, Chapter 5 contains the conclusions of the developed work as well as some suggestions for future work.

Chapter 2

Anesthesia

In surgical procedures where the absence of pain, feeling/sensation and/or movement is essential, anesthesia is an important aspect to consider. There are four main types of anesthesia:

- *Local*: numbs one small area of the body. The patient stays awake and alert;
- *Conscious sedation*: uses a mild sedative to relax the patient and pain medicine to relieve pain. The patient stays awake but may not remember the procedure afterwards;
- *Regional anesthesia*: blocks pain in an area of the body, such an arm or leg. Epidural anesthesia is a type of regional anesthesia;
- *General anesthesia*: affects the whole body. The patient goes to sleep and feels nothing, and should have no memory of the procedure afterwards.

For the purpose of this work only the general anesthesia will be assessed.

2.1 General Anesthesia

2.1.1 General Anesthesia

General anesthesia enables a patient to tolerate surgical procedures that would otherwise inflict unbearable pain, potentiate extreme physiologic exacerbations, and result in unpleasant memories. General anesthesia uses intravenous and inhaled agents to allow adequate surgical access to the operative site. Anesthesiologists are responsible for assessing all factors that influence a patient's medical condition and for selecting the optimal anesthetic technique accordingly.

The main features of general anesthesia are the following::

- It allows proper muscle relaxation for prolonged periods of time;

- It can be administered rapidly and is reversible;
- It reduces intraoperative patient awareness and recall;
- It can be adapted easily to procedures of unpredictable duration or extent;
- It facilitates control of the airway, breathing, and circulation.

2.1.2 General Anesthesia Components

General anesthesia can be defined as a reversible state of unconsciousness where three variables must be controlled: hypnosis, analgesia, and areflexia. For this purpose, the anesthesiologists need to administrate three different types of drugs: hypnotics, opioids and muscle relaxants.

Hypnotic drugs are a class of psychoactives whose primary function is to produce unconsciousness and sedation in surgical anesthesia. Examples of hypnotics are *isoflurane*, *sevoflurane* and *desflurane* (volatile drugs), and *propofol* (intravenous drug).

Opioid (or analgesic) are drugs used to relieve pain and to achieve analgesia. The analgesic effects of opioids are due to decreased perception of pain, decreased reaction to pain as well as increased pain tolerance once they act in the opioid receptors, from the central and peripheral nervous system. Examples of analgesics are *fentanil*, *sufentanil* and *remifentanil* (all intravenous drugs).

Finally, areflexia is defined as the lack of movement produced by muscle relaxants, or neuromuscular blockade agents (NMBA) [Esteves (2008)]. This drugs affect skeletal muscle function and decrease the muscle tonus by blocking nerve impulses so that muscles cannot contract, creating paralysis, a desired condition to facilitate tracheal intubation and to maintain good surgical conditions. Examples of muscle relaxants are *succinylcholine*, *atracurium*, *cis-atracurium*, *vecuronium* and *rocuronium* (all intravenous drugs).

The administration of a particular hypnotic, analgesic or muscle relaxant during a surgical procedure is a choice of the anesthesiologist, where patient characteristics, hospital protocols and the type of surgery are aspects to take into account.

2.1.3 Manipulation of the General Anesthesia Components

In order to manipulate the drug input the anesthesiologist has to supervise some physiological signals from the patient, so that he knows when and how to act in order to preserve a good level of anesthesia. It is based on the interpretation of specific patient signals that the anesthesiologist makes the decision of increase, decrease or cease the drug input.

In order to manipulate the depth of anesthesia (DoA) the anesthesiologist uses information from several physiological signals and depending on the stage of the surgery decides the anesthetic protocol, i.e., the amount of hypnotic and analgesic to deliver. However it was need to take into account that the analgesic drugs only affect the DoA if used in large amounts, which does not usually occur.

Nowadays, *BIS* signal (Bispectral index, which is a measure of the level of consciousness) is widely used in surgery rooms and it is accepted as a measure of the DoA: if the *BIS* signal is to high the anesthesiologist knows he needs to increase the drug input, and if it presents too low values he knows that he needs to decrease the drug input (Note that the *BIS* signal must be preserved between 40 and 60%). Fig. 2.1 illustrates the *BIS* sensor and its way to use and Fig. 2.2 the main *BIS* monitor. Nevertheless, this signal presents some drawbacks namely the noise from the surgical instruments and electromyography that affect the quality of the signal. Therefore, signals such the *BIS-SQI* signal, quality index of the *BIS*, and the*BIS-EMG* signal, interference of the electromyographic signal into the *BIS* signal, are used. In the first one, if the value is too low the anesthesiologist ignores the correspondent *BIS* value and waits for further values with better *BIS-SQI* to validate any action. In the second one, if the value is high the anesthesiologist ignores the *BIS* values, and if the value is low it means a clean *BIS* signal that can be used to make a decision. Finally signals like the heart rate and the blood pressure are supervised to restrain the increase of input of the hypnotic and the analgesic.

Figure 2.1: BIS Sensor.

In summary, the anesthesiologist uses the *BIS* signal as a measure for the hypnotic, which influences directly the DoA, but also the analgesic, which interacts with the hypnotic drug in the DoA. Signals like the *BIS-SQI* and *BIS-EMG* are used to validate the *BIS* signal, and heart rate and blood pressure are used to supervise the hypnotic and analgesic administration.

Figure 2.2: BIS Signal.

As for arreflexia, its manipulation is much easier and direct than the one of the depth of anaesthesia [da Silva et al. (2010), da Silva et al. (2012), Esteves (2008)]. Here the anesthesiologist only has to use the *TOF* signal (*Train-Of-Four*), which is used to measure the level of NMB. This signal is the patient's response to an evoked EMG obtained at the hand of the patient by a sequence of four external stimulations of the ulnar nerve (Fig. 2.3). A baseline measurement is done before paralytic agent is administered in order to determine the current necessary to obtain twitch. The signal (Fig. 2.3) is graphically represented by four bars, which correspond to the four twitches. At the beginning those bars are full (full muscle capacity) and when the muscle relaxant is administered the values decrease and come to zero by full paralysis. The value used to control the muscle relaxant input is T_1, the first twitch of the *TOF* signal. If T_1 is too high the anesthesiologist knows that more muscle relaxant is needed in order to increase muscle paralysis. Another value used, but in order to reverse the muscle paralysis is the *TOF Ratio*, given as the quotient between T_4, the four twitch of the *TOF* signal, and T_1. This value allows the anesthesiologist to know if he can apply the muscle relaxant reverser. If the *TOF Ratio* is different from 0 he knows that the reverser can be administered.

In summary, the anesthesiologist uses the *TOF* signal to handle the input of the NMB agents, using the T_1 to manipulate the muscle paralysis agent and the *TOF Ratio* to identify when the reverser can be administered.

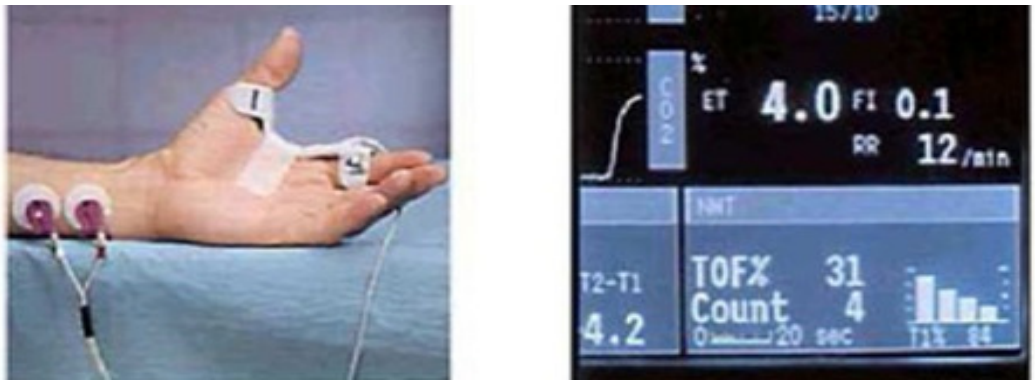

Figure 2.3: Mechanotransducer in the left image and TOF signal in the right image.

The values of the signals *BIS*, *BIS-SQI*, *BIS-EMG*, T_1, T_2, T_3, T_4, and *TOF Ratio* range from 0 to 100.

2.2 Neuromuscular Blockade (NMB)

2.2.1 NMB Anatomy and Physiology

The normal neuromuscular junction (NMJ) consists of a presynaptic neuron, a Schwann cell (covering the neuron), and a postsynaptic muscle fiber [Guyton and Hall (2000)]. The presynaptic neuron stores and releases Acetylcholine (ACh). ACh receptors exist at both junctional and extrajunctional areas of muscle fibers. When a nerve impulse reaches the end of a presynaptic neuron, N-type Ca++ channels increase the intracellular calcium concentration, which causes the synaptic vesicles to release ACh into the endplate. These ACh molecules then bind to the junctional receptors allowing for Na+ and Ca++ influx into the muscle cell, which ultimately leads to contraction due to the end plate potential. Aceytlcholinesterases quickly degrade available synaptic ACh, preventing prolonged contraction [Appiah-Ankam and Hunter (2004)].

A decrease in the binding of acetylcholine leads to a decrease in its effect and neuron transmission to the muscle is less likely to occur [Appiah-Ankam and Hunter (2004)]. There are two types of drugs that affect the transmission at the neuromuscular junction: depolarizing and non-depolarizing. Depolarizing drugs, such as *succinylcholine*, have the same effect on the muscle fibre as does acetylcholine. The difference between these drugs and acetylcholine is that they are not metabolized by acetylcholinesterase so a prolonged activation of the acetylcholine receptors is produced (depolarization of the motor end plate). Non-depolarizing drugs, such as *atracurium* and *rocuronium* block the action of acetylcholine on the acetylcholine receptor sites and therefore no end plate potential is developed [Appiah-Ankam and Hunter (2004)].

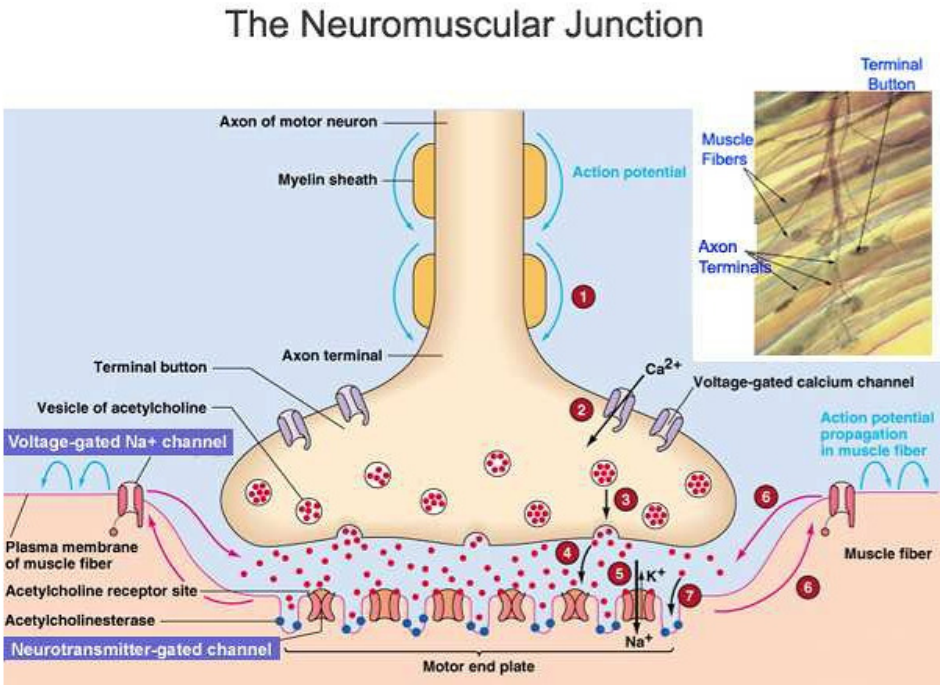

Figure 2.4: Neuromuscular junction [Magalhães (2006)].

The infusion of muscle relaxant drugs is frequent during the surgery for three reasons:

- To eliminate spontaneous breathing and promote mechanical ventilation (e.g. to eliminate the urge to fight the vent);
- To cause a pharmacological restraint so patients do not harm themselves;
- To decrease oxygen consumption.

Neuromuscular blockade agents (NMBA) are non-depolarizing drugs that block the neuromuscular transmission and consequently lead to muscle paralysis. Besides the already mentioned drugs *atracurium* and *rocuronium*, examples of such agents are *cis-atracurium* and *vecuronium* [Appiah-Ankam and Hunter (2004)].

After an infusion of an initial *bolus* of NMBA to facilitate tracheal intubation, the maintenance of muscle relaxation for long periods of time during surgery can be achieved by two different ways, namely it may be ensured either by continuous infusion or by further increments (top-ups) of NMBA at regular instants of time.

A *bolus* is a single dose B of drug usually injected into a blood vessel over a short period of time. It can be mathematically represented by:

$$u(t) = B\delta(t) \quad [\mu g/kg], \tag{2.1}$$

where $\delta(t)$ is the Dirac δ function. On the other side, the continuous infusion of a quantity k_i of drug in a time interval $[t_i, t_{i+1}]$ may be represented by the following step function:

$$u(t) = k_i \quad [\mu g.kg^{-1}.min^{-1}], \quad t_i \leq t \leq t_{i+1}, \quad i = 0, 1, 2, 3, \ldots. \tag{2.2}$$

For the purpose of this work *atracurium* and *rocuronium* will be the NMBA's under consideration.

2.2.2 Atracurium

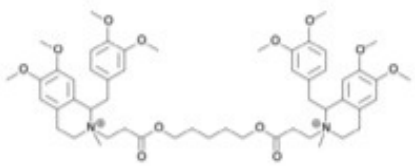

Figure 2.5: Atracurium molecular structure.

Atracurium is a benzyl isoquinolonium ester and can be defined as an intermediate-duration, short-acting relaxant in the category of non-depolarizing neuromuscular blocking drugs, used adjunctively in anesthesia to facilitate endotracheal intubation and to provide skeletal muscle relaxation during surgery or mechanical ventilation. An initial *atracurium* dose of 0.4 to 0.5 mg/kg provides a maximum neuromuscular blockade within 3 to 5 minutes of injection, and a recovery occurs approximately between 20 to 35 minutes after injection. The neuromuscular blocking action of *atracurium* is enhanced in the presence of potent inhalation anesthetics like *isoflurane* and *enflurane* which increase the potency of *atracurium* and prolong neuromuscular block by approximately 35%. The onset time decreases and the duration of maximum effect increases with the increase of the dose. *Atracurium* is rapidly broken down by the body to inactive metabolites by ester hydrolysis (minor pathway) and spontaneous Hoffman degradation (major pathway). The reversion of the neuromuscular blockade can be achieved with an anticholinesterase agent such as *neostigmine* in conjunction with an anticholinergic agent such as *atropine*. Reversal can usually be attempted when recovery of muscle twitch has started. Complete reversal is usually attained within 8 to 10 minutes of the administration of reversing agents.

Schematically, *atracurium* can be characterized as follows:

- *Structure*: Benzyl isoquinolinium compound;

- *Dosing*: 0.4-0.5 mg/kg with onset in 3-5 minutes, lasting 20-35 minutes;
- *Metabolism*: Cleared by non-enzymatic degradation (Hofmann elimination) as well as ester hydrolysis by plasma esterases.

2.2.3 Rocuronium

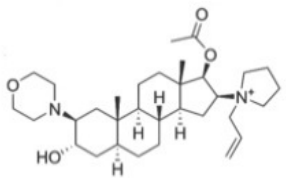

Figure 2.6: Rocuronium molecular structure.

Rocuronium is an aminosteroid non-depolarizing muscle relaxant used in modern anesthesia, to facilitate endotracheal intubation and to provide skeletal muscle relaxation during surgery or mechanical ventilation. As a non-depolarizing drug, *rocuronium* binds competitively to cholinergic receptors on motor end-plate to antagonize action of acetylcholine, resulting in a blockade of neuromuscular transmission. An initial *rocuronium* dose of 0.6 mg/kg provide a maximum neuromuscular blockade within 1 to 2 minutes of injection, and a recovery occurs approximately between 20 to 35 minutes after injection. The neuromuscular blocking action of this drug is enhanced in the presence of inhalation anesthetics like *sevoflurane*, *isoflurane*, and *enflurane*. The onset time decreases and the duration of maximum effect increases with the increase of the dose. *Rocuronium* is mostly cleared in the bile (essentially unchanged) (major pathway) although up to 30% may be excreted renally (minor pathway). Note that individual responses to this drug are highly variable, and the duration of the effect is difficult to predict. The main advantage of using *rocuronium* instead of *atracurium* is the existence of *sugammadex*, a reversal of neuromuscular blockade without relying on inhibition of acetylcholinesterase, like *neostigmine*. Also the reversion can be realized at any point, and so it is not necessary to expect the initial recovery from the initial *bolus*. The reversion of the neuromuscular blockade can also be achieved with an anticholinesterase agent such as *neostigmine* in conjunction with an anticholinergic agent such as *atropine*, but reversal can only be attempted when recovery of muscle twitch has started. In comparison with *atracurium*, *rocuronium* has a lower potency and so a higher dose is required for the same effect, leading to a higher concentration gradient and a faster onset.

Schematically, *rocuronium* is characterized as follows:

- *Structure*: Aminosteroid compound;

- *Dosing*: 0.6 mg/kg onset in 1-2 minutes and lasts 20-35 minutes;
- *Metabolism*: *Rocuronium* is mostly cleared in the bile (essentially unchanged) although up to 30% may be excreted renally. Note that individual responses to *rocuronium* are highly variable, and the duration of the effect is difficult to predict.

Chapter 3

Automatic Control

The manual control is tedious, slower and more prone to errors. Thus, the development of procedures for automatic control is of great interest, once they allow a more rigorous determination of the drug infusion dose and decrease the tendency to error.

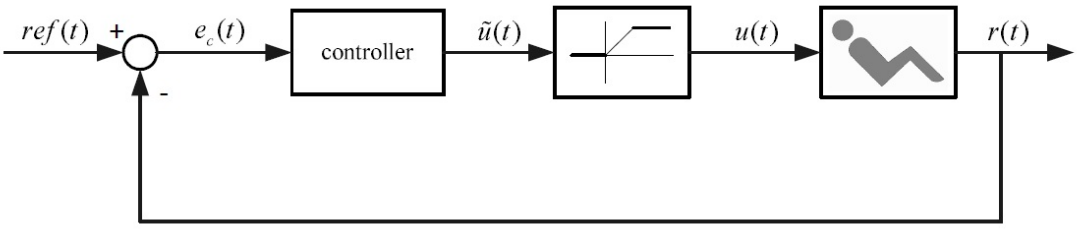

Figure 3.1: Elementary NMB control scheme.

To achieve an efficient automatic control, the effect of drug on the NMB of a patient needs to be modelled by means of mathematical equations.

3.1 Drug Effect Models

In order to model the effect of a drug, pharmacologists often separate their analysis into two steps. First, they built a pharmacokinetic (PK) model that relates the administered drug dose with the blood drug concentration. In a second stage, a pharmacodynamic (PD) model is derived, describing the relation between the former concentration and the drug concentration in the relevant part of the body, known as the effect concentration, as well as the relation between the effect concentration and the actual drug effect (here the NMB level).

3.1.1 Drug Dose/Effect Concentration Models

The most common models for the effect concentration of a drug are compartmental systems.

A system is a set of interconnected elements that are dependent on each other and form a unit that has specific characteristics and functions [Magalhães (2006), Marques (2008)]. Each system admits states that are defined as a set of variables capable to describe the system in any instant of time. Systems can be classified as continuous or discrete-time systems according to the continuous or discrete nature of the time-line over which their variables are defined. Also, systems can be classified as open, closed or isolated. An open system can exchange matter or energy with its surroundings, while a closed system can only exchange energy, but not matter with its surroundings. In contrast, an isolated system cannot exchange neither energy nor matter.

Compartmental systems are widely used to model the pharmacodynamics and pharmacokinetics of intravenously administered drugs [Godfrey (1983), Hof (1996), Beck et al. (2007)]. A compartmental system is a system that has a finite number of homogeneous, well-mixed subsystems, called compartments that exchange material among them and with the environment. These models are based on the principle of mass conservation.

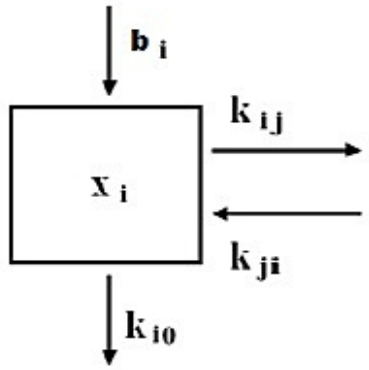

Figure 3.2: Representation of a compartment.

Fig. 3.2 represents a compartment, in this case compartment i; here b_i represents the input rate (the drug infusion rate to a patient if we consider the specific case of anesthesia), x_i is the concentration of material in the compartment i, k_{ij} represents the rate of mass transfer from compartment i to compartment j, and k_{i0} represents the rate of material output from compartment i to the environment.

The input to compartment i, is given by $b_i.u$, where u is the total system input. This input, the state x_i and all the rate constants are assumed to be non-negative.

At each time instant t, the variation $\dot{x}(t)$ in the concentration of material in compartment i

is then given by [Marques (2008), Almeida (2010)]:

$$\dot{x}_i(t) = -\sum_{\substack{j=0,\\ j\neq i}}^{n} k_{ij}x_i(t) + \sum_{\substack{j=1,\\ j\neq i}}^{n} k_{ji}x_j(t) + b_i u \qquad i = 1, \ldots, n. \tag{3.1}$$

where n is the total number of compartments.

Assuming that a linear combination of the concentrations of material in each compartment is the relevant system feature, the output of the system is defined as:

$$y(t) = c_1 x_1(t) + \ldots + c_n x_n(t). \tag{3.2}$$

It is assumed that the c_i are non-negative and that at least one of them is strictly positive.

Gathering in a vector all the concentrations x_i of material in the different compartments, i.e., defining

$$x(t) = \begin{bmatrix} x_1(t) \\ \vdots \\ x_n(t) \end{bmatrix},$$

the previous equations can be written in matrix form as:

$$\begin{cases} \dot{x}(t) = Ax(t) + Bu(t) \\ y(t) = Cx(t) \qquad\qquad t \geq 0, \end{cases} \tag{3.3}$$

where

$$B = \begin{bmatrix} b_1 \\ \vdots \\ b_n \end{bmatrix},$$

is a $n \times 1$ (column) and $C = \begin{bmatrix} c_1, \ldots, c_n \end{bmatrix}$, is a $1 \times n$ matriz (row), and A is a matrix $n \times n$ of the form:

$$A = (a_{ij}),$$

with:

$$\begin{aligned} a_{ii} &= -\sum_{\substack{j=0\\ j\neq 1}}^{n} k_{ij} \\ a_{ij} &= k_{ji} \qquad\qquad (i \neq j). \end{aligned} \tag{3.4}$$

Models as the one given by equations (3.3) are known as linear state-space models.

The vector $x(t)$ is known as the state vector, as it collects all the necessary information about the system at time instant t in order to describe its future behaviour, under the influence of a given input u.

Due to the fact that matrix A is of the form (3.4) and that B and C have nonnegative components and that at least one of its components is strictly positive, it turns out that when the initial state and the input are nonnegative, then the states and the output remain nonnegative.

The particular features of the matrix A are that:

- It is a Metzler Matrix, i.e., all off-diagonal values are non-negative:

$$a_{ij} \geq 0, \qquad \forall i, j \quad with \quad i \neq j$$

- All its diagonal values are non-positive:

$$a_{ii} \leq 0, \qquad \forall i$$

- It is a diagonally dominant matrix, i.e.,

$$\mid a_{ii} \mid \geq \sum_{j \neq i} a_{ij} \qquad \forall i.$$

A matrix with these properties is called a compartmental matrix.

In the sequel, two types of compartmental models are introduced in order to describe the administration of NMBA, namely the *PK/PD* compartmental model and the *reduced parameter* model. These will be used as basis for the implementation of automatic control procedures.

***PK/PD* Compartmental Model**

In this specific work, a mammillary model is considered to represent the pharmacokinetics and pharmacodynamics of NMBA infusion (Fig. 3.3) [Magalhães (2006), Beck et al. (2007), Almeida (2010)]. A mammillary model is a compartmental model in which a central compartment is surrounded by p-1 peripheral compartments which exchange matter only with the central compartmental and not with one another. Pharmacokinetics describes the path that a drug does inside the body, whereas pharmacodynamics means the pharmacology field that studies the physiological effects of drugs, and their mechanisms of action.

Here this model includes three compartments, where two of them represent the pharmacokinetic model (C_1 and C_2), combined with an effect compartment (Ce) which is part of the pharmacodynamic part of the model, see Fig. 3.3.

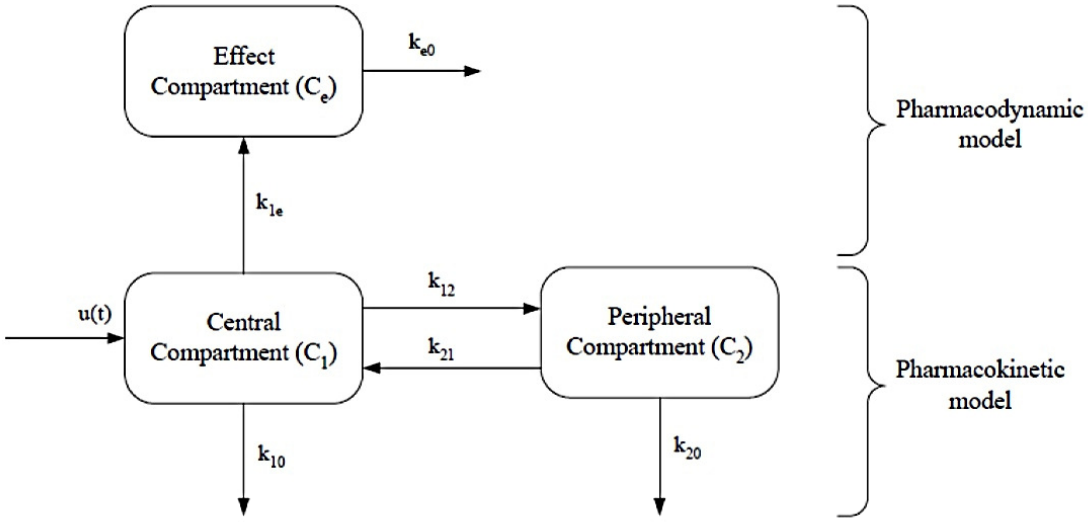

Figure 3.3: Block diagram of the mammillary *PK/PD* compartmental model.

The corresponding state-space equations are as follows.

$$\underbrace{\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \end{bmatrix}}_{\dot{x}}(t) = \underbrace{\begin{bmatrix} -(k_{12}+k_{10}+k_{1e}) & k_{21} & 0 \\ k_{12} & -(k_{21}+k_{20}) & 0 \\ k_{1e} & 0 & -k_{e0} \end{bmatrix}}_{A} \cdot \underbrace{\begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}}_{x}(t) + \underbrace{\begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}}_{B} u(t)$$

$$y(t) = \underbrace{\begin{bmatrix} 0 & 0 & 1 \end{bmatrix}}_{C} \cdot \underbrace{\begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}}_{x}(t). \tag{3.5}$$

This model has six patient dependent parameters, k_{10}, k_{12}, k_{1e}, k_{20}, k_{21}, and k_{e0} (min^{-1}) that must be identified for each particular patient. The state variables x_1 and x_2 $(\mu g/kg)$ correspond to the drug concentrations in compartments C_1 and C_2, whereas x_3 $(\mu g/kg)$ is the drug concentration in the effect compartment, also known as effect concentration. The input $u(t)$ $(\mu g/kg/min)$ corresponds to the delivery rate of drug concentration with respect to the central compartment, and is computed as $u(t) = \tilde{u}(t)/V_1$, where $\tilde{u}(t)$ is the drug delivery rate, and V_1

is the volume of the central compartment. The output $y(t)$ $(\mu g/kg)$ corresponds to the effect concentration.

***Reduced Parameter* Compartmental Model**

The *reduced parameter* model described in da Silva et al. (2012) has been derived by system identification techniques, rather than by pharmacological considerations. It can be described as a compartmental system as seen in Fig. 3.4. Although it is equally a mammillary tri-compartmental model, with two compartments representing the pharmacokinetic model (C_1 and C_2), combined with an effect compartment (C_3) which represents the pharmacodynamic model, the rate exchange constants are not the same as in the previous model. This leads to a different system, with different state-space equations. More concretely the model equations are the following [da Silva et al. (2012), Almeida (2010)].

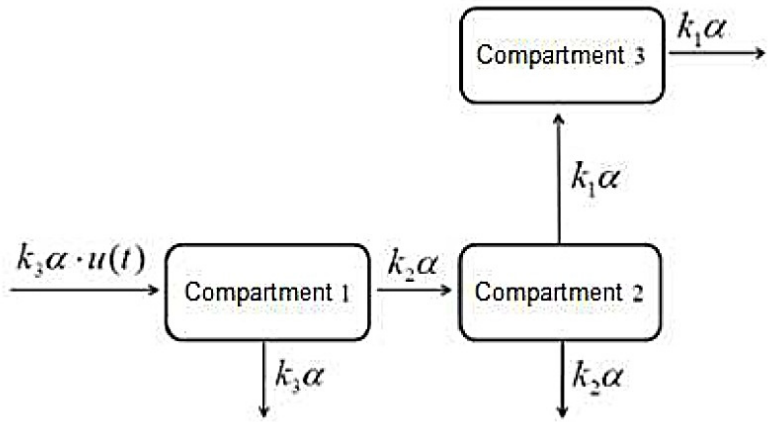

Figure 3.4: Block diagram of the mammillary *reduced parameter* compartmental model.

$$\underbrace{\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \end{bmatrix}}_{\dot{x}}(t) = \underbrace{\begin{bmatrix} -k_3\alpha & 0 & 0 \\ k_2\alpha & -k_2\alpha & 0 \\ 0 & k_1\alpha & -k_1\alpha \end{bmatrix}}_{A} \cdot \underbrace{\begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}}_{x}(t) + \underbrace{\begin{bmatrix} k_3\alpha \\ 0 \\ 0 \end{bmatrix}}_{B} u(t)$$

$$y(t) = \underbrace{\begin{bmatrix} 0 & 0 & 1 \end{bmatrix}}_{C} \cdot \underbrace{\begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}}_{x}(t). \tag{3.6}$$

This model has four parameters, k_1, k_2, k_3 (min^{-1}) and α (alpha) (dimensionless) that, as in the previous case, must be identified for each particular patient. However good results are obtained if the parameters k_1, k_2, k_3 are fixed, based on previous knowledge on the patient population, and only α is identified for each particular patient. This constitutes a great advantage. The values of k_1, k_2 and k_3 used in the sequel are $k_1 = 1$, $k_2 = 4$ and $k_3 = 10$, [da Silva (2011)]. The state variables x_1 and x_2 ($\mu g/kg$) correspond to the drug concentrations in compartments 1 and compartment 2, whereas x_3 ($\mu g/kg$) is the drug concentration in the compartment 3, also known as effect concentration. The input $u(t)$ ($\mu g/kg/min$) corresponds to the drug delivery rate with respect to the central compartment, and is computed as $u(t) = \tilde{u}(t)/V_1$, where $\tilde{u}(t)$ is the drug delivery rate, and V_1 is the volume of the central compartment. The output $y(t)$ ($\mu g/kg$) corresponds to the effect concentration.

3.1.2 Effect Concentration/ Drug Effect Models

As an output from the models presented in the previous subsection, the effect concentration, y(t), is obtained. The effect concentration corresponds to the percentage of administered drug that will produce effect in the NMB level, $r(t)$.

The relationship between the effect concentration and the NMB level is given by the Hill Equation; this is a nonlinear static equation, which assumes slightly different forms for the *PK/PD* model and for the *reduced parameter* model.

For the *PK/PD* model [Beck et al. (2007)]:

$$r(t) = \frac{100}{1 + \left(\frac{y(t)}{C_{50}}\right)^{\gamma}}, \tag{3.7}$$

where C_{50} ($\mu g/kg$) and γ (dimensionless) are patient-dependent parameters.

For the *reduced parameter* model [Almeida (2010)]:

$$r(t) = \frac{100}{1 + \left(\frac{y(t)}{k.C_{50}}\right)^{\gamma}}, \tag{3.8}$$

where C_{50} ($\mu g/kg$) is a fixed parameter equal to 0.6487 for *atracurium* and 1/5 for *rocuronium*, γ (dimensionless) is patient-dependent parameter and $k = 5$ (dimensionless) [Almeida (2010)].

The responses of the *PK/PD* model and of the *reduced parameter* model to a drug *bolus* are illustrated in Fig. 3.5. As can be seen, the behavior of the *reduced parameter* models differs from

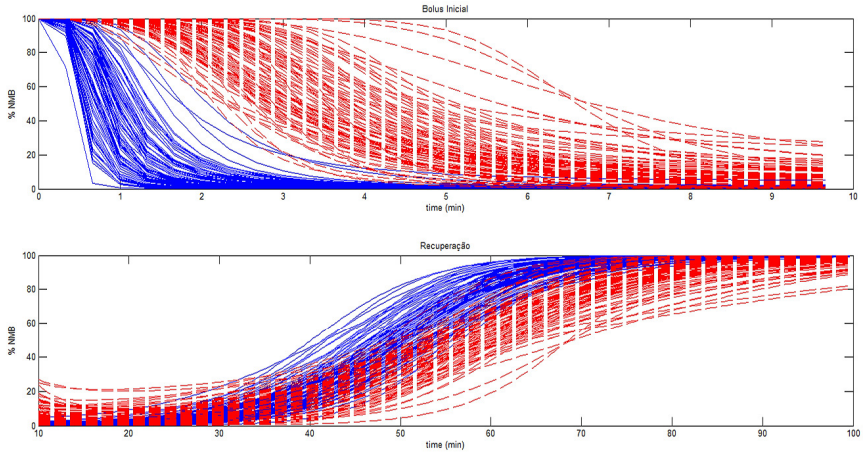

Figure 3.5: Simulation of 100 *atracurium PK/PD* models (blue lines) and 100 *atracurium reduced parameter* models (red lines) to a *bolus* of $500\mu g/kg$ during 100 minutes. The upper plot represents the first 10 minutes, and the lower plot represents the remaining time.

the one of the *PK/PD* models mainly in the transient stage.

3.2 NMB Control Law

In this section two strategies are used in order to control NMB to a desired level. The first one is an open-loop strategy, whereas the second one is based on a closed-loop control law.

3.2.1 Open-Loop Control - TCI

An open-loop controller, also called a non-feedback controller, is a type of controller which computes its input into a system using only a model of the system and the knowledge of the desired control objective.

Open-loop control is useful for well-defined systems where the relationship between input and the resultant state or output can be modelled by a mathematical formula. Such control does not use the comparison between the obtained output and the desired one in order to autocorrect itself. This means that the control variable input does not depend on the observation of the output of the processes under control.

