Control of Neuromuscular blockade with Gaussian Process Models

João M. Lourenço  João M. Lemos  Jorge S. Marques

July 5, 2011
This work has been supported by FCT-Portugal under project GALENO PTDC/SAU-BEB/103667/2008,
Abstract

This paper presents Nonlinear Model Predictive Control (NMPC) of neuromuscular blockade induced by atracurium on patients subject to general anaesthesia. In order to tackle the high levels of uncertainty in the process behavior, probabilistic and non-parametric gaussian process models are used in the NMPC approach. The control structure was tested in wide range of patients with extreme behaviors and has shown to very reliable as it presents very satisfactory performances for all of them.
## Contents

0.1 Introduction .................................................. 1  
0.2 Gaussian Processes ........................................... 2  
    0.2.1 GP Modeling Procedure Summary ..................... 6  
0.3 Model Predictive Control .................................... 7  
0.4 Neuromuscular Blockade Model .............................. 8  
0.5 Simulation Results .......................................... 10  
    0.5.1 GP Model ............................................ 10  
    0.5.2 Control ............................................. 14  
0.6 Conclusions ................................................ 22
0.1 Introduction.

This work presents a study on the use of Gaussian process models to implement a Nonlinear Model Predictive Control algorithm in automatic drug delivery for inducing anesthesia, particularly muscular relaxation, in human patients.

A high degree of uncertainty is present in the dynamics of biological systems due to its nonstationary, nonlinear behavior and inter and intra individual variability. In fact, a large variability of the responses to the infusion of atracurium has been observed. The Gaussian process (GP) models provide a flexible, probabilistic and nonparametric approach to tackle the high uncertainty of these nonlinear systems. With this methodology we obtain a predictive distribution of the output (instead of a point prediction). The predicted output is the mean of the distribution and the uncertainty of the prediction is given by its variance [23]. The prediction value variance, that depends on the data density and noise, can be viewed as the prediction level of confidence and is the principal advantage of this method when compared to neural network or fuzzy models. Another advantage of this kind of model is that its structure is determined only by the selection of the covariance function and the regressors. However, the GP model suffers from high computational cost due to the need to invert the covariance matrix, whose dimensions depend on the size of the data set and the number of regressors, at every iteration of the optimization algorithm. Some applications of the GP model to nonlinear system identification are [7, 10, 3, 11, 21, 9, 19].

The GP model was used with model predictive control since this control algorithm does not require a specific model form. Along with its predictive features, MPC also allows to include constraints and feedforward from accessible disturbances, key aspects for the control of anaesthesia. Some applications of Model Predictive Control based on Gaussian process models are described in [12, 14, 8, 4].
0.2. GAUSSIAN PROCESSES

The main contribution of this paper is to present a study in the application of Nonlinear Model Predictive Control to neuromuscular blockade, with atracurium as blocking agent, based on Gaussian process models.

This work is organized as follows: Dynamic gaussian processes models are described in section 2. Section 3 introduces the Model Predictive Control algorithm principle and in section 4 the empirical model for neuromuscular blockade is presented. Section 5 illustrates, the GP modeling of neuromuscular blockade and the simulations results obtained for Model Predictive Control based on GP model. Conclusions are summarized at the end of the paper in section 6.

0.2 Gaussian Processes

A Gaussian process is a collection of random variables, which have a joint multivariate Gaussian distribution. Assuming a relationship of the form \( y = f(x) \) between an input \( x \in \mathbb{R}^D \) and output \( y \in \mathbb{R} \), the output can be viewed as a collection of random variables \( y(1), \ldots, y(n) \sim N(0, \Sigma) \) which have a joint multivariate Gaussian distribution. The covariance matrix \( \Sigma \) can be parameterized and computed by means of a function \( \Sigma_{pq} = \text{Cov}(y(p), y(q)) = C(x(p), x(q)) \) that determines the covariance between output points corresponding to input points \( x(p) \) and \( x(q) \). The Gaussian process can be fully specified by its mean \( \mu(x) \) (usually assumed to be zero) and its covariance function \( C(x(p), x(q)) \). It is remarked that, although not all data can be modeled as a zero-mean process, this assumption is correct if the data is properly scaled and detrended [5].

