

Supervised Multi-model Adaptive Control of Neuromuscular Blockade with Off-set Compensation

Pedro Oliveira, João P. Hespanha, João M. Lemos and Teresa Mendonça

Abstract—The paper considers the control of neuromuscular blockade (NMB) induced by *atracurium* on patients subject to general surgery. The high levels of uncertainty in the process model are tackled with supervised multi-model adaptive control, for which an innovative supervisor based on least squares estimation is presented to deal with constant unknown offsets in output measurement. In order to avoid excessive switching among the controllers, a model clustering is made, with a common controller in each cluster. Furthermore, an anti-windup algorithm that may be applied to an integrator that is common to a bank of controllers is proposed. The control structure is tested on simulation using realistic nonlinear models for NMB with parameters estimated from actual patient data.

Index Terms—Adaptive Control, Biomedical Systems, Anaesthesia, Computer Control, Control Applications.

I. INTRODUCTION

A major difficulty in the control of the variables involved in general anaesthesia of patients undergoing surgery is the high level of uncertainty in the models relating the manipulated and process variables. Although the structure of the models is relatively well known, amounting to pharmacokinetic/pharmacodynamic (PK/PD) models of Wiener type [1], [2], there is a strong parameter variability from patient to patient that suggests the use of adaptive methods thereby yielding a performance increase while stabilizing different dynamics. In this paper, the control of neuromuscular blockade (NMB) level is considered [3].

Although a variety of adaptive algorithms may be considered, some of which have been successfully tried in anaesthesia, either in simulations or actual clinical cases [4], [5], [6], this paper focus its attention on the class of Supervised Multi-Model Adaptive Controllers (SMMAC). This approach is not new [7], [8] (see a tutorial on [9]), but continues to attract attention, both for theoretical extensions [10], [11], [12], [13] and applications [14], [15]. In what concerns neuromuscular blockade control, successful clinical results of the application of variants of SMMAC are reported in [16], [17]. In this respect, a number of issues remain open, including the methodology for finite controller covering, the design of local controllers and the existence of measure offsets.

The work reported in this paper was made in the framework of project *IDEA – Integrated Design for Anaesthesia Automation*, financed by FCT Portugal under contract PTDC/EEA-ACR/69288. All correspondence regarding this paper should be sent to J. M. Lemos, INESC-ID, R. Alves Redol 9, 1000-029 Lisboa, Portugal, jlm1@inesc.pt. P. Oliveira (pedro.s.oliveira@ist.utl.pt) and J. M. Lemos are with INESC-ID/IST/UTL, Lisboa Portugal. J. P. Hespanha hespanha@ece.ucsb.edu is with University of California at Santa Barbara, USA. T. Mendonça tmendo@fc.up.pt is with Faculdade de Ciências da Universidade do Porto, Portugal.

The main contribution of this paper consists in an innovative supervisor algorithm based on mean squares minimization and its demonstration on neuromuscular blockade. Other contributions include the adaptation of the wind-up algorithm to a single integrator connected to multiple controllers and the use of clustering algorithms to apply SMMAC to NMB.

The paper is organized as follows: After the Introduction in which the problem is formulated and the state of the art reviewed, the neuromuscular blockade pharmacokinetic/pharmacodynamic model for *atracurium* is presented in section II. Section III describes the control algorithm, including the new supervisor for dealing with constant offsets and section IV illustrates its performance by simulations with the nonlinear neuromuscular blockade model. Finally, section V presents a final discussion and draws conclusions.

II. NEUROMUSCULAR BLOCKADE

The dynamic response of the neuromuscular blockade for *atracurium* may be modeled [2], [3] using a compartmental model, resulting in the Wiener type model shown in Figure 1. Here, the linear pharmacokinetic model with transfer function

$$G_{PK}(s) = \frac{a_1}{\tau_1 s + 1} + \frac{a_2}{\tau_2 s + 1} \quad (1)$$

relates the drug infusion rate $u(t)$ [$\mu\text{g kg}^{-1} \text{min}^{-1}$] with the plasma concentration $c_p(t)$ [$\mu\text{g ml}^{-1}$], where a_i [$\text{kg ml}^{-1} \text{min}^{-1}$] and τ_i [min] ($i = 1, 2$) are patient dependent parameters.

