



UNIVERSITY
of
GLASGOW

Kocijan, J. and Murray-Smith, R. (2005) Nonlinear predictive control with a gaussian process model. *Lecture Notes in Computer Science* 3355:pp. 185-200.

<http://eprints.gla.ac.uk/3730/>

Nonlinear Predictive Control with a Gaussian Process Model

Juš Kocijan^{1,2} and Roderick Murray-Smith^{3,4}

¹ Jozef Stefan Institute, Jamova 39, SI-1000 Ljubljana, Slovenia
jus.kocijan@ijs.si

² Nova Gorica Polytechnic, Nova Gorica

³ Dept. Computing Science, University of Glasgow, Glasgow

⁴ Hamilton Institute, National University of Ireland, Maynooth

Abstract. Gaussian process models provide a probabilistic non-parametric modelling approach for black-box identification of nonlinear dynamic systems. The Gaussian processes can highlight areas of the input space where prediction quality is poor, due to the lack of data or its complexity, by indicating the higher variance around the predicted mean. Gaussian process models contain noticeably less coefficients to be optimized. This chapter illustrates possible application of Gaussian process models within model predictive control. The extra information provided by the Gaussian process model is used in predictive control, where optimization of the control signal takes the variance information into account. The predictive control principle is demonstrated via the control of a pH process benchmark.

1 Introduction

Model Predictive Control (MPC) is a common name for computer control algorithms that use an explicit process model to predict the future plant response. According to this prediction in the chosen period, also known as the prediction horizon, the MPC algorithm optimizes the manipulated variable to obtain an optimal future plant response. The input of chosen length, also known as control horizon, is sent to the plant and then the entire sequence is repeated again in the next time period. The popularity of MPC is to a great extent owed to the ability of MPC algorithms to deal with constraints that are frequently met in control practice and are often not well addressed by other approaches. MPC algorithms can handle hard state and rate constraints on inputs and states that are usually, but not always incorporated in the algorithms via an optimization method. Linear model predictive control approaches [13] started appearing in the early eighties and are well-established in control practice (e.g. [18] for an overview). Nonlinear model predictive control (NMPC) approaches [1] started to appear about ten years later and have also found their way into control practice (e.g. [19,23]) though their popularity can not be compared to linear model predictive control. This fact is connected with the difficulty in nonlinear model construction and with the lack of the necessary confidence in the model. There

were a number of contributions in the field of nonlinear model predictive control dealing with issues like stability, efficient computation, optimization, constraints and others. Some recent work in this field can be found in [2,12]. NMPC algorithms are based on various nonlinear models. Often these models are developed as first principles models, but other approaches, like black-box identification approaches are also popular. Various predictive control algorithms are based on neural network models e.g. [17], fuzzy models e.g. [8] or local model networks e.g. [6]. The quality of control depends on the quality of the model. New developments in NMPC approaches are coming from resolving various issues: from faster optimization methods to different process model. The contribution of this chapter is to describe a NMPC principle with a Gaussian process model. The Gaussian process model is an example of a probabilistic non-parametric model that also provides information about prediction uncertainties which are difficult to evaluate appropriately in nonlinear parametric models. The majority of work on Gaussian processes shown up to now considers modelling of static nonlinearities. The use of Gaussian processes in modelling dynamic systems is a recent development e.g. [15,14,3,21,10,11] and some control algorithms based on such are described in [16,5].

The chapter is organized as follows. Dynamic Gaussian process models are briefly introduced in the next section. The control algorithm principle is described in Section 3 and illustrated with the benchmark pH process control in Section 4. Conclusions are stated at the end of the chapter.

2 Modelling of Dynamic Systems with Gaussian Processes

A Gaussian process is an example of the use of a flexible, probabilistic, non-parametric model which directly provides us with uncertainty predictions. Its use and properties for modelling are reviewed in [22].

A Gaussian process is a collection of random variables which have a joint multivariate Gaussian distribution. Assuming a relationship of the form $y = f(\mathbf{x})$ between an input \mathbf{x} and output y , we have $y^1, \dots, y^n \sim \mathcal{N}(0, \Sigma)$, where $\Sigma_{pq} = \text{Cov}(y_p, y_q) = C(\mathbf{x}_p, \mathbf{x}_q)$ gives the covariance between output points corresponding to input points \mathbf{x}_p and \mathbf{x}_q . Thus, the mean $\mu(\mathbf{x})$ (usually assumed to be zero) and the covariance function $C(\mathbf{x}_p, \mathbf{x}_q)$ fully specify the Gaussian process. Note that the covariance function $C(., .)$ can be any function with the property that it generates a positive definite covariance matrix.

A common choice is

$$C(\mathbf{x}_p, \mathbf{x}_q) = v_1 \exp \left[-\frac{1}{2} \sum_{d=1}^D w_d (x_p^d - x_q^d)^2 \right] + v_0 \quad (1)$$

where $\Theta = [w_1 \dots w_D \ v_0 \ v_1]^T$ are the ‘hyperparameters’ of the covariance functions and D the input dimension. Other forms of covariance functions suitable

for different applications can be found in [20]. For a given problem, the parameters are learned (identified) using the data at hand. After the learning, one can use the w parameters as indicators of ‘how important’ the corresponding input components (dimensions) are: if w_d is zero or near zero it means that the inputs in dimension d contain little information and could possibly be removed.

