



Heriot-Watt University
Research Gateway

Offset-free control of a pH system using Multiple Model Predictive Control

Citation for published version:

Hermansson, AW & Syafii, S 2020, 'Offset-free control of a pH system using Multiple Model Predictive Control', *IOP Conference Series: Materials Science and Engineering*, vol. 778, 012072.
<https://doi.org/10.1088/1757-899X/778/1/012072>

Digital Object Identifier (DOI):

[10.1088/1757-899X/778/1/012072](https://doi.org/10.1088/1757-899X/778/1/012072)

Link:

[Link to publication record in Heriot-Watt Research Portal](#)

Document Version:

Publisher's PDF, also known as Version of record

Published In:

IOP Conference Series: Materials Science and Engineering

Publisher Rights Statement:

Published under licence by IOP Publishing Ltd

General rights

Copyright for the publications made accessible via Heriot-Watt Research Portal is retained by the author(s) and / or other copyright owners and it is a condition of accessing these publications that users recognise and abide by the legal requirements associated with these rights.

Take down policy

Heriot-Watt University has made every reasonable effort to ensure that the content in Heriot-Watt Research Portal complies with UK legislation. If you believe that the public display of this file breaches copyright please contact open.access@hw.ac.uk providing details, and we will remove access to the work immediately and investigate your claim.

PAPER • OPEN ACCESS

Offset-free control of a pH system using Multiple Model Predictive Control

To cite this article: A W Hermansson and S Syafii 2020 *IOP Conf. Ser.: Mater. Sci. Eng.* **778** 012072

View the [article online](#) for updates and enhancements.

Offset-free control of a pH system using Multiple Model Predictive Control

A W Hermansson¹, S Syafiie^{2,3}

¹ School of Engineering and Physical Sciences, Heriot-Watt University Malaysia, No. 1 Jalan Venna P5/2, Precinct 5, 62200 Putrajaya, Wilayah Persekutuan Putrajaya, Malaysia

² Department of Chemical and Material Engineering, Faculty of Engineering, King Abdulaziz University - Rabigh, Kingdom of Saudi Arabia, 21911

³ Department of Industrial Engineering, Faculty of Engineering, Syiah Kuala University, 23111 Darussalam - Banda Aceh, Indonesia

a.hermansson@hw.ac.uk

Abstract. In this paper, a novel combination of a multi-model predictive controller (MMPC) and an adaptive integral controller is used to achieve offset-free control of a nonlinear process. The idea is to avoid the more complex tuning that comes with an offset-free control based on an observer. To create an easily tuned controller based on a piecewise linear (PWL) description of an MPC setup, which utilizes a Bayesian weighting approach. The PWL models are also used to design the separate the I-controller that is made adaptive by using the Bayesian weighting again. The MPC and the I-controller are then acting in parallel. The setup is implemented and tested using a simulation of a pH neutralization process.

1. INTRODUCTION

Model Predictive Control (MPC) is considered the most widely used advanced control techniques in the industry [12], [4], [3], [17]. One of the reasons for the spread of the application is the fact that the method originated from the industry with packages like IDCOM [25] and DMC [5]. Three steps can summarize the MPC algorithm. Firstly, use a model to predict the behavior of the process. Then, apply an optimal control consideration over a time horizon into the future. Finally, after obtaining the optimal input sequence utilize only the first input in the sequence, which is a process that is repeated at each time step moving the horizon of prediction with it.

Traditionally, linear models have been used although most systems exhibit some non-linearity. The development over the years has been to be able to deal with increasingly nonlinear systems as well as faster dynamics. Even though the model structure is nonlinear, it is commonly not applied inside the optimal control part as that would render an intractable optimization problem, particularly non-convexity with no guarantee for a global solution. Most of the time a linear model is used in the optimization although at least some part of the model is nonlinear.

The initial assumptions of the system, whether linear or nonlinear, were; no mismatch between the system and the model, no unknown, unmeasurable or stochastic disturbances as well as that the internal variables, the states, are known. To deal with these uncertainties and to obtain offset-free control further steps have to be incorporated into the MPC setup. The starting point to deal with unknown states as well as estimating the disturbance has been to incorporate an observer. The



inclusion of the observer has also been shown to introduce integral action to the MPC by augmentation of the state with the disturbance ([19], [24], [1], [22], [13], [6]). The disturbance estimation can also be based on a velocity form [26], [7] or done without augmentation [27]. Pannochia [21], [23] show they are all interchangeable and just a matter of preference of implementation. The different techniques have also been implemented into nonlinear systems [11], [18], [15], [16]. However, there are some drawbacks of the observer-based approaches. The tuning of the observer gain adds an additional tuning step that is not straightforward [27], [21], [8] and cannot guarantee offset-free control [7], [27] and gives an increase in computational load [28], [15]. This is particularly the case when dealing with nonlinear systems.

We propose a more straightforward approach utilizing integral action directly to obtain offset-free control, but without the additional computational load and complexity of setup, the conventional techniques rely on, which is the reason that offset-free MPC applications to nonlinear systems are quite rare in industry [14]. The system is kept simple by using multiple models, creating a PWL description that is weighted together using a Bayesian weighting to keep a good description of the nonlinear behavior. The Bayesian weighed MMPC is combined with a separate integral action feedback controller to offset-free, that can deal with any type of bounded uncertainties, whether parametric, input, output or unstructured.