The pharmacodynamic part comprises a linear model that relates the plasma concentration with the so called effect concentration $c_e(t)$, described by the transfer function

$$G_{PD}(s) = \frac{1}{(\tau_3 s + 1)(\tau_4 s + 1)} \quad (2)$$

and a nonlinear static relation between the effect concentration and the NMB level $r(t)$ [%], assumed to be given by the Hill equation, written as

$$r(t) = \frac{100C_{50}^\gamma}{C_{50}^\gamma + c_e^\gamma(t)} \quad (3)$$

where parameters τ_i [min] ($i = 3, 4$), C_{50} [$\mu\text{g ml}^{-1}$] and γ (dimensionless) are also patient-dependent. The variable $r(t)$, normalized between 0 and 100, measures the level of the neuromuscular blockade, 0 corresponding to full paralysis and 100 to full muscular activity.

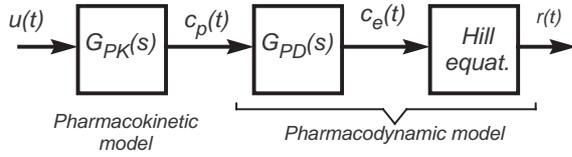


Fig. 1: Block diagram of the neuromuscular blockade model.

III. SUPERVISED MULTI-MODEL ADAPTIVE CONTROL

Figure 2 shows the basic architecture employed by the supervisory multi-model adaptive control (SMMAC) structure. It consists of a bank of N_c controllers and a supervisor. The supervisor decides at each time what is the index ϕ of the controller to actually apply to the plant (the patient). This decision is based on a multiple model based estimator that compares the outputs of a bank of models with the actual plant dynamics. The index of the model that best fits the observed plant dynamics is then used to select the controller to apply.

Although the original approaches used as many controllers as models, it is also possible to associate one controller to a class of models, thereby reducing excessive switching that would lead to performance degradation.

In order to ensure stability, a dwell time condition [7] is used. This amounts to impose that, once a controller is applied to the plant, it will remain so for at least a minimum period of time.

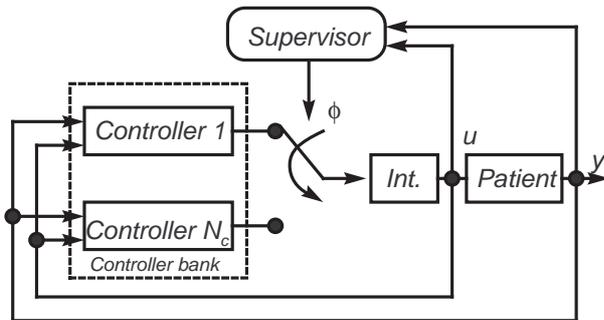


Fig. 2: Supervisory multi-model adaptive control structure.

Hereafter, the different elements of SMMAC are described in detail, in what concerns its specificities relating the problem at hand here.

A. Local controller design

Each controller C_i in the bank of controllers is designed by pole placement, using a polynomial approach [18]. This is a well known design algorithm, briefly sketched below for the sake of completeness. Let the patient dynamics be described, around an operation point, by the linear incremental model

$$A(q)r(k) = B(q)u(k) \quad (4)$$

where A and B are coprime polynomials in the forward shift operator q and k is an integer that denotes discrete time. The admissible control laws are given by

$$R(q)u(k) = T(q)r^c(k) - S(q)r(k) \quad (5)$$

where $R(q)$, $S(q)$ and $T(q)$ are polynomial in the forward-shift operator. The problem to solve consists in designing a causal control law such that the closed loop dynamics is given by

$$A_m(q)r(k) = B_m(q)u(k) \quad (6)$$

with A_m and B_m conveniently specified.