Consider a set of N D -dimensional input vectors $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N]$ and a vector of output data $\mathbf{y} = [y^1, y^2, \dots, y^N]^T$. Based on the data (\mathbf{X}, \mathbf{y}) , and given a new input vector \mathbf{x}^* , we wish to find the predictive distribution of the corresponding output y^* . Unlike other models, there is no model parameter determination as such, within a fixed model structure. With this model, most of the effort consists in *tuning* the parameters of the covariance function. This is done by maximizing the log-likelihood of the parameters, which is computationally relatively demanding since the inverse of the data covariance matrix ($N \times N$) has to be calculated at every iteration.

The described approach can be easily utilized for regression calculation. Based on training set \mathbf{X} a covariance matrix \mathbf{K} of size $N \times N$ is determined. As already mentioned before the aim is to find the distribution of the corresponding output y^* at some new input vector $\mathbf{x}^* = [x_1(N+1), x_2(N+1), \dots, x_D(N+1)]^T$.

For a new test input \mathbf{x}^* , the predictive distribution of the corresponding output is $y^*|\mathbf{x}^*, (\mathbf{X}, \mathbf{y})$ and is Gaussian, with mean and variance

$$\mu(\mathbf{x}^*) = \mathbf{k}(\mathbf{x}^*)^T \mathbf{K}^{-1} \mathbf{y} \quad (2)$$

$$\sigma^2(\mathbf{x}^*) = k(\mathbf{x}^*) - \mathbf{k}(\mathbf{x}^*)^T \mathbf{K}^{-1} \mathbf{k}(\mathbf{x}^*) + v_0 \quad (3)$$

where $\mathbf{k}(\mathbf{x}^*) = [C(\mathbf{x}^1, \mathbf{x}^*), \dots, C(\mathbf{x}^N, \mathbf{x}^*)]^T$ is the $N \times 1$ vector of covariances between the test and training cases and $k(\mathbf{x}^*) = C(\mathbf{x}^*, \mathbf{x}^*)$ is the covariance between the test input and itself.

For multi-step ahead prediction we have to take account of the uncertainty of future predictions which provide the ‘inputs’ for estimating further means and uncertainties.

If we now consider a new random input, $\mathbf{x}^* \sim \mathcal{N}(\mu_{\mathbf{x}^*}, \Sigma_{\mathbf{x}^*})$, Girard *et. al.* [3], have shown that, within a Gaussian approximation the predictive distribution is again Gaussian with mean and variance

$$m(\mu_{\mathbf{x}^*}, \Sigma_{\mathbf{x}^*}) = E_{\mathbf{x}^*} [\mu(\mathbf{x}^*)] \quad (4)$$

$$v(\mu_{\mathbf{x}^*}, \Sigma_{\mathbf{x}^*}) = E_{\mathbf{x}^*} [\sigma^2(\mathbf{x}^*)] + E_{\mathbf{x}^*} [\mu(\mathbf{x}^*)^2] - (E_{\mathbf{x}^*} [\mu(\mathbf{x}^*)])^2 \quad (5)$$

The more detailed derivation can be found in the previous chapter of this book [4]. Equations (4) and (5) can be applied to calculation of multi-step ahead prediction with propagation of uncertainty.

Gaussian processes can, like neural networks, be used to model static nonlinearities and can therefore be used for modelling of dynamic systems if delayed input and output signals are used as regressors. In such cases an autoregressive

model is considered, such that the current output depends on previous estimated outputs, as well as on previous control inputs.

$$\begin{aligned} \mathbf{x}(k) &= [\hat{y}(k-1), \hat{y}(k-2), \dots, \hat{y}(k-L), u(k-1), \\ &\quad u(k-2), \dots, u(k-L)]^T \\ \hat{y}(k) &= f(\mathbf{x}(k)) + \epsilon \end{aligned} \tag{6}$$

Where k denotes consecutive number of data sample. Let \mathbf{x} denote the state vector composed of the previous estimated outputs \hat{y} and inputs u up to a given lag L and ϵ is white noise.

Iterative multi-step ahead prediction is done, as described in the previous chapter [4], by feeding back the predictive mean, as well as the predictive variance at each time-step, thus taking the uncertainty attached to each intermediate prediction into account. The Gaussian process model now not only describes the dynamic characteristics of the non-linear system, but at the same time provides information about the confidence in the predictions. The Gaussian process can highlight areas of the input space where prediction quality is poor, due to the lack of data, by indicating the higher variance around the predicted mean.

It is worthwhile noting that the derivatives of means and variances can be calculated in straightforward manner. For more details see [21].