The outline of the paper is as follows: section 2 describes the MPC setup and algorithm. Section 3 introduces the proposed approach, presenting how the piecewise linear description is incorporated into the MPC scheme and the incorporation of the integral action into the MPC system. Section 4, introduces a pH neutralization system that is used as a case study to test the control setup, which is followed by the simulation testing of the proposed controller and the results thereof. Section 5 summarizes the findings and offers a conclusion.

2. Preliminaries

2.1 Model predictive control setup

The principle of the MPC in its most basic form is to use a model, or a set of models, to predict the behavior of the process, which is done over a predetermined length into the future, the prediction horizon, N_p . The deviation between the actual output, y , and the set-point over that horizon is minimized by choosing a control sequence; $\mathbb{U} = \{u(k+0|k), \dots, u(k+N_c|k)\}$, that will minimize a cost function. The cost functions used may vary, though it is generally on quadratic form, with the example following penalizing the deviation of the controlled variable as well as the control action as seen in:

$$J(k) = \sum_{j=0}^{N_p} \|\tilde{y}(k+j|k) - y_{sp}\|_Q^2 + \sum_{j=0}^{N_c} \|\tilde{u}(k+j|k) - \tilde{u}(k+j-1|k)\|_R^2 \quad (1)$$

where R and Q are positive definite weighting matrices, $\tilde{y}(k+j|k)$ is the control variable predicted by the model, where the prediction is based on the knowledge of the variable at instant k and using it to predict the state at the future state $k+j$. The change in the manipulated variable, $\tilde{u}(k+j+1|k) - \tilde{u}(k+j|k)$ are minimized over the control horizon, N_c , with $\tilde{u}(k+j-1|k)$ being the control input from the previous time step. To cost function is then minimized by selecting the optimum control sequence based on the following optimization problem

$$\min_{\mathbb{U}(k)} J(k) \quad (2)$$

The optimization may be subject to various constraints. Constraints on the manipulated variables are commonly needed to make sure possible variables are obtained in the optimization. States and controlled variables constraints may easily be included as well. After the optimization the first control input in the sequence, $u(k+0|k)$, is applied as the control input at instant k , given as $u(k)$. The process with a new prediction, minimization, obtaining of control sequence and finally the application of the first control input, is repeated at each time step where the horizons keep receding

2.2 Models of the system

The model of the system is the single most crucial component in the MPC setup to the extent that another common description of MPC is Model-Based Predictive Control. The type of model span the whole range of models from linear to nonlinear, utilizing various techniques to describe and obtain them. The most commonly used model type in the research literature is the state-space model, though transfer functions, as well as other forms, have been used as well.

1) Nonlinear models: The state space model of a nonlinear process of interest, can be described on a discrete form as:

$$\begin{aligned}x(k+1) &= f(x(k), u(k), w(k)) \\ y(k) &= h(x(k), v(k))\end{aligned}\quad (3)$$

where, $x \in \mathbb{R}^{n_x}$, $u \in \mathbb{R}^{n_u}$ and $y \in \mathbb{R}^{n_y}$ denote the vectors of state, manipulated input and measured output respectively, all at sample time k . The vectors w and v contains any disturbances or uncertainties which are assumed to be bounded and part of subsets $\mathcal{W} \subset \mathbb{R}^{n_w}$ and $\mathcal{V} \subset \mathbb{R}^{n_v}$ respectively. The system is subject to constraints of the state $x \in \mathcal{X}$ and the manipulated input $u \in \mathcal{U}$, where \mathcal{X} is a compact set that containing the origin in its interior and $\mathcal{U} : \mathcal{X} \rightarrow \mathcal{U}$ is a compact set-valued map containing zero in its interior. The function $f: \mathcal{X} \times \mathcal{U} \times \mathcal{W} \rightarrow \mathbb{R}^{n_x}$ is twice continuously differentiable and the function $h: \mathcal{X} \times \mathcal{V} \rightarrow \mathbb{R}^{n_y}$ is continuous.

2) Linear Description of the system: Using a nonlinear description of the system, (3), generally generates an MPC problem that has some undesired properties, particularly it leads to a non-convex optimization problem. The MPC setup can be made more applicable, maintaining a convex optimization problem, by using a linear representation of (3) instead. Linear representations range from using a single linear, model, to a set of linear models, separating the model into two parts, linear and nonlinear, or carry a linearization of the nonlinear model at each time step. Irrespective of the linear approximation used the common linear state space description of (3) is given as:

$$\begin{aligned}\tilde{x}(k+1) &= A\tilde{x}(k) + B\tilde{u}(k) \\ \tilde{y}(k) &= C\tilde{x}(k)\end{aligned}\quad (4)$$

where $A \in \mathbb{R}^{n_x \times n_x}$, $B \in \mathbb{R}^{n_x \times n_u}$ and $C \in \mathbb{R}^{n_y \times n_x}$. The models utilizes deviation variables, \tilde{x} , \tilde{u} and \tilde{y} of x , u and y defined as $\tilde{x} = \hat{x} - x_k$, $\tilde{u} = \hat{u} - u_k$ and $\tilde{y} = \hat{y} - y_k$ respectively, where $(\hat{x}, \hat{u}, \hat{y})$ are (x, u, y) as approximated by the models and (x_k, u_k, y_k) are the desired set points at the instant k .