In order to solve this problem, perform the factorizations $A = A^+A^-$ and $B = B^+B^-$ with A^+ and B^+ monic and hurwitz. The controller is then obtained by the solution of the following diophantine equation with respect to the unknowns \bar{R} and \bar{S} :

$$A^-R_d\bar{R} + B^-S_d\bar{S} = A_mA_0^c \quad (7)$$

subject to the following order constraints

$$\begin{cases} \partial\bar{S} = \partial A^- + \partial R_d - 1 \\ \partial S = \partial R = \partial A^- + \partial A^+ + \partial S_d + \partial R_d - 1 \\ \partial\bar{R} = \partial R - \partial B^+ - \partial R_d \\ \partial A_0^c = 2\partial A^- - \partial A_m + \partial A^+ + \partial S_d + \partial R_d - \partial B^+ - 1 \end{cases} \quad (8)$$

The term $R_d = q - 1$ includes an integral action to help rejecting steady state error and constant disturbances. The term $S_d = q + 1$ corresponds to a pole in the Nyquist frequency that ensures that measurement noise does not generate large signals [18]. The polynomials defining the controller are then computed by

$$\begin{cases} R = B^+\bar{R}R_d \\ S = A^+\bar{S}S_d \\ T = \bar{B}_mA^+A_0^c \end{cases} \quad (9)$$

B. Anti-windup algorithm

In order to ensure smooth switching between controllers the integrator is common to all controllers. Since the actuator has saturation limits (the pump usually operates between (0 and 20 ml/h) an anti-wind-up mechanism has to be included. The algorithm used is a simple extension to multiple controllers of the one presented in [18]. Adding $A_{aw}(q)u(k)$ in both members of (5), where $A_{aw}(q)$ is the characteristic polynomial of the anti-windup observer, results in

$$\begin{cases} A_{aw}v = Tr^c - Sr + (A_{aw} - R)u \\ u = \text{sat}(v) \end{cases} \quad (10)$$

where v is the control signal before the saturation. Figure 3 shows the anti-windup scheme with an integrator common to all controllers. When the control signal is not saturated the anti-windup scheme reduces to an integrator.

C. Model clustering

According to the approach followed [10], the number of controllers in the controller bank is much smaller than the number of possible models. The association of a set of models to one controller (clustering) may be done in

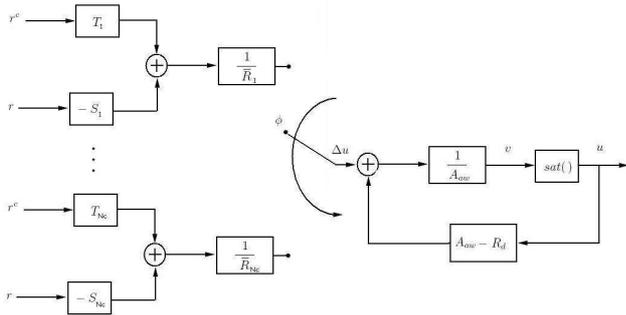


Fig. 3: Block diagram of the anti-windup scheme.

several ways. Since the controller associated to each cluster of models must stabilize all the models in the cluster, this is taken into account. For that sake, when a distance between models is required, use is made of the v -gap metric [10]. This metric is such that, if two models are close, a controller that stabilizes one model in closed loop is likely to stabilize the other as well.

Three model clustering methods are considered:

- 1) An "heuristic" method based on specific structure of NMB;
- 2) A clustering method based on a modification of the k -means algorithm [19];
- 3) A clustering method based on the Complete Link algorithm for graphs [19].

The data used consists in the parameters estimated from patient data. This set contains 100 models that are representative of a population of patients undergoing elective surgery [3].

The so called "heuristic" method explores known facts about the response of NMB to a bolus [3]. The NMB response induced by a *bolus* administration in the induction phase varies significantly from patient to patient, but may be characterized, in a simplified way, by two features. One of these features is $T10 - T80$, the difference between the time instants for the time response of the NMB level r to be at $r = 10\%$ and $r = 80\%$ after a bolus. The other, P , provides a measure of how long does the drug remain effective in the patient, and is defined as the amount of time the NMB level is below 5%. As shown in [3], these features are correlated with acceptable (stabilizing) gains for a PID controller and therefore it is expected that they provide significant information on controller design. This means that it is expected that, if two models are close in the space of these features, then they admit a common stabilizing controller that yields a closed-loop performance that is similar in both cases. Off-course this is a merely heuristic argument that had to be verified by simulation. Hence, clusters have been build by trial and error in the space of the features $T80 - T10$ and P . These clusters can be seen in figure 4 for a partition with 12 classes. The dashed lines in red represent the class separation and the * are the centroids, chosen such that the corresponding controller stabilizes every