TCI control (Target Controlled Infusion) for neuromuscular blockade level is an example of an open-loop control [Bressan et al. (2010)], where the input is stationary, i.e. $u(t) \equiv u_{ref}, t \geq 0$. The adequate value of u_{ref} is computed as follows. Given a desired reference value r_{ref} for the

NMB level, invert the Hill Equation in order to obtain the corresponding reference value y_{ref} for the effect concentration. Due to the characteristics of the compartmental systems that model the drug dose/effect concentration relationship, it turns out that these models have stationary solution $x(t) \equiv x_{ref}, u(t) \equiv u_{ref}$ and $y \equiv y_{ref}$. More concretely $x_{ref} = -A^{-1}.B.u_{ref}$ and $u_{ref} = -(C.A^{-1}.B)^{-1}.y_{ref}$. The TCI control strategy consists in taking the drug dose constantly equal to u_{ref}. Since the patient's initial state is null, and not equal to x_{ref}, the effect concentration is not immediately equal to y_{ref}, implying that the NMB level $r(t)$ does not immediately coincide with the desired value r_{ref}. However, $r(t)$ tends to r_{ref} as time evolves.

3.2.2 Closed-Loop Control - Total Sistem Mass Control Law

In this section it is first shown that it is possible to stabilize the "total mass" of a compartmental system (M) at a given set-point $M^* > 0$ using a suitable feedback control law presented in [Bastin and Provost (2002)].

The total mass of a compartmental system with n compartments is defined as [Marques (2008), Sousa et al. (2010)]:

$$M\left(x(t)\right) = \sum_{i=1}^{n} x_i. \tag{3.9}$$

Note that the states of the compartmental models used in this work do not correspond to drug masses, but rather to drug concentrations. Therefore their sum cannot be interpreted as a mass. However the terminology "total mass" is used in order to keep in line with the literature on this subject.

The objective is to track a constant reference y_{ref} by leading the system to an equilibrium point $X^e = [x_1^e \quad x_2^e \quad x_3^e]^T$ where $x_3^e = y_{ref}$. This is achieved by leading the system mass to an adequate value M^*.

For the *PK/PD* model M^* is given by:

$$M^* = (\alpha_1 + \alpha_2 + 1)y_{ref}, \tag{3.10}$$

where

$$\alpha_1 = \frac{k_{e0}}{k_{1e}},$$

$$\alpha_2 = \frac{k_{12}k_{e0}}{k_{1e}(k_{21}+k_{20})},$$

as shown in [Magalhães, 2006].

For the *reduced parameter* model:

$$M^* = 3.y_{ref}, \tag{3.11}$$

[Almeida, 2010], meaning that the total mass is equally distributed by the three compartments.

The control law is of the form:

$$\tilde{u}(t) = \tilde{u}(x(t)) = Kx(t) + L, \tag{3.12}$$

where the 1×3 matrix K and the real number L must be suitably chosen.

Since a compartmental system admits only positive values as inputs, the control law should provide only such type of values. Therefore, once a theoretical feedback control law $\tilde{u}(t)$ is defined, the input value to be applied is given by:

$$u(t) = max(0, \tilde{u}(t)). \tag{3.13}$$

The choice of K and L is made based on the following considerations.

Let $\dot{x}(t) = Ax(t) + Bu(t)$ be a compartmental system. Its total mass corresponding to a state x can be written as $M(x) = [1 \cdots 1]x$. Then:

$$[1 \cdots 1]\dot{x}(t) = [1 \cdots 1]Ax(t) + [1 \cdots 1]Bu(t), \tag{3.14}$$

which is equivalent to

$$\widehat{\dot{M(x(t))}} = [1 \cdots 1]Ax(t) + (\Sigma b_i)u(t), \tag{3.15}$$

Once the convergence of total mass to the value M^* is desired, the idea is, if possible, to determine $u(t)$ in such way that (3.15) takes the form:

$$\widehat{\dot{M(x(t))}} = -\lambda\left(M(x(t)) - M^*\right), \tag{3.16}$$

where λ is a positive value. Since $\dot{\widehat{M^*}} = 0$ because M^* is a constant, this i.e.:

$$\dot{\widehat{M(x(t))}} - M^* = \dot{\widehat{M(x(t))}} = -\lambda \left(M(x(t)) - M^* \right). \tag{3.17}$$

In this way, defining:

$$\Delta M(t) = M(x(t)) - M^*, \tag{3.18}$$

it holds that

$$\dot{\widehat{\Delta M}} = -\lambda \Delta M(t), \tag{3.19}$$

and hence:

$$M(x(t)) - M^* = \Delta M(t) = e^{-\lambda t} \Delta M(0) \underset{t \to \infty}{\longrightarrow} 0, \tag{3.20}$$

meaning that:

$$M(x(t)) \underset{t \to \infty}{\longrightarrow} M^*. \tag{3.21}$$

as desired.

Equating the right-hand sides of (3.15) and (3.16) and solving in order to $u(t)$, yields:

$$\begin{aligned}
&[1 \cdots 1] A x(t) + (\Sigma b_i) u(t) = -\lambda (M(x(t)) - M^*) \\
&\Leftrightarrow u(t) = (\Sigma b_i)^{-1} \left[-\lambda (M(x(t)) - M^*) - [1 \cdots 1] A x(t) \right] \\
&\Leftrightarrow u(t) = (\Sigma b_i)^{-1} \left[-\lambda M(x(t)) + \lambda M^* - [1 \cdots 1] A x(t) \right] \\
&\Leftrightarrow u(t) = (\Sigma b_i)^{-1} \left[-\lambda [1 \cdots 1] x(t) + \lambda M^* - [1 \cdots 1] A x(t) \right] \\
&\Leftrightarrow u(t) = (\Sigma b_i)^{-1} \left[[1 \cdots 1](-\lambda I - A) x(t) + \lambda M^* \right].
\end{aligned} \tag{3.22}$$

This shows that the control law:

$$u(t) = u(x(t)) = (\Sigma b_i)^{-1} \left[[1 \cdots 1](-\lambda I - A) x(t) + \lambda M^* \right], \tag{3.23}$$

allows the mass stabilization of a compartmental system around a set-point M^*.

The previous equation (3.23) can be rewritten as:

$$u(t) = u(x(t)) = \underbrace{(\Sigma b_i)^{-1}[1 \cdots 1](-\lambda I - A)}_{K} x(t) + \underbrace{(\Sigma b_i)^{-1}\lambda M^*}_{L}, \tag{3.24}$$

as in (3.12).

Finally it is shown in Magalhães (2006), Sousa et al. (2007) and Almeida (2010) that taking $u(t) = max(0, \tilde{u}(t))$, with $\tilde{u}(t)$ given by (3.12), not only leads the total system mass $M(x(t)) = \sum_{i=1}^{3} x_i(t)$ to a value M^*, but also leads the effect concentration, $y(t)$, to a value y^* (3.10) (3.11).

This can be used to control the NMB level in the following way. Given a desired reference value $r_{ref} = r^*$ for the NMB, compute the corresponding reference level for the effect concentration y^* by inverting the Hill equation, i.e.

$$y^* = \left(\frac{100}{r^*} - 1 \right)^{1/\gamma} \times C_{50}, \tag{3.25}$$

for the *PK/PD* model.

$$y^* = \left(\frac{100}{r^*} - 1 \right)^{1/\gamma} \times k.C_{50}, \tag{3.26}$$

for the *reduced parameter* model.

Then obtain M^* as in (3.10) or (3.11), according to the model that is used. Finally, use this value of M^* in the control law (3.24). This guarantees that the NMB follows the desired reference level, r^*.

Note however that this control strategy strongly rests on knowledge of the patient-dependent parameter γ, which is unknown in practical cases. Moreover, for the *PK/PD* model it also depends on the parameters k_{12}, k_{1e}, k_{20}, k_{21}, k_{e0}, and C_{50}. In order to overcome this situation, a switching control strategy is introduced in the next section.

3.3 NMB Control Strategy

Due to the very high degree of uncertainty in the system dynamics and to the intra- and inter-variability in the patients, multiple models have emerged over the past years as an alternative to the PID, adaptive, fuzzy and robust controls in order to overcome those problems.

The study of multiple model switching control is not new. Authors like Morse (1996), Morse (1997), Narendra and Balakrishnan (1994), Narendra and Balakrishnan (1997) and others have studied this over the past years.

Over the next sections this control scheme will be approached. The strategy corresponding to this scheme will be addressed together with the selection criterion to perform the switching during the NMB control. Finally an improvement to the reference tracking will be proposed in order to overcome some issues with the switching control strategy.

It is important to note that for this work only the *reduced parameter* models will be addressed. As has been seen during this chapter the *PK/PD* models differ from the *reduced parameter* models in the number of patient-dependent parameters: the former models have eight patient-dependent parameters and the latter ones have only two. This implies much more variability in the *PK/PD* models and much more uncertainty in the NMB control since more parameters have to be identified in order to perform the total system control mass law (3.24).

3.3.1 Switching Strategy

The requirements of a good controller are speed, accuracy and stability relative to the target. Achieving these features in complex systems, in the presence of large uncertainty in the process to be controlled is not always possible with classical adaptive controllers, and therefore switching control appears in order to overcome the problems faced by previous controllers.

Multiple model switching control [Morse (1996), Morse (1997), Narendra and Balakrishnan (1994), Narendra and Balakrishnan (1997), Neves et al. (2000), Neves (2003), Magalhães (2006)] uses a bank $\mathcal{P}$ of possible models for the process in order to obtain a bank $\mathcal{K}$ of controllers each of them tuned for a specific model. At each time instant a controller is chosen to be active. The controller selection criterion is based on a pre-specified measure of proximity between the process and the model for which the controller is tuned. The outcome of this proximity measure usually varies in time, leading to the switching from one controller to another.

Since *atracurium* and *rocuronium* are the two main NMBA used during surgeries, these were the drugs considered for the purpose of this work.

Therefore a bank $\mathcal{P} = \{\mathcal{P}_1, \ldots, \mathcal{P}_N\}$ of representative *reduced parameter* models is considered, together with a bank of the corresponding total system mass controllers $\mathcal{K} = \{\mathcal{K}_1, \ldots, \mathcal{K}_N\}$ each of them tuned according to what has been explained in section 3.2.2.

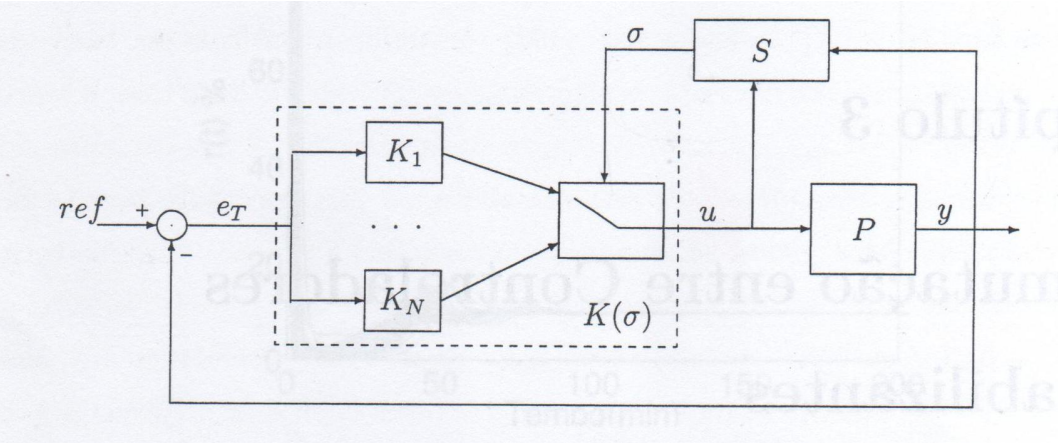

Figure 3.6: Basic scheme of a switching controller for a process $\mathcal{P}$: $\mathcal{K}_1 ... \mathcal{K}_N$ are the controllers in the bank $\mathcal{K}$, and S is the controller selection procedure.

More concretely if $\mathcal{P}_i = \mathcal{P}(\alpha_i, \gamma_i)$ then $\mathcal{K}_i$ produces the control law:

$$u_i(t) = max(0, \tilde{u}_i(t)),$$

with

$$\tilde{u}_i(t) = (\Sigma b_i)^{-1}[1 \cdots 1](-\lambda I - A_i)x(t) + (\Sigma b_i)^{-1}\lambda M_i^*, \tag{3.27}$$

For the action of *atracurium* the bank $\mathcal{P}$ was built taking into account real data acquired during surgeries. Based on that data the joint distribution for the parameters (α, γ) is considered as follows [Rocha et al. (2011)]:

$$(\ln(\alpha), \ln(\gamma)) \sim BN(\mu, \Sigma), \tag{3.28}$$

where

$$\mu = \begin{bmatrix} -3.2870 \\ 0.9812 \end{bmatrix}$$

is the mean vector and

$$\Sigma = \begin{bmatrix} 0.0250 & -0.0179 \\ -0.0179 & 0.1196 \end{bmatrix}$$

is the covariance matrix.

The standard bank of representative models, consisting of a linear part (*compartmental model*) plus a nonlinear part, the (*Hill Equation*), was generated from this distribution (See Appendix C.1).

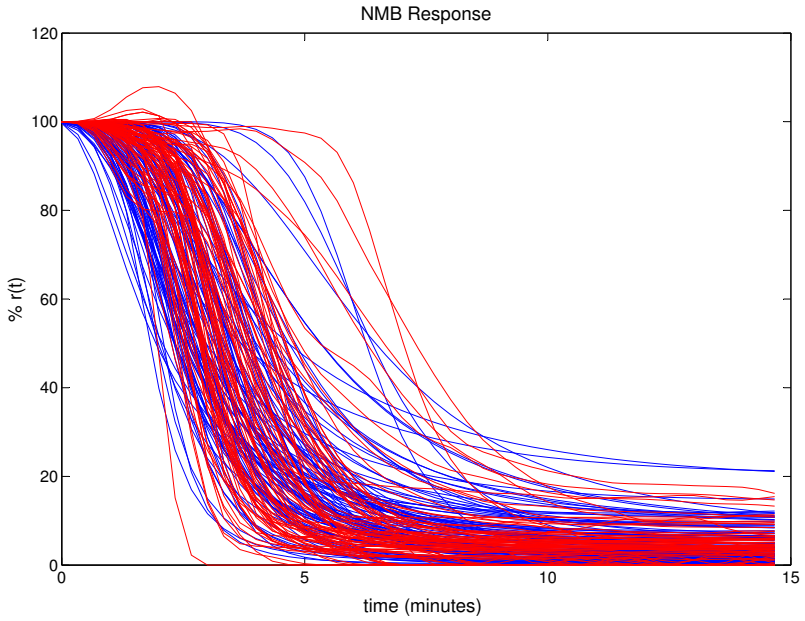

Figure 3.7: Red: real NMB responses acquired during surgery performed with *atracurium*. Blue: NMB responses of the *atracurium* models from the bank $\mathcal{P}$. In both NMB responses an initial *bolus* of $500\mu g/kg$ was administered.

A similar study of the action of *rocuronium* is still being performed, and at this point only a set of real data acquired during 41 surgeries is available. For each of these real cases, the parameters α and γ were identified, and the corresponding models were taken as the standard bank of models to perform switching control.

Note that for this work only 37 models of the original 41 will be used since four models were eliminated due to unsatisfactory results for the NMB control via switching (See Appendix C.2).

After the controller bank $\mathcal{K}$ is obtained, the switching strategy only needs a switching criterion in order to select the controller that should be made active to achieve a better reference tracking.

3.3.2 Switching Criterion

In order to perform switching among the controllers $\mathcal{K}_i$ contained in the bank $\mathcal{K}$ previously mentioned, it is necessary to establish a criterion so that the system "knows" when a switching is advisable.

It is important to notice that due to clinical constraints the automatic control of the NMB does not start immediately after the beginning of the surgery, but only after the recovery from an

initial *bolus*, t^* (the time instant t^* is detected automatically by the OLARD algorithm introduced in [da Silva et al. (2009)]). And so, from that moment forward the switching controller computes at each instant the "nearest model" to the patient and applies the control input corresponding to the associated controller to the patient (See Fig. 3.8).

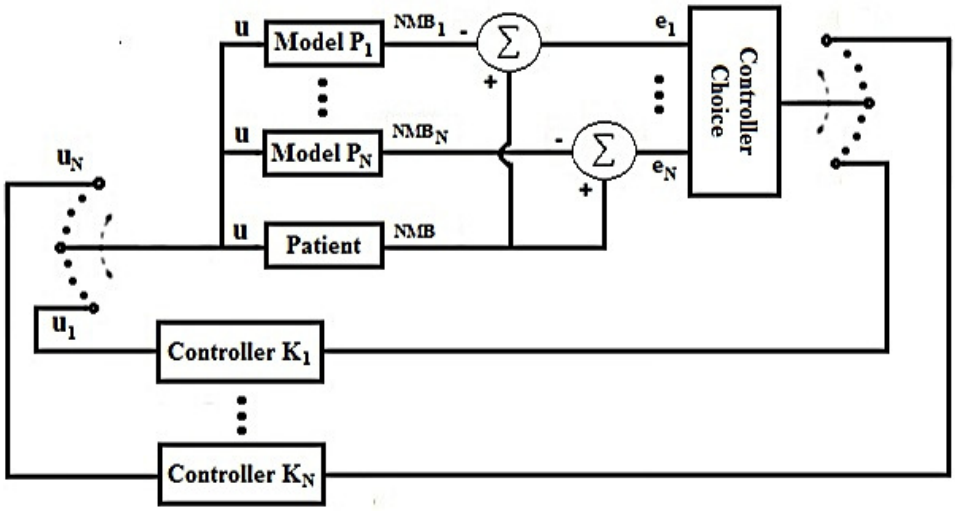

Figure 3.8: Switching control scheme.

The procedure to choose a controller (represented by the box *Controller Choice* Fig. 3.8) has as main goal to select the model that minimizes the following function of the error identification:

$$e_j = r_{patient} - r_j$$

where $r_{patient}$ is the patient's NMB response and the r_j is the NMB response of the model $\mathcal{P}_j$, $j \subset 1, \ldots, N$. More concretely, proximity is here measured by the cumulative quadratic error.

The controller to be selected is the one corresponding to the following minimizing model:

$$f_j(k) = \sum_{l=t^*}^{k} |e_j(l)|^2. \tag{3.29}$$

In order to initialize the procedure at the instant time $t = t^*$ (that corresponds to the recovery from the initial *bolus*) a random controller from bank $\mathcal{K}$ is chosen.

The process of switching control can thus be summarized as:

- **Step 1** $t = 0$: An initial *bolus* is given to the patient.
- **Step 2** $0 < t < t^*$ (before recovery from the initial *bolus*): The control input remains zero.
- **Step 3** $t = t^*$ (t^* computed by the OLARD algorithm introduced in da Silva et al. (2009)): A random controller from bank $\mathcal{K}$ is chosen in order to start the control of the NMB response of the patient.
- **Step 4** $t \geq t^*$: A controller $\mathcal{K}_i$ is chosen at each time instant based on the minimization of the cumulative error between the patient response and the responses of each of the models in the bank $\mathcal{P}$.

For the purpose of simulation, a model $\mathcal{P}_j$ from the bank $\mathcal{P}$ is chosen as describing the real patient dynamics and the corresponding controller $\mathcal{K}_j$ removed from the bank $\mathcal{K}$ yielding a new bank of controllers $(\mathcal{K}\backslash\{\mathcal{K}_j\})$. This prevents the switching strategy to "catch" the controller associated to the patient model, and makes simulation more realistic, since in real cases the patient model and hence the corresponding controller are not available. The initial drug *bolus* is taken as $500\mu g/kg$ for *atracurium* or $600\mu g/kg$ for *rocuronium* and is given to the patient. The remaining steps are kept the same as above (from Step 2 to Step 4).

3.3.3 Switching Results

Figs. 3.9, 3.10, 3.11 and Table 3.1 show results of the application of the switching strategy among a bank of total system mass control laws in order to control the NMB level. For these cases a bank $\mathcal{P}$ with 100 models for *atracurium* (see Appendix C.1) was used.

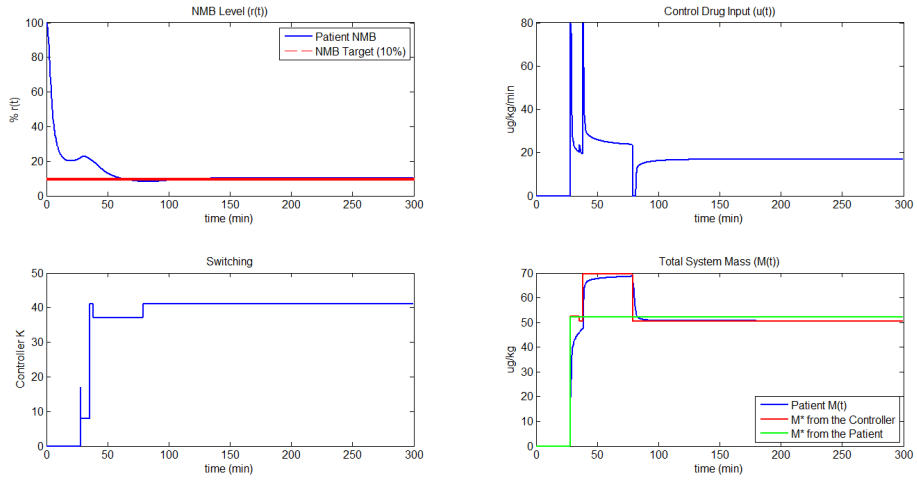

Figure 3.9: NMB control simulation with $\mathcal{M}_{30}$ from the bank $\mathcal{P}$ of *atracurium* models.

Table 3.1: Switching results with *atracurium* models

Patient Model	30	59	92
Recovery to the Initial *Bolus* (min)	28	32	37
First Controller $\mathcal{K}$ (Random)	17	58	6
Last Controller $\mathcal{K}$	41	97	18
Patient U_{ss} $(\mu g/kg/min)$	17,37	5,58	7,34
Real U_{ss} $(\mu g/kg/min)$	16,85	5,6	7,36
ΔNMB (%)	0,37	0,14	0,06

In the first case (Fig. 3.9), the model $\mathcal{P}_{30}$ was used to simulate the patient dynamics, and so the controller $\mathcal{K}_{30}$ was removed from the bank $\mathcal{K}$ of controllers. As can be seen in Fig. 3.9 and confirmed in Table 3.1, using model $\mathcal{M}_{30}$ and applying a typical *bolus* of $500\mu g/kg$ of *atracurium*, the recovery time (instant at which the controller is activated) is t^*=28 min and the switching strategy started with the controller $\mathcal{K}_{17}$. After 300 minutes of simulation the switching strategy ended with controller $\mathcal{K}_{41}$ with a steady-state dose of 16.85 $\mu g/kg/min$. The steady-state dose necessary to achieve the NMB target of 10% for model $\mathcal{P}_{30}$ is in fact 17.37 $\mu g/kg/min$. Since the difference in the steady-state doses is small, the difference between the final NMB level and the desired NMB target is only of 0.37% (See Table 3.1).

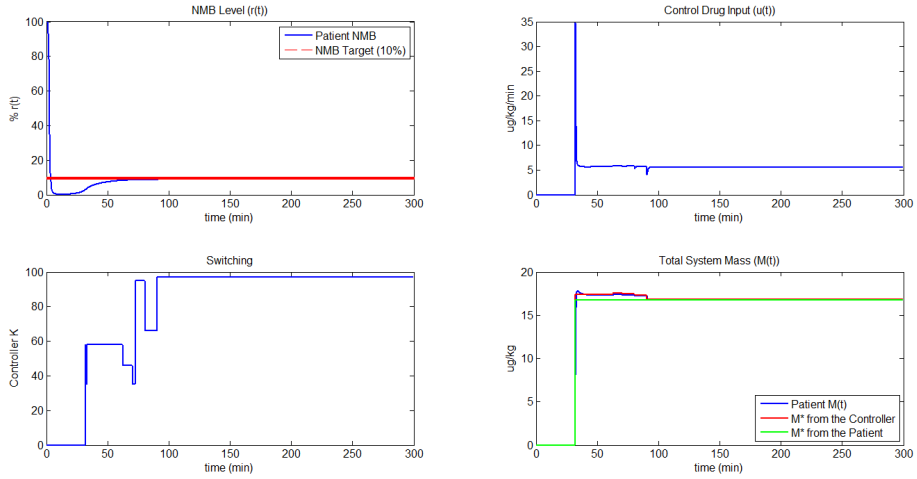

Figure 3.10: NMB control simulation with $\mathcal{M}_{59}$ from the bank $\mathcal{P}$ of *atracurium* models.

In the second case (Fig. 3.10) the model $\mathcal{M}_{59}$ was used to simulate a patient model and as a consequence the controller $\mathcal{K}_{59}$ was removed from the controller bank. In this model the recovery to the initial *bolus* of 500 $\mu g/kg$ of *atracurium* occurred at t^*=32 min and the started up controller was $\mathcal{K}_{58}$. After 300 minutes of simulation the last controller applied was $\mathcal{K}_{97}$, with a steady-state dose of 5.6 $\mu g/kg/min$. Once the steady-state dose needed for this model in order to follow the NMB target is 5.58 $\mu g/kg/min$ the difference between the achieved NMB level and the desired NMB target is only of 0.14% (See Table 3.1).

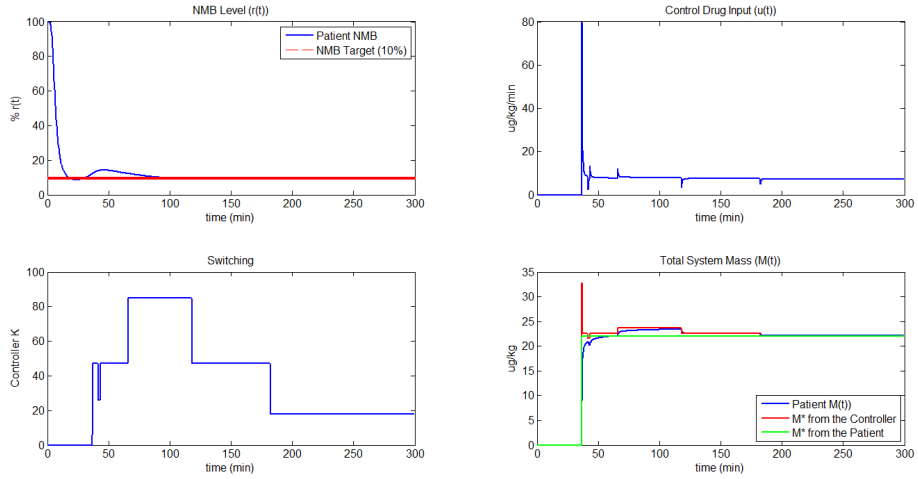

Figure 3.11: NMB control simulation with $\mathcal{M}_{92}$ from the bank $\mathcal{P}$ of *atracurium* models.

Finally, in Fig. 3.11, the model $\mathcal{M}_{92}$ was used to simulate the patient dynamics and the respective controller $\mathcal{K}_{92}$ was removed from the controller bank. After applying a typical *bolus* of 500 $\mu g/kg$ of *atracurium* the recovery happened at t^*=37 min and the first applied controller was $\mathcal{K}_6$. After 300 minutes of simulation the switching strategy ended with the controller $\mathcal{K}_{18}$ and a steady-state dose of 7.36 $\mu g/kg/min$. Comparing with the steady-state dose needed for the model to achieve the NMB target, 7.34 $\mu g/kg/min$, the difference is very small, and so the difference between the achieved NMB level and the desired NMB target is only of 0.06% (See Table 3.1).

In summary, the switching strategy shows good results for the control of the NMB under administration of *atracurium*. As can be seen in the left upper plots of Figs. 3.9, 3.10 and 3.11 the NMB level is driven with only small error to the NMB target value (10%) over time. The left lower plot of each image shows that switching occurs over time, which means that this strategy responds to the patient dynamics, searching better controllers over time. The right lower plots show the total system mass of the patient (in blue), the value M^* that corresponds to the NMB target according to the patient model (in green), and the different values of M^* chosen by the switching strategy over time (in red). As can be seen in these figures the total system mass of the patient follows the values of M^* with good accuracy and the controller computes proper values for M^*, and hence the switching strategy produces good results.

Figs. 3.12, 3.13, 3.14 and Table 3.2 show results concerning the use of the switching strategy among a bank of total system mass control laws in order to control the NMB level by administration of *rocuronium*. For these cases a bank $\mathcal{P}$ with 37 models for *rocuronium* (see Appendix C.2) was used.

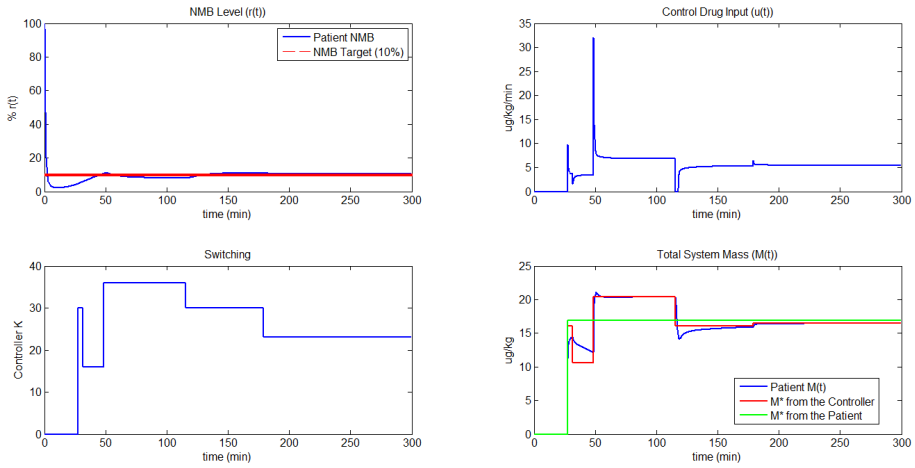

Figure 3.12: NMB control simulation with $\mathcal{M}_1$ from the bank $\mathcal{P}$ of *rocuronium* models.

Table 3.2: Switching results with *rocuronium* models

Patient Model	1	19	23
Recovery to the Initial *Bolus* (min)	28	28	32
First Controller $\mathcal{K}$ (Random)	30	29	31
Last Controller $\mathcal{K}$	23	34	13
Patient U_{ss} $(\mu g/kg/min)$	5,64	4,4	3,92
Real U_{ss} $(\mu g/kg/min)$	5,48	4,4	3,94
ΔNMB (%)	0,34	0,007	0,08

In Fig. 3.12, the model used to simulate the patient dynamics is $\mathcal{M}_1$, and the respective controller $\mathcal{K}_1$ was removed from the controller bank. As can be seen in Fig. 3.12 and in Table 3.2, after applying a typical *bolus* of 600 $\mu g/kg$ of *rocuronium*, the recovery time is t^*=28 min, and the switching strategy started up with the controller $\mathcal{K}_{30}$. After 300 minutes of simulation the last controller applied to this model was $\mathcal{K}_{23}$, corresponding to a steady-state dose of 5.48 $\mu g/kg/min$. Since the steady-state dose necessary for model $\mathcal{M}_1$ in order to lead the NMB level to the NMB target of 10% is 5.64 $\mu g/kg/min$, the difference between the achieved NMB level and the desired NMB target is only of 0.34% (See Table 3.2).

In the second case (Fig. 3.13) the model $\mathcal{M}_{19}$ was used in order to simulate the patient dynamics, and as a consequence the controller $\mathcal{K}_{19}$ was removed from the controller bank. After applying a typical *bolus* of 600 $\mu g/kg$ of *rocuronium* the recovery time occurred at t^*=28 min. The first controller applied by the switching strategy was $\mathcal{K}_{29}$ and at the end of the simulation the last applied controller was $\mathcal{K}_{34}$ with a steady-state dose of 4.4 $\mu g/kg/min$, the same value as the steady-state dose needed for this model to drive the NMB level to the desired target, and so the final NMB level was the NMB target of 10% (See Table 3.2).

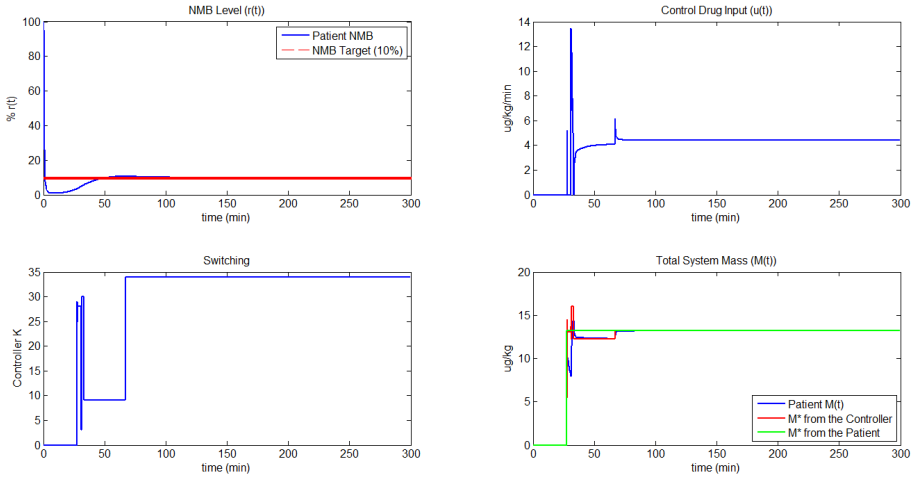

Figure 3.13: NMB control simulation with $\mathcal{M}_{19}$ from the bank $\mathcal{P}$ of *rocuronium* models.

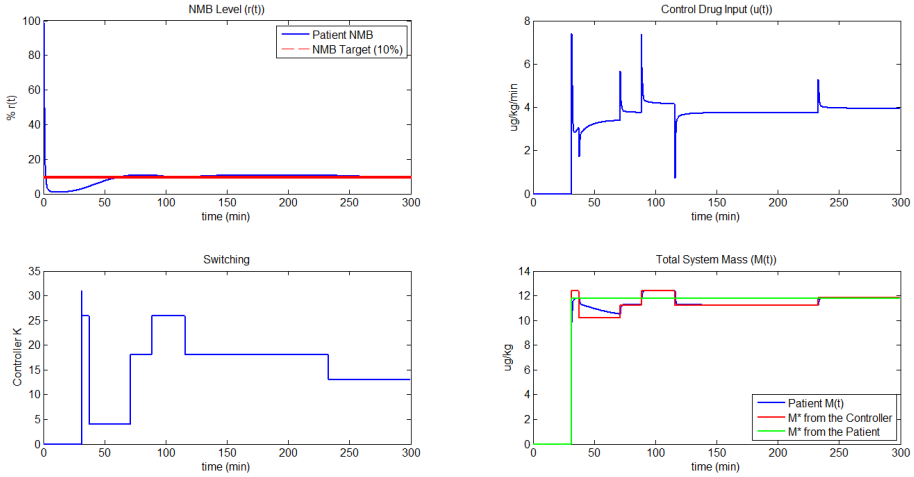

Figure 3.14: NMB control simulation with $\mathcal{M}_{23}$ from the bank $\mathcal{P}$ of *rocuronium* models.