3. THEORY AND METHODS

In this section, the proposed approach for the offset-free MPC for a nonlinear system is presented. Firstly to be able to control a nonlinear system, multiple linear models are used to describe the nonlinear behavior, capturing the nonlinearity without the need of feeding a nonlinear function into the minimization problem (2). Instead of applying all models in (1) they are combined using Bayesian weighting. Secondly, the I-control part, that will run in parallel to the MPC is to generate a combined manipulated variable u .

3.1 MPC for nonlinear system using Multiple Models

The basic idea is to use multiple linear models to describe a nonlinear system for MPC, allowing the optimization problem (2) to use a linear model. The minimization problem remains convex while keeping some of the nonlinear behaviour in the model. The convexity ensures that the global minimum is obtained swiftly. Switching between the models would be the most straightforward technique, which works by selecting the model with the linearization point closest to the current value of the process. Switching, however, tends to cause a slightly erratic behavior mainly when the system is close to the switching points as it could cause the continuous switching between two models on either side of the switching point. The chattering can be reduced by treating the multiple models as different random models. These models would then be combined based on studying each models closeness to the actual value currently as well as how well it worked in the past. Weighting generates a smoother transition

between different models, better control in areas close to the model intersects as well as generally a better description away from the operating point.

The first step would be to obtain a linear model (4) for some operating points. The number of models, as well as their respective operating point, would be selected based on capturing the nonlinear behavior as well as the likely setpoints for the system. The matrices A_i , B_i and C_i are generated for each model, $i = 1, \dots, L$ getting a set of models (4) describing (3). The different models would then be combined to a single model by weighing the different matrices together according to:

$$\begin{aligned} \bar{A} &= \sum_{i=1}^L \lambda_i A_i, \quad \bar{B} = \sum_{i=1}^L \lambda_i B_i, \quad \bar{C} = \sum_{i=1}^L \lambda_i C_i \\ 0 < \lambda_i < 1 & \quad \sum_{i=1}^L \lambda_i = 1 \end{aligned} \quad (5)$$

where L is the number of models and λ_i represents the weight applied to the i th model. Aufderheide and Bequette [2] proposed that Bayesian weighting could be used to obtain the different weights required used in (5). Firstly, the measured output y is compared to the outputs \hat{y}_i predicted by each of the different models, to obtain a residual ϵ_i for each model:

$$\epsilon_i(k) = y(k) - \hat{y}_i(k|k) \quad (6)$$

where a small residual indicates that the model is currently describing the process well. A small residual is not necessarily a good indicator that the model is representing the system (3) accurately. For example, small residuals could be obtained when the linear model is normal to the nonlinear behavior. The Bayesian weighting takes into account how accurately the model has described the system in the past. Thus, there are three components in the computation of the probability, $P_i(k)$, that a model is the correct model at the current time. Firstly, the probability, $P_i(k-1)$, that model i was the exact model up until the previous time step. Secondly, the correction for the accuracy of the model at the current time step, through the exponential value of $-0.5\epsilon_i^T(k)K_i\epsilon_i(k)$. A small residual would mean that this term would be close to one, while a large residual would bring it close to zero. K_i is a diagonal scaling matrix, which ideally should be the covariance of each model. As the covariance is not known it is used as a tuning parameter to set the speed at which the probability will evolve. With large values in K_i a model with large residual would have a quickly reducing probability, while a small K_i would mean a slow transition instead. Lastly, the product of the values from the first to steps are normalized by the total probabilities of all models as seen in (7).

$$P_i(k) = \frac{P_i(k-1) \times \exp(-0.5 \epsilon_i^T(k) K_i \epsilon_i(k))}{\sum_{j=1}^L P_j(k-1) \times \exp(-0.5 \epsilon_j^T(k) K_j \epsilon_j(k))} \quad (7)$$

The weights are then computed by normalizing the current probabilities according to:

$$\lambda_i(k) = \frac{P_i(k)}{\sum_{j=1}^L P_j(k)} \quad (8)$$

The recursive setup of (7) means that if P_i becomes zero, it would remain zero and hence that particular model would become inactive. Instead of repeatedly resetting the system, this effect is countered by setting all small probabilities to at least take the value $\delta > 0$ by the following condition:

$$P_i = \begin{cases} P_i & \text{if } P_i \geq \delta \\ \delta & \text{if } P_i < \delta \end{cases} \quad (9)$$

This ensure that all models remain live so that they can be used in the continuous probability iterations (7). Another effect of having recursive computations is that the initial values of the different probabilities are required and when no a priori knowledge is available uniform distribution of the probabilities is used to initialize the calculations.

The setup of the multiple model predictive controller (MMPC) is thus complete. The steps (6)-(9) generates the weights based on the value and set-point of the controlled variable as the "weight calculator" in figure 1. The models are combined to a single model following (5). The linear system than obtained, (4), is then used in a standard MPC setup (1)-(2) to obtained the manipulated variable.