A common choice of covariance function, when we assume that the process is stationary (the mean is constant and the covariance function only depends on the distance between the inputs \( x(i) \)), that has been proven to work well in practice
Control of Neuromuscular Blockade

[23] is:

\[
C(x(p), x(q)) = \nu_1 \exp \left[ -\frac{1}{2} \sum_{d=1}^{D} w_d (x^d(p) - x^d(q)) \right] + \nu_0 \delta(p, q) \tag{1}
\]

where \(x^d(p)\) denotes the \(d^{th}\) component of the \(D\)-dimensional input vector \(x(p)\), \(\Theta = [\nu_1, w_1, \ldots, w_D, \nu_0]^T\) is the vector of hyperparameters and \(\delta(p, q)\) is the Kronecker operator defined as

\[
\delta(p, q) = \begin{cases} 
1 & p = q \\
0 & p \neq q 
\end{cases} \tag{2}
\]

The \(w_1, \ldots, w_D\) parameters control the scaling of the distances in each input dimension \(x^1(i), \ldots, x^D(i)\). The parameter \(\nu_1\) is the overall scale of correlations and \(\nu_0\) expresses the process noise variance. The exponential term suggests that less distant input vectors lead to highly correlated outputs while more distant inputs generate low correlated outputs. The \(w_1, \ldots, w_D\) parameters can be used to evaluate the relative importance of the corresponding input components (dimensions), i.e., a high or low \(w_i\) value means that the inputs in dimension \(i\) contain high or low information, respectively. There are other forms of covariance functions that can be chosen [1], the only restriction being that these covariance functions must generate non-negative definite covariance matrices, for any set of input points.

Assume a statistical model

\[
y(k) = f(x(k)) + \epsilon(k) \tag{3}
\]

with an additive uncorrelated Gaussian white noise with variance \(\nu_0, \epsilon \sim N(0; \nu_0)\).

Given a set of training data pairs of input data \(X = [x(1), x(2), \ldots, x(n)]\) and the corresponding vector of output data \(y = [y(1), y(2), \ldots, y(n)]\) and a GP prior on \(f(x)\), with zero-mean and Gaussian covariance function such as (1) we wish to get the predictive distribution of \(y(n + 1)\) corresponding to a new given input...
For the random variables $y(1), \ldots, y(n), y(n+1)$ we can write:

$$y, y(n+1) \sim N(0, K_{n+1}),$$

(4)

where $K_{n+1}$ is the covariance matrix made of submatrices as follows:

$$K_{n+1} = \begin{bmatrix}
K & k(x(n+1)) \\
[k(x(n+1))^T] & [K(x(n+1))]
\end{bmatrix}.$$

(5)

The matrix $K$ is the $n \times n$ covariance matrix for the training data such that

$$K = \Sigma_{pq} = C(x(p), x(q)),$$

(6)

and the vector

$$k(x(n+1)) = [C(x(1), x(n+1)), \ldots, C(x(n), x(n+1))]^T$$

(7)

is the $n \times 1$ vector of covariances between the training inputs and the new input. The expression

$$K(x(n+1)) = C(x(n+1), x(n+1))$$

(8)

is the autocovariance of the new input.

The conditional distribution of (4) allows to obtain the predictive distribution of $y(n+1)$, which is also Gaussian [22],

$$P(y(n+1)|y, X, x(n+1)) \sim N(\mu(x(n+1)), \sigma^2(x(n+1))),$$

(9)

where $\mu(x(n+1))$ and $\sigma^2(x(n+1))$ are the mean and variance of the gaussian predictive distribution, and are given by:

$$\mu(x(n+1)) = k(x(n+1))^T K^{-1} y$$

(10)

$$\sigma^2(x(n+1)) = K(x(n+1)) - k(x(n+1))^T K^{-1} k(x(n+1)).$$

(11)
We may say that, given a new input vector $x(n + 1)$, the predicted model output $\hat{y}(n + 1)$ is the mean of the gaussian distribution, i.e., $\hat{y}(n + 1) = \mu(x(n + 1))$ and the uncertainty of this prediction is given by the variance of the gaussian distribution $\sigma^2(x(n + 1))$. The predictive mean (10) can be interpreted as a weighted sum of the training outputs $y$, to make a prediction at the test point $x(n + 1)$.