Finally in the third case (Fig. 3.14) the model $\mathcal{M}_{23}$ was used in order to simulate the patient dynamics and the respective controller $\mathcal{K}_{23}$ was removed from the controller bank. After giving a *bolus* of 600 $\mu g/kg$ of *rocuronium* the recovery time happened at t^*=32 min, and the switching strategy started with the controller $\mathcal{K}_{31}$. At the end of the simulation (300 minutes) the controller used was $\mathcal{K}_{13}$, with a steady-state dose of 3.94 $\mu g/kg/min$. Comparing with the steady-state dose needed by model $\mathcal{M}_{23}$ to achieve the NMB target, 3.92 $\mu g/kg/min$, one can see that the difference is very small. Hence the difference between the achieved NMB level and the desired NMB target was also very small: only 0.08% (See Table 3.2).

In summary, the switching strategy shows good results for the control of the NMB under administration of *rocuronium*. As can be seen in the left upper plots of Figs. 3.12, 3.13 and 3.14 the NMB level is driven with good accuracy to the NMB target value (10%) over time. The left

lower plot of each image shows that switching occurs over time, as happened for the *atracurium* cases. The right lower plots show the total system mass of the patient (in blue), the value M^* of the patient model corresponding to the desired NMB target (in green), and the different values of M^* chosen by the switching strategy over time (in red). As can be seen in these figures the total system mass of the patient follows the value M^* with good accuracy and the controller computes proper values for M^*, and hence the switching strategy produces good results.

3.3.4 Switching Vs. Parameter Identification via EKF

In order to analyze the switching strategy a comparison was made with the *Extended Kalman Filter* (*EKF*). The *EKF* performs the identification of the model parameters used in the total system mass control law (3.24) in order to control the NMB level.

The *Extended Kalman Filter* (*EKF*) is an adaptation of the filter originally proposed by Kalman [Kalman (1960)] in order to estimate the states of a linear system. This extension is aimed at application to nonlinear systems and allows to estimate model parameters by incorporating them as states (which gives rise to a nonlinear system). The explanation of the *Kalman filter* and of the *EKF* is omitted since it falls out of the scope of this work. Some work related with the application of the *EKF* to model parameter identification was made inside the GALENO project [Almeida (2010), da Silva et al. (2012)].

The *EKF* algorithm used here was used before in [Almeida (2010), da Silva et al. (2012)] and properly adapted to this work in order to obtain the necessary data to perform the comparative study.

For the purpose of obtaining data the bank $\mathcal{P}$ with 100 models for *atracurium* (see Appendix C.1) and the bank $\mathcal{P}$ with 37 models for *rocuronium* (see Appendix C.2) were used. A simulation of 300 minutes was performed for each model, where each one was used to simulate the patient dynamics as previously mentioned. Thereafter the two strategies were analyzed using the following as comparison features: *total amount of used drug normalized by the patient weight*, *set-point*, *settling time*, and *tracking quality*. Taking into account that discretized models are used, these parameters are defined as follows:

- The *normalized drug amount* is given by:

$$\sum_{k=t^*}^{t_{end}} u(k), \tag{3.30}$$

  where $u(k)$ $(\mu g/kg/min)$ is the dose of administered drug at step k, and t^* is the recovery time, detected by OLARD.

- The *set-point* corresponds to the value (in %) where NMB level stabilizes;
- The *settling time* is the time that patient's NMB level takes to reach the *set-point* value, with an error not greater than 2%;
- The *tracking quality* is measured by:

$$\frac{1}{n}\sum_{k=settlingtime}^{t_{end}}\sqrt{(r(k)-r_{ref})^2}, \tag{3.31}$$

where $n{=}t_{end}$ - $settlingtime$, $r(k)$ is the NMB level at step k and r_{ref} is the desired reference value of 10%.

The results of applying the switching strategy and the *EKF* followed by a total system mass control law can be visualized in Tables 3.3 (for *atracurium*), and 3.4 (for *rocuronium*).

Table 3.3: Comparison between total mass control with *Switching* and with *Extended Kalman Filter* (*EKF*) (*atracurium*)

		Total Input ($\mu g/kg$)		Set-Point (%)		Settling Time (min)		Ref Track Error (%)	
		Switching	EKF	Switching	EKF	Switching	EKF	Switching	EKF
N	Valid	100	100	100	100	100	100	100	100
Mean		6846	8293	9,90	6,33	96	56	0.27	3,90
Median		6118	7629	9,90	6,38	81	58	0.14	3,68
Std. Deviation		2583	2872	0,62	1,64	51	21	0,48	1,70
Range		11292	16342	7,30	8,25	220	86	3,29	8,26
Minimum		3751	4725	6,10	1,40	29	23	0,01	0,49
Maximum		15043	21067	13,40	9,65	249	108	3,31	8,75
Percentiles	25	5014	6177	9,80	5,08	64	37	0,06	2,66
	75	8100	9758	10,05	7,59	113	73	0,23	5,12

In Table 3.3 it can be seen that in average, the total mass control with switching spends less dose than the total mass control with *EKF* (*EKF+TMC*). Also the input values in the switching are much closer among themselves than the *EKF+TMC* input values. This can be seen by the smaller value of the standard deviation, and of the range. The lowest minimum, and the lowest maximum values also occur for the switching.

As for the set-point reached with these two techniques, switching presents a better mean NMB value: 9.90% for a desired target of 10%. *EKF+TMC* presents a mean value of 6.33%, which is not a good result concerning to the tracking of the NMB reference of 10%. Moreover the switching has set-point values closer to the average than the *EKF+TMC*, as can be seen by the smaller value of the corresponding standard deviation. The set-points achieved in the switching vary between 6.10% and 13.40%, and the *EKF+TMC* between 1.40% and 9.65%, showing again better results with switching.

The bad results verified in the set-point with the *EKF+TMC* can be explained by the bad quality in the parameter identification in cases where the recovery to the initial *bolus* takes to

long too happen. Such situation is put into evidence in Fig. 3.15. Here it can be seen that with the increase of the recovery time the achieved set-point is lower.

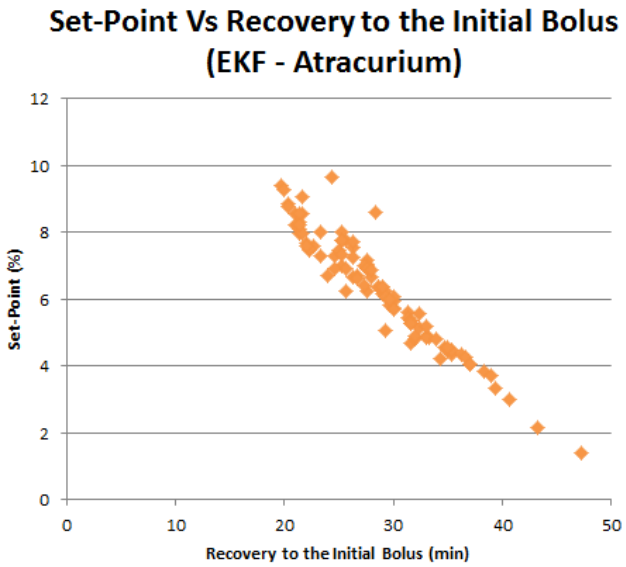

Figure 3.15: Set-Point achieved with the *EKF+TMC* as a function of the recovery time from the initial *bolus* with *atracurium* models.

The settling time mean value is higher with the switching than with the *EKF+TMC*. This conclusion is enhanced by the fact that the lowest minimum and the lowest maximum settling times occur in the *EKF+TMC*. These results can be explained due to the nature of the *EKF+TMC* strategy, which identifies the parameters to be used in the control law when the recovery to the initial *bolus* is detected; in opposition, as the name suggests, the switching strategy causes changes between controllers over time, and only when the last controller is selected the final set-point is achieved and the settling time is measured.

Finally, analyzing the tracking error (*Ref Track Error* column) the mean value for switching is much lower than the one verified for the *EKF+TMC*. The same happens to all the other variables in Table 3.3, showing that the switching is better than the *EKF+TMC*. So it can be concluded that the switching strategy has a better reference tracking quality than the *EKF+TMC* strategy.

Table 3.4: Comparison between total mass control with *Switching* and with *Extended Kalman Filter* (*EKF*) (*rocuronium*)

		Total Input ($\mu g/kg$)		Set-Point (%)		Settling Time (min)		Ref Track Error (%)	
		Switching	EKF	Switching	EKF	Switching	EKF	Switching	EKF
N	Valid	37	37	37	37	37	37	37	37
Mean		3577	3627	10,14	9,38	141	94	0,82	1,38
Median		3205	3125	10,10	9,95	137	89	0,67	0,72
Std. Deviation		1903	2093	1,11	2,33	64	32	0,68	1,97
Range		6610	9226	5,35	10,20	248	112	3,29	8,49
Minimum		911	1200	7,85	1,60	48	39	0,02	0,03
Maximum		7521	10425	13,20	11,80	296	151	3,31	8,52
Percentiles	25	2107	2151	9,23	8,98	94	69	0,31	0,20
	75	4303	4223	10,63	10,68	170	123	1,20	1,37

Now comparing the switching strategy and the *EKF+TMC* strategy using *rocuronium* mod-

els, it can be seen in Table 3.4 that in average the switching consumes less dose than the *EKF+TMC*. Moreover the doses administered during control with switching are more similar among themselves than with the *EKF+TMC*, as is evidenced by the lowest value of the standard deviation and range for the total drug input using switching. The lowest minimum and maximum drug dose are also obtained with the switching strategy, showing once again that the switching consumes less dose.

As for the set-point, here the *EKF+TMC* shows better results for the mean value than in the case of the *atracurium* models (9.38% instead of the previous 6.33% for a desired target of 10%). Comparing the mean value of the switching with the one of the *EKF+TMC*, both of them are similar but the value of switching is closer to the NMB target of 10%. Analyzing the standard deviation and the range the switching strategy has better results for the set-point than the *EKF+TMC*: the set-points for the different cases (models) are more similar for the switching, also the proximity between the minimum and maximum set-point values show that same conclusion. Once again the higher value obtained for the standard deviation and range with the *EKF+TMC* strategy is due to the bad quality in the parameter identification in cases where the recovery to the initial *bolus* takes to long too happen, similar to what happened for the *atracurium* models (see Fig. 3.16): as a consequence the reference tracking is not good (as will be seen more ahead).

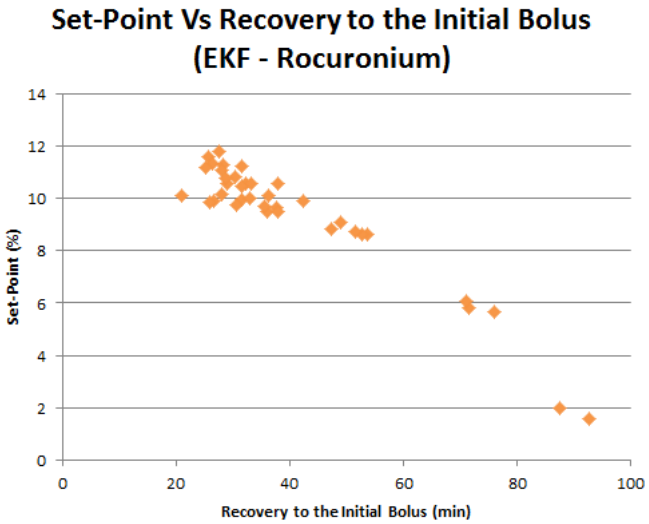

Figure 3.16: Set-Point achieved with the *EKF+TMC* as a function of the recovery time from the initial *bolus* with *rocuronium* models.

Studying the settling time, the same that was observed in Table 3.3 prevails here: the settling time of the switching strategy is higher than for the *EKF+TMC*. This can be concluded from the highest value in the corresponding mean value, in the minimum and maximum, as well as in the range. Once again such result is due to the nature of the switching strategy, as was previously mentioned.

Finally, addressing the tracking error (*Ref Track Error* column), the switching strategy presents, as for the *atracurium* models, lower values than the *EKF+TMC* strategy except for the *percentile 25*. Despite that, the differences between the obtained values for the two strategies is smaller than the one verified with *atracurium* models. Therefore it can be said that the switching strategy presents better reference tracking quality than the *EKF+TMC* but this difference is not so large as for the *atracurium* models.

In summary, for the *atracurium* models, the switching strategy has better results than the *EKF+TMC* strategy with respect to the total dose used during NMB control, follows better the NMB target with a better reference tracking quality, specially when the recovery time to the initial *bolus* is prolonged. As for the settling time, the *EKF+TMC* obtained better results due to the nature of the switching strategy.

With the *rocuronium* models, the switching strategy also has better results than the *EKF+TMC* strategy with respect to the total dose required for NMB control, follows better the NMB target because the values obtained for switching are more closer to the target than the ones obtained with the *EKF+TMC*. The reference tracking quality is better for the switching with an advantage over the *EKF+TMC*, which is smaller than the one verified for the *atracurium* models. Finally the settling time is better for the *EKF+TMC* strategy due to the nature of the switching strategy.

3.3.5 Analyzing the Switching

In order to further study the switching strategy, another comparative study was made using fixed total mass controllers chosen from different criteria. More concretely the last choice of the switching controller was compared with the other controllers, in order to find out if the switching strategy converge to a good controller.

For the purpose of this study both the bank $\mathcal{P}$ of 100 *atracurium* models (see Appendix C.1), and the bank $\mathcal{P}$ of 37 *rocuronium* models (see Appendix C.2) were used.

The criteria used to choose a controller for the model that simulates the patient dynamics are the following:

- The last controller resulting from the switching strategy (Switching) (see Appendix A.2);
- The best controller in the controller bank $\mathcal{K}$ (containing all the controllers except the one of the model used to simulate the patient system) to follow the desired reference of 10% (Ref Track) (see Appendix A.3);

- The controller tunned for the model with the value of α closest to the one of the patient model (Alpha) (see Appendix A.4);
- The controller tunned for the model with the value of γ closest to the one of the patient model (Gamma) (see Appendix A.4);
- The controller tunned for the model with the value of U_{SS} (steady-state dose) closest to the one of the patient model (U_{SS}) (see Appendix A.5);
- The controller tunned for the model with the value of *Norm 2* closest to the one of the patient model (Norm2) (see Appendix A.6);
- The controller tunned for the model with the value of the *Vinnicombe* metric closest to the one of the patient model (Vinnicombe) (see Appendix A.7);
- The controller tunned for the model with the closest *Impulse Response* to the one of the patient model (Impulse) (see Appendix A.8);
- The controller tunned for the model with the closest *Step Response* to the one of the patient model (Step) (see Appendix A.8);

Remark: In the last two criteria, the controllers, Impulse and Response, are chosen in function of the smaller quadratic mean error.

The corresponding chosen controllers are displayed in the tables of Appendix D (for the *atracurium* models see Table D.1, and for the *rocuronium* models see Table D.2).

After the simulation of the NMB control with the total system mass controllers selected by the previous criteria with the purpose of comparing the controllers choice, the *total amount of used drug normalized by the patient weight*, the *set-point*, the *settling time*, and the *tracking quality* (described in Section 3.3.4) were analyzed.

As a previous note, it should be noticed that the selected controllers with the Alpha proximity, Norm2 and Vinnicombe metrics, as well as Impulse and Step response proximity are the same, and so the values obtained with the application of these controllers are the same also, as will be seen in the following tables. First the results obtained using the *atracurium* models (see the models at Appendix C.1) will be presented.

Regarding the total input amount spent during the NMB control, it can be seen in Table 3.5 that in average the last controller of the switching strategy, the controller with the best reference tracking, the controller with the most similar gamma, and the controller with the most similar

Table 3.5: Total input amount during total mass control with different total mass controllers chosen by different metrics (*atracurium*)

Total Input Control (in $\mu g/kg$)										
		Switching	Ref Track	Alpha	Gamma	Uss	Norm2	Vinnicombe	Impulse	Step
N	Valid	100	100	100	100	100	100	100	100	100
Mean		6833	6833	7231	6839	6836	7231	7231	7231	7231
Median		6065	6064	6376	6062	6062	6376	6376	6376	6376
Std. Deviation		2555	2591	2998	2581	2582	2998	2998	2998	2998
Range		11309	11481	16149	11481	11481	16149	16149	16149	16149
Minimum		3679	3679	3825	3679	3679	3825	3825	3825	3825
Maximum		14988	15160	19974	15160	15160	19974	19974	19974	19974
Percentiles	25	5013	5013	4959	5013	5013	4959	4959	4959	4959
	75	8073	8065	8399	8065	8065	8399	8399	8399	8399

U_{SS} exhibit the lowest values. The same happens for the standard deviation showing that, aside from the lower consumption, these controllers have the most similar values among them. This conclusion is reinforced by the smaller values in the range. Analyzing the minimum and maximum values these controllers present the best values. Summarizing the last controller from the switching along with the controller with the best reference tracking, the gamma controller, and the U_{ss} controller lead to smaller drug input during the NMB total system mass control.

Table 3.6: Set-Point of the NMB level during total mass control with different total mass controllers chosen by different metrics (*atracurium*)

Set-Point (in %)										
		Switching	Ref Track	Alpha	Gamma	Uss	Norm2	Vinnicombe	Impulse	Step
N	Valid	100	100	100	100	100	100	100	100	100
Mean		9,92	9,93	10,93	9,88	9,90	10,93	10,93	10,93	10,93
Median		9,90	9,93	9,40	9,85	9,90	9,40	9,40	9,40	9,40
Std. Deviation		0,55	0,54	8,69	0,55	0,55	8,69	8,69	8,69	8,69
Range		5,85	5,85	35,9	5,90	5,90	35,9	35,9	35,9	35,9
Minimum		7,55	7,55	0,10	7,55	7,55	0,10	0,10	0,10	0,10
Maximum		13,40	13,40	36,0	13,45	13,45	36,0	36,0	36,0	36,0
Percentiles	25	9,76	9,81	4,01	9,75	9,75	4,01	4,01	4,01	4,01
	75	10,05	10,05	16,73	10,05	10,05	16,73	16,73	16,73	16,73

Analyzing the achieved set-point (see Table 3.6), once again the last controller from switching, the controller with the best reference tracking, the controller with the most similar gamma, and the controller with the most similar U_{SS} show the closest mean values to the desired reference during the NMB control, of 10%, although the mean values obtained with the other controllers are not much different. Besides that, the values for the standard deviation show that the other controllers produce set-point values that have wide variability and so the reference tracking is not quite satisfactory (this situation will be proven ahead with the results for the tracking quality). The higher values for the range, as well as the smaller values in the minimum and the higher values in the maximum that are obtained using the other controllers corroborate that same conclusion. The standard deviation, range, minimum and maximum values from the first controllers are quite good. Concluding, once again, the last controller from switching together with the controller with the best reference tracking, the controller with the most similar gamma, and the one with the most similar U_{SS} present better results in the set-points achieved during the NMB control with total system mass control law.

Table 3.7: Settling Time of the NMB level during total mass control with different total mass controllers chosen by different metrics (*atracurium*)

Settling Time (in minutes)										
		Switching	Ref Track	Alpha	Gamma	Uss	Norm2	Vinnicombe	Impulse	Step
N	Valid	100	100	100	100	100	100	100	100	100
Mean		64	65	70	66	66	70	70	70	70
Median		67	67	68	67	67	68	68	68	68
Std. Deviation		21	22	21	22	22	21	21	21	21
Range		86	87	90	87	87	90	90	90	90
Minimum		25	24	27	24	24	27	27	27	27
Maximum		111	111	117	111	111	117	117	117	117
Percentiles	25	46	47	56	47	47	56	56	56	56
	75	81	81	87	82	82	87	87	87	87

Regarding the results for the settling time, in Table 3.7, despite the lower values with the last controllers of switching, the controllers with the best reference tracking, the controllers with the most similar gamma, and the controllers with the most similar U_{SS}, the difference in relation with the other controllers is not so significant since the mean values differ from each other from at most 6 minutes. Also the values from the standard deviation, range, minimum and maximum are very similar. Therefore the settling time is not enough to distinguish the controller performance.

Table 3.8: Reference tracking error during total mass control with different total mass controllers chosen by different metrics (*atracurium*)

Reference Tracking Error (%)										
		Switching	Ref Track	Alpha	Gamma	Uss	Norm2	Vinnicombe	Impulse	Step
N	Valid	100	100	100	100	100	100	100	100	100
Mean		0,26	0,24	7,13	0,24	0,24	7,13	7,13	7,13	7,13
Median		0,14	0,12	6,17	0,11	0,11	6,17	6,17	6,17	6,17
Std. Deviation		0,46	0,46	5,19	0,46	0,46	5,19	5,19	5,19	5,19
Range		3,29	3,29	25,94	3,29	3,29	25,94	25,94	25,94	25,94
Minimum		0,02	0,02	0,21	0,02	0,02	0,21	0,21	0,21	0,21
Maximum		3,31	3,31	26,15	3,31	3,31	26,15	26,15	26,15	26,15
Percentiles	25	0,07	0,06	3,37	0,06	0,06	3,37	3,37	3,37	3,37
	75	0,26	0,21	9,42	0,21	0,20	9,42	9,42	9,42	9,42

Finally, considering the results for the reference tracking quality (see Table 3.8), it can be seen that the last controllers from switching, the controllers with the best reference tracking, the controllers with the most similar gamma, and the controllers with the most similar U_{SS} present the best mean values, once the errors are the lowest. Analyzing the standard deviation and the range, the lower values presented for these controllers show once again that better reference tracking can be achieved in comparison with the other controllers (Alpha, Norm2, Vinnicombe, Impulse and Step). The smaller values for the minimum and maximum error reinforce the same conclusion, a better reference tracking is achieved with the first controllers, as was previously seen with respect to the set-point results.

As a conclusion, for *atracurium* models, the last controller from the switching together with the controller with the best reference tracking, the controller from the model with the most similar value of gamma to the one of the patient model, and the controller from the model with the most similar U_{SS} to the one of the patient model present better control because lower

input doses are required, the reference tracking quality is better, and the set-points achieved are quite satisfactory in comparison with the desired NMB target of 10%. So one can say that, for *atracurium* models, the switching strategy converges to a good controller since this one provides a good control of the NMB level.

Table 3.9: Total input amount during total mass control with different total mass controllers chosen by different metrics (*rocuronium*)

Total Input Control (in $\mu g/kg$)										
		Switching	Ref Track	Alpha	Gamma	Uss	Norm2	Vinnicombe	Impulse	Step
N		37	37	37	37	37	37	37	37	37
Mean		3575	3604	3123	3603	3549	3123	3123	3123	3123
Median		3121	3284	3087	3284	3284	3087	3087	3087	3087
Std. Deviation		1954	1986	1607	1978	1921	1607	1607	1607	1607
Range		6661	6816	6798	6816	6816	6798	6798	6798	6798
Minimum		893	893	924	893	893	924	924	924	924
Maximum		7554	7709	7722	7709	7709	7722	7722	7722	7722
Percentiles	25	2091	2213	1977	2214	2214	1977	1977	1977	1977
	75	4370	4442	3637	4331	4331	3637	3637	3637	3637

Analyzing now the results obtained with *rocuronium* models (see the models at Appendix C.2), it can be seen that for the total input spent during the NMB control (see Table 3.9) the controllers of the Alpha proximity, Norm2 and Vinnicombe metrics, and Impulse and Step proximity lead to less input doses than the other controllers. Despite the minimum and maximum values being quite similar for all the controllers, the mean values for the first controllers as well as the standard deviation values, are smaller than the ones for the others. So in a global appreciation, the last controllers from switching together with the controllers with best reference tracking, the controllers with the most similar gamma, and the controllers with the most similar U_{SS} show to spend more input dose during the NMB control.

Table 3.10: Set-Point of the NMB level during total mass control with different total mass controllers chosen by different metrics (*rocuronium*)

Set-Point (in %)										
		Switching	Ref Track	Alpha	Gamma	Uss	Norm2	Vinnicombe	Impulse	Step
N	Valid	37	37	37	37	37	37	37	37	37
Mean		10,14	10,14	13,05	10,08	10,17	13,05	13,05	13,05	13,05
Median		10,15	10,05	11	10,05	10,05	11	11	11	11
Std. Deviation		1,07	0,97	10,30	0,98	0,97	10,30	10,30	10,30	10,30
Range		5,7	5,7	43	5,7	5,7	43	43	43	43
Minimum		7,9	7,9	0,05	7,9	7,9	0,05	0,05	0,05	0,05
Maximum		13,6	13,6	43,05	13,6	13,6	43,05	43,05	43,05	43,05
Percentiles	25	9,55	9,73	4,75	9,63	9,65	4,75	4,75	4,75	4,75
	75	10,65	10,65	20,25	10,575	10,65	20,25	20,25	20,25	20,25

With respect to the results for the set-points achieved (see Table 3.10), the last controller from switching, the controller with the best reference tracking, the gamma proximity controller, and the U_{SS} controller present better results, once the mean set-point values for these controllers are much closer to the NMB reference of 10%, used during the NMB control, than the mean set-point values for the other controllers; also the standard deviation of the first controllers is much smaller than the one verified for the Alpha, Norm2, Vinnicombe, Impulse and Step controllers, showing that not only the mean behavior of the first controllers is closer to the NMB target, but

also the range of set-points obtained are closer to the mean values; this situation is reinforced by the higher values of minimum and the lower values of maximum. Therefore it can be said that the last controller from switching along with the controller with the best reference tracking, and the gamma and U_{SS} controllers present better set-point results. This better tracking quality will be confirmed ahead when the results of Table 3.12 are analyzed.

Table 3.11: Settling Time of the NMB level during total mass control with different total mass controllers chosen by different metrics (*rocuronium*)

Settling Time (in minutes)										
		Switching	Ref Track	Alpha	Gamma	Uss	Norm2	Vinnicombe	Impulse	Step
N	Valid	37	37	37	37	37	37	37	37	37
Mean		86	85	94	88	88	94	94	94	94
Median		76	76	80	85	88	80	80	80	80
Std. Deviation		40	40	43	40	40	43	43	43	43
Range		146	146	172	146	146	172	172	172	172
Minimum		35	35	37	35	35	37	37	37	37
Maximum		180	180	209	180	180	209	209	209	209
Percentiles	25	57	46	58	52	52	58	58	58	58
	75	113	117	124	119	119	124	124	124	124

Regarding the settling time (see Table 3.11) once again the controllers from the last switching, the controller with the best reference tracking, the gamma proximity controller, and the U_{SS} proximity controller have lower mean values, between 85 and 88 minutes, unlike the other controllers that have a mean value of 94 minutes for settling the NMB level. About the standard deviation and the minimum values, they are very similar for all the controllers, but when the range and the maximum values are analyzed the first controllers show again better results, once they register lower values. Overall the last controller from switching along with the best offline controller, the controller with the most similar gamma, and the controller with the most similar U_{SS} have better settling times because they reach the set-point earlier than the other controllers.

Table 3.12: Reference tracking error during total mass control with different total mass controllers chosen by different metrics (*rocuronium*)

Reference Tracking Error (%)										
		Switching	Ref Track	Alpha	Gamma	Uss	Norm2	Vinnicombe	Impulse	Step
N	Valid	37	37	37	37	37	37	37	37	37
Mean		0,79	0,58	8,21	0,56	0,58	8,21	8,21	8,21	8,21
Median		0,68	0,31	5,74	0,38	0,38	5,74	5,74	5,74	5,74
Std. Deviation		0,66	0,67	6,96	0,65	0,68	6,96	6,96	6,96	6,96
Range		3,19	3,19	32,24	3,19	3,19	32,24	32,24	32,24	32,24
Minimum		0,02	0,02	1,02	0,02	0,02	1,02	1,02	1,02	1,02
Maximum		3,21	3,21	33,26	3,21	3,21	33,26	33,26	33,26	33,26
Percentiles	25	0,28	0,14	3,60	0,18	0,18	3,60	3,60	3,60	3,60
	75	1,17	0,71	10,82	0,71	0,71	10,82	10,82	10,82	10,82

Finally considering the tracking quality with respect to the NMB reference level of 10% (see Table 3.12) it can be seen that the mean error values for the last controller of switching, the controller with the best reference tracking, the most similar controller by gamma proximity, and the controller with the most closer U_{SS} are smaller. Analyzing the standard deviation, once again the former controllers show better results with smaller values, as well as for the range, showing a more similar reference tracking performance and with better quality in all the simulated cases

with *rocuronium* models. The lower values for minimum and maximum tracking error reinforce once again this same conclusion. Therefore the last controller from switching together with the controller with the best reference tracking, the controller with the most similar gamma, and the controller with the most similar U_{SS} present better reference tracking quality, a main goal desired in the NMB control.

As a conclusion, for *rocuronium* models, the last controller from the switching together with the controller with the best reference tracking, the controller from the model with the most similar value of gamma to the one of the patient model, and the controller from the model with the most similar U_{SS} to the one of the patient model do not show the best results regarding the total input required to control the NMB level, but once again (as for the *atracurium* models) the reference tracking quality is better, the set-points achieved are closer to the NMB target of 10%, and now the settling time allows to differentiate the controllers because the former controllers have lower values. So one can say that, for *rocuronium* models, the switching strategy converges to a good controller since this one provides a good control of the NMB level.

3.3.6 Reference Tracking Improvement

Although the switching control scheme can take into account some problems related with the variations on the patient behavior, like all the previously developed controllers, it is also nonoptimal, and carries some issues especially related with the reference tracking. Indeed, since the parameters (α, γ) as well as the value of M^* for a real patient are unknown, and the bank has a finite number of controllers, it is expected that the parameters of the control law are not strictly equal to the real patient parameters, and so a reference steady state tracking error is expected.

In order to overcome this drawback a scheme to improve the tracking quality is here proposed. This strategy, relying on the NMB response at steady state, performs the online tunning of the patient parameter γ [Alonso et al. (2008)].

For this purpose the input applied to the patient must be at steady state, u_{ss}, and the corresponding steady state NMB level response, r_{ss}, is supposed to be reached.

Notice that, in steady-state, equation (3.8) becomes:

$$r_{ss} = \frac{100}{1 + \left(\frac{y_{ss}}{k.C_{50}}\right)^{\gamma^*}}, \tag{3.32}$$

where γ^* is the correct value of the parameter γ.

Moreover,

$$y_{ss} = G(0).u_{ss},$$

where

$$G(s) = \frac{k_1 k_2 k_3 \alpha^3}{(s+\alpha k_1)(s+\alpha k_2)(s+\alpha k_3)}$$

is the *reduced parameter* model transfer function. Since $G(0) = 1 \rightarrow y_{ss} = u_{ss}$ and hence:

$$r_{ss} = \frac{100}{1 + \left(\frac{u_{ss}}{k.C_{50}}\right)^{\gamma^*}}, \tag{3.33}$$

Now solving for γ^* yields:

$$\gamma^* = \frac{\ln\left(\frac{100}{r_{ss}} - 1\right)}{\ln\left(\frac{u_{ss}}{k.C_{50}}\right)}. \tag{3.34}$$

In practice it is assumed that u_{ss} and r_{ss} are attained when the variation of the values of $u(t)$ and $r(t)$ are smaller than an adequate threshold.

It turns out that a correct identification of this parameter will lead to the improvement of the control law, since the value of $M^* = 3.y_{ref}$ only depends of the parameter γ (see (3.11) and (3.26)). After this step the parameter γ is fixed as γ^* and each control law $\tilde{u}_i$ is adapted as:

$$\begin{aligned} u_i(t) &= max(0, \tilde{u}_i(t)) \\ \tilde{u}_i(t) &= (\Sigma b_j)_i^{-1}[1 \cdots 1](-\lambda I - A_i)x(t) + (\Sigma b_i)^{-1}\lambda M^*. \end{aligned} \tag{3.35}$$

Moreover the choice of the controller to be applied at each instant is now made based on the cumulative quadratic error between the effect concentration response of the patient ($y_{patient}$) and the effect concentration response of the i-th model in the bank ($i = 1, \ldots, N$) (y_i).

$$e_i = y_{patient} - y_i$$

This cumulative error is obtained from the instant where the controller of the patient NMB level starts using the control law (3.35).

In order to incorporate this improvement strategy a further step is added to the switching strategy leading to a fifth step:

- **Step 5** (if applied): Corresponds to the reference tracking improvement where γ is fixed as γ^* given by (3.34), and the selection of the controller $\mathcal{K}_i$ is then based on the minimization of the cumulative error between the y response of the patient and the y_i responses of each of the models in the bank $\mathcal{P}$ in order to obtain α_i for b_i and A_i of the control law (3.35).

3.3.7 Reference Tracking Improvement Results

Figs. 3.17, 3.18 and 3.19 show the three worst results obtained with the switching strategy using the bank $\mathcal{P}$ of *atracurium* models (see Appendix C.1). The models that produced these results where $\mathcal{M}_{27}$, $\mathcal{M}_{34}$, and $\mathcal{M}_{38}$. In Table 3.13 the results obtained in these simulations can be observed in columns where there is no value for *Calibration Time*. In all the three cases an initial *bolus* of 500 $\mu g/kg$ was used, and the total time of simulation was 300 minutes.

Table 3.13: Comparison between switching results with *atracurium* models with and without reference tracking improvement

Patient Model	27	27	34	34	38	38
Recovery to the Initial *Bolus* (min)	41	41	47	47	24	24
First Controller $\mathcal{K}$ (Random)	74	74	5	5	32	32
Last Controller $\mathcal{K}$	43	80	27	49	30	60
Patient U_{ss} $(\mu g/kg/min)$	4,72	4,72	4,5	4,5	23,19	23,19
Real U_{ss} $(\mu g/kg/min)$	4,92	4,66	4,72	4,44	17,37	21,03
ΔNMB (%)	1,94	0,75	2,6	0,7	3,3	1,03
Calibration Time (min)	-	109	-	200	-	74

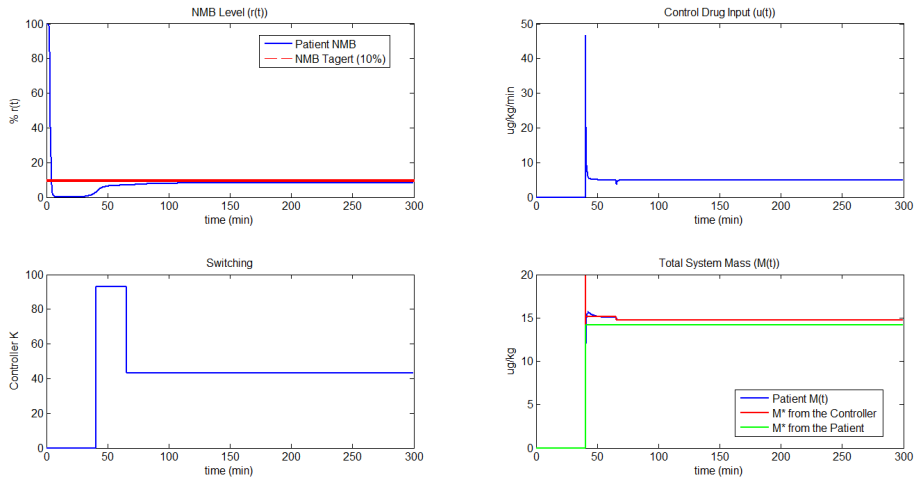

Figure 3.17: NMB control simulation with $\mathcal{M}_{27}$ from the bank $\mathcal{P}$ of *atracurium* models with no calibration.