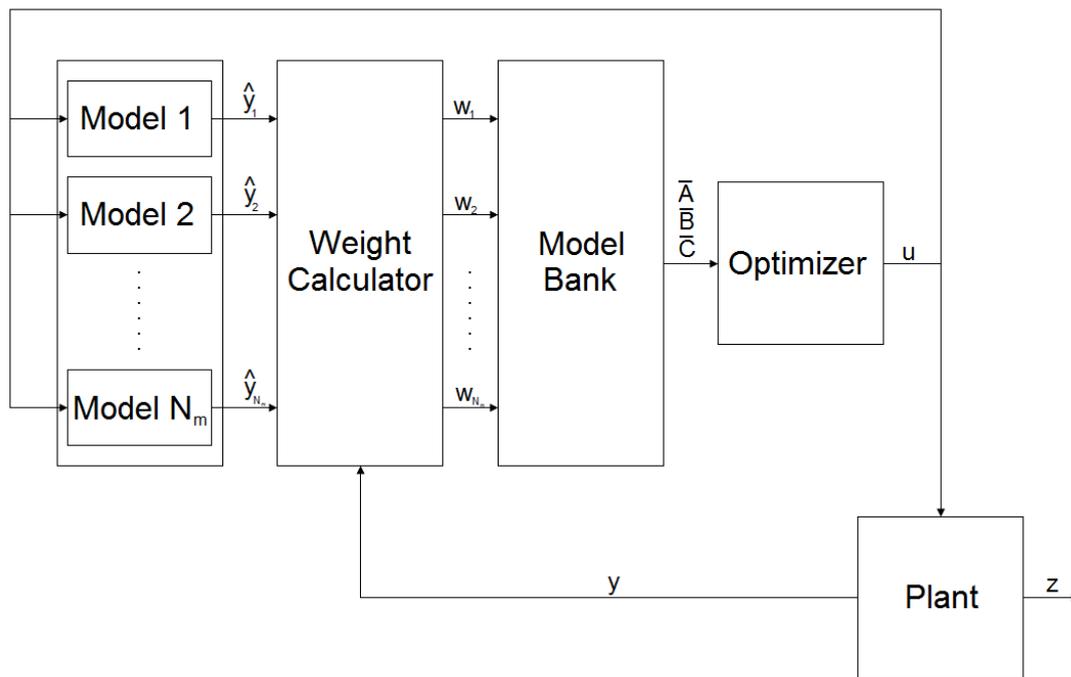


Figure 1. Multi-Model Predictive Control (MMPC) block diagram

3.2 MMPC with I-controller

The MMPC setup would generally not be able to achieve offset-free control. One reason would be the nonlinearity of the system but also because that the states are usually not known. This combination means that there would be a discrepancy between the model values and the system that. This discrepancy would mean that the MPC would generate a suboptimal solution. On top of that, other uncertainties like disturbances may occur as well. Therefore, whenever there are any uncertainties, there will be an offset of the desired controlled variable. This offset is treated in the same manner as in classical control theory by introducing integral action to the controller. In this case, the integral action is incorporated into the system as an entry parallel to the MMPC approach, resulting in a control scheme as presented in figure 2.

The integral controller used is based will be based on an adaptive approach whereby an I-controller is designed for each model and then weighted together utilizing the Bayesian weighting (6)-(9). As the control system following the MPC is on a discrete basis, the I-controller used will be in a discrete form as well:

$$u_I(k) = \sum_{i=1}^L \frac{\lambda_i}{t_s \tau_{I,i}} \sum_{j=0}^k e(j) \quad (10)$$