3 Nonlinear Model Predictive Control

Nonlinear model predictive control as it was applied with the Gaussian process model can be in general described with a block diagram, as depicted in Fig. 1. The model used is fixed, identified off-line, which means that used control

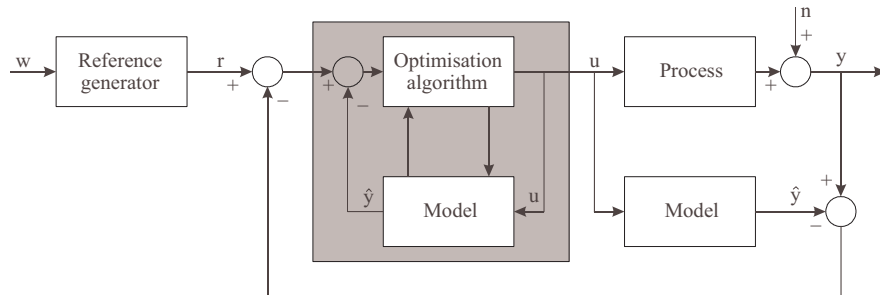


Fig. 1. Block diagram of model predictive control system

algorithm is not an adaptive one. The structure of the entire control loop is therefore less complex than in the case where the model changes with time. The following items describe the basic idea of predictive control:

- Prediction of the system output signal $y(k+j)$ is calculated for each discrete sample k for a large horizon in future ($j = N_1, \dots, N_2$). Predictions are denoted as $\hat{y}(k+j|k)$ and represent j -step ahead prediction, while N_1 and N_2 determine lower and upper bound of prediction horizon. Lower and upper bound of output signal prediction horizon determine coincidence horizon, within which a match between output and reference signal is expected. Output signal prediction is calculated from process model. Predictions are dependent also on the control scenario in the future $u(k+j|k), j = 0, \dots, N_u-1$, which is intended to be applied from a moment k onwards.
- The reference trajectory is determined $r(k+j|k), j = N_1, \dots, N_2$, which determines reference process response from present value $y(k)$ to the setpoint trajectory $w(k)$.
- The vector of future control signals $\mathbf{U}(k)$ containing $u(k+j|k), j = 0, \dots, N_u-1$ is calculated by minimization of cost function (also called objective function) such that predicted error between $r(k+j|k)$ and $\hat{y}(k+j|k), j = N_1, \dots, N_2$ is minimal. Structuring of future control samples can be used in some approaches.
- Only the first element $u(k|k)$ of the optimal control signal vector $u(k+j|k), j = 0, \dots, N_u-1$ is applied.

In the next sample a new measured output sample is available and the entire procedure is repeated. This procedure is called a receding horizon strategy.

The control objective is to be achieved by minimization of the cost function. The cost function penalizes deviations of the predicted controlled outputs $\hat{y}(k+j|k)$ from a reference trajectory $r(k+j|k)$. This reference trajectory may depend on measurements made up to time k . Its initial point may be the output measurement $y(k)$, but also a fixed set-point, or some predetermined trajectory. The minimization of cost function, in which future control signal ($\mathbf{U}(k)$) is calculated, can be subject to various constraints (e.g. input, state, rates, etc).

The process model for calculation of predicted outputs is in our case a Gaussian process model, which provides not only the mean value $\hat{y}(k+j|k)$ but also the corresponding variance.

There are many alternative ways of how NMPC with Gaussian process models can be realized.

Cost function. The cost function used (11) is just one of many possible ones.

It is well known that selection of the cost function has a major impact on the amount of computation.

Optimization problem for $\Delta\mathbf{U}(k)$ instead of $\mathbf{U}(k)$. This is not just a change of formalism, but also enables other forms of MPC. One possibility is a DMC controller with nonlinear model, e.g. [8] - a frequently used principle, that together with appropriate cost function enables problem representation as a least squares problem that can be solved in one iteration in which an explicit solution is found. This is, as in the case with other special case simplifications, not a general case solution.

Process model. The process model can be determined off-line and fixed for the time of operation or determined on-line during the operation of controller

[16]. However, the problem of increasing covariance matrix dimension with incoming data has to be dealt with. Safety related issues also need to be considered thoroughly in the case of adaptive version application.

Soft constraints. Using constraint optimization algorithms is very demanding for computation and soft constraints, namely weights on constrained variables in cost function, can be used to decrease the amount of computation. More on this topic can be found in [9,24].

Linear MPC. It is worth to remark that even though this is a constrained nonlinear MPC problem it can be used in its specialized form as a robust linear MPC.

There are several issues of interest for applied NMPC. Let us mention some of them.

Efficient numerical solution. Nonlinear programming optimization algorithm is very demanding for computation. Various approximations and other approaches (e.g. approximation of explicit solution) exist to decrease computational load, mainly for special cases, like linear process models or special cost functions.

One possibility to decrease the computational load necessary for optimization is with the incorporation of prediction derivation (and variance) into optimization algorithm. When using Gaussian process models the prediction and variance derivation can be calculated in a straightforward manner.

Stability. At present no stability conditions have been derived for Gaussian processes as a representative of probabilistic non-parametric models.

Robustness. This issue has a major impact on the applicability of the algorithm in practice. The fact that the process model contains the information about the model confidence enables controller to optimize the manipulative variable to “avoid” regions where the confidence in the model is not high enough. This possibility itself makes the controller robust if applied properly. MPC robustness in the case of other algorithms is usually not some specially built feature of the MPC algorithms, but was more an issue of assessment for particular MPC algorithms.