For the first case, Fig. 3.17, the controller $\mathcal{K}_{27}$, was removed from the controller bank. The recovery time, and so the instant for the controller started up, was t^*=41 min, and the control begins with controller $\mathcal{K}_{74}$. By the end of the simulation the controller chosen was $\mathcal{K}_{43}$ with a steady-state dose of 4.92 $\mu g/kg/min$. Comparing with the steady-state dose of 4.72 $\mu g/kg/min$, required to lead the NMB level of this model to the NMB target of 10%, it can be seen that there is a difference between these doses, leading to a difference between the final NMB level and the NMB target level (ΔNMB) ΔNMB=1.94% (See Table 3.13).

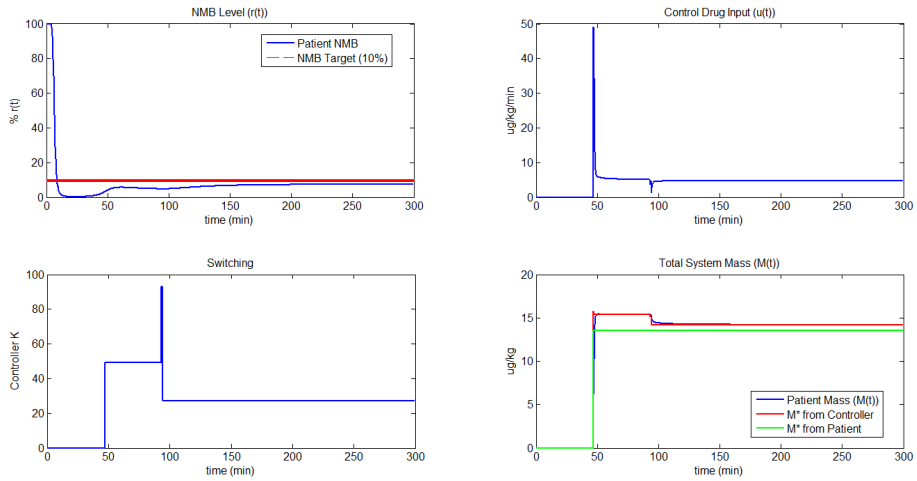

Figure 3.18: NMB control simulation with $\mathcal{M}_{34}$ from the bank $\mathcal{P}$ of *atracurium* models with no calibration.

In the second case, Fig. 3.18 the controller $\mathcal{K}_{34}$ was removed from the controller bank and the recovery time occured at t^*=47 min. The control started up with controller $\mathcal{K}_5$ and ended with the controller $\mathcal{K}_{27}$ with a steady-state dose of 4.72 $\mu g/kg/min$, different from the required dose of 4.5 $\mu g/kg/min$ needed to lead the NMB level to the target value of 10%. Therefore a difference ΔNMB=2.6% between the final NMB level and the desired NMB target was observed (See Table 3.13).

For the third case, Fig. 3.19, the controller removed from the controller bank was $\mathcal{K}_{38}$. The recovery time was t^*=24 min, and at this time the control started up with the controller $\mathcal{K}_{32}$. By the end of the simulation time the last applied controller was $\mathcal{K}_{30}$ with a steady-state dose of 17.37 $\mu g/kg/min$, very different of the 21.03 $\mu g/kg/min$, required to lead the NMB level to the target of 10%, and so a difference ΔNMB=3.3% between the final NMB level obtained and the desired NMB target was observed (See Table 3.13).

Figs. 3.20, 3.21, and 3.22 show the results of applying the reference tracking improvement technique to the cases of Figs. 3.17, 3.18, and 3.19 respectively. The values obtained for these

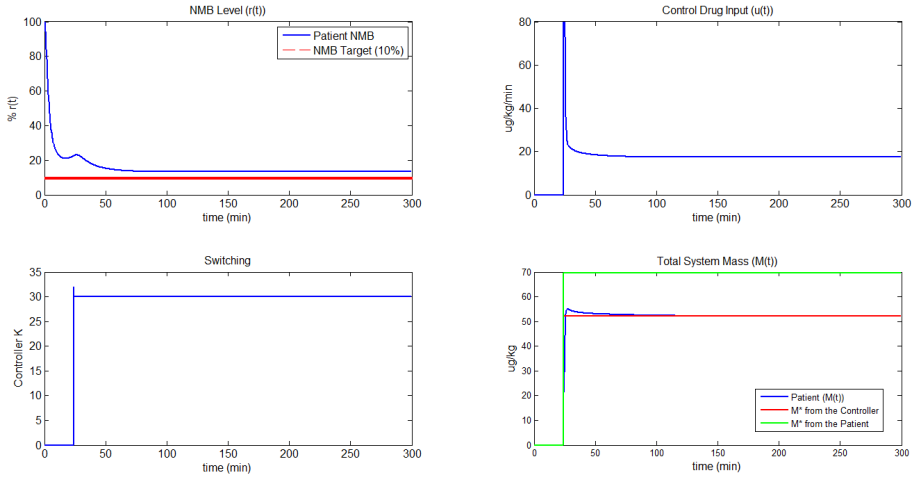

Figure 3.19: NMB control simulation with $\mathcal{M}_{38}$ from the bank $\mathcal{P}$ of *atracurium* models with no calibration.

cases can be visualized in Table 3.13 in the columns where there are values for the *Calibration Time*. This time corresponds to the instant where the reference tracking improvement was applied. As expected the last controller to be used was not the same as previously, the same happens with the final steady-state dose applied, which obviously leads to different values of ΔNMB.

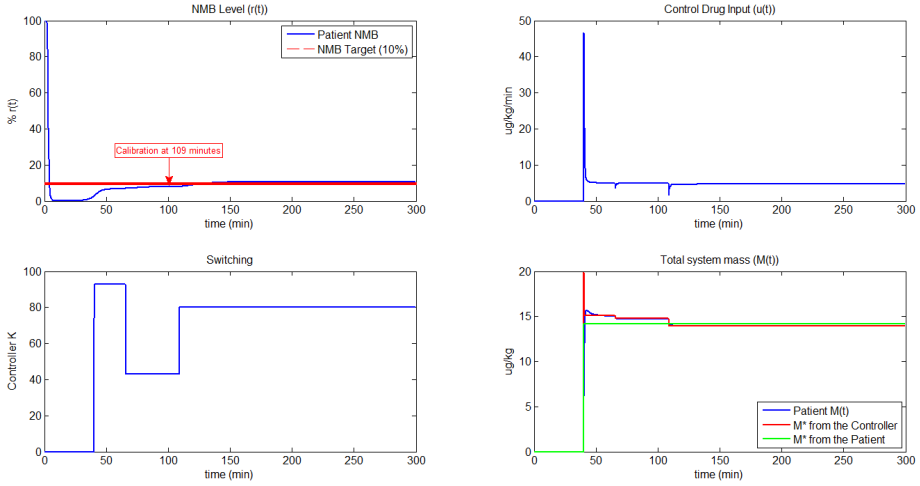

Figure 3.20: NMB control simulation with $\mathcal{M}_{27}$ from the bank $\mathcal{P}$ of *atracurium* models with calibration at 109 minutes.

For the first case, Fig. 3.20, a reference tracking improvement to the case of Fig. 3.17 ($\mathcal{M}_{27}$), the calibration occurred at t=109 min, and the last controller applied after the 300 minutes of simulation was $\mathcal{K}_{80}$, with a steady-state dose of 4.66 $\mu g/kg/min$ instead of the dose of 4.92 $\mu g/kg/min$ (without calibration). As a consequence the difference between the final NMB level and the NMB target level was decreased from ΔNMB=1.95% to ΔNMB=0.75%, which is a significant improvement (See Table 3.13).

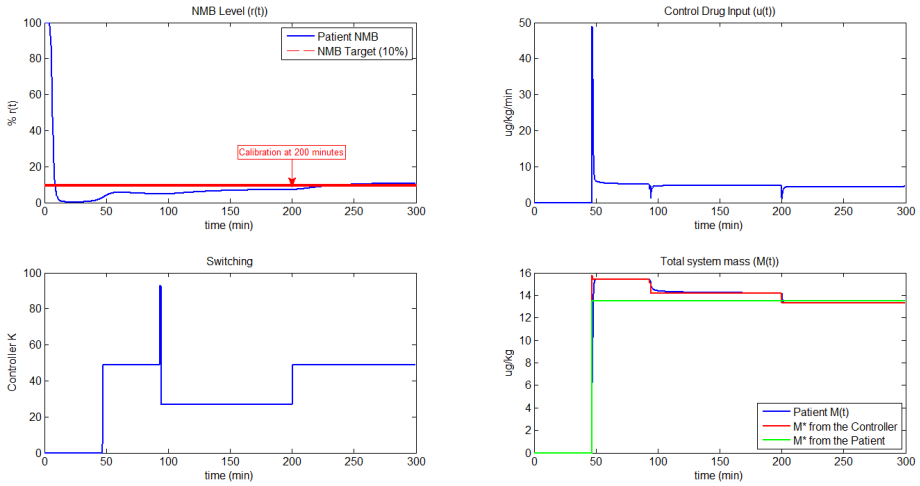

Figure 3.21: NMB control simulation with $\mathcal{M}_{34}$ from the bank $\mathcal{P}$ of *atracurium* models with calibration at 200 minutes.

In the second case, Fig. 3.21, a reference tracking improvement to the case of Fig. 3.18 ($\mathcal{M}_{34}$), the calibration was applied at minute 200. The final controller was $\mathcal{K}_{49}$ with a steady-state dose of 4.44 $\mu g/kg/min$ instead of the dose of 4.72 $\mu g/kg/min$ (without calibration). Therefore the difference between the final NMB level and the NMB target level decreased from ΔNMB=2.6% to ΔNMB=0.7%, once again showing a significant improvement (See Table 3.13).

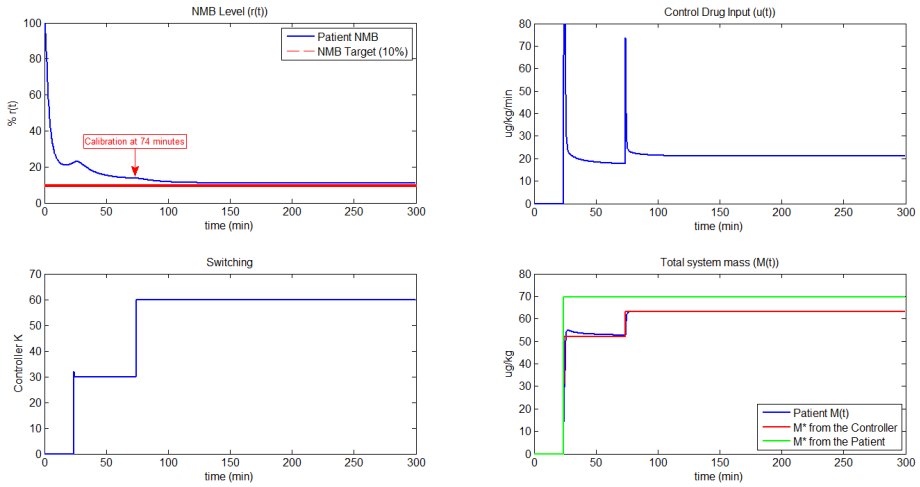

Figure 3.22: NMB control simulation with $\mathcal{M}_{38}$ from the bank $\mathcal{P}$ of *atracurium* models with calibration at 74 minutes.

In the third case, Fig. 3.22, a reference tracking improvement to the case of Fig. 3.19 ($\mathcal{M}_{38}$), the calibration happened at minute 74, and as a consequence the last controller applied was $\mathcal{K}_{60}$ with a steady-state dose of 21.03 $\mu g/kg/min$ instead of the dose of 17.37 $\mu g/kg/min$ (without calibration). Thus the difference between the final NMB level and the NMB target level decreased from the ΔNMB=3.3% to ΔNMB=1.03%, once again showing a significantly better performance (See Table 3.13).

Summarizing, the reference tracking improvement, when properly applied (the input must remain in a steady-state dose, and the steady-state NMB response to that dose must be achieved), produces significant improvements to the reference tracking when *atracurium* models are used, since the difference between the final NMB level and the NMB target level decrease.

Figs. 3.23, 3.24 and 3.25 show the three worst results obtained with the switching strategy using the bank $\mathcal{P}$ for *rocuronium* models (see Appendix C.2). The models that produced these results were $\mathcal{M}_{14}$, $\mathcal{M}_{22}$, and $\mathcal{M}_{28}$. In Table 3.14 the results obtained in these simulations can be observed in columns where there is no value for the *Calibration Time*. In all the three cases an initial *bolus* of 600 $\mu g/kg$ was used, and the total time of simulation was 300 minutes.

Table 3.14: Comparison between switching results with *rocuronium* models with and without reference tracking improvement

Patient Model	14	14	22	22	28	28
Recovery to the Initial *Bolus* (min)	93	93	88	88	33	33
First Controller $\mathcal{K}$ (Random)	18	18	2	2	15	15
Last Controller $\mathcal{K}$	21	30	14	10	33	14
Patient U_{ss} $(\mu g/kg/min)$	1,37	1,37	1,42	1,42	13,09	13,09
Real U_{ss} $(\mu g/kg/min)$	1,42	1,36	1,37	1,39	9,03	11,37
ΔNMB (%)	1,71	0,82	1,84	1,23	3,24	1,14
Calibration Time (min)	-	200	-	180	-	109

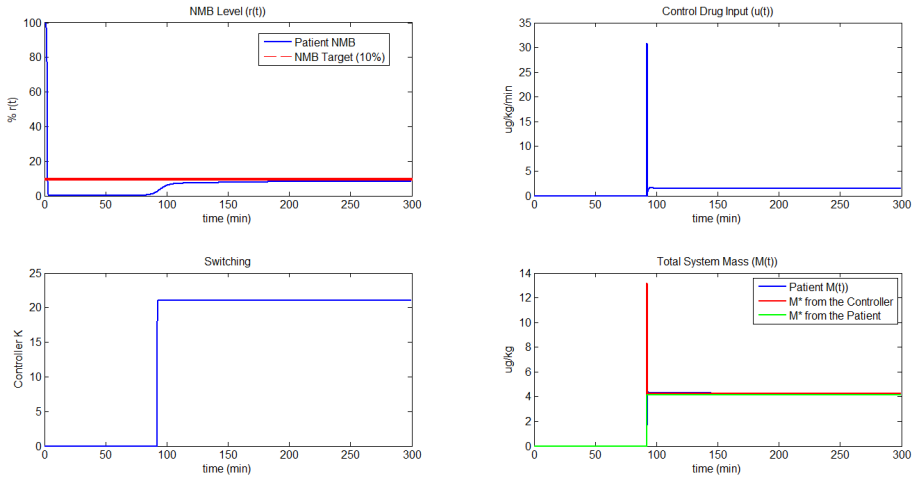

Figure 3.23: NMB control simulation with $\mathcal{M}_{14}$ from the bank $\mathcal{P}$ of *rocuronium* models with no calibration.

In the first case, Fig. 3.23, where the model $\mathcal{M}_{14}$ is used to simulate the patient dynamics, the respective controller $\mathcal{K}_{14}$ was removed from the controller bank. The recovery time was t^*=93 min, and the control was initiated with the controller $\mathcal{K}_{18}$, and ended with the controller $\mathcal{K}_{21}$ with a steady-state dose of 1.42 $\mu g/kg/min$ leading to a difference between the final NMB level and the desired NMB target ΔNMB=1.71% (see Table 3.14).

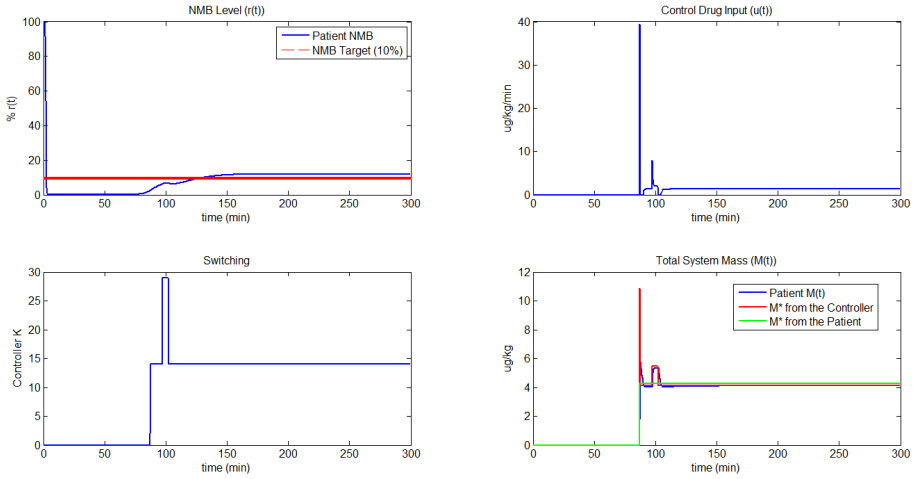

Figure 3.24: NMB control simulation with $\mathcal{M}_{22}$ from the bank $\mathcal{P}$ of *rocuronium* models with no calibration.

For the second case, Fig. 3.24, the controller corresponding to the model used to simulate the patient dynamics, $\mathcal{K}_{22}$, was removed from the controller bank, and when the recovery happened, t^*=88 min, the control initiate with the controller $\mathcal{K}_2$, and ended with the controller $\mathcal{K}_{14}$ with a steady-state dose of 1.37 $\mu g/kg/min$. In comparison with the steady-state dose of 1.42 $\mu g/kg/min$, required to lead the NMB level to the desired NMB target, a difference ΔNMB=1.84% was observed (see Table 3.14).

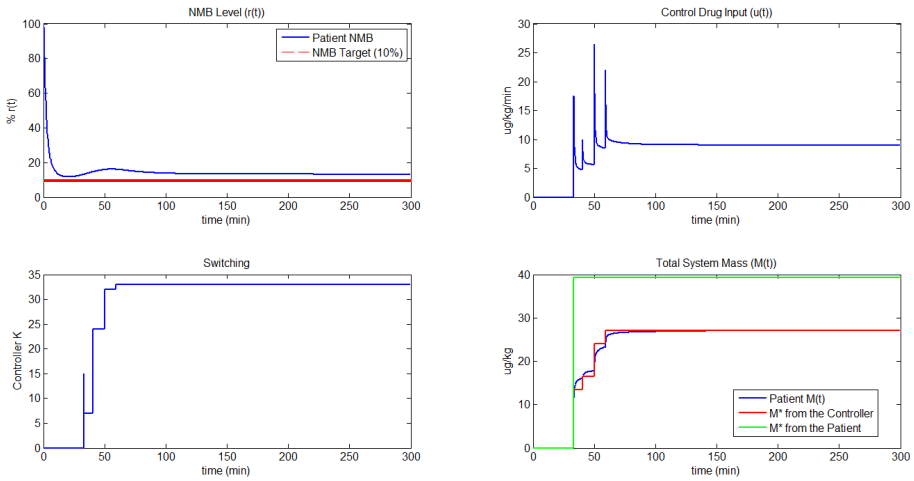

Figure 3.25: NMB control simulation with $\mathcal{M}_{28}$ from the bank $\mathcal{P}$ of *rocuronium* models with no calibration.

Then, in the third case, Fig. 3.25 the respective controller to the model used to simulate the patient dynamics, $\mathcal{K}_{28}$, was removed from the controller bank. The recovery time was t^*=33 min, and the control started with the controller $\mathcal{K}_{15}$, and at the end of the simulation the controller applied was $\mathcal{K}_{33}$ with a steady-state dose of 9.03 $\mu g/kg/min$. This dose is significantly

different from the dose of 13.09 $\mu g/kg/min$, required to drive the NMB level to the desired NMB target. Therefore a final difference between the achieved NMB level and the desired NMB target ΔNMB=3.24% occurred (see Table 3.14).

Figs. 3.26, 3.27, and 3.28 show the results of the application of the reference tracking improvement technique to the cases of Figs. 3.23, 3.24, and 3.25, respectively. The values obtained with this improvement technique are in the columns where there are values for the *Calibration Time* in Table 3.14. As happened in the case of *atracurium*, the calibration time corresponds to the instant were the reference tracking improvement was applied. Once more, as expected, the last controller was not the same as previously, and the same occurred with the final steady-state dose applied, which obviously leads to different values of ΔNMB.

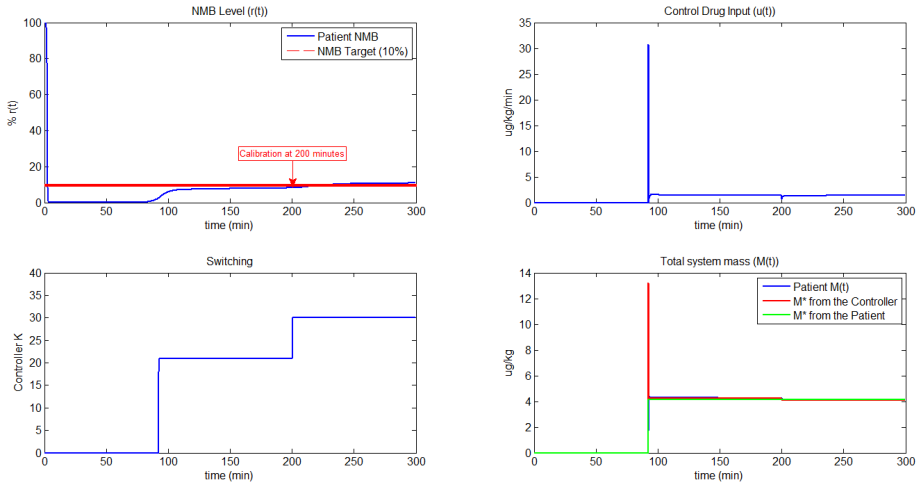

Figure 3.26: NMB control simulation with $\mathcal{M}_{14}$ from the bank $\mathcal{P}$ of *rocuronium* models with calibration at 200 minutes.

For the first case, Fig. 3.26, the calibration was applied at minute 200. This time the final controller was $\mathcal{K}_{30}$, with a steady-state dose of 1.36 $\mu g/kg/min$ instead of the dose of 1.42 $\mu g/kg/min$ (without calibration). Therefore the difference between the final NMB level achieved and the desired NMB target decreased from ΔNMB=1.71% to ΔNMB=0.82%, showing better tracking results (see Table 3.14).

In the second case, Fig. 3.27, the calibration was applied at minute 180, and the last controller applied was $\mathcal{K}_{10}$ with a steady-state dose of 1.39 $\mu g/kg/min$ instead of the previously dose of 1.37 $\mu g/kg/min$ (without calibration). The difference between the final NMB level achieved and the NMB target level decreased from ΔNMB=1.84% to ΔNMB=1.23%, revealing an enhancement of the reference tracking quality once again (see Table 3.14).

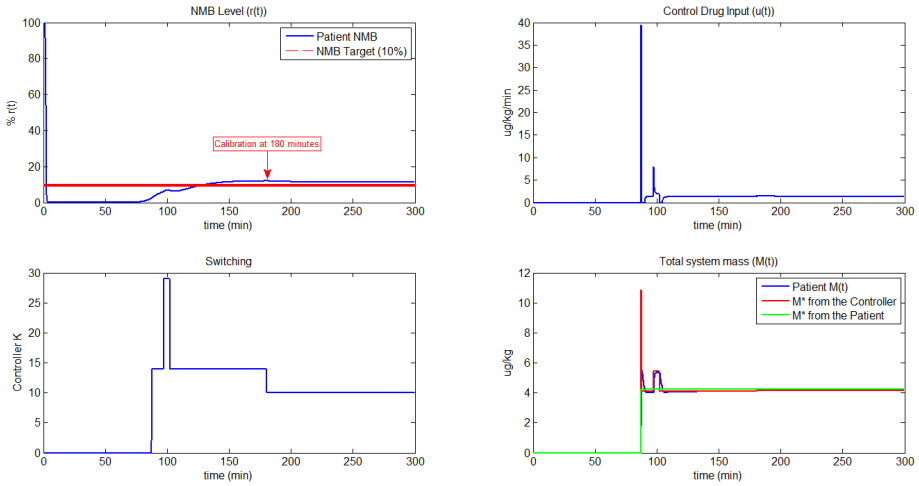

Figure 3.27: NMB control simulation with $\mathcal{M}_{22}$ from the bank $\mathcal{P}$ of *rocuronium* models with calibration at 180 minutes.

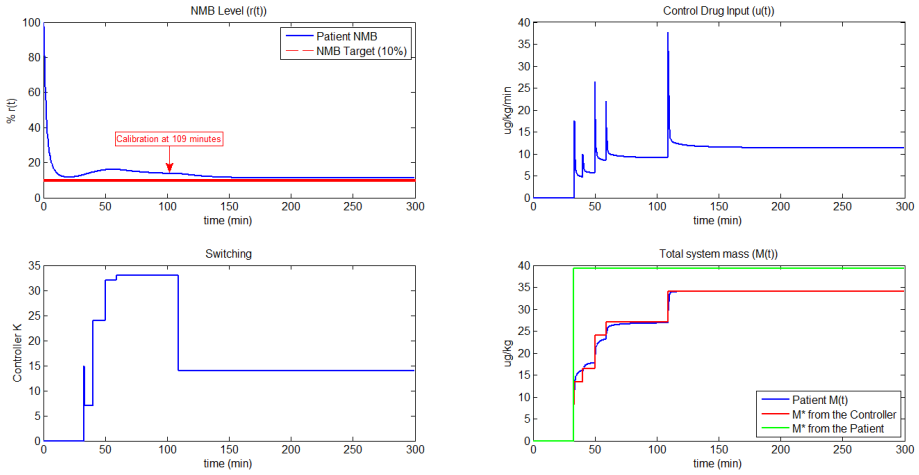

Figure 3.28: NMB control simulation with $\mathcal{M}_{28}$ from the bank $\mathcal{P}$ of *rocuronium* models with calibration at 109 minutes.

For the third case, in Fig. 3.28, the calibration was applied at minute 109. The final controller applied was $\mathcal{K}_{14}$ with a steady-state dose of 11.37 $\mu g/kg/min$ instead of the dose of 9.03 $\mu g/kg/min$ (without calibration). Therefore the difference between the achieved NMB level and the desired NMB target decreased from ΔNMB=3.24% to ΔNMB=1.14%, showing once more a significant improvement in the reference tracking quality (See Table 3.14).

In summary, once again, when properly applied (the input must remain in a steady-state dose, and the steady-state NMB response to that dose must be achieved), the reference tracking improvement produces significant improvements in the reference tracking quality for *rocuronium* models, since the difference between the final NMB level and the desired NMB target is reduced.

Chapter 4

Clinical Environment

The NMB control strategy presented in this work based on a switching total system mass control law was implemented at the *Unidade Local de Saúde de Matosinhos - Hospital Pedro Hispano* (ULSM-HPH) with the supervision of the anesthesiologist Dr. Rui Rabiço.

In order to apply the NMB control to real patients it was necessary to use the appropiate NMB control hardware and software, and to follow a suitable protocol defined by the anesthesiologist. The protocol varies according to the surgery type and the patient characteristics.

4.1 Hardware for NMB Control

Figure 4.1: Datex-Ohmeda modular system for anesthesia monitoring.

The NMB control system applied at the ULSM-HPH consists of:

- A Datex-Ohmeda device (a modular monitoring family with a full line-up of monitors and accessories, like the anesthesia monitor, a complex solution for anesthesia patient

Figure 4.2: Alaris GH Pump for drug infusion.

monitoring during surgeries), in Fig. 4.1, along with a NMT (neuromuscular transmission) module (Fig. 2.3) in order to obtain the NMB level of the patient during surgery;

- A drug delivery system: an Alaris GH syringe pump (Fig. 4.2) is used in order to administer the muscle relaxant to the patient;
- A computer: a portable computer with two USB serial ports for connecting the Datex-Ohmeda device and the infusion pump, in order to run the NMB control software.

The Datex-Ohmeda system and the Alaris GH Pump are connected to the portable computer by RS232-USB cables.

4.2 Software for NMB Control

To perform the NMB control a proper software is necessary in order to read the patient NMB level from the Datex-Ohmeda system, and then compute the respective NMB control input using the switching total system mass control strategy. The software must allow the transmission of that input value to the Alaris GH pump.

For that purpose two programs were used namely the *Data Acquisition System* and the *Monitoring Control Anesthesia*. These programs were developed by researchers of the *Galeno Project*.

The first software (Fig. 4.3) does the collection of all the data acquired by the Datex-Ohmeda system, registers the infusion information related to the syringe pump, and also allows to control the pump's infusion rate. It also permits the user to insert special notes about the surgery, like intubation and incision, and to mark the administration of a manual *bolus* (when drugs are administered manually). This software creates three .txt files, one with the values collected from the Datex-Ohmeda, another with the information about the infusion given by the Alaris GH pump, and a last one with all the information related to the patient (Gender, Age, Height and Weight) and to the surgery (protocol, surgery name, special notes and manual *bolus*).

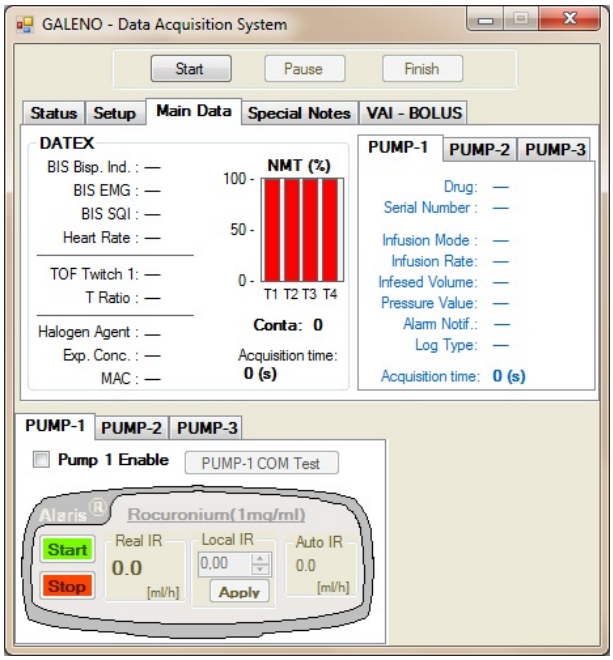

Figure 4.3: Data Acquisition System program developed by Galeno researchers.

The second program (Fig. 4.4) has the main objective to perform the NMB control but it can also realize the plot of some interest variables related with the Datex-Ohmeda and the Alaris GH pump. This program was developed in the MATLAB environment and was designed in a modular way so that any user can develop his own NMB control strategy as a MATLAB function and then incorporate it in the main software.

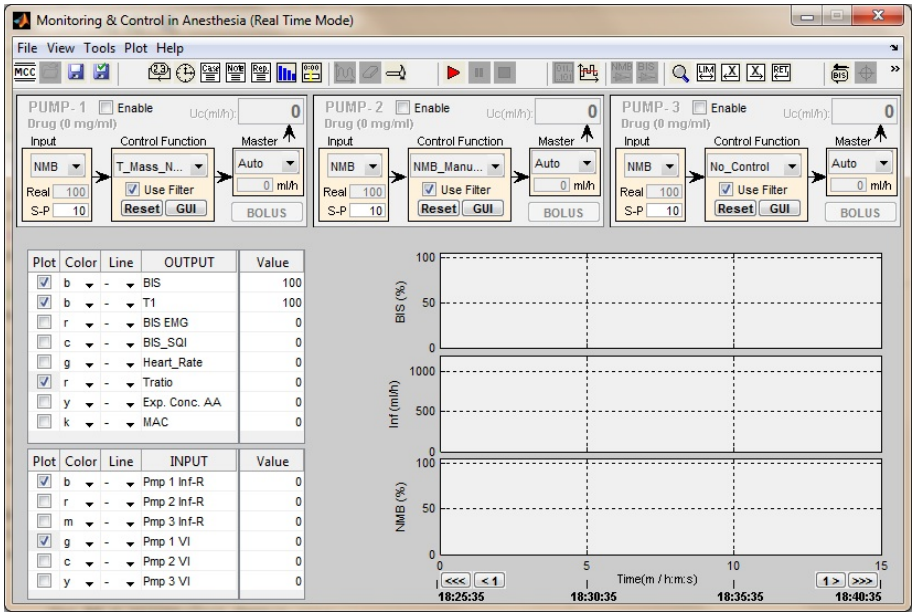

Figure 4.4: Monitoring and Control in Anesthesia program developed by Galeno researchers.

The program makes a connection with the former one in order to obtain from it the collected values. In this way the information necessary to perform the NMB control is provided and the

control input can be determined. Afterwards the *Monitoring and Control in Anesthesia* program sends out that input value to the former program so that it imposes it to the Alaris GH Pump.

A more detailed explanation about these two programs can be read in Appendix B.

4.3 Protocol

The protocol used during the surgeries with the application of the switching strategy to control the NMB level based on a bank of total system mass control laws was designed by the anesthesiologist. It varies according to the characteristics of the patient, the type of surgery and the protocols implemented by the hospital. Since the real cases control was performed at the *Unidade Local de Saúde de Matosinhos - ULSM*, the specific protocol was made by the anesthesiologist *Rui Rabiço* according to the general protocols implemented at this hospital.

For the hypnotic induction a *bolus* of *propofol* was used in order to achieve a quick induction. After this induction the maintenance was made with *sevoflurane*, a volatile drug. The choice of a volatile drug for the maintenance is due to the possibility of using the *Datex Ohmeda* device that is able to make the automatic administration of this drug; the anesthesiologist only needs to define the infusion rate. For analgesia a single *bolus* of *fentanil* was used at the beginning. Finally, for the arreflexia, the drug *rocuronium* was used. this drug is nowadays the most widely used due to its special properties, in particular the existence of *sugamadex*, a drug that makes the full reversion of the NMB level possible at any time. At the end of the surgery either the NMB reverser *neostigmine* with *atropine* or the *sugamadex* were used.

The process of general anesthesia with NMB control during surgeries can be summarized as follows:

- **Step 1** Administration of a *fentanil bolus*.
- **Step 2** Administration of a *propofol bolus* followed by *sevoflurane* administration when the BIS signal starts to recover from the *propofol bolus*.
- **Step 3** Calibration of the NMB Sensor.
- **Step 4** $t = 0$: An initial *bolus* of *rocuronium* is given to the patient (Automatic administration performed by the *Monitoring and Control in Anesthesia software*).
- **Step 5** $0 < t < t^*$ (before recovery from the initial *bolus* of *rocuronium*): The control input remains zero.

- **Step 6** $t = t^*$ (t^* computed by the OLARD algorithm introduced in da Silva et al. (2009)): A random controller from bank $\mathcal{K}$ is chosen in order to start the control of the NMB response of the patient.

- **Step 7** $t \geq t^*$: A controller $\mathcal{K}_i$ is chosen at each time instant based on the minimization of the cumulative error between the patient response and the responses of each of the models in the bank $\mathcal{P}$.

- **Step 8** At the end of the surgery: Administration of either the NMB reverser or the *rocuronium* antagonist according to the patient characteristics and his/her NMB level at this stage.