To be able to make predictions, based on (10), the vector of hyperparameters $\Theta$ has to be provided either as prior knowledge or estimated from the available data. This may be done by performing the maximization of the log-likelihood of the hyperparameters:

$$L(\Theta) = \log P(y/X) = -\frac{1}{2} \log(\det K) - \frac{1}{2} y^T K^{-1} y - \frac{n}{2} \log(2\pi).$$

This optimization requires the computation of the derivative of $L(\Theta)$ with respect to each parameter $w_1, \ldots, w_D, \nu_0, \nu_1$ of the vector of hyperparameters $\Theta$, which is given [16] as

$$\frac{\partial L(\Theta)}{\partial \Theta_j} = -\frac{1}{2} \text{tr} \left[ K^{-1} \frac{\partial K}{\partial \Theta_j} \right] + \frac{1}{2} y^T K^{-1} \frac{\partial K}{\partial \Theta_j} K^{-1} y,$$

where $\text{tr}$ denotes the trace.

Gaussian processes can be used for the modeling of dynamic systems of the form (3) if delayed input and output signals are used as regressors [6], [10]

$$x(k) = [y(k-1), y(k-2), \ldots, y(k-L), u(k-1), u(k-2), \ldots, u(k-L)].$$

In such cases an autoregressive model is considered, such that the current output depends on previous outputs $y$, as well as on previous control inputs, up to a given lag $L$.

The Gaussian process model not only describes the dynamic characteristics of the system but also provides information about the confidence in the predictions. This advantage can be used to point out predictions of poor quality, indicated by the corresponding variance high values.
The multiple-step-ahead predictions can be performed by repeating iteratively one-step ahead predictions up to a desired step ahead (iterative method). This method can be performed by feeding back to the input of the GP model only the mean values of the predicted output (naive approach), or the complete output distributions (exact approach), together with the future control inputs in both approaches. In the naive approach, a $l$-step-ahead prediction, knowing the $L$ past input and output signals as well as the future $l$ control input values, can be obtained if at time steps $k, \ldots, k + l$ the following regressors are feed into the GP model:

$$
\begin{align*}
\mathbf{x}(k) &= [y(k-1), \ldots, y(k-L), u(k-1), \ldots, u(k-L)] \\
\mathbf{x}(k+1) &= [\hat{y}(k), y(k-1), \ldots, y(k-L+1), u(k), \ldots, u(k-L+1)] \\
&\vdots \quad \vdots \\
\mathbf{x}(k+l) &= [\hat{y}(k+l-1), \hat{y}(k+l-2), \ldots, y(k+l-L), u(k+l-1), \\
&\quad \ldots, u(k+l-L)],
\end{align*}
$$

where $\hat{y}$ is the predicted model output which is computed with (10).

The naive approach does not account for the uncertainty generated by each intermediate step prediction giving worse results especially in terms of predicted variance which is usually narrower than the obtained by the exact approach [2], [3]. More on the GP model simulation and differences of approaches can be found in [6], [10] and [11].

### 0.2.1 GP Modeling Procedure Summary

Assume that a training data set is known and consists of a $n \times d$ matrix $\mathbf{X}$ of input measurements and a $n \times 1$ vector $\mathbf{y}$ of output or target values. Use the training data set to develop a model that can be used to make prediction with new data. Assume that the new data, called the testing data, is given by an $1 \times d$ input vector.
The control of neuromuscular blockade is crucial in various medical applications. The goal is to predict the value of $y^*$ given $X, y,$ and $x^*$.