4 Example

4.1 pH Process

A simplified schematic diagram of the pH neutralization process taken from [7] is given in Fig. 2. The process consists of an acid stream (Q_1), buffer stream (Q_2) and base stream (Q_3) that are mixed in a tank T_1 . Prior to mixing, the acid stream enters the tank T_2 which introduces additional flow dynamics. The acid and base flow rates are controlled with flow control valves, while the buffer flow rate is controlled manually with a rotameter. The effluent pH (pH) is the measured variable. Since the pH probe is located downstream from the tank T_1 , a time delay (T_d) is introduced in the pH measurement. In this study, the pH

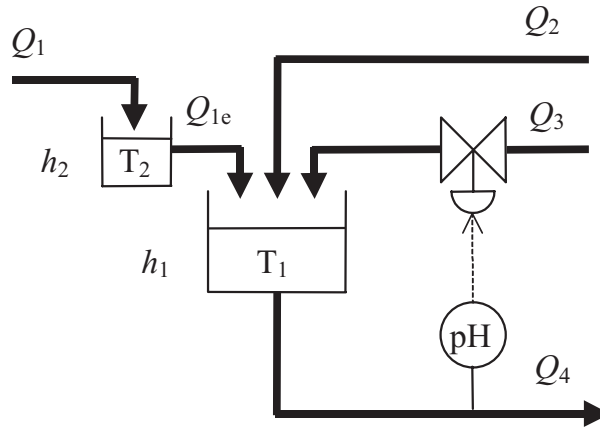


Fig. 2. The pH neutralization system scheme

is controlled by manipulating the base flow rate. A more detailed description of the process with mathematical model and necessary parameters is presented in [7].

The dynamic model of the pH neutralization system shown in Fig. 2 is derived using the conservation equations and equilibrium relations. The model also includes valve and transmitter dynamics as well as hydraulic relationships for the tank outlet flows. Modelling assumptions include perfect mixing, constant density, and complete solubility of the ions involved. The simulation model of pH process, which was used for necessary data generation contains therefore various nonlinear elements as well as implicitly calculated function which is value of highly nonlinear titration curve.

4.2 Model Identification

Based on responses and iterative cut-and-try procedure a sampling time of 25 seconds was selected. The sampling time was so large that the dead-time mentioned in the previous section disappeared.

The chosen identification signal of 400 samples was generated from a uniform random distribution and rate of 50 seconds.

Obtained hyperparameters of the third order Gaussian process model were:

$$\begin{aligned} \Theta &= [w_1, w_2, w_3, w_4, w_5, w_6, v_0, v_1] \\ &= [-6.0505, -2.0823, -0.4785, -5.3388, -3.4206, -8.7080, \\ &\quad 0.8754, -5.4164] \end{aligned} \quad (7)$$

where hyperparameters from w_1 to w_3 denote a weight for each output regressor, from w_4 to w_6 denote a weight for each input regressor, v_0 is estimated noise variance and v_1 is the estimate of the vertical scale of variation.

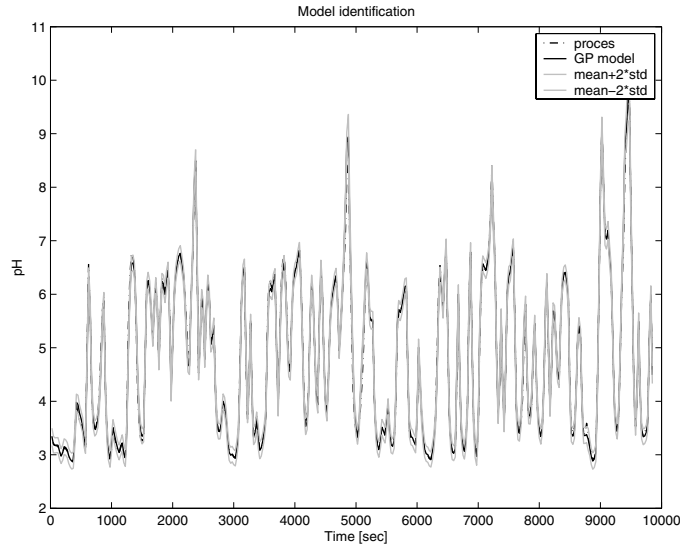


Fig. 3. Response of Gaussian process model on excitation signal used for identification

The region in which the model was obtained can be seen from Fig. 3. A very good fit can be observed for the identification input signal which was used for optimization. However, the obtained model contains information mainly in the region below $\text{pH}=7$ as can be concluded from the response in Fig. 3. The validation signal was from the same region as the identification signal. The identification and validation signal were obtained with generator of random noise with different initial values. Response of the model to validation signal and comparison with process response is depicted in Fig. 4. Fitting of the response for validation signal:

– average absolute test error

$$AE = 0.0691 \quad (8)$$

– average squared test error

$$SE = 0.0109 \quad (9)$$

– log density error

$$LD = -0.7130 \quad (10)$$

After model validation the model was utilized for control design. See [11] for more issues on pH process modelling with Gaussian process models.