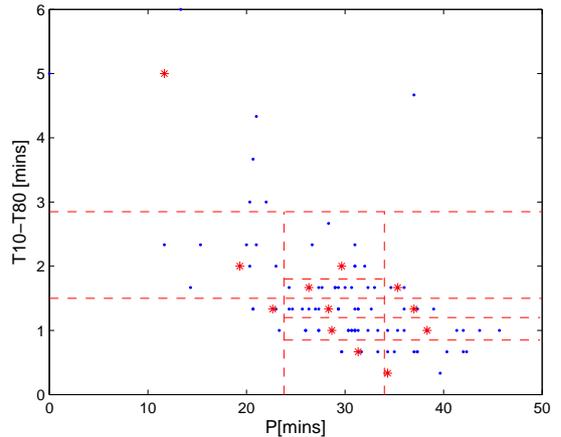


Fig. 4: Clustering with 12 classes using $T80 - T10$ and P .

model of the class.

The second method is the well known k -means modified such that, instead of using random initializations, it starts at the cluster obtained by the heuristic method.

The last method was proposed in order to overcome the need to start from an initial approximative solution obtained from problem knowledge. For that sake the Complete Link algorithm was used [19].

With these methods, 6 controller banks were build, one for each clustering methods and number of classes (either 7 or 12, resulting from a refinement of the partition with 7 classes).

D. Supervisor

The supervisor proposed in [7] consists of the following elements:

- Bank of model based output estimators;
- Prediction error performance index computation;
- Switching logic.

For ease of reference, this supervisor will hereafter be named PEPI (Prediction Error Performance Index).

The model bank is made from N models M_j , each represented by the ARX model

$$A_j(q^{-1})y_j(k) = B_j(q^{-1})u(k) + \bar{e}_j(k), \quad (11)$$

in which

$$A_j(q^{-1}) = 1 + \sum_{i=1}^{n_a} a_{j,i}q^{-i} \quad B_j(q^{-1}) = \sum_{i=1}^{n_b} b_{j,i}q^{-i}$$

are polynomials in the unit delay operator q^{-1} , with A_j monic and B_j of fixed degrees n_a and n_b respectively, for all $j = 1, \dots, N$. These models are obtained from the linearization of the Wiener NMB model described in section II. From these models, the estimator for the output assuming model M_j is given by

$$\hat{y}_j(k) = (A_o^p - A_j) \frac{1}{A_o^p} y(k) + \frac{B_j}{A_o^p} u(k). \quad (12)$$

where $A_o^p(q^{-1}) = 1 + \sum_{i=1}^{n_a} a_{o,i}q^{-i}$ is a monic hurwitz polynomial with the same degree as the A_j 's, that will hereafter be referred as the prediction observer polynomial. If a dead-beat ("fast") observer is selected (all the roots of A_o^p at the origin) no filtering occurs. As confirmed by the experiments below, a much better choice consists in making the observer dynamics much slower, by selecting the roots of A_o^p in the real segment between 0 and 1, but away from 0. The importance of the selection of A_o^p when controlling the NMB level is shown, through actual clinical results in [17].

For each model M_j , the prediction error, e_{p_j} is by subtracting \hat{y}_j from y_j . A performance index $\pi_{p_j}, j = 1, \dots, N$, is then computed by low pass filtering according to

$$\pi_{p_j}(k) = \lambda_p \pi_{p_j}(k-1) + (1 - \lambda_p) e_{p_j}^2(k), \quad (13)$$

where λ_p is a parameter that can be either constant or variable. The variable solution aims at adjusting λ_p as a function of a measure of the input excitation according to the following rationale: when the control error (it could be the prediction error as well) is below a specified threshold, the adaptation is suspended and $\lambda_p = 1$, if not the λ_p is set to $\lambda_0 = 0.85$ which implies a faster adaptation.

Since the discrete models $M_j = (A_j, B_j)$ are linearized around a working point, they are affected by a constant (in practice exactly unknown) off-set. A solution to this problem with the PEPI estimator is to differentiate the control and NMB signals before entering the supervisor. This has the disadvantage of increasing the noise level. Another possibility is explained hereafter.