4.4 Results

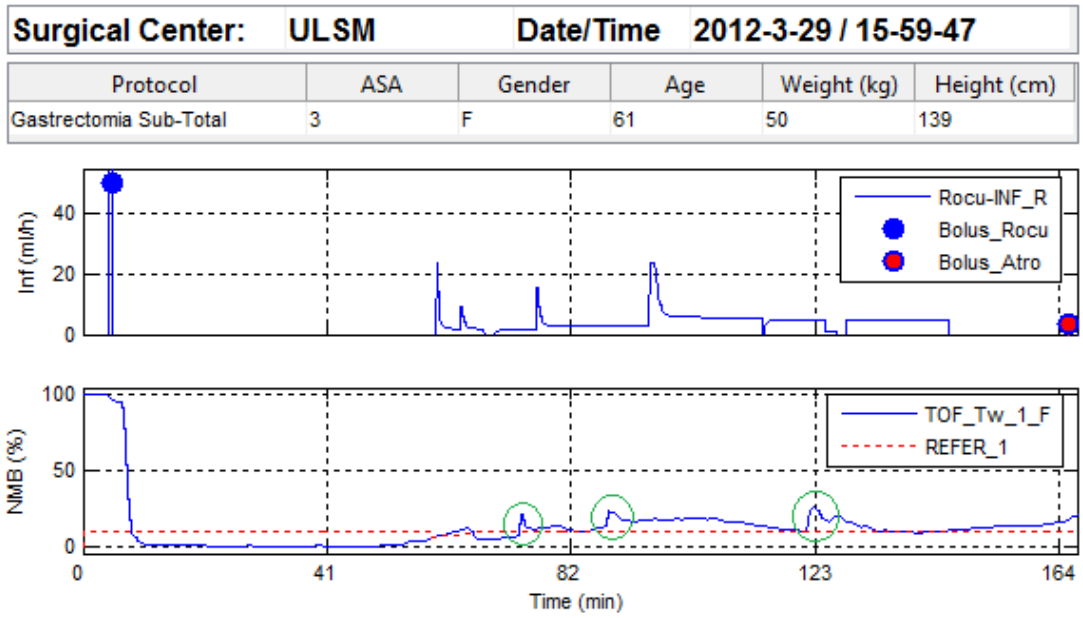

Figure 4.5: Results of the application of the switching strategy with a bank of total system mass control laws in a surgery at ULSM-HPH (2012/03/29 Case 2)

In Fig. 4.5 the case collected on March 29^{th} 2012 at ULSM-HPH is represented. At the top of this figure some information regarding the clinical case can be seen: *Protocol* (name of the surgery), *ASA*, *Gender*, *Age*, *Weight (kg)*, and *Height (cm)* of the patient.

Remark: The ASA information that appears in Figs. 4.5, 4.6, 4.7, 4.8, and 4.9 corresponds to the ASA classification system that represents the evaluation done by the anesthesiologist about the general health and well-being of the patient. Usually there are six classes:

1. A normal healthy patient;

2. A patient with mild systemic disease;
3. A patient with severe systemic disease;
4. A patient with severe systemic disease that is a constant threat to life;
5. A moribund patient who is not expected to survive without the operation;
6. A declared brain-dead patient whose organs are being removed for donor purposes.

Remark: In these real cases the switching signal was not accessible, so the only information about the occurrence of switching is the change in the *rocuronium* infusion rate after the recovery to the initial *bolus* happened (a sudden change in the drug infusion rate means a switching in the controller).

The upper plot of Fig. 4.5 contains all the information about the drugs administered to the patient that have effect on the NMB level. So, initially, around minute 10, the *rocuronium bolus* of 50 mg is represented, then, around minute 60, the recovery occurred and the controller started up with the infusion of *rocuronium*. At the end, the reverser, *neostigmine* together with *atropine*, was delivered to the patient (around the minute 170).

In the lower plot of Fig. 4.5 the NMB level of the patient during the surgery is plotted in blue and the NMB target in red. For this case, three artefacts occurred during the surgery, and they are marked by a green circle. These sudden changes have no reasonable explanation, since if they were sensor faults these artefacts should have been momentary, and after a while they should have come back to the previous values, but in this case after the sudden changes the NMB level stays in those news values. Despite this, the switching control shows to have a good behavior, once the NMB level was stabilized around the NMB target, although after the second artefact the stabilization took more time to happen. Therefore in this case the switching control shows reasonable results, once no abnormal values were registered. The anesthesiologist considered it a good performance as the NMB level remained within an acceptable interval of NMB values, in any moment reach high values. The surgeons did not complain about the muscle relaxation level, and the controller did not deliver an excess of muscle relaxant to the patient.

In Fig. 4.6 the first case collected on April 12^{th} 2012 at ULSM-HPH is represented. At the top of this figure once again the same information regarding the patient as in Fig. 4.5 can be seen.

The upper plot contains all the information about the drugs related with the NMB response that were administered to the patient. In this plot it can be seen that an initial *bolus* of 50

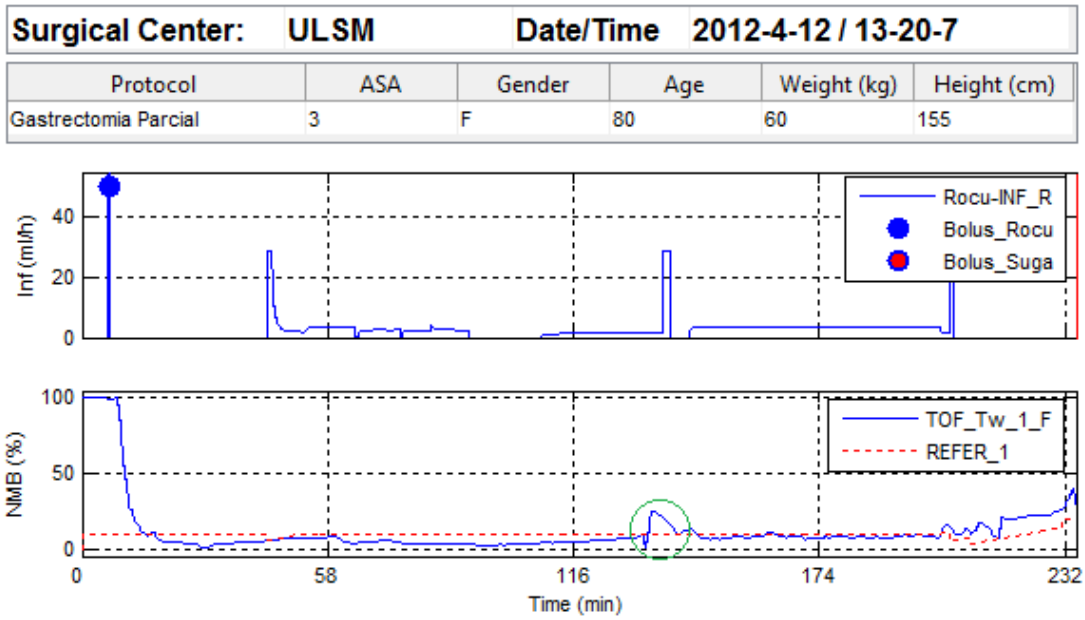

Figure 4.6: Results of the application of the switching strategy with a bank of total system mass control laws in a surgery at ULSM-HPH (2012/04/12 Case 1).

mg was given to the patient around minute 10, at about minute 45 the OLARD algorithm identified the recovery of the NMB level, and as a consequence the controller began with the *rocuronium* infusion. At the end of the surgery, around minute 235, the antagonist *sugamadex* was administered to the patient.

The lower plot shows the NMB level of the patient during the surgery (in blue), and the NMB target (in red). In a quick preview of the NMB signal, it can be seen that the NMB level remains closely to the NMB target. In a detailed analysis, since the beginning of the control procedure and until minute 60 the NMB response was stabilized around the target. After that the NMB level remained slightly under the target, which means that a larger dose than the required was administered to the patient, but over time the controller was able to follow again the target (around minute 120). Once again, around minute 140, an artefact appears in the signal, in the same way as in the case of Fig. 4.5, but once again the controller was able to adjust the NMB level to the target thereafter. From this point on, and until the end of the surgery, the NMB level was constantly kept within a small range of values around the target. The anesthesiologist considered this controller performance good because the NMB level was kept within a very reasonable interval, and once again no complains from the surgeons were registered about the muscle paralysis of the patient. The NMB level did not reach any abnormal value, and the control did not give an overall excessive dose of *rocuronium* to the patient. So the goal of liberating the anesthesiologist for more important tasks was achieved. As a final note, the changes in the *rocuronium* infusion rate during the control are a sign that switching among the controllers in the bank occurred.

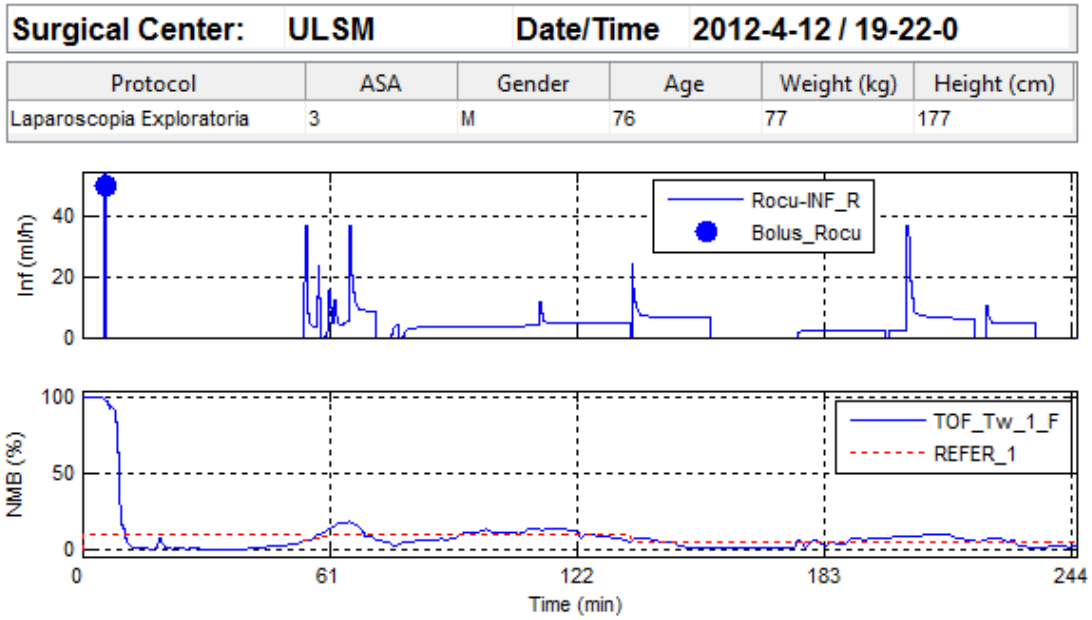

Figure 4.7: Results of the application of the switching strategy with a bank of total system mass control laws in a surgery at ULSM-HPH (2012/04/12 Case 2).

In Fig. 4.7 the second case collected on April 12^{th} 2012 at ULSM-HPH is represented. At the top of this figure once again the same patient information as in the previous figures is displayed.

The upper plot represents all the information about the NMB drugs administered to the patient. For this case a *rocuronium bolus* of 50 mg was administered around minute 10, and the recovery to this *bolus* occurred around the minute 50, and therefore the controller was initiated by that time. In this case there is no record of any reverser or antagonist having been administered to the patient at the end of the surgery.

The lower plot shows the NMB level of the patient during the surgery (in blue), as well as the NMB target desired (in red). For this case the NMB control remains in an acceptable interval from 0 to 20 % approximately. Around minute 60 the controller did some switching, specially due to the overshoot in the NMB level registered in the beginning of the control; in order to compensate this, more dose than the required was administered, but afterwards, until the change of the NMB target from 10% to 5% around minute 140, the NMB level of the patient followed the NMB target quite well. When the target was reduced, the NMB level decreased more than expected and remained close to 0% from minute 150 until minute 170. At the end of the procedure, the same happened again. In general the switching controller was not optimal, but for the anesthesiologist *Dr. Rui Rabiço* it was satisfactory, since he never had to be concerned about the patient NMB level during the surgery, no excessively high values were verified, and once again the surgeons did not complain about undesired patient movements. However this time the controller gave a little more dose than the required one. Once again, by the changes

occurred in the *rocuronium* infusion rate, the occurrence of switching can be inferred.

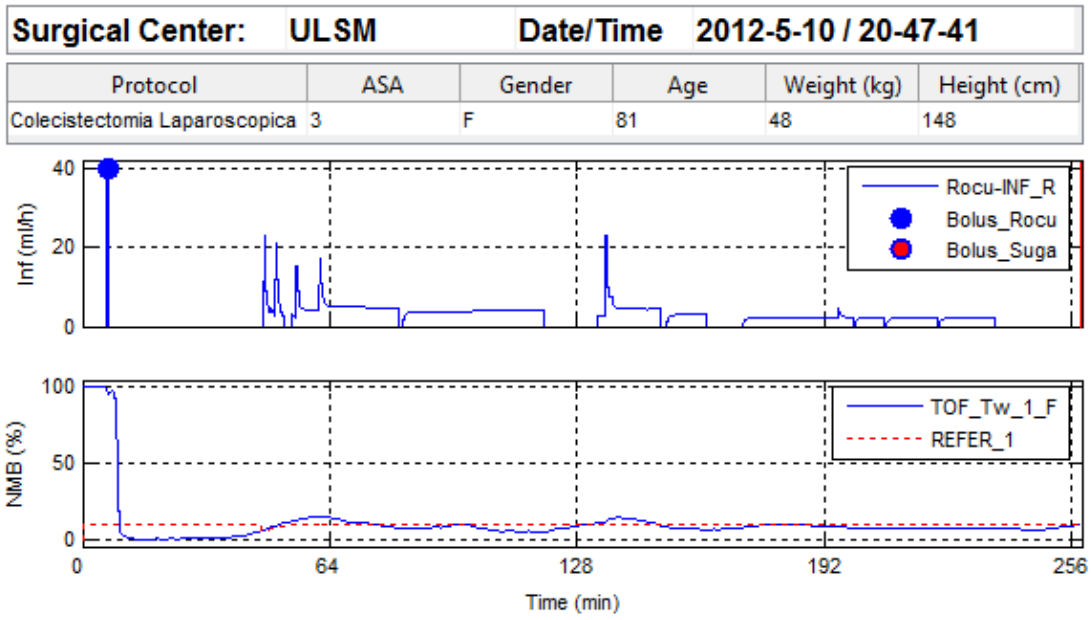

Figure 4.8: Results of the application of the switching strategy with a bank of total system mass control laws in a surgery at ULSM-HPH (2012/05/10 Case 2).

Fig. 4.8 contains the results acquired on May 10^{th} 2012 at ULSM-HPH. Once again, at the top of this figure some information about the patient and the surgery is shown.

The upper plot, which presents all the information regarding to the NMB drugs administered to the patient, shows that an initial *bolus* of 40 mg was given to the patient around minute 10, and the recovery occurred about minute 50. At this same moment the controller begins to work by applying the continuous infusion of *rocuronium* to the patient. In this case, at the end of the surgery, the antagonist, *sugamadex*, was administered to the patient (around minute 260).

The lower plot shows the NMB response of the patient during the surgery (in blue), as well as the NMB target applied to this case (in red). This case shows a good performance of the controller. In this surgery the NMB level was always kept between 6 and 14%, and no abnormal value was registered during the surgery. The infusion rate of *rocuronium* was neither excessive not scarce. The surgeons did not complain at any time about the muscle paralysis of the patient, the anesthesiologist did not have to worry about the muscle relaxation of the patient, and so this case can be considered very satisfactory. Once again, through the several changes in the infusion rate of *rocuronium* it can be inferred that the switching occurred.

Fig. 4.9 shows the case collected on May 31^{th} 2012 at ULSM-HPH. At the top of this figure once again the same patient information as in the previous figures is displayed.

The upper graphic of Fig. 4.9 displays all the drugs related to the NMB level that were

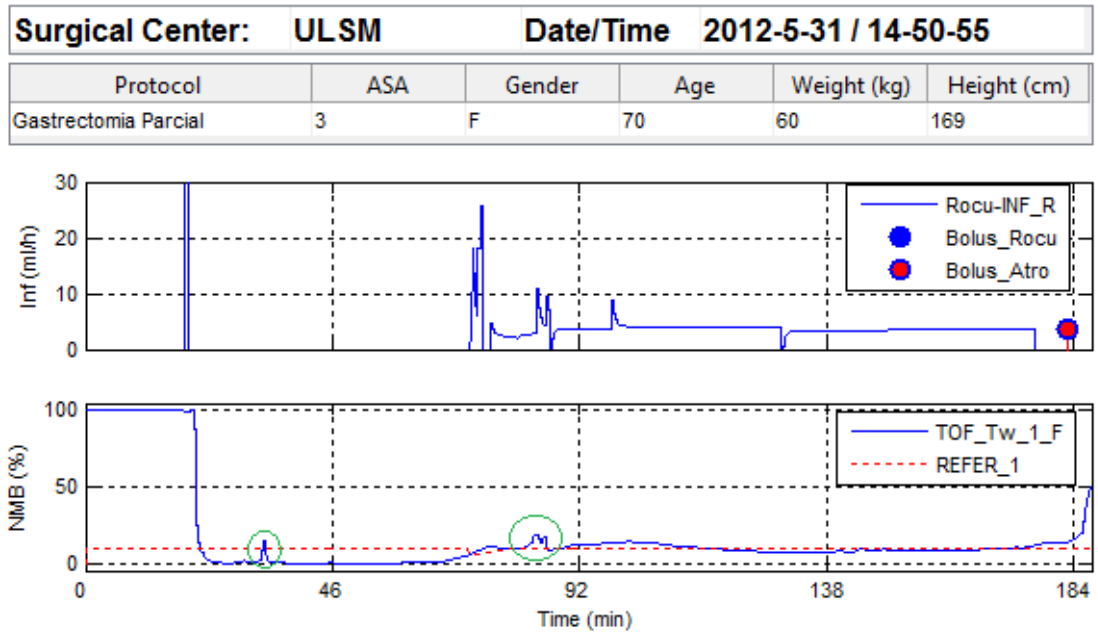

Figure 4.9: Results of the application of the switching strategy with a bank of total system mass control laws in a surgery at ULSM-HPH (2012/05/31 Case 1).

administered to the patient. For this case an initial *bolus* of 50 mg of *rocuronium* was given to the patient around minute 20, around minute 75 the recovery to the initial *bolus* was detected and so the NMB control started up with the infusion of *rocuronium*. At the end of the surgery, around minute 180, the reverser, *neostigmine* together with *atropine*, was administered for a full reversion of the NMB.

In the lower plot, the patient NMB response is represented (blue line) as well as the NMB target (red line). During the surgery two artefacts were registered due to the action of the anesthesiology, who needed to make new accesses for the intravenous drug administration (green circles): the first one occurred quite after the initial *bolus* administration, and the second one right after the recovery to the initial *bolus*. Although the second one appeared during the NMB control procedure, the controller was able to adjust to the new situation and to control the NMB level around the desired target. In general, the NMB level was kept around 6 and 14% (except when the second artefact happened), and the patient relaxation was never a problem for the anesthesiologist and for the surgeons. Also the anesthesiologist never needed to take any extra action to adjust the patient muscle relaxation. Overall it can be considered that the controller had a good performance. Once again the *rocuronium* infusion rate changes during the NMB control indicate that switching has occurred.

As a conclusion it can be said that the switching strategy shows to provide good results for the NMB level control during general anesthesia. The anesthesiologist needed not interfere to adjust the level of muscle relaxation of the patient, the surgeons did not complain about the patient

muscle relaxation, the NMB level was kept between good minimum and maximum values, and the controller was able to respond to artefacts and sudden changes in the patients NMB level.

Despite these results, some precautions have to be taken into account to ensure a good NMB level control with the switching strategy. The first one is related with the calibration of the sensor used to measure the NMB level of the patient. This step of calibration is essential to know the normal behavior of the patient without the presence of neuromuscular blockade agents (NMBA) and so to be able to quantify the NMB level during the administration of NMBA. If the calibration is wrong the values measured during the NMB control will also be wrong. Another aspect to take into account is the distance between the NMBA access and the intravenous access. This distance must be as short as possible in order to assure that the NMBA dose computed by the controller and administered by the infusion pump is delivery to the patient, and not in the access that connects the NMBA access to the patient intravenous system. Prolonged sensor faults and external interference in the patient NMB response must be avoided in order not to compromise the NMB switching criterion and the respective NMB control. Also, it was necessary to put valves at the end of the NMBA access in order to prevent the reflux of other drugs administered intravenously, which would modify the NMBA concentration. Finally, if the patient's arm could be placed open rather than along the body the NMB signal would be free of noise and the computation of the error between the patient's response and the responses of the models, necessary to implement the switching criterion, would be more accurate. This problem with the patient's arm positioning occurred with the signals acquired in the *Hospital Geral de Santo António* where the surgeries are made with the patient's arm along the body and the NMB signal has noise presence. The noise was constant even when the NMB sensor was changed, but not when the surgery took place with the patient's arm open.

After starting to take into account the precautions, an improvement of the NMB control performance was achieved during surgeries.

Chapter 5

Conclusions

The aim of this work was to study the performance of a switching strategy based on a bank of total system mass control laws to perform the NMB level control of patients during surgery.

Chapter 2 presented information related with medical aspects, like the definition of general anesthesia, neuromuscular blockade (NMB) anatomy and physiology, and neuromuscular blockade agents (NMBA) like *atracurium* and *rocuronium*.

In Chapter 3 all the theoretical work used in this thesis were presented. The compartmental models, for the *PK/PD* model and for the new *reduced parameter* model, describing the linear part that relates the drug input with the effect concentration, together with the non-linear part, also for both models, which allow to establish a relationship between the effect concentration and the obtained NMB level, were described. It is important to notice that the modelling for *atracurium* and *rocuronium* only differs in the patient-dependent parameters and in the value of the fixed parameter C_{50}. Then the NMB open-loop control law (TCI) was presented together with the closed-loop control law with real interest for this work, the total system mass control law. After the NMB control was explained, the switching strategy was described along with the criterion used to perform the switching within a bank of pre-designed controllers.

Hereafter the simulated results of the application of this strategy to control the NMB level are presented. Good control performances with a good reference tracking quality were achieved. The total system mass of the patient was able to follow the value M^* performed by the controller and this switching strategy allowed to compute a better M^* value over time (closer to the real M^* value required to drive the patient NMB level to the NMB target level) and so the patient NMB level was driven to the NMB target level as desired. Therefore it can be said that theoretically the main goal of having a good closed-loop control was achieved.

After that, a study to evaluate the switching strategy was made through a comparison with

a well known technique of parameter identification, the *Extended Kalman Filter* (EKF). So for the banks of available models (*atracurium* and *rocuronium*) a simulation of NMB control with total system mass control was performed using these two techniques. As a result the switching strategy shows to perform a better NMB control, specially when the recovery to the initial *bolus* is prolonged, this situation is very usual in quick NMB inductions where the initial *bolus* is higher than the normal. In these cases the EKF showed to perform a bad identification, which made the NMB control law lead the NMB level far from the desired target. The switching strategy presents the advantage not to be dependent from the recovery time to the initial *bolus*. Also the switching strategy shows to spend less total input dose, during the NMB control, than the control law corresponding to the model parameters identified with the EKF. Despite that the settling time with the switching strategy is much longer, but such situation is due to the way the strategy is conceived (switching the controller over time). The differences between these two techniques are more evident with the *atracurium* models; with the *rocuronium* models they are lower but still exist.

Afterwards a study in order to analyze if the switching strategy provides a good controller over time was made. Several total system mass controllers selected offline with different metrics, among which the last controller used during the switching strategy, were applied. The last controller of the switching strategy along with the best controller from the controller bank chosen offline, the controller with the most similar gamma to the one of the patient model, and the controller with the most similar U_{SS} to the one of the patient model show better results regarding the total input amount required for the NMB control with *atracurium* models because they spend less dose, but not with *rocuronium* models where they spend more dose. Despite that, these controllers show better reference tracking quality for both NMBA models and so, even if with *rocuronium* models these controllers spend more dose, the reference tracking quality justifies their qualification as better controllers. As for the settling time, the results obtained are similar for all the controllers studied. Summarizing, the last controller of the switching procedure shows to provide a good total system mass control leading to consequent good NMB control results both for *atracurium* and *rocuronium*, proving that the switching provides in the course of time a good controller for the NMB level control.

Finally in Chapter 3 a technique to perform the reference tracking improvement was presented. For this purpose a steady-state dose, U_{SS}, along with the corresponding achieved steady-state NMB level are used in order to identify a better approximation for the parameter γ of the total system control law and thus perform a recalibration of the controller. Therefore the three worst results obtained with the switching strategy with *atracurium* models and the three worst cases with *rocuronium* models were considered and the respective recalibration was applied to those cases. It was possible to see that this recalibration strategy provided a better reference

tracking quality. Although the off-set between the NMB level achieved and the NMB target desired was not completely eliminated, it has been substantially reduced. For practical purposes this technique is not optimal because it is hard to obtain steady-state doses and NMB levels and so a good calibration of the parameter γ is hard to achieve.

In Chapter 4 the practical work related with this thesis was presented. Here all the information related with hardware, software and protocol used during the real NMB control cases was initially explained, and then the results obtained were illustrated graphically. Five real cases were presented with good NMB control using the switching strategy. After analyzing those cases it can be concluded that this strategy presents good results when applied in real surgeries. The NMB level was kept inside an acceptable range (between 5 and 15% more or less), the patient muscle stiffness was never a point of discussion for the surgeons, the anesthesiologist never have to perform any manual control on the patient muscle relaxation, and even when sudden changes in the NMB level occurred the controller was able to respond and again drive the NMB level closer to the desired target. Also when some momentary sensor faults were verified the controller continues to work correctly. Another aspect of this control strategy is its ability to adapt to changes in the patient dynamics in the course of time. Despite these encouraging results some care is needed in order to avoid a bad controller performance. For example, a good sensor calibration is essential, it is desirable to place the patient arm opened instead of along the body (to avoid noise), to use a short access as possible to connect the NMBA access to the patient intravenous access, and to use reflux valves in order to avoid changes in the NMBA concentration.

In summary, this work shows the performance of a NMB control strategy based on switching among controllers with a total system mass control law. More important than the exploitation of this technique are the good results obtained both in theory and in practice. The total system mass control shows to work properly with switching and this strategy is able to select a good controller for the patient. Moreover this strategy provides good results even in the presence of momentary sensor faults and sudden changes, and is also able to respond to the variations in the patient dynamics over time, which was one of the major problems with the previously used controllers. Therefore this control scheme is able to overcome not only the intervariability among patients but also the intravariability.

For the future, the construction of a population model for the action of *rocuronium* is suggested, in a similar way to what has been done for *atracurium* models. Thus will allow to generate any desired bank $\mathcal{P}$ of *rocuronium* models, and as a consequence any desired controller bank $\mathcal{K}$. This study will make the switching strategy for the NMB control with *rocuronium* more robust,

once the bank of controllers will be more complete, and hence more able to fit the wide inter- and intra-patient system variability.

Another aspect is related to pursuing the application of the switching strategy to real cases in the operating room, in order to increase the number of acquired data and so to be able to perform a statistic study to evaluate the switching performance in real cases.

Finally a further study of the filter used to remove artefacts from the NMB signal needs to be made. Actually a combined *FIR+Median+Non Linear filter* is used in order to perform such action, but still some undesired noise or sensor faults are not corrected. The obstacle to the use of a higher order filter is the increasing of the delay that the filter will incorporate in the NMB level, which is an undesired situation as it will affect the performance of the NMB control.

Bibliography

H. Magalhães, "Contributions to neuromuscular blockade control," PhD Thesis, Faculdade de Ciências da Universidade do Porto, Mar 2006.

J. Almeida, "Estratégia de controlo positivo para sistemas compartimentais do bloqueio neuromuscular," MSc Thesis, Faculdade de Ciências da Universidade do Porto, 2010.

J. W. D. Vries, H. H. Ros, and L. H. Booij, "Infusion of vecuronium controlled by a closed-loop system," *British Journal of Anaesthesia*, vol. 58, no. 10, pp. 1100–1103, Oct 1986.

C. M. Wait, V. A. Goat, and C. E. Blogg, "Feedback control of neuromuscular blockade: A simple system for infusion of atracurium," *Anesthesia*, vol. 42, no. 11, pp. 1212–1217, Nov 1987.

P. Lago, T. Mendonca, and L. Goncalves, "Online autocalibration of a pid controller of neuromuscular blockade," *Proceedings of the 1998 IEEE International Conference on Control Applications*, vol. 1, pp. 363–367, Sep 1998.

N. M. Cass, D. G. Lampard, W. A. Brown, and J. R. Coles, "Computer controlled muscle relaxation: a comparison of four muscle relaxants in the sheep," *Anaesthesia and Intensive Care*, vol. 4, no. 1, pp. 16–22, Feb 1976.

A. J. Asbury and D. A. Linkens, "Clinical automatic control of neuromuscular blockade," *Anaesthesia*, vol. 41, no. 3, pp. 316–320, Mar 1986.

D. A. O'Hara, G. J. Derbyshire, F. J. Overdyk, D. K. Bogen, and B. Marshall, "Closed-loop infusion of atracurium with four different anesthetic techniques," *Anesthesiology*, vol. 74, no. 2, pp. 258–263, Feb 1991.

H. S. Bradlow, P. C. Uys, and L. B. Rametti, "On-line control of atracurium induced muscle relaxation," *Journal of Biomedical Engineering*, vol. 8, no. 1, pp. 72–77, Jan 1986.

R. R. Jaklitsch and D. R. Westenskow, "A model-based self-adjusting two-phase controller for vecuronium-induced muscle relaxation during anesthesia," *IEEE Transactions on Biomedical Engineering*, vol. 34, no. 8, pp. 583–594, Aug 1987.

P. C. Uys, D. F. Morrell, H. S. Bradlow, and L. B. Rametti, "Self-tuning, microprocessor-based closed-loop control of atracurium-induced neuromuscular blockade," *British Journal of Anaesthesia*, vol. 6, no. 1, pp. 685–692, Dec 1988.

M. Lendl, U. Schwarz, H. Romeiser, R. Unbehauen, M. Georgieff, and G. Geldner, "Nonlinear model-based predictive control of non-depolarazing muscle relaxants using neural networks," *Journal of Clinical Monitoring and Computing*, vol. 15, no. 5, pp. 271–278, 1999.

D. G. Mason, D. A. Linkens, M. F. Abbod, N. D. Edwards, and C. S. Reilly, "Automated delivery of muscle relaxants using fuzzy logic control," *IEEE Engineering in Medicine and Biology Magazine*, vol. 13, no. 5, pp. 678–685, 1994.

D. G. Mason, N. D. Edwards, D. A. Linkens, and C. S. Reilly, "Performance assessment of a fuzzy controller for atracurium-induced neuromuscular block," *British Journal of Anaesthesia*, vol. 76, no. 3, pp. 396–400, Mar 1996.

D. G. Mason, J. J. Ross, N. D. Edwards, D. A. Linkens, and C. S. Reilly, "Self-learning fuzzy control with temporal knowledge for atracurium-induced neuromuscular block during surgery," *Computers and Biomedical Research*, vol. 32, no. 3, pp. 187–197, Jun 1999.

J. J. Ross, D. G. Mason, D. A. Linkens, and N. D. Edwards, "Self-learning fuzzy logic control of neuromuscular block," *British Journal of Anaesthesia*, vol. 78, no. 4, pp. 412–415, 1997.

N. D. Edwards, D. G. Mason, and J. J. Ross, "A portable self-learning fuzzy logic control system for muscle relaxation," *Anaesthesia*, vol. 53, no. 2, pp. 136–139, Feb 1998.

J. S. Shieh, S. Z. Fan, L. W. Chang, and C. C. Liu, "Hierarchical rule-based monitoring and fuzzy logic control for neuromuscular block," *Journal of Clinical Monitoring and Computing*, vol. 16, no. 8, pp. 583–592, 2000.

J. M. Lemos, R. N. Silva, and J. S. Marques, "Adaptive control in the presence of sensor measure outliers," in *Proceedings of the 10th Mediterranean Conference on Control and Automation (MED'02)*, Lisbon, Portugal, Jul 2002.

J. M. Lemos, R. Silva, and J. S. Marques, "Adaptive control of the ball and beam plant in the presence of sensor measure outliers," in *Proceedings of the 2002 American Control Conference (ACC'02)*, vol. 6, Anchorage, Alaska, U.S.A., 2002, pp. 4612–4613.

J. M. Lemos, H. Magalhães, and R. Dionísio, "Control of physiological variables in the presence of interrupted feedback measurements," in *Proceedings of the 16th Internation Conference on Systems Engineering*, Coventry, England, 2003, pp. 433–438.

J. M. Lemos, H. Magalhães, T. Mendonça, P. Rocha, S. Esteves, and J. Gaivão, "Observer dynamics in switching control: application to neuromuscular blockade," in *Proceedings of the WISP2005 IEEE International Symposium on Intelligent Signal Processing*, Faro, Portugal, 2005, pp. 211–216.

J. M. Lemos, H. Magalhães, T. Mendonça, and R. Dionísio, "Control of neuromuscular blockade in the presence of sensor faults," *IEEE Transactions on Biomedical Engineering*, vol. 52, no. 11, pp. 1902–1911, Nov 2005.

J. M. Lemos, H. Magalhães, T. Mendonça, and P. Rocha, "Observer design in switching control of neuromuscular blockade: clinical cases," in *28th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBS '06)*, New York City, New York, U.S.A., 2006, pp. 5436–5439.

H. Magalhães, T. Mendonça, and P. Lago, "A matlab smart adaptive system for the control of neuromuscular blockade," in *Proceedings of the 2nd European Medical and Biological Engineering Conference*, Wien, Austria, 2002, pp. 664–665.

H. Magalhães, T. Mendonça, P. Lago, and P. Rocha, "A matlab tool for switching control," in *Proceedings of the 5th Portuguese Conference on Automatic Control*, Aveiro, Portugal, 2002, pp. 284–288.

H. Magalhães, J. M. Lemos, and T. Mendonça, "Hidden markov models based supervision of sensor faults: application to neuromuscular blockade control," in *Proceedings of the 6th Portuguese Conference on Automatic Control*, Faro, Portugal, 2004, pp. 343–348.

H. Magalhães, T. Mendonça, and P. Rocha, "Identification and control of positive and compartmental systems applied to neuromuscular blockade," in *Proceedings of the 16th IFAC World Congress*, Prague, Czech Republic, 2005.

C. Sousa, T. Mendonça, and P. Rocha, "Control of uncertain compartmental systems," in *Proceeding of the 15th Mediterranean Conference on Control and Automation*, Athens, Greece, Jun 2007.

C. Sousa, T. F. Mendonça, and P. Rocha, "Total mass control in uncertain compartmental systems," in *8th Portuguese Conference on Automatic Control*, Vila Real, Portugal, Jul 2008.

C. Sousa, T. Mendonça, and P. Rocha, "Target mass control for uncertain compartmental systems," *International Journal of Control*, vol. 83, no. 7, pp. 1387–1396, Jun 2010.

J. Almeida, M. M. da Silva, T. Wigren, and T. Mendonça, "Contributions to the initialization of online identification algorithms for anaesthesia: the neuromuscular blockade case study,"

in *18th Mediterranean Conference on Control and Automation (MED)*, Marrakech, Morocco, Jun 2010, p. 1341.1346.

J. Almeida, T. Mendonça, and P. Rocha, "An improved strategy for neuromuscular blockade control with parameter uncertainty," in *50th IEEE Conference on Decision and Control and European Control Conference (CDC-ECC)*, Orlando, Florida, U.S.A., Dec 2011, pp. 867–872.