In the Gaussian process approach, the prediction of $y^*$ involves the selection of a positive semidefinite covariance function $C(x(i), x(j))$, where $x(i)$ and $x(j)$ are vectors with $d$ components. The covariance function can be used to construct the $n \times n$ covariance matrix $K$ with entries $K_{i,j} = C(x(i), x(j))$, where $x(i)$ and $x(j)$ are rows of $X$, and also the $1 \times n$ cross covariance vector $k$ with entries $k_j = C(x^*, x(j))$. The prediction $\hat{y}^*$ of $y^*$ is obtained with the Gaussian processes equation (10). Besides the prediction $\hat{y}^*$, the Gaussian process approach also leads to the prediction estimated variance (11), where $K(x^*) = C(x^*, x^*)$.

Note that the hyperparameters of the covariance function are not known in advance and they can be estimated via the maximization of the log-likelihood (12) using the training data.

For further details, a presentation of Gaussian processes can be found in [18].

### 0.3 Model Predictive Control

The Model Predictive Control (MPC) algorithm used [15], aims at minimizing in discrete time and in a receding horizon sense the quadratic cost,

$$ J(k, k + T_p) = \sum_{i=1}^{T_p} \|\hat{y}(k + i) - r(k + i)\|^2_Q + \sum_{i=1}^{T_u} \|u(k + i - 1)\|^2_R $$

(16)

where $k$ is current time, $\hat{y}(k + i)$ is the $i$-step ahead prediction, $r$ is the reference to track, $T_p$ is the prediction horizon, $T_u$ is the control horizon, and $Q$ and $R$ are error and control effort weighting matrices, respectively. In the previous equation, $\|x\|^2_B = x^TBx$, where $B$ is symmetric and positive definite. The quadratic cost function (16) is minimized subjected to constraints imposed, among others, upon...
0.4. NEUROMUSCULAR BLOCKADE MODEL

Control of Neuromuscular Blockade

by saturations of actuators and their rates of change, and output magnitude:

\[ u_{\text{min}} \leq u(k) \leq u_{\text{max}} \]  \hspace{1cm} (17)

\[ \Delta u_{\text{min}} \leq \Delta u(k) \leq \Delta u_{\text{max}} \]  \hspace{1cm} (18)

\[ y_{\text{min}} \leq y(k) \leq y_{\text{max}} \]  \hspace{1cm} (19)

The predictive control principle can be summarized as follows: At each current time \( k \), the process output \( y(t + k) \) is predicted over a time horizon \( k = 1, \ldots, T_p \). The predicted values are indicated by \( \hat{y}(t + k) \) and the value \( T_p \) is called the prediction horizon. The prediction is done by a model of the process, in our case a GP model, and it is assumed that this model is available. The prediction depends on the past inputs and outputs, but also on the future control signals \( u(k + i), k = 1, \ldots, T_u \) (i.e. the control actions which is intended to be applied from a moment \( k \) onwards); At each time \( k \), among the admissible control scenarios, a set of future control signals is chosen \( u(k + i), k = 1, \ldots, T_u \) by optimization of a given criterion, depending on the predicted control errors \( [\hat{y}(k+i) - r(k+i)] \), \( k = 1, \ldots, T_p \). In most of optimization criteria the control effort is also included. The resulting control \( u(k) \) is then applied to the process, but only at instant time \( k \). According to a receding horizon strategy at the next sampling period a new measured output sample is available and the whole procedure is repeated.

0.4 Neuromuscular Blockade Model

The dynamic response of the Neuromuscular Blockade Model (NMBM) induced by a bolus (i.e., a sudden infusion performed with a syringe) of atracurium may be modeled, as shown in Fig. 1, by a wiener structure [20], [17]. The linear compartmental pharmacokinetic model relates the drug infusion rate \( u(t) \) \([\mu g kg^{-1} min^{-1}]\) with the plasma concentration \( c_p(t) \) \([\mu g ml^{-1}]\), and may be described by the state
Control of Neuromuscular Blockade

Figure 1: Block diagram of the neuromuscular blockade model.