E. Dealing with constant offsets

In order to derive a Least Mean Squares estimator (LMS) based supervisor able to deal with unknown constant offsets, let the linearized patient model be written as

$$\begin{cases} As(k) = Bu(k) \\ y(k) = s(k) + b \end{cases} \quad (14)$$

where A and B are unknown polynomials that define the patient model and b is an unknown but constant bias term (only y can be measured, not s). The following predictor is asymptotically correct for the model (14)

$$\hat{s} = \left(1 - \frac{A}{\omega}\right)s + \frac{B}{\omega}u \quad (15)$$

where ω is any hurwitz polynomial. From (14) and (15)

$$\hat{y} = \hat{s} + b = \left(1 - \frac{A}{\omega}\right)s + \frac{B}{\omega}u + b \quad (16)$$

Let the multiple models be defined by

$$\begin{cases} A_i s_i = B_i u \\ y_i = s_i + b_i \end{cases} \quad (17)$$

For these models, we can use predictors of the form

$$\hat{y}_i = \left(1 - \frac{A_i}{\omega}\right)s + \frac{B_i}{\omega}u + b_i \quad (18)$$

Since s is not available, it is necessary to express the predictors in terms of the measured output y . To this effect, replace s by $y - b_i$ which is correct for the actual model:

$$\hat{y}_i = \left(1 - \frac{A_i}{\omega}\right)y + \frac{B_i}{\omega}u + \frac{A_i}{\omega}b_i \quad (19)$$

Hence the prediction error is given by

$$\hat{y}_i - y = -\frac{A_i}{\omega}(y - b_i) + \frac{B_i}{\omega}u \quad (20)$$

This predictor has the key property that if there is model matching (in the sense that $A_i = A, B_i = B$ e $b_i = b$) one has

$$\hat{y}_i - y = -\frac{A_i}{\omega}(y - b_i) + \frac{B_i}{\omega}u = -\frac{A}{\omega}s + \frac{B}{\omega}u \rightarrow 0$$

Once obtained an expression for the predictor, the optimization problem is formulated. At time K_f one seeks to find the model that minimizes

$$\arg \min_{1 \leq i \leq N} \sum_{k=0}^{K_f} (\hat{y}_i(k) - y(k))^2 \quad (21)$$

To avoid having one model for each value of b_i , one can perform the minimization over b_i in closed form. To do this, linearity is used in order to generate the output estimation error as follows:

$$\hat{y}_i - y = \alpha_i + b_i \beta_i \quad (22)$$

Comparing (20) and (22) it is concluded that

$$\alpha_i := -\frac{A_i}{\omega}y + \frac{B_i}{\omega}u \quad \beta_i := -\frac{A_i}{\omega}\mathbf{1} \quad (23)$$

where $\mathbf{1}$ denotes the unit step function. This allows to express model selection as

$$\arg \min_{1 \leq i \leq N} \sum_{k=0}^{K_f} (\alpha_i(k) + b_i \beta_i(k))^2 \quad (24)$$

or equivalently

$$\begin{aligned} & \arg \min_{1 \leq i \leq N} \left(\sum_{k=0}^{K_f} \alpha_i^2(k) \right) + \\ & + 2b_i \left(\sum_{k=0}^{K_f} \alpha_i(k) \beta_i(k) \right) + b_i^2 \left(\sum_{k=0}^{K_f} \beta_i^2(k) \right) \end{aligned} \quad (25)$$

The minimization over all $b_i \in \mathbb{R}$ can be done in closed form:

$$b_i^{opt} = -\frac{\sum_{k=0}^{K_f} \alpha_i(k) \beta_i(k)}{\sum_{k=0}^{K_f} \beta_i(k)^2} \quad (26)$$

Substituting the optimal solution (26) in the model selection equation (25), results in

$$\arg \min_{1 \leq i \leq N} \sum_{k=0}^{K_f} \alpha_i^2(k) - \frac{\left(\sum_{k=0}^{K_f} \alpha_i(k) \beta_i(k) \right)^2}{\sum_{k=0}^{K_f} \beta_i(k)^2} \quad (27)$$

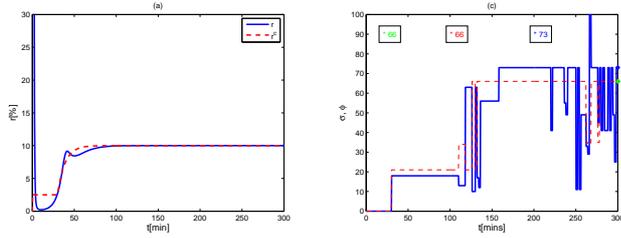


Fig. 5: Simulation results obtained for M_{56} with $\tau_D = 5$. (a) - NMB signal (b) - switching (σ) and class index (ϕ) signals.