4.3 Control

A moving-horizon minimization problem of the special form [13]

$$\min_{\mathbf{U}(k)} [r(k+P) - \hat{y}(k+P)]^2 \quad (11)$$

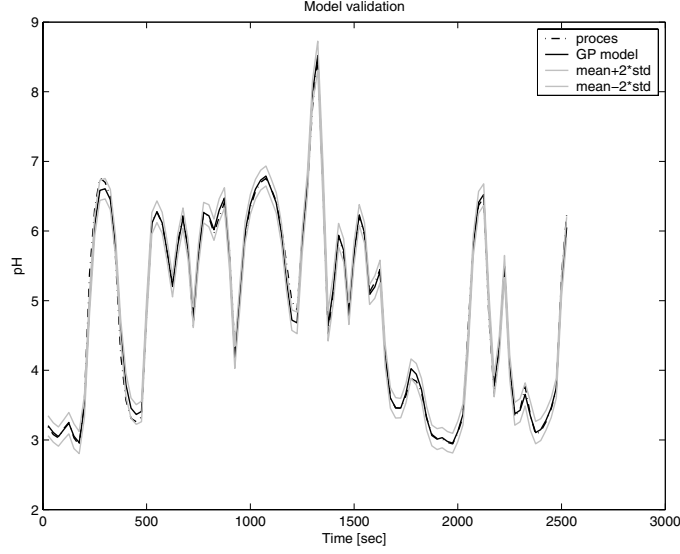


Fig. 4. Response of Gaussian process model on excitation signal used for validation

subject to:

$$\text{var } \hat{y}(k+P) \leq k_v \quad (12)$$

$$| \mathbf{U}(k) | \leq k_{ih} \quad (13)$$

$$| \dot{\mathbf{U}}(k) | \leq k_{ir} \quad (14)$$

$$| \mathbf{x}(k) | \leq k_{sh} \quad (15)$$

$$| \dot{\mathbf{x}}(k) | \leq k_{sr} \quad (16)$$

is used in our case, where $\mathbf{U}(k) = [u(k) \dots u(k+P)]$ is input signal, P is the coincidence point (the point where a match between output and reference value is expected) and inequalities from (12) to (16) represent constraint on output variance k_v , input hard constraint k_{ih} , input rate constraint k_{ir} , state hard constraint k_{sh} and state rate constraint k_{sr} respectively. The process model is a Gaussian process.

The optimization algorithm, which is constrained nonlinear programming, is solved at each sample time over a prediction horizon of length P , for a series of moves which equals to control horizon. In our case control horizon was chosen to be one and to demonstrate constraint on variance the rest of constraints was not taken into the account. Nevertheless, all this modifications do not change the generality of solution, but they do affect the numerical solution itself.

The control algorithm described above was tested for the pH process with simulation. The reference trajectory r is defined so that it approaches the set-point exponentially from the current output value. The coincidence point was

chosen to be 8 samples and, as already mentioned, the control horizon is one sample. The results of unconstrained control are given in Figs. 5 and 6.

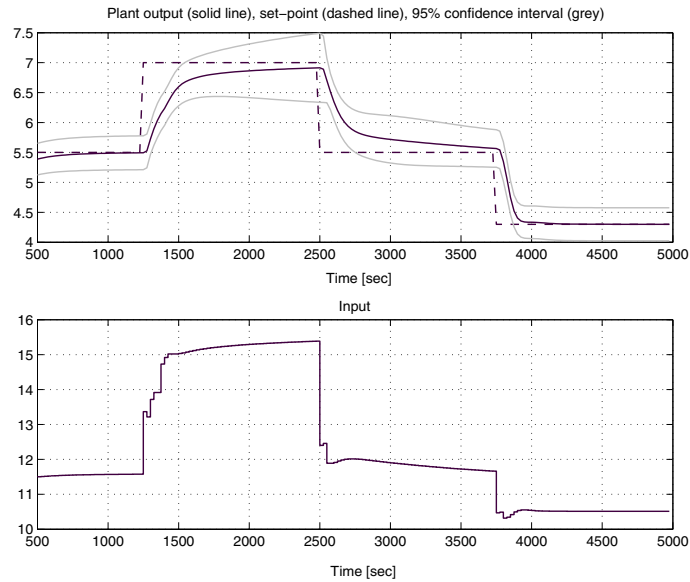


Fig. 5. Non-constrained case: response of Gaussian process model based control (upper figure) and control signal (bottom figure)

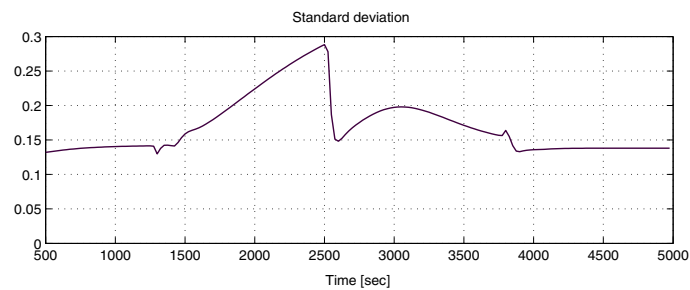


Fig. 6. Non-constrained case: standard deviation corresponding to the previous figure

It can be seen from different set-point responses that the model differs from the process in different regions. It can be clearly seen that the variance increases as output signal approaches regions which were not populated with enough identification data. It should be noted however that variances are sum of variances

that correspond to information about regions where there are varying degrees of confidence in the model accuracy, depending upon local density of available identification data and of output response variances. When variances increase too much, one design option is that the response can be optimized with constrained control. Results can be seen in Figs. 7 and 8.