\begin{align*}
\dot{x}_1 &= -\lambda_1 x_1(t) + a_1 u(t) \quad (20) \\
\dot{x}_2 &= -\lambda_2 x_2(t) + a_2 u(t) \quad (21) \\
c_p(t) &= \sum_{i=1}^{2} x_i(t) \quad (22)
\end{align*}

where $\lambda_i [min^{-1}]$, and $a_i [Kg\,ml^{-1}]$ ($i = 1, 2$) are patient-dependent parameters. The pharmacodynamic effect for *atracurium*, relates the plasma concentration $c_p(t)$ to the level of NMB (normalized between 0 and 100, with 0 corresponding to full paralysis and 100 to normal muscular activity), $r(t)$ (in percent). This involves a nonlinearity described by the Hill equation

$$r(t) = \frac{100C_{50}^{\beta}}{C_{50}^{\beta} + C_e^{\beta}(t)},$$

and the relation between the plasma concentration $c_p(t)$ and the effect compartment concentration $c_e(t) [\mu g\,ml^{-1}]$,

\begin{align*}
\dot{c}(t) &= -\lambda c(t) + \lambda c_p(t) \quad (24) \\
\dot{c}_e(t) &= -\frac{1}{\tau}c_e(t) + \frac{1}{\tau}c(t), \quad (25)
\end{align*}

where $\tau [min]$, $\lambda [min^{-1}]$, $C_{50} [\mu g\,ml^{-1}]$, and $\beta$ (dimensionless) are all patient-dependent parameters. The equation (25) was included to allow a better clinical data replication [13]. To cover a wide range of neuromuscular blockade behaviors, a set of non-linear dynamic models, $M_j$, $j = 1, \ldots, 100$, has been generated (see Figure 2) using the probabilistic model for *atracurium* discussed previously [13].
Figure 2: One hundred simulated (noise free) responses induced by a bolus of 500 $\mu g Kg^{-1}$, at $t = 0$, of atracurium. Responses in black represent the models used to test the control algorithm.

The patient behaviors are characterized by a large dynamic variability, which creates the need for a control strategy with a high degree of reliability and robustness.

0.5 Simulation Results

0.5.1 GP Model

As already mentioned in section 0.3, to use a MPC strategy a prediction model is necessary. In order to be useful for the control strategy the model should be capable of representing the large uncertainty of the neuromuscular blockade dynamic.
Figure 3: Simulated responses induced by an atracurium bolus of 500 $\mu g Kg^{-1}$ at $t = 0$ for 5 patients. In the left is shown a zoom of the neuromuscular blockade for $t \leq 8$.

After some simulations, using a trial and error approach, the model $M_{94}$ was chosen to generate the identification data, needed for the learning of the gaussian process model. In Figure 3, where are represented the extremes ($M_{12}$ and $M_{69}$) and the average ($M_{41}$ and $M_{44}$) neuromuscular blockade dynamic behaviors when excited by a bolus of 500$\mu g Kg^{-1}$. We can see that the model $M_{94}$ has an almost central dynamic behavior. For the modeling of a system, it is desirable to be able to predict the response to inputs that cover most or all of the operating range of the system.

To obtain the identification data, the NMB model was initially excited by a bolus of 500$\mu g Kg^{-1}$ and after that by a pseudo random binary signal with amplitude
values between 0 and 11, multiplied by a random number with normal distribution. Figure 4-a) shows the simulated neuromuscular blockade response and the excitation signal that induced it. The output signal is contaminated with normally distributed measurement noise. The maximum limit of the infusion rate $u(t)$ was set as 20 $\mu gKg^{-1}min^{-1}$ considering the limit of the infusion pumps used for the model data collection and in use in the operating room. To avoid excessive computation time (the number of samples used in the identification determines the dimension of the covariance matrix) the identification data is only made of 250 samples.