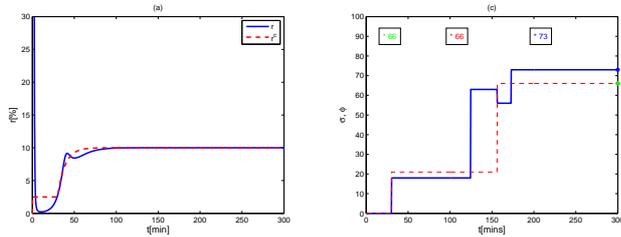


Fig. 6: Simulation results obtained for M_{56} with $h = 1$. (a) - NMB signal (b) - switching (σ) and class index (ϕ) signals.

IV. SIMULATION RESULTS

A number of simulations are shown hereafter that illustrate the results obtained with SMMAC including the options explained above. In these simulations an initial *bolus* of $500 \mu kg^{-1}$ is applied at the beginning. As a consequence, the NMB index decreases to a value between 0% and 5%, depending on the patient model. In this initial phase, the system is almost saturated and therefore the controller is only started after 30 min .

A. Dwell-time vs. Hysteresis

In the first example the switching logic based on dwell-time is compared with the one that uses hysteresis [15]. In both cases, the PEPI supervisor is used. Figure 5, shows the results with *dwell-time*, and figure 6 shows the results when *hysteresis* is used to implement the decision logic. Patient model M_{56} is used. Although the input/output performance is similar, hysteresis leads to a much lower number of switchings, which is an advantage.

B. Constant vs. variable prediction filter

This example illustrates the advantage of using a variable performance index computing filter. The PEPI supervisor is used. The simulation results for the case with constant and variable λ_p are shown in Figures 7 and 8, respectively. With a constant λ_p the adaptation was too sluggish, leading to a settling time much slower than the one obtained with variable λ_p .

C. LMS estimator

Figure 9 shows a typical result obtained with the LMS estimator. The LMS based estimator usually yields a good

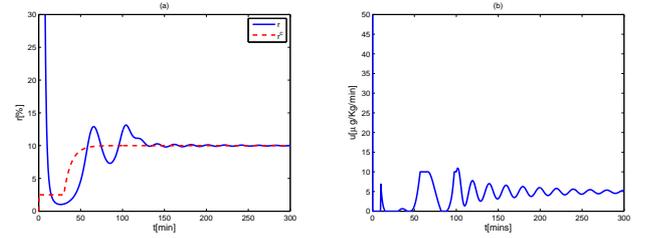


Fig. 7: Simulation results obtained for M_{69} with constant $\lambda_p = 0.975$. (a) - NMB signal (b) - manipulated variable (c) - switching (σ) and class index (ϕ) signals.

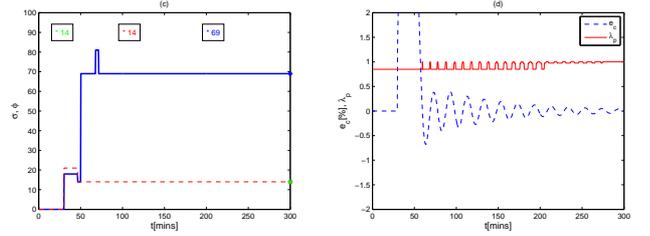
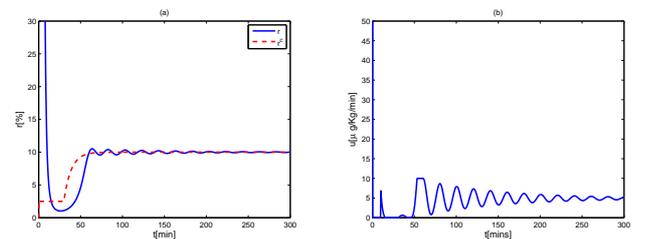


Fig. 8: Simulation results obtained for M_{69} with variable λ_p , with $\lambda_0 = 0.85$ and $e_c(t)$ threshold of 0.1%. (a) - NMB signal (b) - manipulated variable (c) - switching (σ) and class index (ϕ) signals, (d) - $\lambda_p(t)$ as a function of $e_c(t)$ variation.

performance and a reduced number of switchings. In the example showed, the patient response was good despite the switching controller did not converge to the controller of the patient's class.