M. Teixeira, T. Mendonça, and P. Rocha, "Tracking the nmb level via a switching system mass control strategy," in *8th IFAC Symposium on Biological and Medical Systems*, Budapest, Hungary, Aug 2012, accepted.

S. Esteves, "Monitorização do bloqueio neuromuscular," *Cadernos Temáticos (2º Caderno): Centro Hospitalar do Porto - Hospital Santo António*, 2008.

M. M. da Silva, J. Almeida, T. Wigren, and T. Mendonça, "Merging pk/pd information in a minimally parameterized model of the neuromuscular blockade," in *32nd Annual International Conference of the IEEE EMBS*, Buenos Aires, Argentina, 2010.

M. M. da Silva, T. Wigren, and T. Mendonça, "Nonlinear identification of a minimal neuromuscular blockade model in anesthesia," *IEEE Transactions on Control Systems Technology*, vol. 20, no. 1, pp. 181–188, Jan 2012.

A. Guyton and J. Hall, *Textbook of medical physiology.* W.B. Saunders Company, 2000.

J. Appiah-Ankam and J. M. Hunter, "Pharmacology of neuromuscular blocking drugs," *Continuing Education in Anaesthesia, Critical Care and Pain*, 2004.

E. Marques, "Sistemas compartimentais," MSc Thesis, Universidade de Aveiro, 2008.

K. Godfrey, *Compartmental Models and their Application.* London: Academic Press Inc, 1983.

J. Hof, "System theory and system identification of compartmental systems," PhD Thesis, University of Groningen, 1996.

C. Beck, H. Lin, and M. Bloom, "Multivariable modeling and control of anesthetic pharmacodynamics," *Lecture Notes in Control and Information Sciences*, vol. 357, pp. 263–289, 2007.

M. M. da Silva, "Prediction error identification of minimally parametrized wiener models in anesthesia," in *18th IFAC World Congress*, Milano, Italy, 2011.

N. Bressan, A. Moreira, P. Amorim, and C. S. Nunes, "Anaesthesia synchronization software: target controlled infusion system evaluation," in *32nd Annual International Conference of the IEEE EMBS*, Buenos Aires, Argentina, 2010.

G. Bastin and A. Provost, "Feedback stabilisation with positive control of dissipative compartmental systems," in *15th International Symposium on Mathematical Theory of Networks and Systems*, Notre Dame, Indiana, USA, 2002.

S. Morse, "Supervisory control of families of linear set-point controllers part 1: exact matching," *IEEE Transactions on Automatic Control*, vol. 41, no. 11, pp. 1413–1431, 1996.

A. S. Morse, "Supervisory control of families of linear set-point controllers part 2: robustness," *IEEE Transactions on Automatic Control*, vol. 42, no. 11, pp. 1500–1515, 1997.

K. Narendra and J. Balakrishnan, "Improving transient response of adaptive control systems using multiple models and switching," *IEEE Transactions on Automatic Control*, vol. 39, no. 9, pp. 1861–1866, 1994.

K. S. Narendra and J. Balakrishnan, "Adaptive control using multiple models," *IEEE Transactions on Automatic Control*, vol. 42, no. 2, pp. 171–187, 1997.

A. Neves, T. Mendonça, and P. Rocha, "Tracking by switching control: a case study," in *Proceedings of the 4th Portuguese Conference on Automatic Control*, Guimarães, Portugal, 2000, pp. 61–65.

A. Neves, "Controlo comutado para a infusão de fármacos," PhD Thesis, Universidade de Aveiro, 2003.

C. Rocha, T. Mendonça, and M. E. Silva, "A simulation model for neuromuscular blockade induced by atracurium," *Internal Report of CIDMA*, 2011, submitted to Mathematical and Computer Modelling of Dynamical Systems.

M. M. da Silva, C. Sousa, R. Sebastião, J. Gama, T. Mendonça, P. Rocha, and S. Esteves, "Total mass tci driven by parametric estimation," in *17th Mediterranean Conference on Control and Automation*, Makedonia Palace, Thessaloniki, Greece, Jun 2009.

R. E. Kalman, "A new approach to linear filtering and prediction problems," *Transactions of the ASME Journal of Basic Engineering*, vol. 82, pp. 35–45, 1960.

H. Alonso, T. Mendonça, and P. Rocha, "A hybrid method for parameter estimation and its application to biomedical systems," *Computer Methods and Programs in Biomedicine*, vol. 89, no. 2, pp. 112–122, Feb 2008.

Appendix A

MATLAB Routines

A.1 Simulation of the response to an initial bolus

```
function Offline_Bolus_Inicial_RED
clear all
clc
%% Load da base de dados com os parâmetros Alpha e Gamma
% load([cd '\Base de Dados\Alpha_Gamma_Atr_100.mat'])
load([cd '\Base de Dados\Alpha_Gamma_Roc_41.mat'])
%% Determinar o numero de modelos na base de dados
num_modelos = length(NEW_THETA_100(1,:));
indice_modelo = [1:1:num_modelos]';
%% Identificação das variáveis iniciais
y_modelos = [];
NMB_modelos = [];
for l=1:num_modelos
   NMB_modelos(1,l) = 100; % Definir o primeiro valor de todos os NMBs
end
h = 20; % Tempo de Discretização em segundos
h = h/60; % Tempo de Discretização em minutos
tempo_simulacao = 300; % Tempo de simulação em minutos
tempo_simulacao = tempo_simulacao/h; % Em instantes de amostragem
% bolus_inicial = 500/h; % Bolus inicial m ug/kg para o Atracurium
bolus_inicial = 600/h; % Bolus inicial em ug/kg para o Rocuronium
% c50 = 0.6487*5; % C50 para o Atracurium
c50 = 1; % C50 para o Rocuronium
```

```
%% Obtenção dos modelos compartimentais.
[Ad_REDM,Bd_REDM,x_modelos,gamma_modelos] = modelo_reduzido_modelos
(NEW_THETA_100,h,num_modelos,indice_modelo);
%% Simulação
for i=1:tempo_simulacao
    for j=1:num_modelos
        x_modelos2(:,1) = x_modelos{i,j};
        Ad_modelos = Ad_REDM{j};
        Bd_modelos = Bd_REDM{j};
        if j==1
            if i==1
                u_RED(i) = bolus_inicial;
            else
                u_RED(i) = 0;
            end
        end
        x_modelos2(:,2) = Ad_modelos*x_modelos2(:,1)+Bd_modelos*u_RED(i);
        x_modelos{i+1,j} = x_modelos2(:,2);
        y_modelos(i+1,j) = [0 0 1]*x_modelos2(:,2);
        NMB_modelos(i+1,j) = [100/(1+((y_modelos(i+1,j)/c50)^
        gamma_modelos(j)))];
    end
end
%% Plot dos Resultados
% Definir a escala de tempo em minutos
t=[0:1:tempo_simulacao]; % Criar escala para o eixo do tempo
t=t*h; % Converter o eixo do tempo em minutos
% Plot das Concentrações de Efeito
subplot(2,1,1),plot(t,y_modelos)
title('Concentração de Efeito')
xlabel('minutos')
ylabel('ug/kg')
% Plot dos Relaxamentos Musculares
subplot(2,1,2),plot(t,NMB_modelos)
title('Relaxamento Muscular')
xlabel('minutos')
```

```
58 ylabel('% NMB')
59 end
60
61 function [Ad_REDM,Bd_REDM,x_modelos,gamma_modelos] = modelo_reduzido_modelos
   (NEW_THETA_100,h,num_modelos,indice_modelo)
62 k1=1; k2=4; k3=10;
63 x_modelos = [];
64 x_modelos2 = [];
65 for k=1:num_modelos
66     % Obtenção dos parâmetros para os modelos simulados
67     alpha_modelos(k) = NEW_THETA_100(1,indice_modelo(k,1));
68     gamma_modelos(k) = NEW_THETA_100(2,indice_modelo(k,1));
69     % Construção das matrizes do modelo compartimental
70     Ac_modelos = [(-k3*alpha_modelos(k)) 0 0;(k2*alpha_modelos(k)) -
       (k2*alpha_modelos(k)) 0;0 (k1*alpha_modelos(k)) -(k1*alpha_modelos(k))];
71     Bc_modelos = [k3*alpha_modelos(k) 0 0]';
72     % Discretização do sistemas compartimentais
73     [Ad_modelos,Bd_modelos] = c2d(Ac_modelos,Bc_modelos,h);
74     % Guardar as matrizes de todos os modelos do banco de dados
75     Ad_REDM{k} = Ad_modelos;
76     Bd_REDM{k} = Bd_modelos;
77     % Definir os estados iniciais de cada modelo
78     x_RED_modelos = Bc_modelos;
79     x_modelos{1,k} = x_RED_modelos;
80 end
81 end
```

A.2 Switching Strategy

```
1 function Control_Law_Switching_RED_Olard
2 clear all
3 clc
4 %% Load da base de dados com os parametros Alpha e Gamma
5 load([cd '\Base de Dados\Alpha_Gamma_Atr_100.mat']) % Para o Atracurium
6 % load([cd '\Base de Dados\Alpha_Gamma_Roc_37.mat']) % Para o Rocuronium
```

```
%% Determinar tamanho da base de dados
length1 = length(NEW_THETA_100(1,:));
%% Definir algumas variáveis
h = 20; % Tempo de discretização em segundos.
h = h/60; % Tempo de discretização em minutos.
bolus_inicial = 500/h; % Em ug/kg, e no tempo discreto, para o Atracurium
% bolus_inicial = 600/h; % Em ug/kg, e no tempo discreto, para o Rocuronium
nmb_ref = 10; % Referência desejada em %
lambda = 0.5; % Parametro intrinseco ao controlador
c50 = 0.6487*5; % Valor de C50 para o Atracurium
% c50 = 1; % Valor de C50 para o Rocuronium
%% Definir tempo de simulação e converter para instantes de 20s
tempo_simulacao = 300; % Em minutos
tempo_simulacao = tempo_simulacao/h; % Em instantes de 20s
%% Limpar algumas variáveis para evitar erros de cálculos
NMB_paciente = 100;
erro = [];
ChangePoint = [];
M_Estrela = [];
flag1 = 0;
%% Definir o modelo que vai ser usado para simular o paciente
paciente = 1; % Definir qual o modelo do banco de dados usado como paciente
parametros_paciente(:,1) = NEW_THETA_100(:,paciente);
%% Definir o banco de dados para identificação e controlo do paciente
for s=1:length1
    if s<paciente
        parametros_modelos(:,s) = NEW_THETA_100(:,s);
    else if s==paciente
        else if s>paciente
                parametros_modelos(:,s-1) = NEW_THETA_100(:,s);
            end
        end
    end
end
%% Identificação do numero de modelos presentes na base de dados
num_modelos = length(parametros_modelos(1,:));
```

```
43 indice_modelo = [1:1:num_modelos]';
44 %% Obter as variaveis necessárias à simulação
45 [Ad_paciente,Bd_paciente,x_paciente,gamma_paciente,M_E_Paciente,U_SS_Real] =
   modelo_reduzido_paciente(parametros_paciente,h,c50,nmb_ref);
46 [Ad_REDM,Bd_REDM,x_modelos,gamma_modelos,Ce_estrela] = modelo_reduzido_
   modelos(parametros_modelos,h,num_modelos,indice_modelo,nmb_ref,c50);
47 %% Simulação
48 for i=1:tempo_simulacao
49     for j=1:num_modelos
50         Ad_modelos = Ad_REDM{j};
51         Bd_modelos = Bd_REDM{j};
52         if isempty(ChangePoint)
53             if i==1
54                 u_RED(i) = bolus_inicial;
55             else
56                 u_RED(i) = 0;
57             end
58             if j==1
59                 x_paciente = Ad_paciente*x_paciente+Bd_paciente*u_RED(i);
60                 y_paciente(i+1) = [0 0 1]*x_paciente;
61                 NMB_paciente(i+1) = [100/(1+((y_paciente(i+1)/c50)
                   ^gamma_paciente))];
62             end
63             x_modelos{1,j} = Ad_modelos*x_modelos{1,j}+Bd_modelos*u_RED(i);
64             y_modelos(i+1,j) = [0 0 1]*x_modelos{1,j};
65             NMB_modelos(i+1,j) = [100/(1+((y_modelos(i+1,j)/c50)
               ^gamma_modelos(j)))];
66         else
67             if j==1
68                 Ad_control = Ad_REDM{Controlador};
69                 Bd_control = Bd_REDM{Controlador};
70                 x_control = x_modelos{1,Controlador};
71                 M_Estrela2(i) = M_E_Paciente;
72                 M_paciente(i) = sum(x_paciente);
73                 if flag1==0
74                     M_Estrela(i) = 3*Ce_estrela(Controlador);
```

```
75                  else
76                      Ce_calibrado = [(((100*(c50^Gamma_Calibrado))-((c50
                       ^Gamma_Calibrado)*nmb_ref))/nmb_ref)^(1/Gamma_Calibrado)];
77                      M_Estrela(i) = 3*Ce_calibrado;
78                  end
79                  u_RED(i) = min(80,max(0,(((sum(Bd_control))^(-1))*[1 1 1]*
                    (lambda*eye(3)-Ad_control)*x_control+(((sum(Bd_control))^(-1))
                    *(1-lambda)*M_Estrela(i)))));
80                  x_paciente = Ad_paciente*x_paciente+Bd_paciente*u_RED(i);
81                  y_paciente(i+1) = [0 0 1]*x_paciente;
82                  NMB_paciente(i+1) = [100/(1+((y_paciente(i+1)/c50)^
                     gamma_paciente))];
83              end
84
85              x_modelos{1,j} = Ad_modelos*x_modelos{1,j}+Bd_modelos*u_RED(i);
86              y_modelos(i+1,j) = [0 0 1]*x_modelos{1,j};
87              NMB_modelos(i+1,j) = [100/(1+((y_modelos(i+1,j)/c50)
                ^gamma_modelos(j)))];
88
89              erro(i+1,j) = ((NMB_paciente(i+1)-NMB_modelos(i+1,j)).^2);
90              erro2(i+1,j) = ((y_paciente(i+1)-y_modelos(i+1,j)).^2);
91          end
92      end
93      %% Função que identifica automaticamente o inicio da recuperação
94      [ChangePoint] = OLARD_bloco(NMB_paciente);
95      %% Escolher o primeiro controldador aleatoriamente
96      if i==(ChangePoint-1)
97          Controlador = randi(length(NMB_modelos(1,:)),1)
98      end
99      %% Escolher os restantes controladores
100     if i>=ChangePoint & flag1==0
101         erro_acumulado(i+1,:) = sum(erro);
102         min_1 = min(erro_acumulado(i+1,:));
103         Controlador = find(erro_acumulado(i+1,:)==min_1);
104         tabela_Controladores(i+1) = Controlador;
105     elseif i>=ChangePoint & flag1==1
```

```
106         erro_acumulado(i+1,:) = sum(erro2);
107         min_1 = min(erro_acumulado(i+1,:));
108         Controlador = find(erro_acumulado(i+1,:)==min_1);
109         tabela_Controladores(i+1) = Controlador;
110     end
111     %% Calibração U Steady State
112     if i==270
113         Gamma_Calibrado = log(100/NMB_paciente(i+1))/log(u_RED(i)/c50);
114         flag1 = 1;
115     end
116 end
117 %% Limar algumas variáveis para o plot
118 NMB_paciente(end) = [];
119 tabela_Controladores(end) = [];
120 erro(end,:) = [];
121 erro_acumulado(end,:) = [];
122 u_RED(1) = 0;
123 %% Construção de algumas variáveis de interesse
124 Lei_Controlo = tabela_Controladores(end)
125 Switching = tabela_Controladores;
126 Inicio_Recuperacao = ChangePoint*20/60
127 Delta_NMB_SS = abs(NMB_paciente(end)-nmb_ref)
128 Dose_SS = u_RED(end)
129 end
130
131 function [Ad_paciente,Bd_paciente,x_paciente,gamma_paciente,M_E_Paciente,
    U_SS_Real] = modelo_reduzido_paciente(parametros_paciente,h,c50,nmb_ref)
132 % Obtenção dos parâmetros para o modelo do paciente
133 k1=1; k2=4; k3=10;
134 alpha_paciente = parametros_paciente(1,1);
135 gamma_paciente = parametros_paciente(2,1);
136 % Construção das matrizes do modelo compartimental
137 Ac_paciente = [(-k3*alpha_paciente) 0 0;(k2*alpha_paciente) -(k2*alpha_paciente)
    0;0 (k1*alpha_paciente) -(k1*alpha_paciente)];
138 Bc_paciente = [k3*alpha_paciente 0 0]';
139 % Discretização do sistema compartimental
```

```
140 [Ad_paciente,Bd_paciente] = c2d(Ac_paciente,Bc_paciente,h);
141 % Definir estado inicial do paciente
142 x_paciente = [];
143 x_paciente = Bc_paciente; % Estado Inicial discreto é igual à matriz B contínua
144 % Obter M* do modelo do paciente
145 M_E_Paciente = 3*((((100*(c50^gamma_paciente))-((c50^gamma_paciente)*nmb_ref))
    /nmb_ref)^(1/gamma_paciente));
146 % Obter U Steady State do Paciente
147 Var1 = (alpha_paciente*k1*k2*k3);
148 Var2 = (Var1/((k2-k1)*(k3-k1)))/(alpha_paciente*k1);
149 Var3 = (Var1/((k1-k2)*(k3-k2)))/(alpha_paciente*k2);
150 Var4 = (Var1/((k1-k3)*(k2-k3)))/(alpha_paciente*k3);
151 Var5 = Var2+Var3+Var4;
152 U_SS_Real = c50*((((100/nmb_ref)-1)^(1/gamma_paciente))/Var5);
153 end
154
155 function [Ad_REDM,Bd_REDM,x_modelos,gamma_modelos,Ce_estrela] =
    modelo_reduzido_modelos(parametros_modelos,h,num_modelos,indice_modelo,
    nmb_ref,c50)
156 k1=1; k2=4; k3=10;
157 x_modelos  = [];
158 for k=1:num_modelos
159     % Obtenção dos parâmetros para os restantes modelos
160     alpha_modelos(k) = parametros_modelos(1,indice_modelo(k,1));
161     gamma_modelos(k) = parametros_modelos(2,indice_modelo(k,1));
162     % Construção das matrizes dos modelos compartimentais
163     Ac_modelos = [(-k3*alpha_modelos(k)) 0 0;(k2*alpha_modelos(k)) -
        (k2*alpha_modelos(k)) 0;0 (k1*alpha_modelos(k)) -(k1*alpha_modelos(k))];
164     Bc_modelos = [k3*alpha_modelos(k) 0 0]';
165     % Discretização dos sistemas compartimentais
166     [Ad_modelos,Bd_modelos] = c2d(Ac_modelos,Bc_modelos,h);
167     % Guardar as matrizes de todos os modelos do banco de dados
168     Ad_REDM{k} = Ad_modelos;
169     Bd_REDM{k} = Bd_modelos;
170     % Definir estado inicial dos modelos
171     x_RED_modelos  = Bc_modelos;
```

```
172     x_modelos{1,k} = x_RED_modelos;
173     % Obter Ce* dos modelos
174     Ce_estrela(k) = [(((100*(c50^gamma_modelos(k)))-((c50^gamma_modelos(k))
        *nmb_ref))/nmb_ref)^(1/gamma_modelos(k))];
175 end
176 end
177
178 function [ChangePoint] = OLARD_bloco(NMB_paciente)
179 ChangePoint=[];
180 index = 0; holdindex = 0; Subindex = 0;
181 if length(NMB_paciente)>54
182     baseline = mean(NMB_paciente(49-4:49+5));
183     for jj=54:length(NMB_paciente)
184         if (NMB_paciente(jj) > NMB_paciente(jj-1)) && (NMB_paciente(jj)
            > baseline+1)
185             index = index + 1; Subindex = Subindex + 1; holdindex =0;
                P(Subindex) = NMB_paciente(jj);
186             if  (NMB_paciente(jj) > 7) && (Subindex >= 3)
187                 index = 15; P(Subindex)=NMB_paciente(jj);
188             end
189         elseif (NMB_paciente(jj) <= NMB_paciente(jj-1)) && (index >= 1)
190             holdindex = holdindex + 1; Subindex = 0;
191             if (NMB_paciente(jj) > 7 ) && (holdindex > 3)
192                 index = 0; holdindex =0;
193             elseif(NMB_paciente(jj) <= 7 ) && (holdindex >= 3)
194                 index = 0; holdindex =0; Subindex = 0;
195             end
196         end
197         if index >= 15
198             ChangePoint=jj;
199             return
200         end
201     end
202 end
203 end
```

A.3 Controller Choice by the Best Reference Tracking

```
function Control_Law_Offline_RED_Olard
clear all
clc
%% Load da base de dados com parametros Alpha e Gamma
% load([cd '\Base de Dados\Alpha_Gamma_Atr_100.mat'])
load([cd '\Base de Dados\Alpha_Gamma_Roc_37.mat'])
%% Determinar numero de modelos presentes no banco de dados
length1 = length(NEW_THETA_100(1,:));
for m=1:length1
    %% Definir variaves iniciais
    h = 20; % Tempo de discretização em segundos
    h = h/60;  % Tempo de discretização em minutos
    tempo_simulacao = 300; % Tempo de simulação em minutos
    tempo_simulacao = tempo_simulacao/h; % Em instantes de amostragem
%     bolus_inicial = 500/h; % Bolus inicial em ug/kg, para o Atracurium
    bolus_inicial = 600/h; % Bolus inicial em ug/kg, para o Rocuronium
    nmb_ref = 10; % Target para o NMB
    lambda = 0.5; % Parametro do controlador da Lei de Controlo de Massa
    ChangePoint = [];
%     c50_red = 0.6487*5; % C50 para o Atracurium
    c50_red = 1; % C50 para o Rocuronium
    NMB_paciente = 100;
    NMB_paciente2 = [];
    NMB_paciente2 = 100;
    erro = [];
    %% Retirar da base de dados um modelo para ser o paciente e construção
    de uma matriz com os restantes modelos que sobram como base de dados
    paciente = m; % Modelo usado como paciente
    parametros_paciente(:,1) = NEW_THETA_100(:,paciente);
    for e=1:length1
        if e<paciente
            parametros_modelos(:,e) = NEW_THETA_100(:,e);
        else if e==paciente
            else if e>paciente
```

```
34                      parametros_modelos(:,e-1) = NEW_THETA_100(:,e);
35                  end
36              end
37          end
38      end
39      %% Identificação do numero de modelos presentes na base de dados
40      num_modelos = length(parametros_modelos(1,:));
41      indice_modelo = [1:1:num_modelos]';
42      %% Obter as matrizes e parametros necessários à simulação
43      [Ad_paciente,Bd_paciente,gamma_paciente,x_paciente] =
        modelo_reduzido_paciente(parametros_paciente,h,num_modelos);
44      [Ad_REDM,Bd_REDM,x_modelos,gamma_modelos,Ce_estrela] =
        modelo_reduzido_modelos(parametros_modelos,h,num_modelos,indice_modelo,
        c50_red,nmb_ref);
45      %% Simulação
46      for i=1:tempo_simulacao
47          for j=1:num_modelos
48              x_paciente2(:,1) = x_paciente{i,j};
49              x_modelos2(:,1) = x_modelos{i,j};
50              Ad_modelos = Ad_REDM{j};
51              Bd_modelos = Bd_REDM{j};
52              if isempty(ChangePoint)
53                  if i==1
54                      u_RED(i,j) = bolus_inicial;
55                  else
56                      u_RED(i,j) = 0;
57                  end
58                  x_paciente2(:,2) = Ad_paciente*x_paciente2(:,1)+Bd_paciente
                    *u_RED(i,j);
59                  x_paciente{i+1,j} = x_paciente2(:,2);
60                  y_paciente(i+1,j) = [0 0 1]*x_paciente2(:,2);
61                  NMB_paciente(i+1,j) = [100/(1+((y_paciente(i+1,j)/c50_red)
                    ^gamma_paciente))];
62                  if j==1
63                      NMB_paciente2(i+1) = NMB_paciente(i+1,j);
64                  end
```

```
65                  x_modelos2(:,2) = Ad_modelos*x_modelos2(:,1)+Bd_modelos
                    *u_RED(i,j);
66                  x_modelos{i+1,j} = x_modelos2(:,2);
67                  y_modelos(i+1,j) = [0 0 1]*x_modelos2(:,2);
68                  NMB_modelos(i+1,j) = [100/(1+((y_modelos(i+1,j)/c50_red)
                    ^gamma_modelos(j)))];
69              else
70                  modelo = indice_modelo(j);
71                  Ad_control = Ad_REDM{modelo};
72                  Bd_control = Bd_REDM{modelo};
73                  x_control = x_modelos{i,j};
74                  M_estrela = 3*Ce_estrela(modelo);
75                  M_paciente = x_paciente2(1,1)+x_paciente2(2,1)+x_paciente2(3,1);
76                  u_RED(i,j) = min(80,max(0,(((sum(Bd_control))^(-1))*[1 1 1]*
                    (lambda*eye(3)-Ad_control)*x_control(:,1)+(((sum(Bd_control))^
                    (-1))*(1-lambda)*M_estrela))));
77                  x_paciente2(:,2) = Ad_paciente*x_paciente2(:,1)+Bd_paciente
                    *u_RED(i,j);
78                  x_paciente{i+1,j} = x_paciente2(:,2);
79                  y_paciente(i+1,j) = [0 0 1]*x_paciente2(:,2);
80                  NMB_paciente(i+1,j) = [100/(1+((y_paciente(i+1,j)/c50_red)
                    ^gamma_paciente))];
81                  x_modelos2(:,2) = Ad_modelos*x_modelos2(:,1)+Bd_modelos
                    *u_RED(i,j);
82                  x_modelos{i+1,j} = x_modelos2(:,2);
83                  y_modelos(i+1,j) = [0 0 1]*x_modelos2(:,2);
84                  NMB_modelos(i+1,j) = [100/(1+((y_modelos(i+1,j)/c50_red)
                    ^gamma_modelos(j)))];
85                  erro(i+1,j) = abs(NMB_paciente(i+1,j)-nmb_ref);
86              end
87          end
88          [ChangePoint] = OLARD_bloco(NMB_paciente2);
89      end
90      %% Determinar qual o melhor controlador
91      Erro_Acumulado = sum(erro);
92      min_1 = min(Erro_Acumulado);
```

```
93      modelo1 = find(Erro_Acumulado==min_1);
94      %% Construir variáveis
95      Leis_de_Controlo(m) = modelo1; % Guardar qual foi o melhor Controlador
96      Inicio_Recuperacao(m) = ChangePoint*h; % Inicio da recurperação em minutos
97      NMB(:,m) = NMB_paciente(:,modelo1); % NMB do paciente com a melhor lei de
        controlo
98      All_NMBs{m} = NMB_paciente; % Guardar o NMB do paciente com a melhor lei
        de controlo
99  end
100 %% Construir variáveis
101 numero_paciente = [1:1:length1];
102 Resultados_Offline_RED(1,:) = numero_paciente;
103 Resultados_Offline_RED(2,:) = Leis_de_Controlo;
104 %% Save das variáveis
105 % save Dados_Offline_RED_Atr Resultados_Offline_RED NMB Inicio_Recuperacao
    All_NMBs
106 save Dados_Offline_RED_Roc Resultados_Offline_RED NMB Inicio_Recuperacao
    All_NMBs
107 end
108
109 function [Ad_paciente,Bd_paciente,gamma_paciente,x_paciente] =
    modelo_reduzido_paciente(parametros_paciente,h,num_modelos)
110 %% Obtenção dos parâmetros para o modelo do paciente
111 k1=1; k2=4; k3=10;
112 alpha_paciente = parametros_paciente(1,1);
113 gamma_paciente = parametros_paciente(2,1);
114 %% Construção das matrizes do modelo compartimental
115 Ac_paciente = [(-k3*alpha_paciente) 0 0;(k2*alpha_paciente) -
    (k2*alpha_paciente) 0;0 (k1*alpha_paciente) -(k1*alpha_paciente)];
116 Bc_paciente = [k3*alpha_paciente 0 0]';
117 %% Discretização do sistema compartimental
118 [Ad_paciente,Bd_paciente] = c2d(Ac_paciente,Bc_paciente,h);
119 %% Definir estado inicial do paciente
120 x_paciente = [];
121 x_paciente2 = [];
122 x_RED_paciente = Bc_paciente; % Estado Inicial discreto = à matriz B contínua
```

```
for gg=1:num_modelos
    x_paciente{1,gg} = x_RED_paciente;
end
end

function [Ad_REDM,Bd_REDM,x_modelos,gamma_modelos,Ce_estrela] =
modelo_reduzido_modelos(parametros_modelos,h,num_modelos,indice_modelo,
c50_red,nmb_ref)
k1=1; k2=4; k3=10;
x_modelos  = [];
x_modelos2 = [];
for k=1:num_modelos
    % Obtenção dos parâmetros para os restantes modelos
    alpha_modelos(k) = parametros_modelos(1,indice_modelo(k,1));
    gamma_modelos(k) = parametros_modelos(2,indice_modelo(k,1));
    % Construção das matrizes dos modelos compartimentais
    Ac_modelos = [(-k3*alpha_modelos(k)) 0 0;(k2*alpha_modelos(k)) -
    (k2*alpha_modelos(k)) 0;0 (k1*alpha_modelos(k)) -(k1*alpha_modelos(k))];
    Bc_modelos = [k3*alpha_modelos(k) 0 0]';
    % Discretização dos sistemas compartimentais
    [Ad_modelos,Bd_modelos] = c2d(Ac_modelos,Bc_modelos,h);
    % Guardar as matrizes de todos os modelos
    Ad_REDM{k} = Ad_modelos;
    Bd_REDM{k} = Bd_modelos;
    % Definir estados iniciais dos modelos
    x_RED_modelos  = Bc_modelos;
    x_modelos{1,k} = x_RED_modelos;
    % Obter Ce* dos modelos
    Ce_estrela(k) = [(((100*(c50_red^gamma_modelos(k)))-((c50_red
    ^gamma_modelos(k))*nmb_ref))/nmb_ref)^(1/gamma_modelos(k))];
end
end

function [ChangePoint] = OLARD_bloco(NMB_paciente2)
ChangePoint=[];
index = 0;  holdindex = 0; Subindex = 0;
```

```
155 if length(NMB_paciente2)>54
156     baseline = mean(NMB_paciente2(49-4:49+5));
157     for jj=54:length(NMB_paciente2)
158         if (NMB_paciente2(jj) > NMB_paciente2(jj-1)) && (NMB_paciente2(jj) >
            baseline+1)
159             index = index + 1; Subindex = Subindex + 1; holdindex = 0;
                P(Subindex) = NMB_paciente2(jj);
160             if (NMB_paciente2(jj) > 7) && (Subindex >= 3)
161                 index = 15;  P(Subindex)=NMB_paciente2(jj);
162             end
163         elseif (NMB_paciente2(jj) <= NMB_paciente2(jj-1)) && (index >= 1)
16 4            holdindex = holdindex + 1; Subindex = 0;
165             if (NMB_paciente2(jj) > 7 ) && (holdindex > 3)
166                 index = 0; holdindex =0;
167             elseif(NMB_paciente2(jj) <= 7 ) && (holdindex >= 3)
168                 index = 0; holdindex =0; Subindex = 0;
169             end
170         end
171         if index >= 15
172             ChangePoint=jj;
173             return
174         end
175     end
176 end
177 end
```

A.4 Model choice by Alpha or Gamma proximity

```
1 function Proximidade_Alpha_Gamma_RED
2 clear all
3 clc
4 %% Load da base de dados com os parametros Alpha e Gamma
5 % load([cd '\Base de Dados\Alpha_Gamma_Atr_100.mat'])
6 load([cd '\Base de Dados\Alpha_Gamma_Roc_37.mat'])
7 %% Determinar o numero de modelos no banco de dados
```

```
length1 = length(NEW_THETA_100(1,:));
%% Retirar da base de dados um modelo para ser o paciente e construção
de uma matriz com os restantes modelos que sobram da base de dados
for i=1:length1
    paciente = i; % Modelo usado para simular o paciente
    parametros_paciente(:,1) = NEW_THETA_100(:,paciente);
    for e=1:length1
        if e<paciente
            parametros_modelos(:,e) = NEW_THETA_100(:,e);
        else if e==paciente
            else if e>paciente
                    parametros_modelos(:,e-1) = NEW_THETA_100(:,e);
                end
            end
        end
    end
    %% Identificação do numero de modelos presentes na base de dados
    num_modelos    = length(parametros_modelos(1,:));
    indice_modelo = [1:1:num_modelos]';
    %% Recolher os parametros Alpha e Gamma
    [alpha_paciente,gamma_paciente] = modelo_reduzido_paciente
    (parametros_paciente);
    [alpha_modelos,gamma_modelos] = modelo_reduzido_modelos
    (parametros_modelos,num_modelos,indice_modelo);
    %% Determinar o modelo com o alpha mais próximo ao do paciente
    delta_alpha = abs(alpha_paciente-alpha_modelos);
    [MIN, modelo2] = min(delta_alpha);
    Alpha_Proximo(i) = modelo2;
    %% Determinar o modelo com o gamma mais próximo ao do paciente
    delta_gamma = abs(gamma_paciente-gamma_modelos);
    [MIN, modelo3] = min(delta_gamma);
    Gamma_Proximo(i) = modelo3;
end
%% Guardar Dados
% save Dados_Proximidade_Alpha_Gamma_RED_Atr Alpha_Proximo Gamma_Proximo
save Dados_Proximidade_Alpha_Gamma_RED_Roc Alpha_Proximo Gamma_Proximo
```

```
41 end
42
43 function [alpha_paciente,gamma_paciente] = modelo_reduzido_paciente
   (parametros_paciente)
44 %% Obtenção dos parâmetros para o modelodo paciente
45 alpha_paciente = parametros_paciente(1,1);
46 gamma_paciente = parametros_paciente(2,1);
47 end
48
49 function [alpha_modelos,gamma_modelos] = modelo_reduzido_modelos
   (parametros_modelos,num_modelos,indice_modelo)
50 for k=1:num_modelos
51     %% Obtenção dos parâmetros para os restantes modelos
52     alpha_modelos(k) = parametros_modelos(1,indice_modelo(k,1));
53     gamma_modelos(k) = parametros_modelos(2,indice_modelo(k,1));
54 end
55 end
```