The regressors (which together with the covariance function are the only choices that have to be made for the GP model), and the hyperparameters of the covariance
function were selected to maximize the likelihood of the training data by trial and error between numerous alternatives, being

$$\mathbf{x}(k) = [y(k-1), y(k-2), y(k-3), u(k-1), u(k-2)]$$

(26)

the vector of regressors that shown better results. As covariance function was chosen (1). The corresponding vector of hyperparameters, is given by:

$$\Theta = [w_1, \ldots, w_D, \nu_1, \nu_0]^T$$

(27)

$$= 10^3[0.0237, 0.3456, 0.1047, 1.2014, 0.5897, 6.6691, 0.0013]$$

(28)

The maximum likelihood approach was used to determine the hyperparameters. A conjugate gradient optimization algorithm with line searches [7] was used to search for their optimal values.
0.5. SIMULATION RESULTS

The response of the Gaussian process model to the validation signal is shown in Figure 4-b). In the same Figure it is shown the input data that is different from the data used in the learning, although they are of the same kind in two sections, namely, shortly after 200 samples and just before the 400 samples. The shaded band represents a 95% confidence level of the GP model predicted mean. In the range of the operating space there is a very good adequation between the prediction and the actual values, with most of the actual values lying within the prediction confidence region. Regions with wider confidence bar indicate that the model is less certain about its prediction.

The GP model was also validated for a predictive horizon of length 6, since the model is intended for predictive control purposes. Figure 5 shows Gaussian process (obtained with $M_{94}$) six-step-ahead prediction, the input data, and the $M_{94}$ and $M_{69}$ responses for comparison. The results show a good adequation between the prediction and the actual values for both simulations. Also in these simulations most of the actual values lie within the prediction confidence region (more than 85% of the values). The six-step-ahead predictions were obtained using the naive approach that produces narrower predicted variance regions than the obtained with the exact approach.

0.5.2 Control

In this section we apply an Nonlinear MPC (NMPC) algorithm based on a Gaussian process model, to the problem of controlling the neuromuscular blockade by means of administration of atracurium.

It is assumed that an initial bolus of 500 $\mu g K^{-1}$ is manually applied to the patient, driving the $r\%$ to a level near 0% in a certain period of time. After this initial period, the control objective consists in following a specific reference. The value of the reference is initially fixed at a low level during approximately 30min.
Control of Neuromuscular Blockade

Figure 6: Predictive control ($T_p = 20$, $T_u = 1$). Simulation results obtained with patient models 12, 41, 44, and 69 showing the reference (black) and the neuromuscular blockade level $r(t)$ (blue).

It is then gradually increased to the final value of 10%. The MPC algorithm is turned on when the reference begins to increase.

The GP-NMPC minimizes the cost function (16) subject to the Gaussian process model (9) and the constraint (17), where $u_{min} = 0$ and $u_{max} = 20$. Several simulations, with different patient models (mainly patient $M_{69}$) have been conducted in order to find the best configuration for the MPC algorithm parameters $T_u, T_p, Q, R$ with a sampling interval of 20s (the one used for real data collection). The parameters were selected as: $T_u = 1$, $T_p = 20$, $Q = diag(1.14^1, \ldots, 1.14^{T_p})$
0.5. *SIMULATION RESULTS*  

Control of Neuromuscular Blockade

![Graphs showing control of neuromuscular blockade](image)

Figure 7: Predictive control ($T_p = 20, T_u = 1$). Infusion rate (controller output) obtained with patient models 12, 41, 44, and 69.

and, $R = I$. A low $T_u$ value was chosen because the cost does not depend much on the control horizon and the computation time is lower. A system noise $n(t)$ with a variance of $\sigma^2 = 0.3$ was superimposed to the output process. The Gaussian process model used in all the simulations is fixed and was *taught* off-line by the model $M_{94}$, as explained in the last section.

The GP prediction of the level of NMB was done by iterative multi-step ahead prediction, feeding back at each time step the predictive mean only (*naive approach*), an approach that has shown similar results, in what concerns the mean predictions, when compared with the *exact approach* [10], [11].
Figures 6, and 7 illustrate the output and the control signal, respectively, corresponding to the application of the NMPC algorithm based on Gaussian process model to the models $M_{12}$, $M_{41}$, $M_{44}$ and, $M_{69}$. It can be seen from these same figures that the quality of the responses is better for model $M_{41}$ and worse for model $M_{69}$. These results were expected, due to the differences between the model that was used in the learning of the GP model and the models used in the simulations (see Figure 3). The oscillatory behavior in model $M_{69}$ and to lesser extent in model $M_{44}$ indicates that the model upon which the GP model was based do not provide a full coverage of the range of patient dynamics. To overcame the output oscillations, mainly in patient $M_{69}$, a filtered output derivative term was added to the NMPC control signal (hereafter called NMPC+D),

$$\frac{u_d(z)}{y(z)} = \frac{K(1 - a)(z - 1)}{z - a}.$$  

(29)