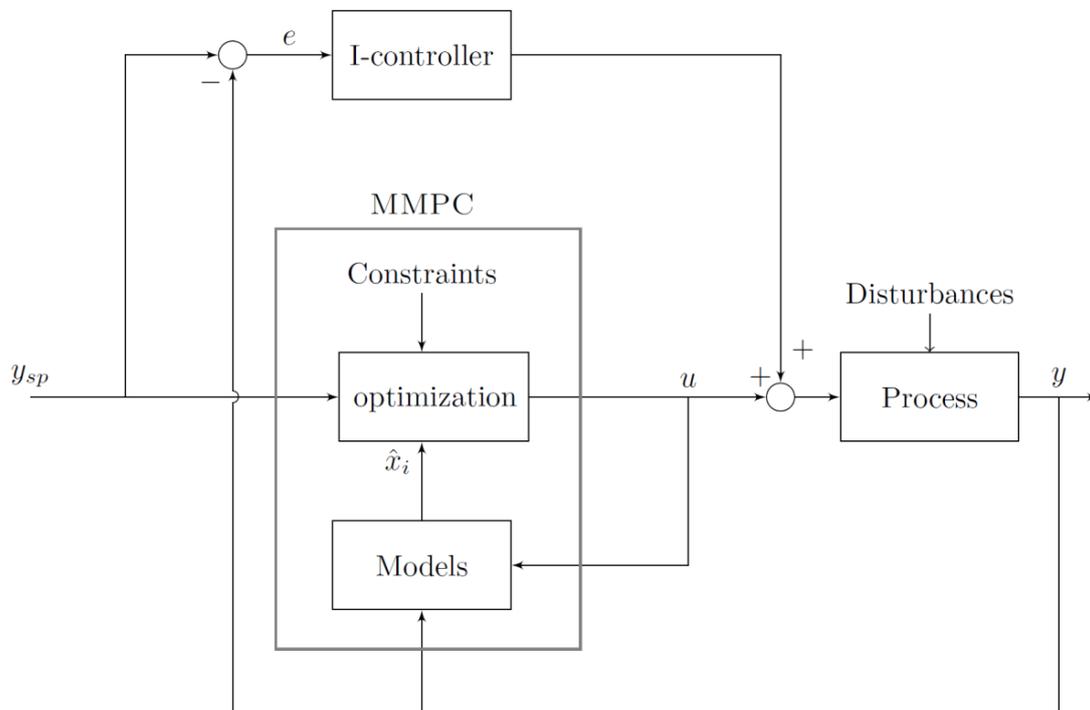


Figure 2. Block-diagram: MMPC with I-control

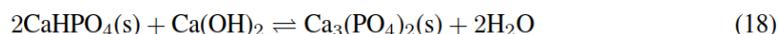
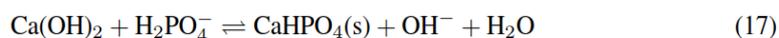
where τ_i is the reset time, the control variable for integral control, for each model i . The factor t_s is the number of sampling intervals per unit time. The reset time for each model is obtained by first changing each of model from state-space into transfer function form, which is used for internal model control tuning. The obtained controller is PI-controller with values K_c^- and τ_i^- . The proportional part is assumed handled by the MMPC setup and its feedback of the state, and hence only the integral part is used with the reset time obtained as the quotient τ_i^-/K_c^- . Lastly, the reset time is reverted to state space and going through discretization to fit in with the form used in (10). The MMPC with I-controller (MMPC-I) setup as depicted in figure 2 is thus described.

4. APPLICATION TO A PH NEUTRALIZATION PROCESS

The process considered is a continuously stirred tank reactor (CSTR) assumed to have a constant volume and that the feed flow has constant magnitude. This will lead to constant retention time, τ , provided that the control stream remains small in comparison to the feed stream, which is also assumed. The feed stream and control streams consist of aqueous solutions of Phosphoric acid (H_3PO_4) and Calcium hydroxide ($\text{Ca}(\text{OH})_2$) respectively. The neutralization reactions are modeled based on the approach proposed by Gustafsson et. al. [10], [9], [20]. In this case, the components of the acid-base system of interest are H^+ , H_3PO_4 , H_2PO_4^- , HPO_4^{2-} , PO_4^{3-} , OH^- . The reactions relating to those components are considered to be instantaneous, and hence the system is at equilibrium all the time and given by:



However, the system is further complicated by the formation of salts (CaHPO_4 and $\text{Ca}_3(\text{PO}_4)_2$) according to:



The salts will either precipitate into a solid form or remain in the solution depending on pH.

The system is then modeled using the reaction invariant approach whereby the vector, X contains the concentration of five variables; charge, phosphate, calcium, and the two salts, given as

$$\begin{aligned} \frac{d}{dt}X(t) &= \frac{1}{\tau} (X_f(t) + X_u(t - d_1) - X(t)) + r(X(t)) \\ Y(t) &= \varphi(X(t - d_2)) \end{aligned} \quad (19)$$

where Y is the pH in the tank, φ is nonlinear function describing the pH, X_f is concentration of phosphoric acid in the feed and considered the disturbance, X_u is the controlled flow rate of calcium hydroxide into the system divided by feed flow rate and $r(X(t))$ is the precipitation of phosphate and calcium into solids. The system is assumed to include two time delays, d_1 , which is the delay in the control actuation, and d_2 , which is a measurement delay. Both time delays are assumed to be of magnitude 0.5 min. (19) is the model used for the simulation. The controller, however, require that a set of discrete linear equations describes the system. The linearized models were derived for the feed having a phosphoric acid concentration of 0.01 mmol/l. The model set was produced by linearizing around five base points (pH=3, 4, 5, 6 and 7) applying deviation variables for X , X_f , X_u and Y for each model. The model set given by [20] is continuous and was discretized with a sampling time of 0.5 min to give a model form for (19) given as

$$\begin{aligned} x(k+1) &= A_i x(k) + B_{iu} u(k-1) \\ y(k) &= C_i x(k-1) \end{aligned} \quad (20)$$

where i is the model number. This model system was the one used in the MMPC-I controller.

4.1 Simulation Results

Matlab was used to simulate the process as well as the controller. The ode-solver in Matlab was used to simulate the system specified by (19) and MMPC-I approach as specified by (1)-(2) using the quadratic optimization solver using the models as specified by (5)-(9) and (20) as well as the description of the I-controller given by (10). For comparison, an MMPC was run without the addition of the I-controller as well as a single model MPC with I-controller. The weighting matrices Q and R , was set to be identity matrices, which puts equal weighting on the state, x , and the control action, u , as they are deviating from their respective set-points.