D. Complete simulation results

Table I, shows performance parameters for the different structures considered. These performance parameters are the number of situations in which the controller converged to the "correct" one, the initial *Overshoot*, the settling time, t_s , between $\pm 1\%$ and the mean squared error at steady state ($t > 75 \text{ min}$).

Both the PEPI and LMS based supervisors yielded good results. In general, the PEPI estimator worked better with 7 controllers, in opposition the LMS estimator that showed to perform better with more controllers.

V. DISCUSSION AND CONCLUSIONS

The SMMAC structure proposed solves a number of difficulties in the control of NMB for patients subject to general anaesthesia. In particular, a supervisor based on

TABLE I: Performance parameters the control systems implemented.

Control System	converging cases	Overshoot [%]		t_s [min]		MSE [%]	
		avg	worst case	avg	worst case	avg	worst case
intuitive-7-PEPI	63	2.84	35	65.22	96	0.0205	0.34
intuitive-7-LMS	27	4.62	39	68.24	286	0.0523	1.6
k-means-7-PEPI	69	2.25	35	66.65	298	0.3098	29.3
k-means-7-LMS	33	4.39	54	67.09	250	0.2804	8.35
agglomerative-7-PEPI	26	23.41	99	110.82	300	11.47	114
agglomerative-7-LMS	34	3.72	36	68.11	197	0.1222	4.47
intuitive-12-PEPI	14	7.09	45	112.07	300	0.7415	8.75
intuitive-12-LMS	14	2.88	40	59.96	286	0.0277	1.4
k-means-12-PEPI	8	4.25	50	86.47	300	0.6008	15.9
k-means-12-LMS	12	3.11	45	61.61	157	0.0707	4.43
agglomerative-12-PEPI	20	9.51	64	107.7	300	1.5882	17.6
agglomerative-12-LMS	22	11.04	99	91.34	300	1.4983	13.2

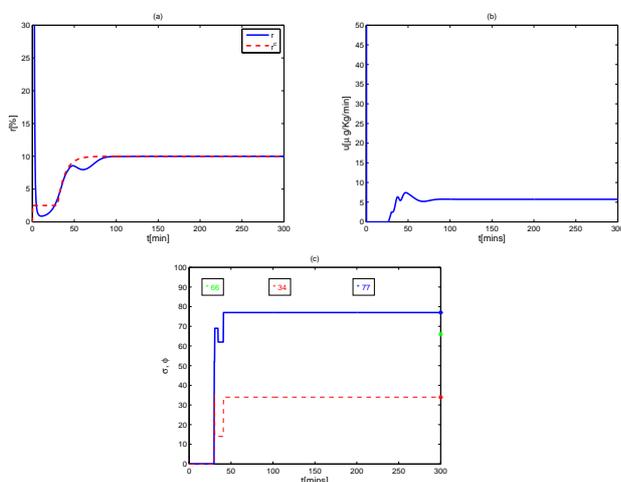


Fig. 9: Simulation results obtained for M_{85} with the LMS estimator. (a) - NMB signal (b) - manipulated variable (c) - switching (σ) and class index (ϕ) signals.

least mean squares solved the problem of the existence of unknown offsets. Comparing it with existing algorithms based on model performance evaluation, it is observed that a reduced number of switches occurs.

A final difficulty is smooth start-up. Due to nonlinearity embodied in the Hill equation, the NMB level becomes saturated for a period after the initial drug bolus. In order to start SMMAC with unacceptable start-up transients, a proportional controller is first applied for some minutes. In order to switch to SMMAC, during 5 samples the state of SMMAC should be initialized with a value consistent with the control yielded by the proportional controller.