After several simulations, with different patient models (mainly patient $M_{69}$) in order to find the best configuration for the derivative term, the prediction horizon $T_p$ was reduced to 14, $K$ was set to 5, and $a = 0.3$, and all the remaining parameters were not changed.

As seen by comparing Figures 6, and 8 with the derivative term introduction the initial transient almost disappeared in models $M_{12}$, $M_{41}$, $M_{44}$, and the output oscillations of model $M_{69}$ were significantly reduced. The infusion rate of the NMPC+D controller, as expected, shows a more random and spiky behavior than the NMPC infusion rate (compare Figures 7 and 9), however, that one has never reached his maximum limit of $u_{\text{max}} = 20 \mu g Kg^{-1}min^{-1}$. Table 1 shows the controller NMPC+D performance analysis for the patients $M_{12}$, $M_{41}$, $M_{44}$ and, $M_{69}$. The Mean Square Tracking Error was calculated for each simulation after the initial 30 min. Variable r(t) is near the reference and the mean infusion rate values are far from maximum limit. Due to his slow recovery after the initial bolus infusion and the chosen reference (that begins to increase after 30 minutes), patient
0.5. SIMULATION RESULTS

Control of Neuromuscular Blockade

Figure 8: Predictive control \( (T_p = 14, T_u = 1) \) plus output derivative control. Simulation results obtained with patient models 12, 41, 44, and 69 showing the reference (black) and the neuromuscular blockade level \( r(t) \) (blue).

\( M_{69} \) takes about 60 minutes to get to \( r(t)\% = 10 \) after the bolus infusion, this patient mean square tracking error will always be higher than the others patients, for similar control conditions.

To test the effect of the time horizons on the performance of the NMPC+D controller, simulations were run with fixed parameters, except the prediction horizon \( T_p \). Figure 10 shows the Mean Square Tracking Error as a function of the prediction horizon \( T_p \). For low values of \( T_p \) a big error is obtained. Due to decreasing prediction precision of the GP model high values of \( T_p \) also yield to a
Figure 9: Predictive control ($T_p = 14$, $T_u = 1$) plus output derivative control. Simulation infusion rate (controller output) obtained with patient models 12, 41, 44, and 69.

decreasing performance. The best $T_p$ values are 11, 11, 11 and 16 for models $M_{41}$, $M_{44}$, $M_{12}$ and $M_{69}$, respectively. For the simulations made with controller NMPC+D the prediction horizon was chosen always $T_p = 14$, a value that is a compromise for all the tested patients.

Because of the inclusion of the derivative term in the control signal and to further test the control algorithm robustness, it was decided to analyze the effect of the output noise in the performance of the controller. Figure 11 illustrates the mean square tracking error as a function of the output noise variance. Although,
0.5. SIMULATION RESULTS  

Control of Neuromuscular Blockade

Figure 10: Predictive control \((T_u = 1)\) plus output derivative control. Mean Square Error obtained with different values of the prediction horizon for the models 12, 41, 44, and 69. The circles indicate the model minimum error.