The simulation studies two different cases. Firstly, a simple set point tracking was carried out stepping 3-4-5-6-7-6-5-4-3 set point tracking, as shown in figure 3. As the only uncertainty, in this case, is in the form of the tracking problem both the MMPC and the MMPC-I are performing well. It should be noted that the MMPC is faster as it does not include the I-controller, which slows down the controller. The single model MPC have some problem being slower than the MMPC-I as a single model would create more modeling error. The PID controller, which was tuned at pH=5 also slows down the response. As the single PID controller had to be able to handle the full pH range a large reset time is needed around pH=5, which will create a sluggish system around pH=7 or pH=3. Secondly, to make things more complicated for the controller, the tracking problem is combined with a disturbance. The disturbance takes the form of a 10% decrease in the flow of the Phosphoric acid occurring between 20 min and 370 min. The responses for the different controller is shown in figure 4, with input behaviour, is shown in figure 5. The disturbance is. The major change seen is that the MMPC does not achieve very good

control any longer as there is no part of the controller that can handle disturbances. The level of control

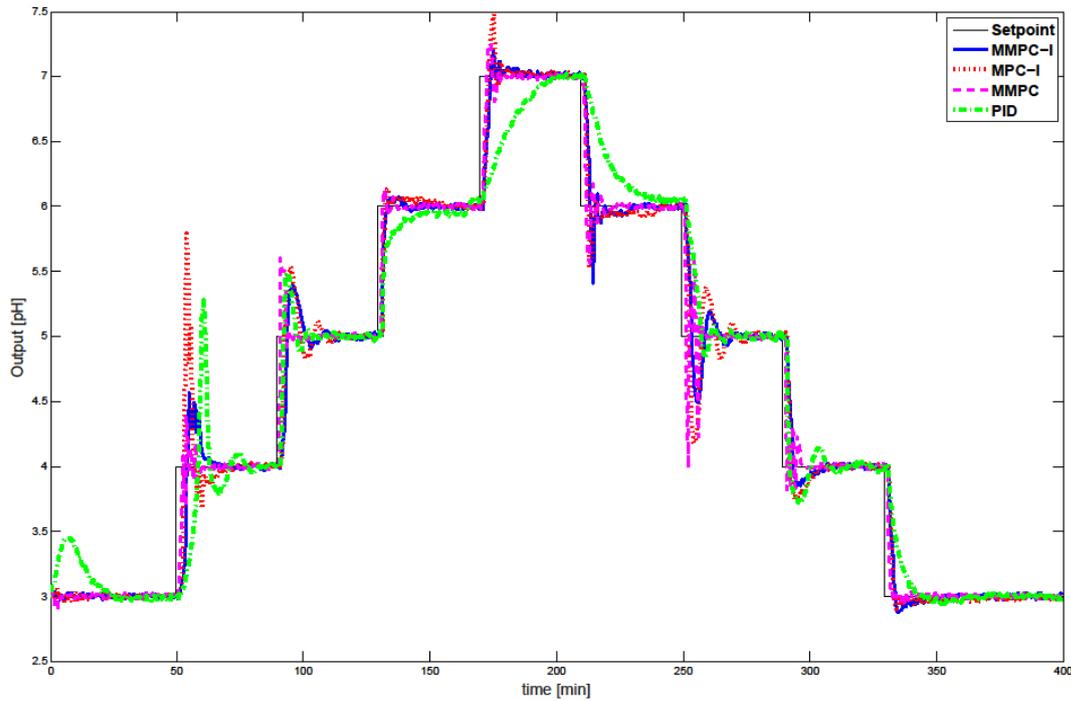


Figure 3. MMPC-I behaviour for the servo problem stepping from 3 to 7 and back again.