A.5 Model choice by Steady-State Input

```
1  function U_SteadyState_RED
2  clear all
3  clc
4  %% Load do banco de dados com os parametros Alpha e Gamma
5  % load([cd '\Base de Dados\Alpha_Gamma_Atr_100.mat'])
6  load([cd '\Base de Dados\Alpha_Gamma_Roc_37.mat'])
7  %% Determinar o numero de modelos na banco de dados
8  length1 = length(NEW_THETA_100(1,:));
9  for i=1:length1
10     Alpha = NEW_THETA_100(1,i);
11     Gamma = NEW_THETA_100(2,i);
12     C50 = 1; % C50 para o Rocuronium
13 %     C50 = 0.6487*5; % C50 para o Atracurium
14     k1=1;k2=4;k3=10;
15     Var1 = (Alpha*k1*k2*k3);
```

```
16      Var2 = (Var1/((k2-k1)*(k3-k1)))/(Alpha*k1);
17      Var3 = (Var1/((k1-k2)*(k3-k2)))/(Alpha*k2);
18      Var4 = (Var1/((k1-k3)*(k2-k3)))/(Alpha*k3);
19      Var5 = Var2+Var3+Var4;
20      U_SS(i) = C50*((((100/nmb_ref)-1)^(1/Gamma))/Var5);
21  end
22  for d=1:length1
23      %% Obtenção do Uss do paciente
24      paciente = d;
25      Uss_paciente = U_SS(d);
26      %% Obtenção dos Uss dos restantes modelos
27      for s=1:length1
28          if s<paciente
29              Uss_modelos(s) = U_SS(s);
30          else if s==paciente
31              else if s>paciente
32                      Uss_modelos(s-1) = U_SS(s);
33                  end
34              end
35          end
36      end
37      for j=1:length1-1
38          Delta_Uss(j) = abs(Uss_paciente-Uss_modelos(j));
39      end
40      %% Determinação do modelo mais próximo
41      [MIN, modelo] = min(Delta_Uss);
42      Modelos_Uss(d) = modelo;
43      %% Guardar dados
44      % save Dados_U_SteadyState_Atr Modelos_Uss U_SS
45      save Dados_U_SteadyState_Roc Modelos_Uss U_SS
46  end
```

A.6 Model choice by the Norm 2 metric

```
function Metrica_Norma_OpenLoop_RED
clear all
clc
%% Load da base de dados com os parametros Alpha e Gamma
% load([cd '\Base de Dados\Alpha_Gamma_Atr_100.mat'])
load([cd '\Base de Dados\Alpha_Gamma_Roc_37.mat'])
%% Determinar o numero de modelos no banco de dados
length1 = length(NEW_THETA_100(1,:));
%% Definir o tempo de discretização
h = 20; % Tempo de discretização em segundos.
h = h/60; % Tempo de discretização em minutos.
%% Retirar da base de dados um modelo para ser o paciente e construção de
uma matriz com os restantes modelos que sobram da base de dados
for i=1:length1
    paciente = i; % Modelo usado para simular o paciente
    parametros_paciente(:,1) = NEW_THETA_100(:,paciente);
    for e=1:length1
        if e<paciente
            parametros_modelos(:,e) = NEW_THETA_100(:,e);
        else if e==paciente
            else if e>paciente
                    parametros_modelos(:,e-1) = NEW_THETA_100(:,e);
                end
            end
        end
    end
    %% Identificação do numero de modelos presentes na base de dados
    if i==1
        num_modelos = length(parametros_modelos(1,:));
        indice_modelo = [1:1:num_modelos]';
    end
    %% Obter os sistemas do modelo do paciente e dos modelos da base de dados
    [sysc_paciente,sysd_paciente] = modelo_reduzido_paciente
    (parametros_paciente,h);
```

```
33      [sysc_modelos,sysd_modelos] = modelo_reduzido_modelos(parametros_modelos,
        num_modelos,indice_modelo,h);
34      %% Norma dos modelos do paciente e dos modelos da base de dados
35      Norma_Paciente1 = norm(sysc_paciente,2);
36      Norma_Paciente2 = norm(sysd_paciente,2);
37      for k=1:num_modelos
38          sysc_modelos2 = sysc_modelos{k};
39          sysd_modelos2 = sysd_modelos{k};
40          Norma_Modelos1(k) = norm(sysc_modelos2,2);
41          Norma_Modelos2(k) = norm(sysd_modelos2,2);
42      end
43      %% Encontrar modelo com Norma2 mais semelhante ao paciente
44      [Min1,Modelo1] = min(abs(Norma_Paciente1-Norma_Modelos1));
45      Modelos(1,i) = Modelo1;
46      [Min2,Modelo2] = min(abs(Norma_Paciente2-Norma_Modelos2));
47      Modelos(2,i) = Modelo2;
48  end
49  %% Construção da variável com resultados finais
50  numero_paciente = [1:1:length1];
51  Resultados_Norma_RED(1,:) = numero_paciente;
52  Resultados_Norma_RED(2,:) = Modelos(1,:);
53  Resultados_Norma_RED(3,:) = Modelos(2,:);
54  %% Guardar Dados
55  % save Dados_Norma_OpenLoop_RED_Atr Resultados_Norma_RED
56  save Dados_Norma_OpenLoop_RED_Roc Resultados_Norma_RED
57  end
58
59  function [sysc_paciente,sysd_paciente] = modelo_reduzido_paciente
 (parametros_paciente,h)
60  %% Obtenção dos parâmetros para o modelo do paciente
61  k1=1; k2=4; k3=10;
62  alpha_paciente = parametros_paciente(1,1);
63  %% Construção das matrizes do modelo compartimental
64  Ac_paciente = [(-k3*alpha_paciente) 0 0;(k2*alpha_paciente) -(k2*alpha_paciente)
 0;0 (k1*alpha_paciente) -(k1*alpha_paciente)];
65  Bc_paciente = [k3*alpha_paciente 0 0]';
```

```
66  Cc_paciente = [0 0 1];
67  Dc_paciente = 0;
68  %% Obtenção do sistema compartimental
69  sysc_paciente = ss(Ac_paciente,Bc_paciente,Cc_paciente,Dc_paciente);
70  sysd_paciente = c2d(sysc_paciente,h,'zoh');
71  end
72
73  function [sysc_modelos,sysd_modelos] = modelo_reduzido_modelos
 (parametros_modelos,num_modelos,indice_modelo,h)
74  %% Identificação das variáveis fixas
75  k1=1; k2=4; k3=10;
76  for k=1:num_modelos
77      num = [];
78      den = [];
79      %% Obtenção dos parâmetros para os restantes modelos
80      alpha_modelos(k) = parametros_modelos(1,indice_modelo(k,1));
81      %% Construção das matrizes dos modelos compartimentais
82      Ac_modelos = [(-k3*alpha_modelos(k)) 0 0;(k2*alpha_modelos(k)) -
        (k2*alpha_modelos(k)) 0;0 (k1*alpha_modelos(k)) -(k1*alpha_modelos(k))];
83      Bc_modelos = [k3*alpha_modelos(k) 0 0]';
84      Cc_modelos = [0 0 1];
85      Dc_modelos = 0;
86      %% Obtenção dos sistemas compartimentais
87      sysc_modelos{k} = ss(Ac_modelos,Bc_modelos,Cc_modelos,Dc_modelos);
88      sysd_modelos{k} = c2d(sysc_modelos{k},h,'zoh');
89  end
90  end
```

A.7 Model choice by the Vinnicombe metric

```
1  function Metrica_Vinnicombe_OpenLoop_RED
2  clear all
3  clc
4  %% Load da base de dados com os parametros Alpha e Gamma
5  % load([cd '\Base de Dados\Alpha_Gamma_Atr_100.mat'])
```

```
load([cd '\Base de Dados\Alpha_Gamma_Roc_37.mat'])
%% Determinar numero de modelos presentes no banco de dados
length1 = length(NEW_THETA_100(1,:));
%% Retirar da base de dados um modelo para ser o paciente e construção
de uma matriz com os restantes modelos que sobram da base de dados
for i=1:length1
    paciente = i; % Modelo usado como paciente
    parametros_paciente(:,1) = NEW_THETA_100(:,paciente);
    for e=1:length1
        if e<paciente
            parametros_modelos(:,e) = NEW_THETA_100(:,e);
        else if e==paciente
            else if e>paciente
                    parametros_modelos(:,e-1) = NEW_THETA_100(:,e);
                end
            end
        end
    end
    %% Identificação do numero de modelos presentes na base de dados
    if i==1
        num_modelos = length(parametros_modelos(1,:));
        indice_modelo = [1:1:num_modelos]';
    end
    %% Obter as equações de transferência
    [tf_paciente] = modelo_reduzido_paciente(parametros_paciente);
    [tf_modelos] = modelo_reduzido_modelos(parametros_modelos,num_modelos,indice_mode
    %% Métrica de Vinnicombe
    for k=1:num_modelos
        tf_modelos2 = tf_modelos{k};
        [gap,nugap] = gapmetric(tf_paciente,tf_modelos2);
        valores_gap(k) = gap;
        valores_vinnicombe(k) = nugap;
    end
    %% Encontrar modelo mais semelhante ao paciente
    min1_nugap = min(valores_vinnicombe);
    modelo_nugap = find(min1_nugap==valores_vinnicombe);
```

```
41      Modelos{1,i} = modelo_nugap;
42      xx = length(modelo_nugap);
43      if xx>1
44          modelo_Vinnicombe(i) = modelo_nugap(1);
45      else
46          modelo_Vinnicombe(i) = modelo_nugap;
47      end
48  end
49  %% Construção da variável com resultados finais
50  numero_paciente = [1:1:length1];
51  Resultados_Vinnicombe_RED(1,:) = numero_paciente;
52  Resultados_Vinnicombe_RED(2,:) = modelo_Vinnicombe;
53  Resultados_Vinnicombe_RED(3,:) = modelo_Gap2;
54  %% Guardar Dados
55  % save Dados_Vinnicombe_OpenLoop_RED_Atr Resultados_Vinnicombe_RED Modelos
56  save Dados_Vinnicombe_OpenLoop_RED_Roc Resultados_Vinnicombe_RED Modelos
57  end
58
59  function [tf_paciente] = modelo_reduzido_paciente(parametros_paciente)
60  num = [];
61  den = [];
62  %% Obtenção dos parâmetros do modelo do paciente
63  k1=1; k2=4; k3=10;
64  alpha_paciente = parametros_paciente(1,1);
65  gamma_paciente = parametros_paciente(2,1);
66  %% Construção das matrizes do modelo compartimental
67  Ac_paciente = [(-k3*alpha_paciente) 0 0;(k2*alpha_paciente) -
    (k2*alpha_paciente) 0;0 (k1*alpha_paciente) -(k1*alpha_paciente)];
68  Bc_paciente = [k3*alpha_paciente 0 0]';
69  Cc_paciente = [0 0 1];
70  Dc_paciente = 0;
71  %% Obtenção da equação de transferência
72  [num,den] = ss2tf(Ac_paciente,Bc_paciente,Cc_paciente,Dc_paciente);
73  tf_paciente = tf(num,den);
74  end
75
```

```
function [tf_modelos] = modelo_reduzido_modelos(parametros_modelos,
num_modelos,indice_modelo)
%% Identificação das variáveis fixas
k1=1; k2=4; k3=10;
for k=1:num_modelos
    num = [];
    den = [];
    %% Obtenção dos parâmetros dos restantes modelos
    alpha_modelos = parametros_modelos(1,indice_modelo(k,1));
    gamma_modelos = parametros_modelos(2,indice_modelo(k,1));
    %% Construção das matrizes dos modelos compartimentais
    Ac_modelos = [(-k3*alpha_modelos) 0 0;(k2*alpha_modelos) -
    (k2*alpha_modelos) 0;0 (k1*alpha_modelos) -(k1*alpha_modelos)];
    Bc_modelos = [k3*alpha_modelos 0 0]';
    Cc_modelos = [0 0 1];
    Dc_modelos = 0;
    %% Obtenção das equações de transferência
    [num,den] = ss2tf(Ac_modelos,Bc_modelos,Cc_modelos,Dc_modelos);
    tf_modelos2 = tf(num,den);
    tf_modelos{k} = tf_modelos2;
end
end
```

A.8 Model choice by Impulse or Step Response proximity

```
function Comp_Impulse_Step_RED
clear all
clc
%% Identificação das variáveis iniciais
h = 20; % Tempo de discretização em segundos
h = h/60; % Tempo de discretização em minutos
tempo_simulacao = 300; % Tempo de simulação em minutos
tempo_simulacao = tempo_simulacao/h; % Em instantes discretizados
% dose_impulso = 500; % Dose do impulso para o Atracurium
dose_impulso = 600; % Dose do impulso para o Rocuronium
```

```
% dose_step = 5; % Dose do degrau para o Atracurium
dose_step = 6; % Dose do degrau para o Rocuronium
% c50_red = 0.6487*5; % C50 para o Atracurium
c50_red = 1; % C50 para o Rocuronium
%% Load da base de dados com os parametros Alpha e Gamma
% load([cd '\Base de Dados\Alpha_Gamma_Atr_100.mat'])
load([cd '\Base de Dados\Alpha_Gamma_Roc_37.mat'])
%% Determinar o numero de modelos no banco de dados
length1=length(NEW_THETA_100(1,:));
%% Retirar da base de dados um modelo para ser o paciente e construção
de uma matriz com os restantes modelos que sobram da base de dados
for i=1:length1
    paciente = i; % Modelo usado para simular o paciente
    parametros_paciente(:,1) = NEW_THETA_100(:,paciente);
    for s=1:length1
        if s<paciente
            parametros_modelos(:,s) = NEW_THETA_100(:,s);
        else if s==paciente
            else if s>paciente
                    parametros_modelos(:,s-1) = NEW_THETA_100(:,s);
                end
            end
        end
    end
    %% Identificação do numero de modelos presentes na base de dados
    num_modelos = length(parametros_modelos(1,:));
    indice_modelo = [1:1:num_modelos]';
    %% Obter as respostas ao impulso e ao degrau do modelo do paciente e
    dos modelos da base de dados
    [imp_cont_paciente,imp_disc_paciente,step_cont_paciente,step_disc_paciente,
    NMB_imp_cont_paciente,NMB_imp_disc_paciente,NMB_step_cont_paciente,
    NMB_step_disc_paciente] = modelo_reduzido_paciente(parametros_paciente,
    h,dose_impulso,dose_step,tempo_simulacao,c50_red);
    [imp_cont_modelos,imp_disc_modelos,step_cont_modelos,step_disc_modelos,
    NMB_imp_cont_modelos,NMB_imp_disc_modelos,NMB_step_cont_modelos,
    NMB_step_disc_modelos] = modelo_reduzido_modelos(parametros_modelos,h,
```

```
    dose_impulso,dose_step,tempo_simulacao,num_modelos,indice_modelo,c50_red);
40  %% Comparar as respostas ao impulso e ao degrau
41  for g=1:tempo_simulacao
42      for f=1:num_modelos
43          imp_cont_modelos2 = imp_cont_modelos{f};
44          erro_imp_cont(g,f) = sqrt((imp_cont_paciente(g)-
            imp_cont_modelos2(g))^2);
45          step_cont_modelos2 = step_cont_modelos{f};
46          erro_step_cont(g,f) = sqrt((step_cont_paciente(g)-
            step_cont_modelos2(g))^2);
47          imp_disc_modelos2 = imp_disc_modelos{f};
48          erro_imp_disc(g,f) = sqrt((imp_disc_paciente(g)-
            imp_disc_modelos2(g))^2);
49          step_disc_modelos2 = step_disc_modelos{f};
50          erro_step_disc(g,f) = sqrt((step_disc_paciente(g)-
            step_disc_modelos2(g))^2);
51
52          NMB_imp_cont_modelos2 = NMB_imp_cont_modelos{f};
53          erro_NMB_imp_cont(g,f) = sqrt((NMB_imp_cont_paciente(g)-
            NMB_imp_cont_modelos2(g))^2);
54          NMB_step_cont_modelos2 = NMB_step_cont_modelos{f};
55          erro_NMB_step_cont(g,f) = sqrt((NMB_step_cont_paciente(g)-
            NMB_step_cont_modelos2(g))^2);
56          NMB_imp_disc_modelos2 = NMB_imp_disc_modelos{f};
57          erro_NMB_imp_disc(g,f) = sqrt((NMB_imp_disc_paciente(g)-
            NMB_imp_disc_modelos2(g))^2);
58          NMB_step_disc_modelos2 = NMB_step_disc_modelos{f};
59          erro_NMB_step_disc(g,f) = sqrt((NMB_step_disc_paciente(g)-
            NMB_step_disc_modelos2(g))^2);
60      end
61  end
62  %% Fazer o somatório dos erros das diferenças
63  erro2_imp_cont = sum(erro_imp_cont);
64  erro2_imp_disc = sum(erro_imp_disc);
65  erro2_step_cont = sum(erro_step_cont);
66  erro2_step_disc = sum(erro_step_disc);
```

```
67
68      erro2_NMB_imp_cont = sum(erro_NMB_imp_cont);
69      erro2_NMB_imp_disc = sum(erro_NMB_imp_disc);
70      erro2_NMB_step_cont = sum(erro_NMB_step_cont);
71      erro2_NMB_step_disc = sum(erro_NMB_step_disc);
72      %% Identificar o somatório mais pequeno
73      erro_min_imp_cont = min(erro2_imp_cont);
74      erro_min_imp_disc = min(erro2_imp_disc);
75      erro_min_step_cont = min(erro2_step_cont);
76      erro_min_step_disc = min(erro2_step_disc);
77
78      erro_min_NMB_imp_cont = min(erro2_NMB_imp_cont);
79      erro_min_NMB_imp_disc = min(erro2_NMB_imp_disc);
80      erro_min_NMB_step_cont = min(erro2_NMB_step_cont);
81      erro_min_NMB_step_disc = min(erro2_NMB_step_disc);
82      %% Identificar o modelo com o somatório mais pequeno
83      modelo_imp_cont = find(erro_min_imp_cont==erro2_imp_cont);
84      modelo_imp_disc = find(erro_min_imp_disc==erro2_imp_disc);
85      modelo_step_cont = find(erro_min_step_cont==erro2_step_cont);
86      modelo_step_disc = find(erro_min_step_disc==erro2_step_disc);
87
88      modelo_NMB_imp_cont = find(erro_min_NMB_imp_cont==erro2_NMB_imp_cont);
89      modelo_NMB_imp_disc = find(erro_min_NMB_imp_disc==erro2_NMB_imp_disc);
90      modelo_NMB_step_cont = find(erro_min_NMB_step_cont==erro2_NMB_step_cont);
91      modelo_NMB_step_disc = find(erro_min_NMB_step_disc==erro2_NMB_step_disc);
92      %% Guardar modelos em variáveis
93      Modelo_Impulso_Continuo(i) = modelo_imp_cont;
94      Modelo_Impulso_Discreto(i) = modelo_imp_disc;
95      Modelo_Degrau_Continuo(i) = modelo_step_cont;
96      Modelo_Degrau_Discreto(i) = modelo_step_disc;
97
98      Modelo_NMB_Impulso_Continuo(i) = modelo_NMB_imp_cont;
99      Modelo_NMB_Impulso_Discreto(i) = modelo_NMB_imp_disc;
100     Modelo_NMB_Degrau_Continuo(i) = modelo_NMB_step_cont;
101     Modelo_NMB_Degrau_Discreto(i) = modelo_NMB_step_disc;
102 end
```

```
103 %% Guardar dados
104 numero_paciente = [1:1:length1];
105 Resultados_Imp_Step(1,:) = numero_paciente;
106 Resultados_Imp_Step(2,:) = Modelo_Impulso_Continuo;
107 Resultados_Imp_Step(3,:) = Modelo_Degrau_Continuo;
108 Resultados_Imp_Step(4,:) = Modelo_Impulso_Discreto;
109 Resultados_Imp_Step(5,:) = Modelo_Degrau_Discreto;
110
111 Resultados_NMB_Imp_Step(1,:) = numero_paciente;
112 Resultados_NMB_Imp_Step(2,:) = Modelo_NMB_Impulso_Continuo;
113 Resultados_NMB_Imp_Step(3,:) = Modelo_NMB_Degrau_Continuo;
114 Resultados_NMB_Imp_Step(4,:) = Modelo_NMB_Impulso_Discreto;
115 Resultados_NMB_Imp_Step(5,:) = Modelo_NMB_Degrau_Discreto;
116
117 % save Dados_Imp_Step_RED_Atr Resultados_Imp_Step Resultados_NMB_Imp_Step
118 save Dados_Imp_Step_RED_Roc Resultados_Imp_Step Resultados_NMB_Imp_Step
119 end
120
121 function [imp_cont_paciente,imp_disc_paciente,step_cont_paciente,
    step_disc_paciente,NMB_imp_cont_paciente,NMB_imp_disc_paciente,
    NMB_step_cont_paciente,NMB_step_disc_paciente] = modelo_reduzido_paciente
    (parametros_paciente,h,dose_impulso,dose_step,tempo_simulacao,c50_red)
122 %% Limpar variáveis
123 imp_cont = [];
124 imp_disc = [];
125 step_cont = [];
126 step_disc = [];
127 NMB_imp_cont_paciente = [];
128 NMB_imp_disc_paciente = [];
129 NMB_step_cont_paciente = [];
130 NMB_step_disc_paciente = [];
131 imp_cont_paciente = [];
132 imp_disc_paciente = [];
133 step_cont_paciente = [];
134 step_disc_paciente = [];
135 %% Obtenção dos parâmetros para o modelo do paciente
```

```
136 k1=1; k2=4; k3=10;
137 alpha_paciente = parametros_paciente(1,1);
138 gamma_paciente = parametros_paciente(2,1);
139 %% Construção das matrizes do modelo compartimental
140 Ac_paciente = [(-k3*alpha_paciente) 0 0;(k2*alpha_paciente) -
    (k2*alpha_paciente) 0;0 (k1*alpha_paciente) -(k1*alpha_paciente)];
141 Bc_paciente = [k3*alpha_paciente 0 0]';
142 Cc_paciente = [0 0 1];
143 Dc_paciente = 0;
144 %% Discretização do sistema compartimental
145 sysc_paciente=ss(Ac_paciente,Bc_paciente,Cc_paciente,Dc_paciente);
146 sysd_paciente=c2d(sysc_paciente,h,'zoh');
147 %% Obtenção das Respostas ao Impulso e ao Degrau para o Modelo do Paciente
148         % Em tempo contínuo
149 [imp_cont,t] = impulse(sysc_paciente,0:1:(tempo_simulacao-1));
150 [step_cont,t] = step(sysc_paciente,0:1:(tempo_simulacao-1));
151 imp_cont_paciente = imp_cont.*dose_impulso;
152 step_cont_paciente = step_cont.*dose_step;
153 for g=1:length(imp_cont)
154     NMB_imp_cont_paciente(g) = [100/(1+(((imp_cont(g)*dose_impulso)/c50_red)
        ^gamma_paciente))];
155     NMB_step_cont_paciente(g) = [100/(1+(((step_cont(g)*dose_step)/c50_red)
        ^gamma_paciente))];
156 end
157         % Em tempo discreto
158 [imp_disc,t] = impulse(sysd_paciente,0:h:(tempo_simulacao-1)/3);
159 [step_disc,t] = step(sysd_paciente,0:h:(tempo_simulacao-1)/3);
160 imp_disc_paciente = imp_disc.*dose_impulso;
161 step_disc_paciente = step_disc.*dose_step;
162 for g=1:length(imp_disc)
163     NMB_imp_disc_paciente(g) = [100/(1+(((imp_disc(g)*dose_impulso)/c50_red)
    ^gamma_paciente))];
164     NMB_step_disc_paciente(g) = [100/(1+(((step_disc(g)*dose_step)/c50_red)
    ^gamma_paciente))];
165 end
166 end
```

```
167
168 function [imp_cont_modelos,imp_disc_modelos,step_cont_modelos,
    step_disc_modelos,NMB_imp_cont_modelos,NMB_imp_disc_modelos,
    NMB_step_cont_modelos,NMB_step_disc_modelos] = modelo_reduzido_modelos
    (parametros_modelos,h,dose_impulso,dose_step,tempo_simulacao,num_modelos,
    indice_modelo,c50_red)
169 %% Limpar variáveis
170 imp_cont = [];
171 imp_disc = [];
172 step_cont = [];
173 step_disc = [];
174 NMB_imp_cont = [];
175 NMB_imp_disc = [];
176 NMB_step_cont = [];
177 NMB_step_disc = [];
178 imp_cont_modelos = [];
179 imp_disc_modelos = [];
180 step_cont_modelos = [];
181 step_disc_modelos = [];
182 %% Identificação das variáveis fixas
183 k1=1; k2=4; k3=10;
184 for k=1:num_modelos
185     %% Obtenção dos parâmetros para os restantes modelos
186     alpha_modelos(k) = parametros_modelos(1,indice_modelo(k,1));
187     gamma_modelos(k) = parametros_modelos(2,indice_modelo(k,1));
188     %% Construção das matrizes dos modelos compartimentais
189     Ac_modelos = [(-k3*alpha_modelos(k)) 0 0;(k2*alpha_modelos(k)) -
    (k2*alpha_modelos(k)) 0;0 (k1*alpha_modelos(k)) -(k1*alpha_modelos(k))];
190     Bc_modelos = [k3*alpha_modelos(k) 0 0]';
191     Cc_modelos = [0 0 1];
192     Dc_modelos = 0;
193     %% Discretização dos sistemas compartimentais
194     sysc_modelos = ss(Ac_modelos,Bc_modelos,Cc_modelos,Dc_modelos);
195     sysd_modelos = c2d(sysc_modelos,h,'zoh');
196     %% Obtenção das Respostas ao Impulso e ao Degrau para os restantes  modelos
197         % Em tempo contínuo
```

```
198     [imp_cont,t] = impulse(sysc_modelos,0:1:(tempo_simulacao-1));
199     [step_cont,t] = step(sysc_modelos,0:1:(tempo_simulacao-1));
200     imp_cont_modelos{k} = imp_cont.*dose_impulso;
201     step_cont_modelos{k} = step_cont.*dose_step;
202     for g=1:length(imp_cont)
203         NMB_imp_cont(g) = [100/(1+(((imp_cont(g)*dose_impulso)/c50_red)
            ^gamma_modelos(k)))];
204         NMB_step_cont(g) = [100/(1+(((step_cont(g)*dose_step)/c50_red)
            ^gamma_modelos(k)))];
205     end
206     NMB_imp_cont_modelos{k} = NMB_imp_cont;
207     NMB_step_cont_modelos{k} = NMB_step_cont;
208         % Em tempo discreto
209     [imp_disc,t] = impulse(sysd_modelos,0:h:(tempo_simulacao-1)/3);
210     [step_disc,t] = step(sysd_modelos,0:h:(tempo_simulacao-1)/3);
211     imp_disc_modelos{k} = imp_disc.*dose_impulso;
212     step_disc_modelos{k} = step_disc.*dose_step;
213     for g=1:length(imp_cont)
214         NMB_imp_disc(g) = [100/(1+(((imp_disc(g)*dose_impulso)/c50_red)
            ^gamma_modelos(k)))];
215         NMB_step_disc(g) = [100/(1+(((step_disc(g)*dose_step)/c50_red)
            ^gamma_modelos(k)))];
216     end
217     NMB_imp_disc_modelos{k} = NMB_imp_disc;
218     NMB_step_disc_modelos{k} = NMB_step_disc;
219 end
220 end
```

Appendix B

GALENO Software

B.1 Datex SAD - Data Acquisition

The program *Datex SAD* allows the user to collect all the data related with the anesthesia performed during a surgery. This program collects all the data obtained from the *Datex Ohmeda* equipment and the drug infusion pumps, and was developed by researchers of the *GALENO* project.

Besides the data acquisition, this program allows the user to see the data values collected, insert notes about the surgery (like instant of intubation, incision, and other things) and any *bolus* administered manually, to register the patient information, the name and the protocol of the surgery, among other aspects related with the surgery.

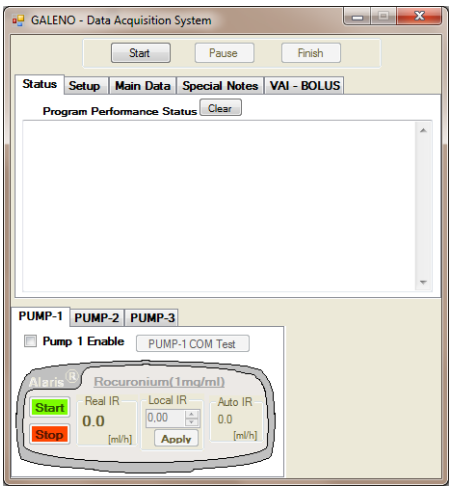

Figure B.1: Datex SAD Program - Status Separator.

Fig. B.1 presents the *Status* separator of the *Datex SAD* program. Here, any information that the program needs to transmit to the user is represented, such as errors occurred, and tests performed to the connection between the computer and the *Datex Ohmeda* equipment and drug

pump infusions.

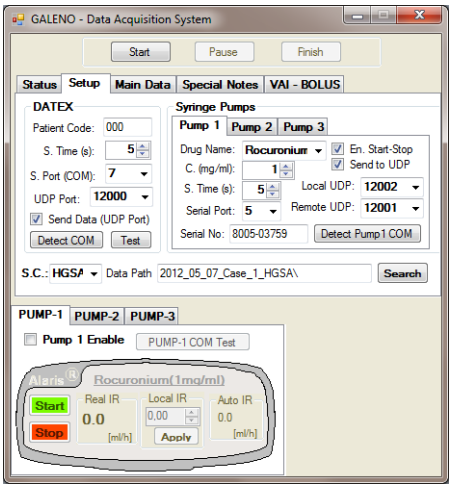

Figure B.2: Datex SAD Program - Setup Separator.

Fig. B.2 presents the *Setup* separator of the *Datex SAD* program. This part allows the user to define the drug of each pump infusion, as well as the drug concentration, the sampling time of the *Datex Ohmeda* and infusion pump data, the path to save all the data collected, among other minor things.

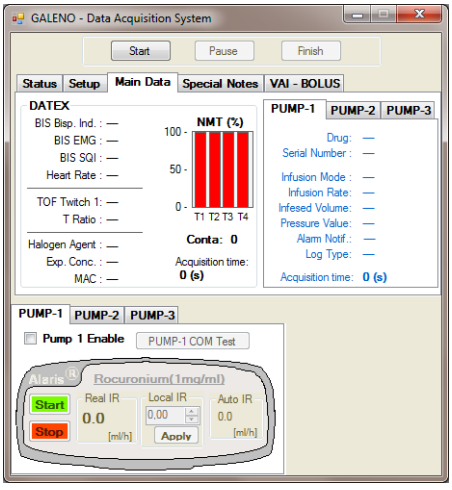

Figure B.3: Datex SAD Program - Main Data Separator.

Fig. B.3 shows the *Separator* that allows the user to see the values acquired from *Datex Ohmeda* equipment like *BIS signal*, *BIS-SQI signal*, *TOF signal*, and from the drug infusion pumps like *infusion rate*.

Fig. B.4 represents the *Special Notes* separator, where the user can register all the information related with the surgery and the patient, like:

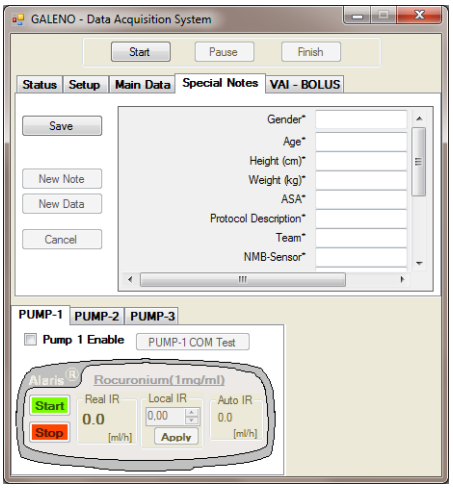

Figure B.4: Datex SAD Program - Special Notes Separator.

- Patient Gender, Age, Height, Weight and ASA;
- Name of the surgery;
- Protocol of the surgery;
- NMB sensor used.

Also, the user is able to insert any *bolus* administered manually to the patient, by recording the name, dose and instant of administration of the drug. Other possibility of this separator is the possibility to insert any desirable note at any instant during the surgery, and so any remarkable aspect of the surgery can be recorded.

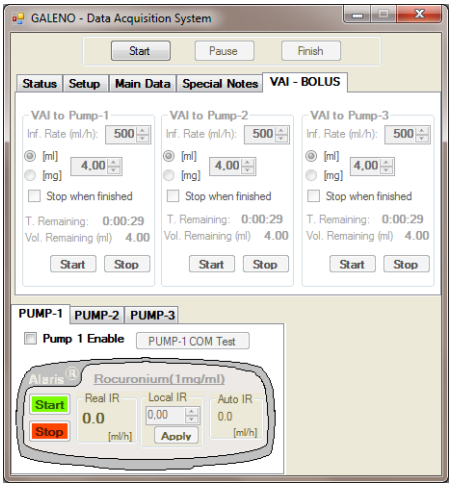

Figure B.5: Datex SAD Program - VAI Bolus Separator.

Fig. B.5 shows the separator that allows the user to give a *bolus* through a direct order sent to the drug infusion pump.

B.2 Monitoring and Control in Anesthesia

The second program, called *Monitoring and Control in Anesthesia* is developed in a MATLAB environment by *GALENO* researchers, and has the main goal to monitoring the process of anesthesia and also to perform the automatic control of the NMB level, if desired.

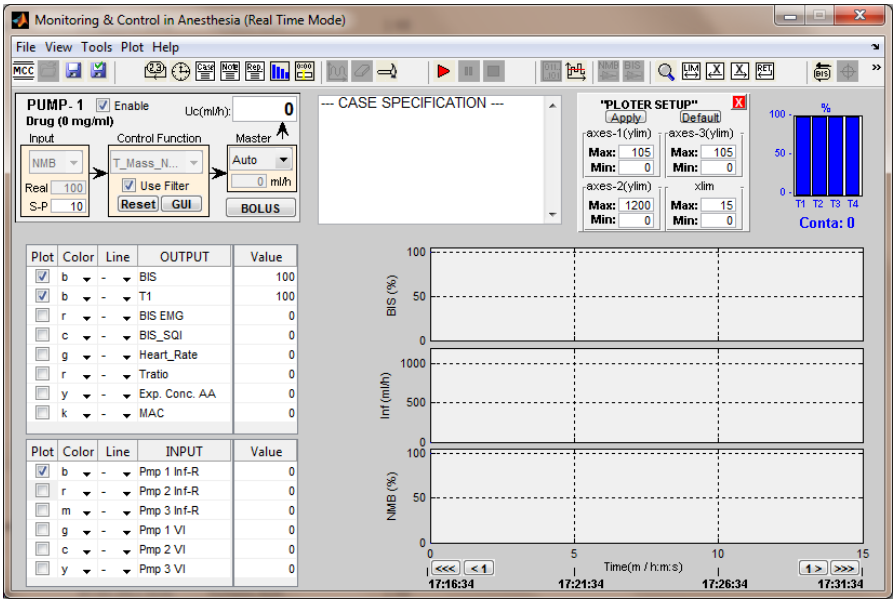

Figure B.6: Monitoring and Control in Anesthesia Program.

In Fig. B.6 the interface of this program can be seen. This program receives data collected by the former program and allows the visualization of that data in function of time. Also, this program allows the control of the NMB; for this purpose, the user only needs to enable the pump (in the upper left corner) and choose the NMB control function. For that, the user must have the NMB control functions, or if desired, create a new one. This program works in a modular way, and in order to perform a new control the user only needs to make the control function and incorporate it in this program.

This program is also able to working offline as an important tool to analyze and study the data collected previously.