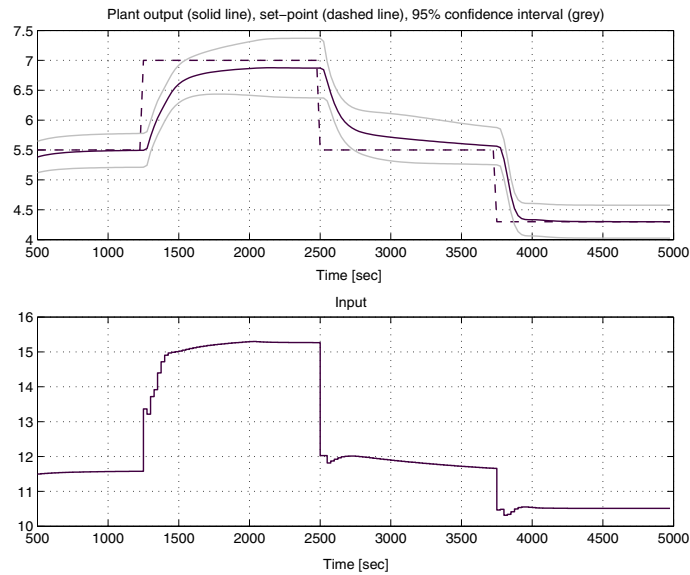


Fig. 7. Constrained case ($\sigma_{max} = 0.25$): response of Gaussian process model based control (upper figure) and control signal (bottom figure)

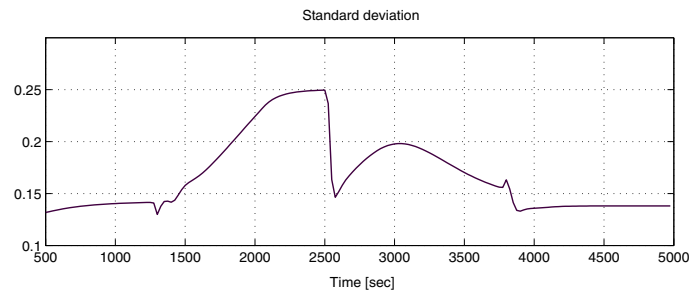


Fig. 8. Constrained case ($\sigma_{max} = 0.25$): standard deviation corresponding to the previous figure

It can be seen from Figs. 7 and 8 that the closed-loop system response now avoids the region with large variance, at the cost of an increase in steady-state

error. This could be interpreted also as trade-off between designed performance and safety.

A possible alternative selection of cost function that avoids constrained optimization and is therefore computationally less demanding would be as follows.

$$\min_{\mathbf{U}(k)} E\{[r(k+P) - \hat{y}(k+P)]^2\} \quad (17)$$

Using the fact that $\text{var}\{\hat{y}\} = E\{\hat{y}^2\} - E\{\hat{y}\}^2$, the cost function can be written as:

$$\min_{\mathbf{U}(k)} [r(k+P) - E\{\hat{y}(k+P)\}]^2 + \text{var}\{\hat{y}(k+P)\} \quad (18)$$

Results with cost function (18) are given in Figs. 9 and 10. It can be again observed from Figs. 9 and 10 that the closed-loop system response avoids the region with large variance, at the cost of increased steady-state error, as was the case with constrained control, but with less computational burden than the constrained control case. The control strategy with cost function (18) is “to avoid” going into regions with higher variance. The term “higher variance” does not specify any specific value. In the case that controller does not seem to be cautious enough, a pragmatic calibration option is that the variance term can be weighted to enable shaping of the closed-loop response according to variance information:

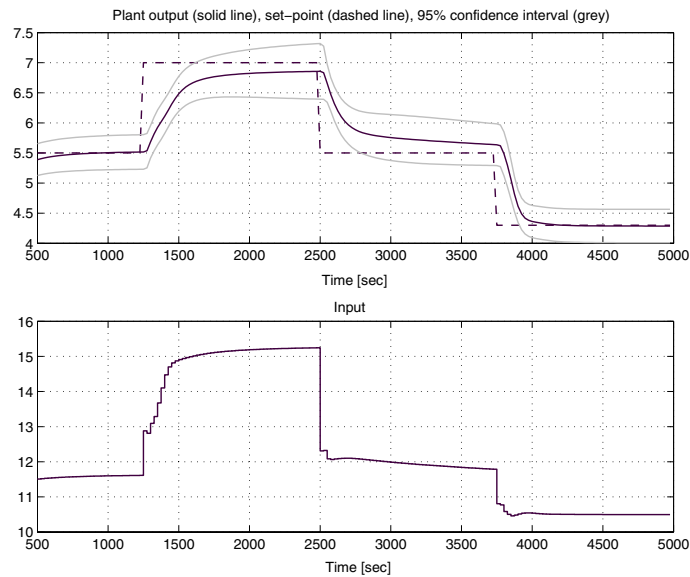


Fig. 9. Response of Gaussian process model based control with “soft constraints” (upper figure) and control signal (bottom figure)