This figure shows a decreasing performance for higher values of the output noise variance, Figure 12 shows that even for high values of the noise \((\sigma^2 = 0.8)\) the response is still quite acceptable and similar to the one obtained with \(\sigma^2 = 0.3\) (Figure 8). Table 2 shows the controller NMPC+D performance analysis for the patients \(M_{12}, M_{41}, M_{44}\) and, \(M_{69}\) with \(\sigma^2 = 0.8\) output noise variance. The results show that the control performance is reduced essentially by an higher output noise that is shown in the output standard deviation, rather than in an inferior control performance. The mean infusion rate is almost equal to the one obtained with \(\sigma^2 = 0.3\), however, is more dispersed and for all the models the infusion rate maximum limit was reached several times.
Table 1: The four models parameters for NMPC with filtered derivative and a random output noise of $\sigma^2 = 0.3$

<table>
<thead>
<tr>
<th></th>
<th>model 12</th>
<th>model 41</th>
<th>model 44</th>
<th>model 69</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Mean Square Tracking Error</strong></td>
<td>0.927</td>
<td>0.465</td>
<td>0.466</td>
<td>2.181</td>
</tr>
<tr>
<td><strong>Mean r% Steady State</strong></td>
<td>10.20</td>
<td>9.55</td>
<td>9.67</td>
<td>9.28</td>
</tr>
<tr>
<td><strong>r% Steady State Standard deviation</strong></td>
<td>0.559</td>
<td>0.592</td>
<td>0.550</td>
<td>0.558</td>
</tr>
<tr>
<td><strong>Mean Infusion Rate</strong></td>
<td>7.197</td>
<td>5.271</td>
<td>5.543</td>
<td>4.562</td>
</tr>
<tr>
<td><strong>Infusion Rate Standard deviation</strong></td>
<td>3.521</td>
<td>3.512</td>
<td>3.554</td>
<td>3.788</td>
</tr>
</tbody>
</table>

Table 2: The four models parameters for NMPC with filtered derivative and a random output noise of $\sigma^2 = 0.8$

<table>
<thead>
<tr>
<th></th>
<th>model 12</th>
<th>model 41</th>
<th>model 44</th>
<th>model 69</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Mean Square Tracking Error</strong></td>
<td>1.397</td>
<td>1.125</td>
<td>1.021</td>
<td>2.835</td>
</tr>
<tr>
<td><strong>Mean r% Steady State</strong></td>
<td>10.13</td>
<td>9.38</td>
<td>9.56</td>
<td>9.17</td>
</tr>
<tr>
<td><strong>r% Steady State Standard deviation</strong></td>
<td>0.904</td>
<td>0.948</td>
<td>0.879</td>
<td>0.857</td>
</tr>
<tr>
<td><strong>Mean Infusion Rate</strong></td>
<td>7.225</td>
<td>5.318</td>
<td>5.585</td>
<td>4.623</td>
</tr>
<tr>
<td><strong>Infusion Rate Standard deviation</strong></td>
<td>5.199</td>
<td>4.963</td>
<td>5.037</td>
<td>4.846</td>
</tr>
</tbody>
</table>
0.6. Conclusions

In this paper, the application of Nonlinear Model Predictive Control to neuromuscular blockade based on Gaussian process models is developed. A simulation study show that this is a competitive approach to model and control nonlinear dynamic systems with high level of uncertainty such as neuromuscular blockade. Gaussian process modelling and the MPC control approach show to be very reliable as they both present very satisfactory performances for a wide range of neuromuscular blockade behaviors and high noise levels associated with the sensor measurements. A major disadvantage of the Gaussian process model is the computational load associated with the need to invert the covariance matrix, at ev-
every iteration of the optimization algorithm. A computational load that is increased with Model Predictive Control high prediction horizon values. However, for neuromuscular blockade, the control algorithm is easily implemented within the used sampling period.
0.6. CONCLUSIONS

Control of Neuromuscular Blockade

Figure 12: Predictive control ($T_p = 14$, $T_u = 1$) plus output derivative control. Simulation results obtained with patient models 12, 41, 44, and 69 for an output noise variance of 0.8. The figure presents reference in black and the neuromuscular blockade level $r(t)$ in blue.
Figure 13: Predictive control ($T_p = 14$, $T_u = 1$) plus output derivative control. Simulation infusion rate (controller output) obtained with patient models 12, 41, 44, and 69 with a output noise variance of 0.8.
0.6. CONCLUSIONS

Control of Neuromuscular Blockade
Bibliography


BIBLIOGRAPHY

Control of Neuromuscular Blockade