REFERENCES

[1] J. Bailey and W. Haddad (2005). Drug dosing in clinical pharmacology: paradigms, benefits, and challenges, *IEEE Control Syst. Mag.*, 25(2):35-51.
 [2] B. Weatherley and S. Williams and E. Neill (1983). Pharmacokinetics, Pharmacodynamics and Dose-Response Relationships of *Atracurium* Administered i. v., *Br. J. Anaesth.*, 55:39s-45s.

[3] P. Lago and T. Mendonça and L. Gonçalves, "On-line autocalibration of a PID controller of neuromuscular blockade", in *Proc. of the 1998 IEEE International Conference on Control Applications*, Trieste, Italy, 1998, pp. 363-367.
 [4] Jaklitsch, R. and D. Westenskov (1987). A model based self adjusting two-phase controller for *vecuronium*-induced muscle relaxation during anesthesia. *IEEE Trans. Biom. Eng.*, BME-34:583-594.
 [5] Kansanaho, M. and K. Olkkola (1996). Performance assessment of an adaptive model-based feedback controller: Comparison between *atracurium*, *mivacurium*, *rocuronium* and *vecuronium*. *J. Clin. Monitor., Comput.*, 13(4):217-224.
 [6] Y. Sawaguchi, E. Furutani, G. Shirakami, M. Araki and K. Fukuda (2008). A model-predictive hypnosis control system under total intravenous anesthesia. *IEEE Trans. Biomed. Eng.*, 55(3):874-887.
 [7] S. Morse (1996). Supervisory control of families of linear set-point controllers - Part 1: Exact matching, *IEEE Trans. Automat. Contr.*, 41(10):1413-1431.
 [8] K. Narendra and J. Balakrishnan (1997). Adaptive control using multiple models, *IEEE Trans. Automat. Contr.*, 42(2):171-187.
 [9] J. P. Hespanha (2001). Tutorial on Supervisory Control. *Lecture notes for the workshop Control Using Logic and Switching for the 40th IEEE CDC, Orlando, FL*, available at (<http://www.ece.ucsb.edu/hespanha/techreps.html>).
 [10] B. Anderson, T. Brinsmead, F. De Bruyne, J. Hespanha, D. Liberzon and A. S. Morse (2000). Multiple Model Adaptive Control. Part 1: Finite Controller Coverings. *Int. J. Robust Nonlinear Control*, 10:909-929.
 [11] B. Anderson, T. Brinsmead, F. De Bruyne, J. Hespanha, D. Liberzon and A. S. Morse (2001). Multiple Model Adaptive Control. Part 2: Switching. *Int. J. Robust Nonlinear Control*, 11:479-496.
 [12] Mosca, E. and T. Agnoloni (2001). Inference of candidate loop performance and data filtering for switching supervisory control. *Automatica*, 37:527-534.
 [13] D. Angeli and E. Mosca (2002). Lyapunov-based switching supervising control of nonlinear uncertain systems. *IEEE Trans. Autom. Control*, 47:500-505.
 [14] B. Chadouri, R. Majumder and B. Pal (2004). Application of multiple-model adaptive control strategy for robust damping of interarea oscillations in power systems. *IEEE Trans. Control Syst. Tech.*, 12:727-736.
 [15] J. Böling, D. Seborg and J. Hespanha. Multi-model adaptive control of a simulated pH neutralization process. *Control Eng. Practice*, doi:10.1016/j.conengprac.2006.11.008, 2007.
 [16] J. M. Lemos and H. Magalhães and T. Mendonça and R. Dionísio, Control of neuromuscular blockade in the presence of sensor faults, *IEEE Trans. Biomed. Eng.*, vol. 52(11), 2005, pp 1902-1911.
 [17] T. Mendonça, H. Magalhães, J. M. Lemos, P. Rocha and S. Esteves (2007). Multiple model adaptive control of neuromuscular blockade: Design guidelines and clinical cases. *Proc. ECC'07*, Kos, Greece.
 [18] K.J. Åström and B. Wittenmark, (1997). *Computer-Controlled Systems: Theory and Design*, Prentice Hall, New Jersey, USA.
 [19] R. Duda, P. Hart and D. Stork (2001). *Pattern Classification*, John Wiley & Sons Inc., New York.