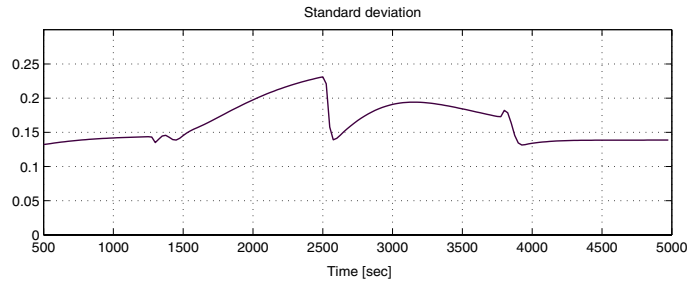


Fig. 10. Standard deviation corresponding to the previous figure

$$\min_{\mathbf{U}(k)} [r(k + P) - E\{\hat{y}(k + P)\}]^2 + \lambda \text{var}\{\hat{y}(k + P)\} \quad (19)$$

NMPC with the second cost function with weight on variance $\lambda = 2$, using unconstrained optimization, gives the results depicted in Figs. 11 and 12, showing a reduction in the standard deviation of the predictions of the closed-loop response, compared to Fig. 10, and minor changes in the mean behaviour.

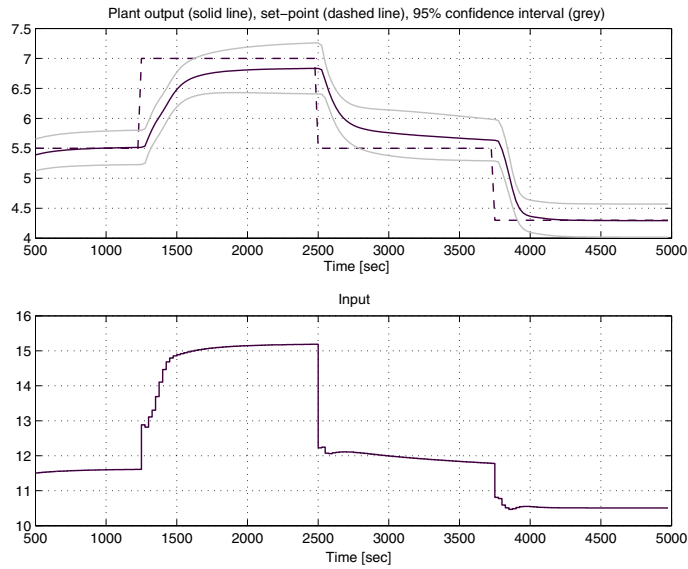


Fig. 11. Response of Gaussian process model based control with “soft constraints” (upper figure) and control signal (bottom figure)

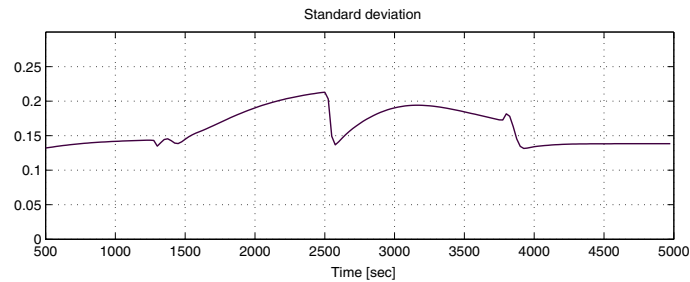


Fig. 12. Standard deviation corresponding to the previous figure

Beside the difference in the optimization algorithm, the presented options give also a design choice on how “safe” the control algorithm is. In the case when it is very undesirable to go into “unknown” regions the constrained version might be the better option.

5 Conclusions

The principle of Model Predictive Control based on a Gaussian process model was presented and illustrated with a pH process control example. In the example, a constraint on model variance was included. This can be complimented also with other constraints when necessary. The use of Gaussian process models makes it possible to include information about the confidence in the model depending on the region.

It was indicated that using Gaussian process models offers an attractive possibility for control design that results in a controller with a higher level of robustness due to information contained in the model. It is necessary to stress that the presented control strategy represents only a feasibility test for Gaussian process application for model predictive control and additional efforts are necessary before this approach will be applicable in engineering practice.

A practical challenge dealing with application of Gaussian process models in control applications is related to the computational burden associated with the number of training data (although recent work in [21] has shown how derivative observations can improve the situation in control contexts). Another interesting issue that is under investigation is disturbance rejection. Despite these current challenges, the Gaussian process approach has a number of exciting advantages. The simple model structure, the reduced sensitivity to the choice of model structure and the uncertainty information one obtains on the predictions are attractions of the Gaussian process approach. The principle shown is quite general and several modifications that accelerate computation can be used and are planned to be derived in the future.

Acknowledgments

This work was made possible by EC funded Multi-Agent Control Research Training Network HPRN-CT-1999-00107, and RM-S by EPSRC project GR/M76379/01, and Science Foundation Ireland grant 00/PI.1/C067. J. Kocijan acknowledges the support of Slovene Ministry of education, science and sport through project P2-0001. The authors are grateful for comments and useful suggestions from Agathe Girard, Daniel Sbarbaro and Gregor Gregorčič.