Another function of this software is their ability to decode data collected from the *Datex Ohmeda* equipment by the *Datex SAD* software. That data is collect in binary code and a translation into decimal code is necessary, in order to become that information easily interpreted. Furthermore the data collected from the drug infusion pumps is placed together with the former decoded data in a same file (a .mat and a .txt file). The user can choose which data is to transcribed to this files. Using the Setup button the desired variables can be chosen from a list

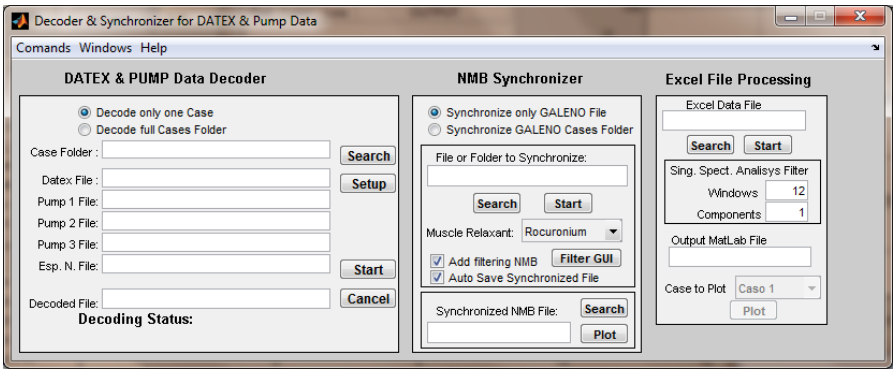

Figure B.7: Data Decoder and NMB Synchronizer.

with all the collected variables (Fig. B.7, DATEX & PUMP Data Decoder).

In this program, the user can also synchronize all the data related with the NMB process (muscle relaxant *bolus* and/or infusions and *TOF Twitch1 signal*) (Fig. B.7, NMB Synchronizer). This program is able to define the beginning of the *TOF Twitch1 signal* (when the first muscle relaxant *bolus* is administered) and the end (when the reverser or antagonist is given to the patient). A *FIR+Median+Nonlinear* filter is applied to the former signal in order to eliminate momentary artefacts. All the information about *bolus* and/or infusions of NMB agents is also synchronized together with the *TOF Twitch1 signal*. After that the user can perform a plot of that same data at any time in order to see the synchronized data.

Appendix C

Bank of Models

C.1 Bank of *Atracurium* Models

Table C.1: Bank $\mathcal{P}$ of 100 *atracurium* models

Model	Alpha	Gamma	Model	Alpha	Gamma	Model	Alpha	Gamma
1	0,044825	2,246444	35	0,035803	3,761446	69	0,031645	3,98684
2	0,041211	2,973995	36	0,035529	3,434949	70	0,039477	3,279347
3	0,037808	1,855902	37	0,03927	2,948437	71	0,044602	2,226665
4	0,039477	2,025008	38	0,034499	1,117047	72	0,038253	2,89103
5	0,033515	4,541045	39	0,037355	4,599891	73	0,041705	2,221494
6	0,048703	1,805011	40	0,048026	2,247864	74	0,037062	2,382454
7	0,037713	2,861148	41	0,042657	1,812593	75	0,051218	3,081781
8	0,049477	1,306051	42	0,038965	1,333677	76	0,04317	2,437369
9	0,038941	2,274889	43	0,032347	3,045493	77	0,028146	2,293676
10	0,042818	1,886052	44	0,036958	5,276774	78	0,037572	4,407872
11	0,029811	2,88261	45	0,03824	2,952315	79	0,028174	2,329105
12	0,03349	3,956473	46	0,034253	3,732466	80	0,043878	2,724829
13	0,045396	3,181219	47	0,035452	2,610205	81	0,039743	1,942959
14	0,033817	4,182405	48	0,039871	3,552253	82	0,041292	2,245287
15	0,040894	2,632453	49	0,047662	1,367506	83	0,042835	2,750104
16	0,035324	3,264164	50	0,025507	4,793415	84	0,049133	3,076152
17	0,036584	1,926236	51	0,031476	3,57673	85	0,035545	2,477095
18	0,030266	2,683396	52	0,04453	2,21595	86	0,036573	3,020124
19	0,030641	4,637471	53	0,040921	2,66494	87	0,04114	2,317859
20	0,04358	3,193266	54	0,035816	4,597099	88	0,043534	2,4034
21	0,037106	3,089448	55	0,039865	2,57324	89	0,031392	2,057052
22	0,032983	1,950344	56	0,032074	3,967723	90	0,034291	2,719511
23	0,033157	3,228414	57	0,031856	1,966477	91	0,037601	4,294171
24	0,042751	2,747081	58	0,039268	3,771905	92	0,025354	2,690111
25	0,03704	2,355365	59	0,047352	4,047775	93	0,044781	1,538782
26	0,040488	2,76778	60	0,042008	2,03076	94	0,031832	4,977276
27	0,039756	5,844767	61	0,034518	2,037591	95	0,044745	3,415907
28	0,042052	2,113837	62	0,042909	1,771183	96	0,033047	3,756672
29	0,039779	1,515875	63	0,035947	3,429255	97	0,041262	1,549427
30	0,030303	1,309313	64	0,033359	2,265309	98	0,042476	4,020024
31	0,039622	2,001249	65	0,034858	2,717931	99	0,036354	2,243404
32	0,044547	1,890872	66	0,033955	1,814073	100	0,040799	1,694015
33	0,041872	1,683731	67	0,036166	3,810206			
34	0,025775	6,717481	68	0,040797	1,959553			

Table C.1 contains the 100 *atracurium* models used to build the bank $\mathcal{P}$ for the *atracurium*

drug. From this bank $\mathcal{P}$ the total mass system control law was tuned for each model leading to a bank $\mathcal{K}$ with total mass system controllers for *atracurium*.

C.2 Bank of *Rocuronium* Models

Table C.2: Bank $\mathcal{P}$ of 41 *rocuronium* models

Model	Alpha	Gamma	Model	Alpha	Gamma
1	0,054415	1,269884	22	0,054164	1,663375
2	0,038213	1,7086	23	0,068183	1,482369
3	0,058058	1,446245	24	0,078358	1,414203
4	0,010252	3,79457	25	0,031073	1,024188
5	0,025788	4,447869	26	0,031736	6,324634
6	0,065598	1,793064	27	0,052867	1,608939
7	0,036549	2,626389	28	0,081144	1,291639
8	0,034114	2,480481	29	0,027712	1,057524
9	0,093514	1,462093	30	0,033816	1,392472
10	0,030824	2,024059	31	0,044985	1,546866
11	0,070261	1,562036	32	0,028992	0,854272
12	0,020998	6,456394	33	0,02354	1,49399
13	0,031634	3,699414	34	0,027631	3,67053
14	0,05355	2,074754	35	0,030673	1,310781
15	0,081388	0,986952	36	0,039037	1,602996
16	0,060556	1,601377	37	0,051239	1,056976
17	0,030002	6,916275	38	0,038301	0,998537
18	0,045182	2,178904	39	0,062052	1,482878
19	0,037963	2,395231	40	0,030789	3,556661
20	0,095264	5,022601	41	0,03945	1,146665
21	0,091863	1,748396			

Table C.3: Bank $\mathcal{P}$ of 37 *rocuronium* models

Model	Alpha	Gamma	Model	Alpha	Gamma
1	0,054415	1,269884	20	0,078358	1,414203
2	0,038213	1,7086	21	0,031073	1,024188
3	0,058058	1,446245	22	0,031736	6,324634
4	0,065598	1,793064	23	0,052867	1,608939
5	0,036549	2,626389	24	0,081144	1,291639
6	0,034114	2,480481	25	0,027712	1,057524
7	0,093514	1,462093	26	0,033816	1,392472
8	0,030824	2,024059	27	0,044985	1,546866
9	0,070261	1,562036	28	0,028992	0,854272
10	0,031634	3,699414	29	0,02354	1,49399
11	0,05355	2,074754	30	0,027631	3,67053
12	0,081388	0,986952	31	0,030673	1,310781
13	0,060556	1,601377	32	0,039037	1,602996
14	0,030002	6,916275	33	0,051239	1,056976
15	0,045182	2,178904	34	0,038301	0,998537
16	0,037963	2,395231	35	0,062052	1,482878
17	0,091863	1,748396	36	0,030789	3,556661
18	0,054164	1,663375	37	0,03945	1,146665
19	0,068183	1,482369			

Table C.2 contains the 41 models obtained from a set of real data acquired during 41 surgeries. For each of these real cases, the parameters α and γ were identified, and the corresponding models were taken as the bank for *rocuronium* drug. The models 4, 5, 12, and 20 (red models in table C.2) were discarded because no good NMB control can be to obtain with the switching strategy.

This led to table C.3. These models were used to build the bank $\mathcal{P}$ for the *rocuronium* drug. From this bank $\mathcal{P}$ the total system mass control law was tuned for each model leading to a bank $\mathcal{K}$ with total system mass controllers for *rocuronium*.

Appendix D

Controller Choice (in Section 3.3.5)

Table D.1: Controllers chosen by different metrics for *atracurium* models

Patient Model	Switching	Ref Track	Alpha	Gamma	Uss	Norm2	Vinnicombe	Impulse	Step
1	39	39	92	81	81	92	92	92	92
2	44	44	96	44	44	96	96	96	96
3	40	9	6	9	9	6	6	6	6
4	59	59	69	59	59	69	69	69	69
5	53	53	11	53	53	11	11	11	11
6	40	40	83	40	40	83	83	83	83
7	10	71	3	10	10	3	3	3	3
8	41	29	83	29	29	83	83	83	83
9	39	63	41	63	63	41	41	41	41
10	31	31	82	31	31	82	82	82	82
11	7	71	17	71	71	17	17	17	17
12	55	55	5	55	55	5	5	5	5
13	19	19	1	19	19	1	1	1	1
14	90	90	65	90	90	65	65	65	65
15	46	52	52	46	46	52	52	52	52
16	69	69	46	69	69	46	46	46	46
17	21	80	85	80	80	85	85	85	85
18	52	52	29	91	91	29	29	29	29
19	53	38	29	38	38	29	29	29	29
20	13	13	87	13	13	87	87	87	87
21	42	74	73	74	74	73	73	73	73
22	56	80	95	80	80	95	95	95	95
23	16	16	95	20	16	95	95	95	95
24	82	82	10	82	82	10	10	10	10

25	73	73	73	78	73	73	73	73	73
26	82	82	67	82	82	67	67	67	67
27	43	43	80	43	43	80	80	80	80
28	59	88	59	88	88	59	59	59	59
29	92	92	27	92	92	27	27	27	27
30	41	8	18	8	8	18	18	18	18
31	4	4	80	4	4	80	80	80	80
32	10	10	51	10	10	51	51	51	51
33	99	99	59	99	99	59	59	59	59
34	27	27	49	27	27	49	49	49	49
35	95	57	53	95	95	53	53	53	53
36	62	62	84	62	62	84	84	84	84
37	44	44	57	44	44	57	57	57	57
38	30	30	60	8	8	60	60	60	60
39	53	53	77	53	53	77	77	77	77
40	1	1	48	1	1	48	48	48	48
41	6	65	24	65	65	24	24	24	24
42	8	30	9	30	30	9	9	9	9
43	85	83	55	85	85	55	55	55	55
44	93	93	25	93	93	25	25	25	25
45	37	37	71	37	37	71	71	71	71
46	35	95	89	95	95	89	89	89	89
47	15	15	36	15	15	36	36	36	36
48	50	50	54	50	50	54	54	54	54
49	42	42	58	42	42	58	58	58	58
50	93	19	91	19	19	91	91	91	91
51	48	48	88	48	48	88	88	88	88
52	72	72	32	72	72	32	32	32	32
53	18	18	15	18	18	15	15	15	15
54	39	39	35	39	39	35	35	35	35
55	47	47	48	47	47	48	48	48	48
56	12	68	56	12	12	56	56	56	56
57	22	22	93	67	67	93	93	93	93
58	35	35	37	35	35	37	37	37	37
59	97	97	49	97	97	49	49	49	49
60	4	4	28	4	4	28	28	28	28
61	4	60	38	60	60	38	38	38	38
62	6	65	82	6	6	82	82	82	82
63	36	36	54	36	36	54	54	54	54
64	98	9	12	9	9	12	12	12	12
65	89	79	61	89	89	61	61	61	61
66	41	41	14	41	41	14	14	14	14
67	95	58	98	58	58	98	98	98	98
68	80	57	99	57	57	99	99	99	99
69	56	97	51	56	56	51	51	51	51

70	16	16	4	16	16	4	4	4	4
71	72	72	32	72	72	32	32	32	32
72	7	11	45	11	11	45	45	45	45
73	52	71	33	71	71	33	33	33	33
74	87	87	25	87	87	25	25	25	25
75	83	83	8	83	83	8	8	8	8
76	87	84	62	87	87	62	62	62	62
77	78	64	78	9	9	78	78	78	78
78	90	5	90	90	90	90	90	90	90
79	77	77	77	86	86	77	77	77	77
80	65	89	20	89	89	20	20	20	20
81	68	22	27	22	22	27	27	27	27
82	1	1	96	1	1	96	96	96	96
83	24	24	10	24	24	10	10	10	10
84	75	75	8	75	75	8	8	8	8
85	76	76	36	76	76	36	36	36	36
86	43	43	17	43	43	17	17	17	17
87	25	79	2	79	79	2	2	2	2
88	74	74	20	74	74	20	20	20	20
89	61	61	51	61	61	51	51	51	51
90	65	80	46	65	65	46	46	46	46
91	14	78	78	14	78	78	78	78	78
92	18	18	50	18	18	50	50	50	50
93	96	96	94	96	96	94	94	94	94
94	50	50	57	50	50	57	57	57	57
95	63	63	93	63	63	93	93	93	93
96	35	35	22	35	35	22	22	22	22
97	93	93	82	93	93	82	82	82	82
98	59	59	41	59	59	41	41	41	41
99	82	82	67	82	82	67	67	67	67
100	33	33	68	33	33	68	68	68	68

Table D.2: Controllers chosen by different metrics for *rocuronium* models

Patient Model	Switching	Ref Track	Alpha	Gamma	Uss	Norm2	Vinnicombe	Impulse	Step
1	23	23	17	23	23	17	17	17	17
2	31	16	33	16	16	33	33	33	33
3	34	6	12	6	6	12	12	12	12
4	2	16	18	16	16	18	18	18	18
5	5	5	15	5	5	15	15	15	15
6	5	5	25	15	15	25	25	25	25
7	19	3	16	3	3	16	16	16	16
8	14	10	35	10	10	35	35	35	35
9	12	26	18	26	26	18	18	18	18
10	35	29	21	29	29	21	21	21	21
11	14	8	17	8	8	17	17	17	17
12	33	33	23	33	33	23	23	23	23
13	22	22	34	31	31	34	34	34	34
14	21	21	30	21	21	30	30	30	30
15	8	8	26	11	11	26	26	26	26
16	6	6	2	6	6	2	2	2	2
17	4	4	7	2	2	7	7	7	7
18	22	2	1	2	2	1	1	1	1
19	34	34	9	34	34	9	9	9	9
20	7	25	23	25	25	23	23	23	23
21	33	33	8	33	33	8	8	8	8
22	14	14	10	14	14	10	10	10	10
23	13	31	11	31	31	11	11	11	11
24	1	1	12	30	30	12	12	12	12
25	32	32	29	32	32	29	29	29	29
26	30	20	6	20	20	6	6	6	6
27	23	9	15	9	9	15	15	15	15
28	12	12	14	12	12	14	14	14	14
29	31	19	29	34	34	29	29	29	29
30	10	10	25	10	10	25	25	25	25
31	26	24	35	24	24	35	35	35	35
32	23	23	36	13	13	36	36	36	36
33	25	25	23	25	25	23	23	23	23
34	21	12	2	12	12	2	2	2	2
35	19	19	13	19	19	13	13	13	13
36	10	30	8	30	30	8	8	8	8
37	25	33	32	25	1	32	32	32	32

Appendix E

Real Case Reports

The reports from the real cases acquired during surgeries are developed by the software *Monitoring and Control in Anesthesia* (see Appendix B). These reports provide a summary from the surgeries where the switching strategy was applied, once the *Gender*, *Age*, *Height* and *Weight* from the patient, the name of the surgery, the protocol used by the anesthesiologist, and the controllers used during the surgery can be promptly seen, and any variable acquired with the software *Datex SAD - Data Acquisition* (see Appendix B) during the surgery can be plotted. Also, this report allows an access to all the information collected in the Special Notes Tab of the *Datex SAD - Data Acquisition* software (for more details see Appendix B).

This tool is very important to provide quick information, easily interpreted by the user. The whole surgery can be easily summarized and, most important, the NMB level control can be easily explained and described through these reports.

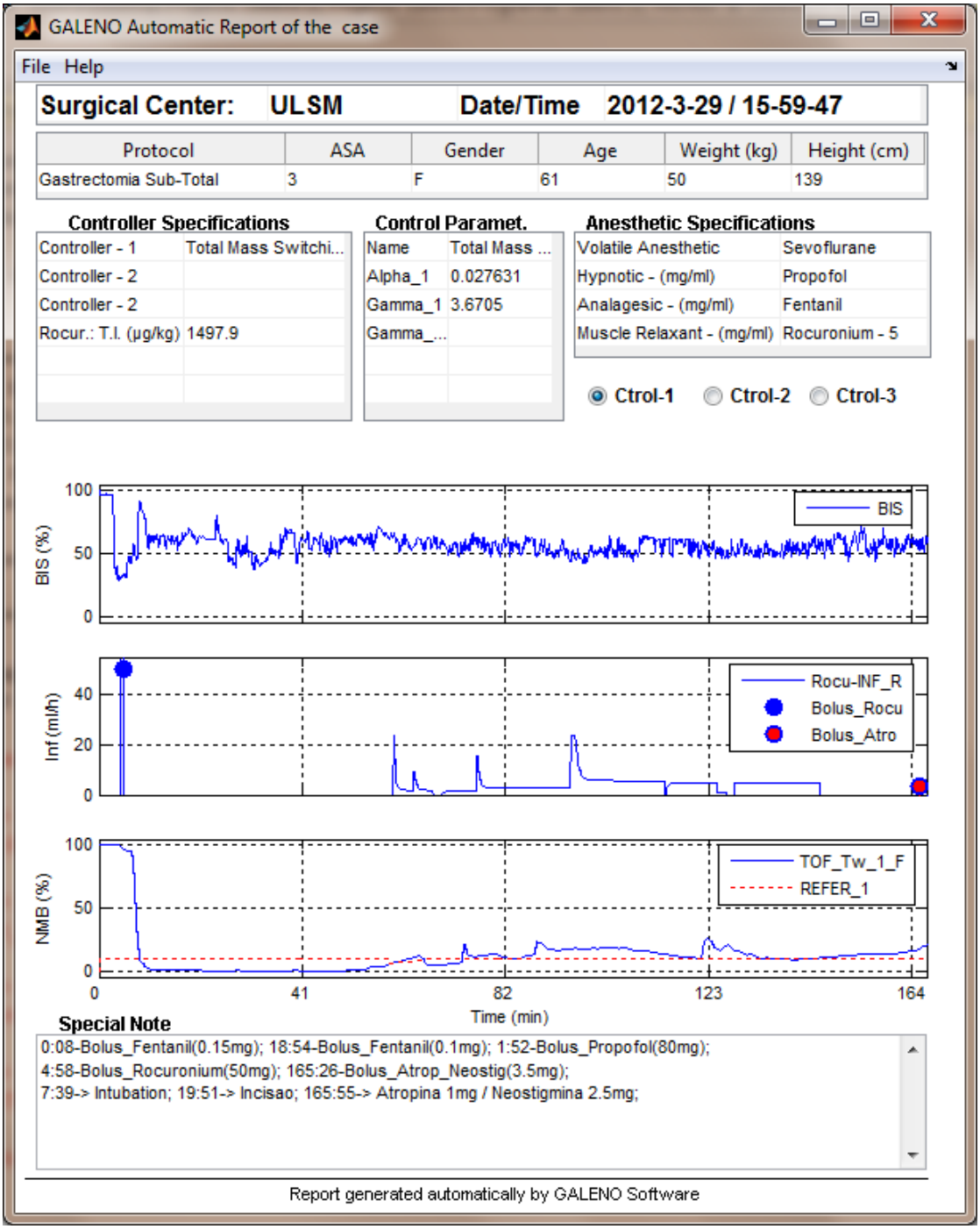

Figure E.1: Report from a real case acquired at ULSM-HPH (2012/03/29 Case 2)

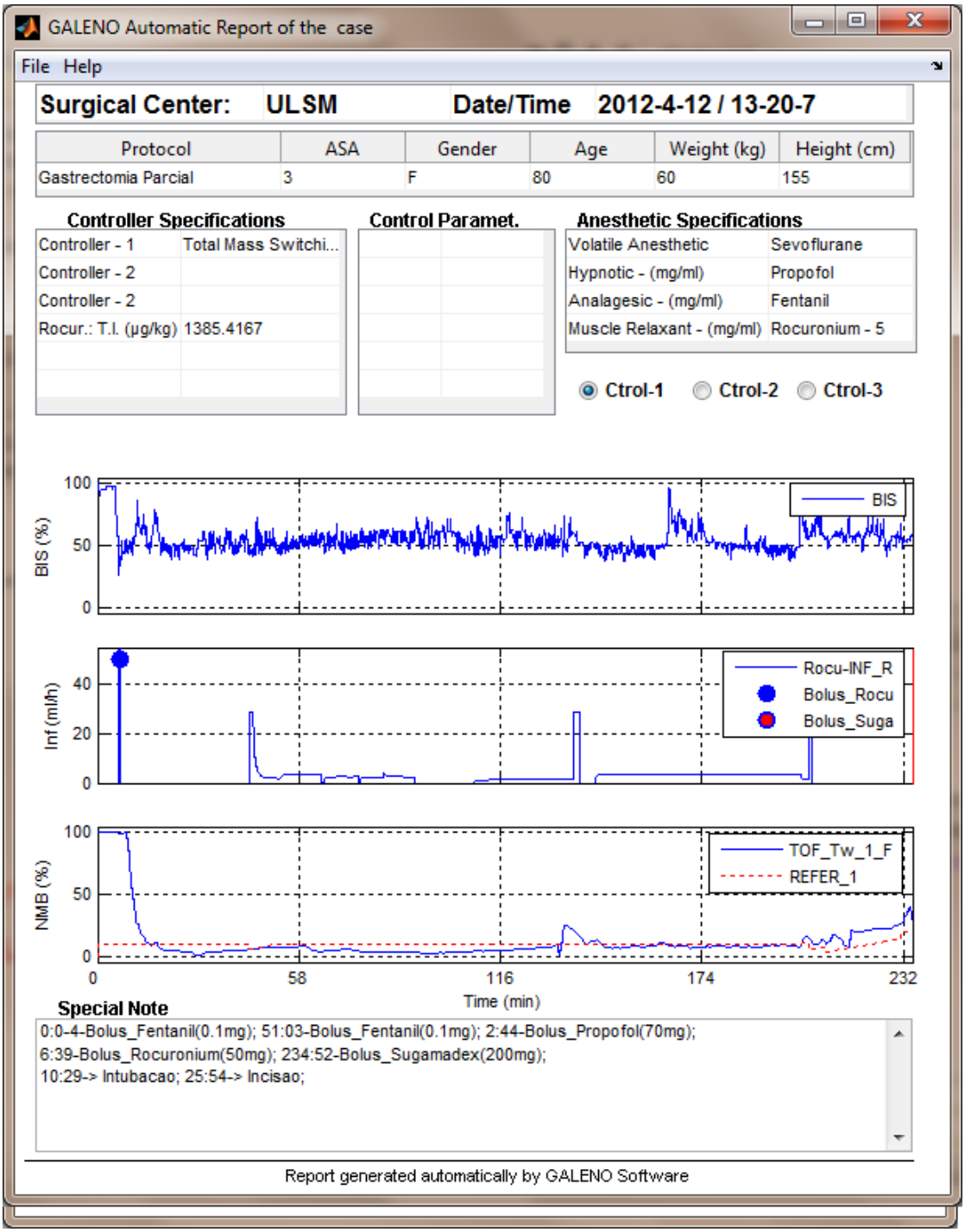

Figure E.2: Report from a real case acquired at ULSM-HPH (2012/04/12 Case 1).

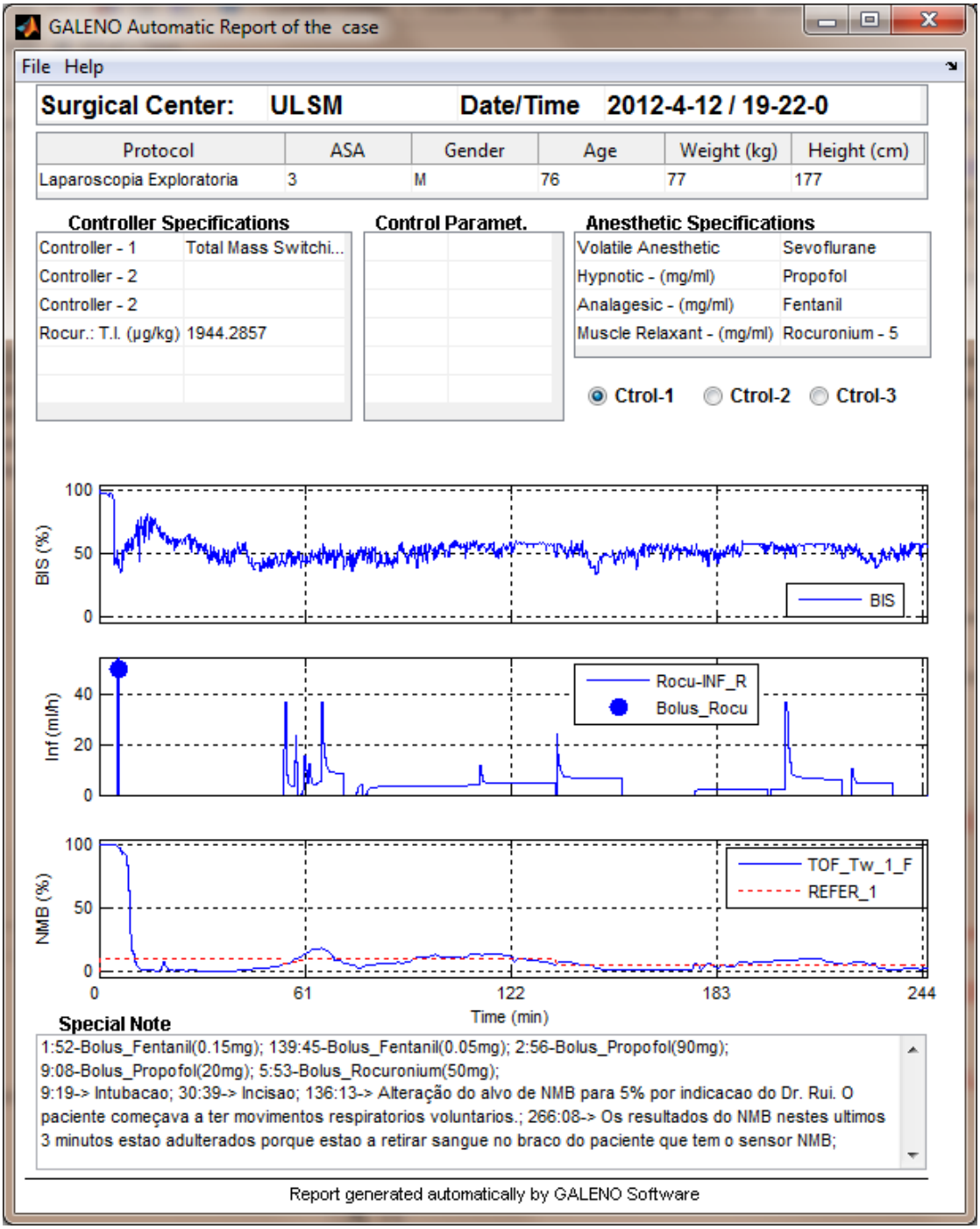

Figure E.3: Report from a real case acquired at ULSM-HPH (2012/04/12 Case 2).

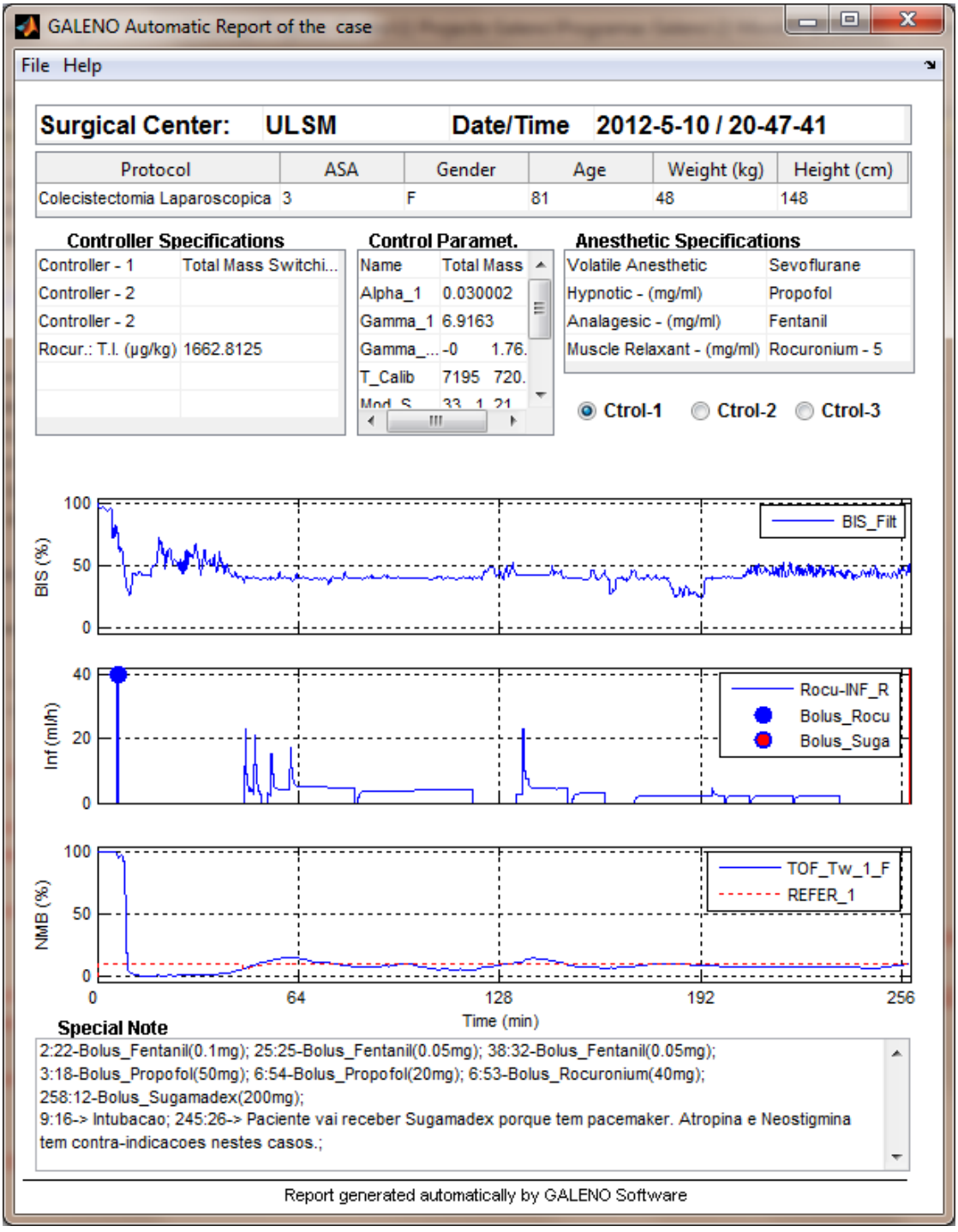

Figure E.4: Report from a real case acquired at ULSM-HPH (2012/05/10 Case 2).

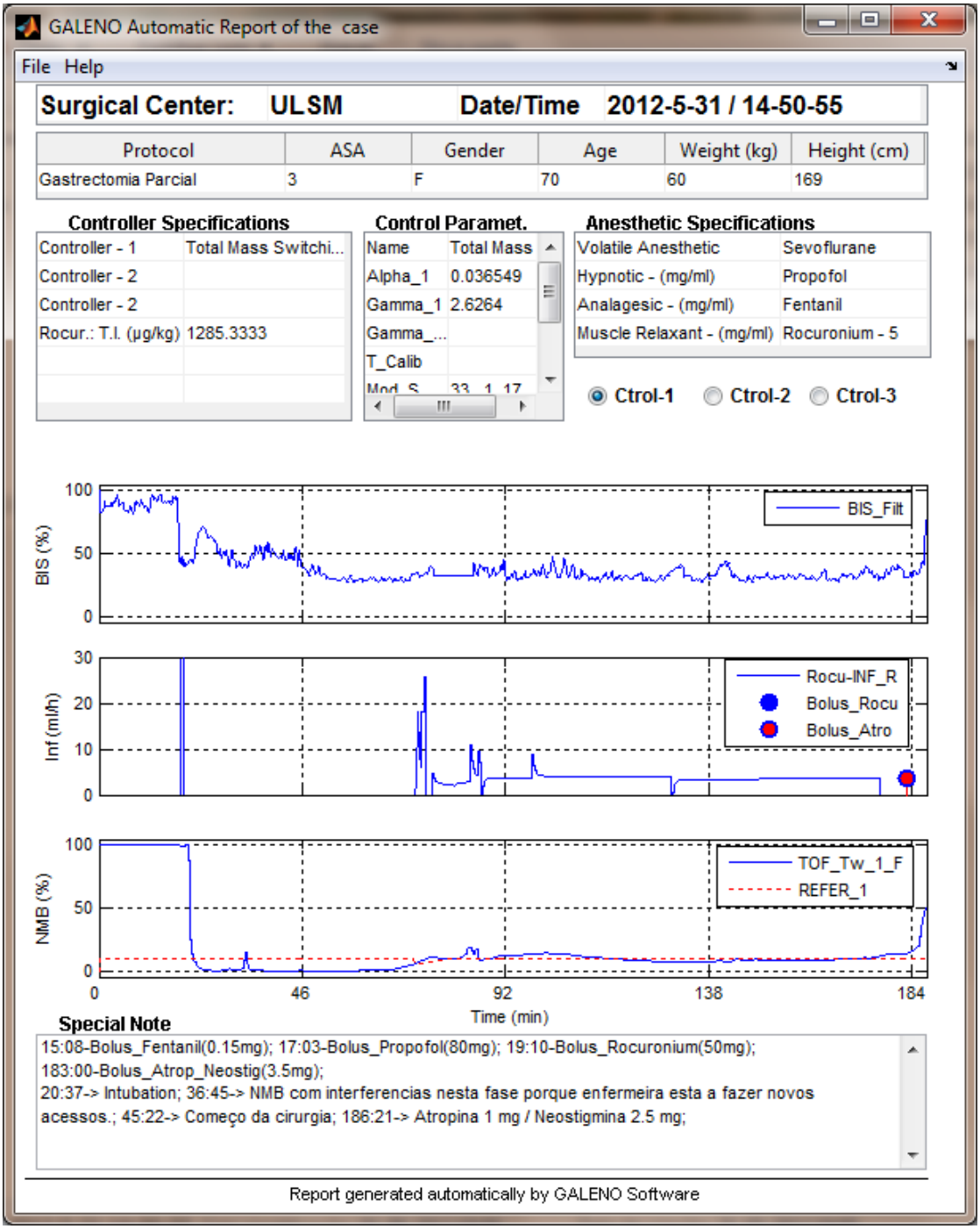

Figure E.5: Report from a real case acquired at ULSM-HPH (2012/05/31 Case 1).