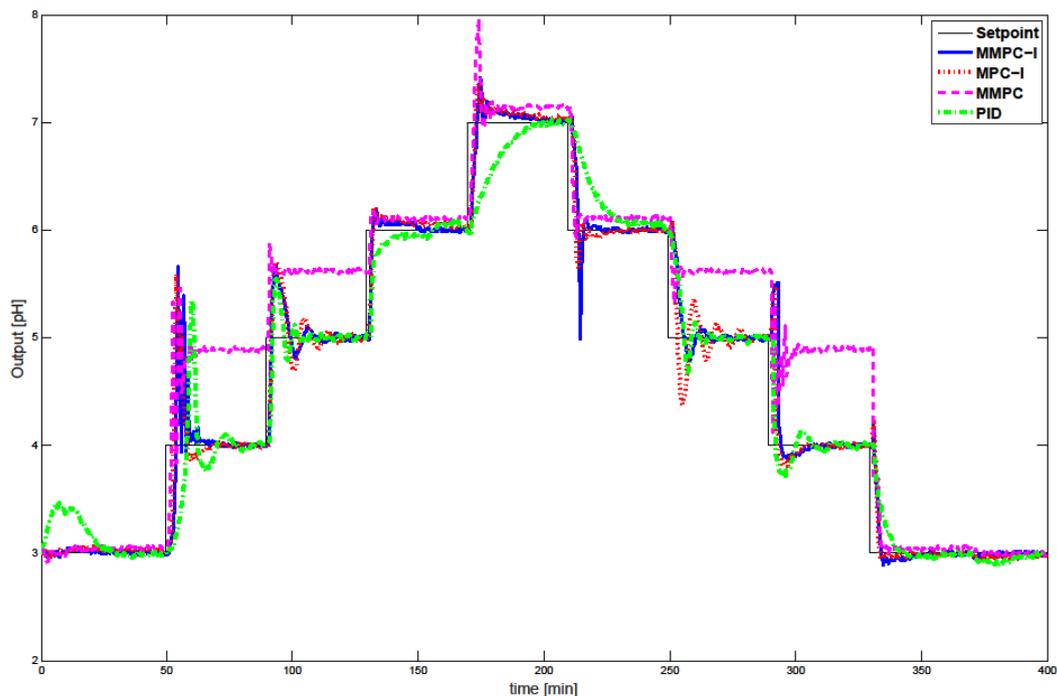


Figure 4. MMPC-I behaviour for the servo and regulatory problem stepping from 3 to 7 and back again. Assuming correct model.

for the other controllers have deteriorated as well, but to a much smaller extent. This is mainly seen in increased oscillations as well as longer settling time especially in the mid-range (pH=4-6). The analysis

is confirmed using root mean squared error (RMSE). The RMSE is 0.25 for the MMPC plus I-controller, 0.22 for the MMPC without I-controller, 0.31 for the single MPC with I-controller and 0.32 for the PID controller in the first case. The RMSE for the second case is 0.25, 0.54, 0.50 and 0.32 for the respective controller. The proposed setup is outperforming the other two techniques, but there are still some problems. Particularly the significant overshoots at pH, 4, 5 and 7, which are caused by the steep gradient of the process at those pH, requiring a small integral action to avoid instability.

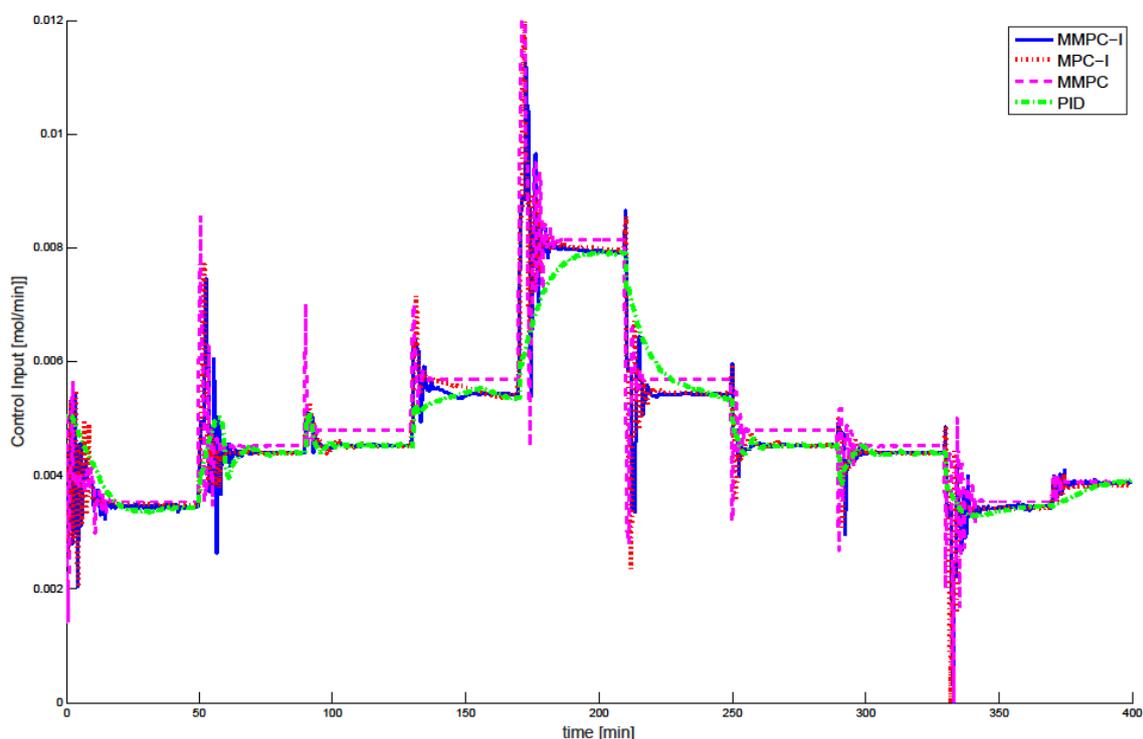


Figure 5. The control inputs for the servo/regulatory-problem specified in figure 4.

5. Conclusion

In this paper, an MMPC approach in parallel with an adaptive I-controller was presented to produce offset-free control. The MMPC-I setup is based on a PWL description to ensure that the nonlinear behaviour is contained in the model while keeping the model linear. The PWL models are weighted together using Bayesian weighting creating an adaptive behaviour increasing accuracy away from the operating points. Integral action is included by running an adaptive I-controller in parallel with the MMPC. The I-controller is designed using classical tuning techniques and made adaptive by utilizing the Bayesian weighting of the different I-controllers.

The performance of the proposed controller is applied to a pH process with multi-protic acid and base including precipitation, creating a good test for the controller through its severe nonlinearity. The MMPC-I proved to outperform the standard PID-controller as well as more straightforward versions of MPC. The performance advantage creates less overshoot as well as faster settling.