References

1. Allgöwer F., Badgwell T.A., Qin S.J., Rawlings J.B., Wright S.J., Nonlinear predictive control and moving horizon estimation - an introductory overview, In: Frank, P.M (edt.), *Advances in control: highlights of ECC'99*, Springer, (1999), 391-449.
2. Allgöwer F., Zheng A. (eds.), *Nonlinear Model Predictive Control, Progress in system and control theory*, Vol. 26, Birkhäuser Verlag, Basel, (2000).
3. Girard A., Rasmussen C.E., Quinero Candela, J. and Murray-Smith R., *Gaussian Process Priors With Uncertain Inputs & Application to Multiple-Step Ahead Time Series Forecasting*, NIPS 15, Vancouver, Canada, MIT Press, (2003).
4. Girard A. and Murray-Smith R., *Gaussian Process: Prediction at a noisy input and application to iterative multiple-step ahead forecasting of time-series*, In: *Switching and Learning in Feedback Systems*, Eds R. Murray-Smith, R. Shorten, Springer, 2004.
5. Gregorčič G., Lightbody G., *Internal model control based on a Gaussian process prior model*, *Proceedings of ACC'2003*, Denver, CO, (2003), 4981-4986.
6. Johansen T.A., Foss B.A., Sorensen A.V., *Non-linear predictive control using local models - applied to a batch fermentation process*, *Control Eng. Practice*, 3(3), (1995), 389-396.
7. Henson M.A., Seborg D.E., *Adaptive Nonlinear Control of a pH Neutralization Process*, *IEEE Trans. Control System Technology*, Vol. 2, No. 3, (1994), 169-183.
8. Kavšek-Biasizzo K., Škrjanc I., Matko D., *Fuzzy predictive control of highly non-linear pH process*, *Computers & chemical engineering*, Vol. 21, Supp. 1997, (1997), S613-S618.
9. Kerrigan E.C., Maciejowski J.M., *Soft constraints and exact penalty functions in model predictive control*, *Control 2000 Conference*, Cambridge, (2000).
10. Kocijan J., Girard A., Banko B., Murray-Smith R., *Dynamic Systems Identification with Gaussian Processes*, *Proceedings of 4th Mathmod*, Vienna, (2003), 776-784.
11. Kocijan J., Likar B., Banko B., Girard A., Murray-Smith R., Rasmussen C.E., *A case based comparison of identification with neural network and Gaussian process models*, *Preprints of IFAC ICONS Conference*, Faro, (2003), 137-142.
12. Kouvaritakis B., Cannon M. (eds.), *Nonlinear predictive control, Theory and practice*, *IEE Control Engineering Series 61*, IEE, (2001).
13. Maciejowski J.M., *Predictive control with constraints*, Pearson Education Limited, Harlow, (2002).
14. Murray-Smith, R., Girard, A., *Gaussian Process priors with ARMA noise models*, *Irish Signals and Systems Conference*, Maynooth, (2001), 147-152.
15. Murray-Smith, R and T. A. Johansen and R. Shorten, *On transient dynamics, off-equilibrium behaviour and identification in blended multiple model structures*, *European Control Conference*, Karlsruhe, (1999), BA-14.

16. Murray-Smith R. and Sbarbaro D., Nonlinear adaptive control using nonparametric Gaussian process prior models, In: Proc. IFAC Congress, Barcelona, (2002).
17. Nørgaard M., Ravn O., Poulsen N.K., Hansen L.K., Neural networks for modelling and control of dynamic systems, Springer, London, (2000).
18. Qin S.J., Badgwell T.A., An overview of industrial model predictive control technology, In: Kantor J.C., Garcia C.E., Carnahan B. (eds.) Fifthe international conference on Chemical process control, AChE and CACHE, (1997), 232-56.
19. Qin S.J., Badgwell T.A., An overview of nonlinear model predictive control applications, In: Allgöwer F., Zheng A. (eds.), Nonlinear model predictive control, Birkhauser Verlag, (2000), 369-392.
20. Rasmussen C.E., Evaluation of Gaussian Processes and other Methods for Non-Linear Regression, Ph.D. Disertation, Graduate department of Computer Science, University of Toronto, Toronto, (1996).
21. Solak, E., Murray-Smith R., Leithead, W.E., Leith, D.J., and Rasmussen, C.E., Derivative observations in Gaussian Process models of dynamic systems, NIPS 15, Vancouver, Canada, MIT Press, (2003).
22. Williams C.K.I., Prediction with Gaussian processes: From linear regression to linear prediction and beyond, In: Learning in Graphical Models, Jordan, M.I., (ed.), Kluwer Academic, Dordrecht, (1998), 599-621.
23. Young R.E., Bartusiak R.D., Fontaine, R.W., Evolution of an industrial nonlinear model predictive controller, Preprints on Chemical Process Control - CPC VI, CACHE, Tucson, AZ, (2001), 399-401.
24. Zheng A., Morari M., Stability of model predictive control with mixed constraints, IEEE Trans. Autom. Control, 40(19), (1995), 1818-1823.