References

- [1] J. Åkesson and P. Hagander. Integral action—a disturbance observer approach. In *European Control Conference (ECC), 2003*, pages 2577–2582. IEEE, 2003.
- [2] B. Aufderheide and B. Bequette. Extension of dynamic matrix control to multiple models. *Computer and Chemical Engineering*, 27:1079–1096, 2003.
- [3] B. W. Bequette. Non-linear model predictive control: A personal retrospective. *The Canadian*

Journal of Chemical Engineering, 85:408–415, 2007.

- [4] E. Camacho and C. Bordons. *Model Predictive Control*. Springer, Berlin, second edition, 2004.
- [5] C. Cutler and B. Ramaker. Dynamic matrix control—a computer control algorithm. In *Proceedings of the joint automatic control conference*, volume 1, pages Wp5–B. American Automatic Control Council Piscataway, NJ, 1980.
- [6] A. Ferramosca, A. González, and D. Limon. Offset-free multi-model economic model predictive control for changing economic criterion. *Journal of Process Control*, 54:1–13, 2017.
- [7] A. González, E. Adam, and J. Marchetti. Conditions for offset elimination in state space receding horizon controllers: A tutorial analysis. *Chemical Engineering and Processing: Process Intensification*, 47(12):2184–2194, 2008.
- [8] G. Goodwin, H. Kong, G. Mirzaeva, and M. Seron. Robust model predictive control: reflections and opportunities. *Journal of Control and Decision*, 1(2):115–148, 2014.
- [9] T. Gustafsson, B. Skrifvars, K. Sandström, and K. Waller. Modeling of pH for control. *Industrial Engineering Chemistry Research*, 34:820–827, 1995.
- [10] T. Gustafsson and K. Waller. Dynamic modeling and reaction invariant control of pH. *Chemical Engineering Science*, 38(3):389–398, 1983.
- [11] R. Huang, L. Biegler, and S. Patwardhan. Fast offset-free nonlinear model predictive control based on moving horizon estimation. *Industrial & Engineering Chemistry Research*, 49(17):7882–7890, 2010.
- [12] J. Maciejowski. *Predictive control with constraints*. Prentice Hall, Englewood Cliffs, NJ, 2002.
- [13] U. Maeder, F. Borrelli, and M. Morari. Linear offset-free model predictive control. *Automatica*, 45(10):2214–2222, 2009.
- [14] D. Mayne. Model predictive control: Recent developments and future promise. *Automatica*, 50(12):2967–2986, 2014.
- [15] M. Mohammadkhani, F. Bayat, and A. Jalali. Two-stage observer based offset-free MPC. *ISA Transactions*, 57:136 – 143, 2015.
- [16] M. Mohammadkhani, F. Bayat, and A. Jalali. Robust output feedback model predictive control: A stochastic approach. *Asian Journal of Control*, 19(6):2085–2096, 2017.
- [17] M. Morari and J. Lee. Model predictive control: Past, present and future. *Computers & Chemical Engineering*, 23:667–682, 1999.
- [18] M. Morari and U. Maeder. Nonlinear offset-free model predictive control. *Automatica*, 48(9):2059–2067, 2012.
- [19] K. Muske and T. Badgwell. Disturbance modeling for offset-free linear model predictive control. *Journal of Process Control*, 12(5):617–632, 2002.
- [20] R. Nyström, K. Sandström, T. Gustafsson, and H. Toivonen. Multimodel robust control of nonlinear plants: a case study. *Journal of Process Control*, 9:135–150, 1999.
- [21] G. Pannocchia. Offset-free tracking MPC: A tutorial review and comparison of different formulations. In *2015 European Control Conference (ECC)*, pages 527–532, July 2015.
- [22] G. Pannocchia and A. Bemporad. Combined design of disturbance model and observer for offset-free model predictive control. *Automatic Control, IEEE Transactions on*, 52(6):1048–1053, 2007.
- [23] G. Pannocchia, M. Gabiccini, and A. A. Offset-free MPC explained: novelties, subtleties, and applications. *IFAC-PapersOnLine*, 48(23):342 – 351, 2015. *5th IFAC Conference on Nonlinear Model Predictive Control NMPC 2015*, Seville, Spain, 17–20 September 2015.
- [24] G. Pannocchia and J. Rawlings. Disturbance models for offset-free model-predictive control. *AIChE journal*, 49(2):426–437, 2003.
- [25] J. Richalet, A. Rault, J. Testud, and J. Papon. Model predictive heuristic control: Applications to industrial processes. *Automatica*, 14(5):413–428, 1978.
- [26] M. Rodrigues and D. Odloak. MPC for stable linear systems with model uncertainty. *Automatica*, 39(4):569–583, 2003.
- [27] P. Tatjewski. Disturbance modeling and state estimation for offset-free predictive control with state-space process models. *International Journal of Applied Mathematics and Computer Science*, 24(2), 2014.
- [28] X. Tian, P. Wang, D. Huang, and S. Chen. Offset-free multistep nonlinear model predictive control under plant–model mismatch. *International Journal of Adaptive Control and Signal Processing*, 28(3-5):444–463, 2014.