Temperature regulation of a pilot-scale batch reaction system via explicit model predictive control

Javier Sanchez-Cossio, Juan David Ortega-Alvarez
Departamento de Ingeniería de Procesos
Universidad EAFIT
Medellín, Colombia
Email: {jsanch10, jortega}@eafit.edu.co

Carlos Ocampo-Martinez
IEEE Senior Member
Institut de Robòtica i Informàtica Industrial (CSIC-UPC)
Universitat Politècnica de Catalunya (UPC)
Barcelona, Spain
Email: cocampo@iri.upc.edu

Abstract—In this paper, the temperature of a pilot-scale batch reaction system is modeled towards the design of a controller based on the explicit model predictive control (EMPC) strategy. Some mathematical models are developed from experimental data to describe the system behavior. The simplest, yet reliable, model obtained is a (1,1,1)-order ARX polynomial model for which the mentioned EMPC controller has been designed. The resultant controller has a reduced mathematical complexity and, according to the successful results obtained in simulations, will be used directly on the real control system in a next stage of the entire experimental framework.

I. INTRODUCTION

Several industrial systems within the process control framework are regulated via traditional control strategies, which have been worked on conveniently during last years according to particular management objectives and operation criteria. However, new strategies and methodologies to improve the performance of processes and devices have raised in order to innovate in the traditional industrial environments. To this end and having available a complete, robust and versatile pilot-scale batch reaction system, it is worthy to perform research towards the potential implementation of existing control techniques in such a way that real-time controllers may be implemented and potentially transferred to the actual implementation within real processes. The Department of Process Engineering at EAFIT University in Medellín (Colombia) has an equipment for studying batch processes from which the temperature loop is here considered. This loop should maintain the temperature inside the reaction vessel close to a reference value by manipulating some system devices. To accomplish this, the model-based predictive control (MPC) strategies are considered since they have had a significant impact in industrial application due to their ability to tackle complex problems, optimizing the behavior of the plant by using a dynamical model to predict its response to the control signals and thus selecting the most suitable value according to the objectives defined. Renowned chemical companies like Repsol YPF have implemented model-based predictive controllers to large-scale batch reactors with advantageous results, e.g., improved product quality, shorter batch duration, and higher equipment availability [1].

Different works have highlighted the limitations of traditional controllers dealing with batch reaction systems: actuator delays, temperature overshoot during the heating-up stage

and significantly different dynamics between the heating and cooling stages, just to mention some of them [1], [2]. Often, these works point out to MPC controllers as the best alternative to overcome that mentioned issues. Moreover, MPC is posed as a suitable alternative to design decentralized multivariable PID controllers in applications like biodiesel reactors, where complex decoupling techniques are required, yet not completely effective [3]. Many of this works provide with evidence of the advantages of MPC in such systems using simulations [4], [5], [6] while some others moved to real-scale implementation with success [1], although few is available on dealing with the computational requirements of such implementation in average industrial control equipment. Ho et al. [3] used centralized adaptive generalized predictive control (CA-GPC) to handle multiple control loops of a transesterification batch reactor, instead of using a decentralized multiple-controller strategy. They draw upon a validated phenomenological model of the transesterification process and used ARX models to both represent and simplify its different components. Results from simulations show that the performance of the CA-GPC controller was far more effective for temperature control than a well-tuned decentralized PID, although just comparable for the control of methyl-ester concentration. Nevertheless, the authors acknowledge the computational complexity inherent to the real implementation of such control scheme and propose the lab-scale implementation as the next step.

The work reported in this paper comprises the development and results analyses of the design of an explicit MPC controller (EMPC) for temperature control of the aforementioned pilot-scale batch reactor. The main contribution of this paper consists in making the designed EMPC controller streamlined enough to be implemented in the equipment as simple as a medium-profile PLC without losing its advantages in performance. The controller reported here has a reduced mathematical complexity and will be used directly on the real control system powered by an S7-300 PLC.

The remainder of this paper is organized as follows. In Section II, a short functional description of the case study is presented in order to clarify the structure of the physical system and establish some guidelines for its modeling. Then, in Sections III and IV, some autoregressive models formulated for the system are presented. From the simplest ARX model obtained, the EMPC controller is designed in Section V. Finally, some conclusions are drawn in Section VI.
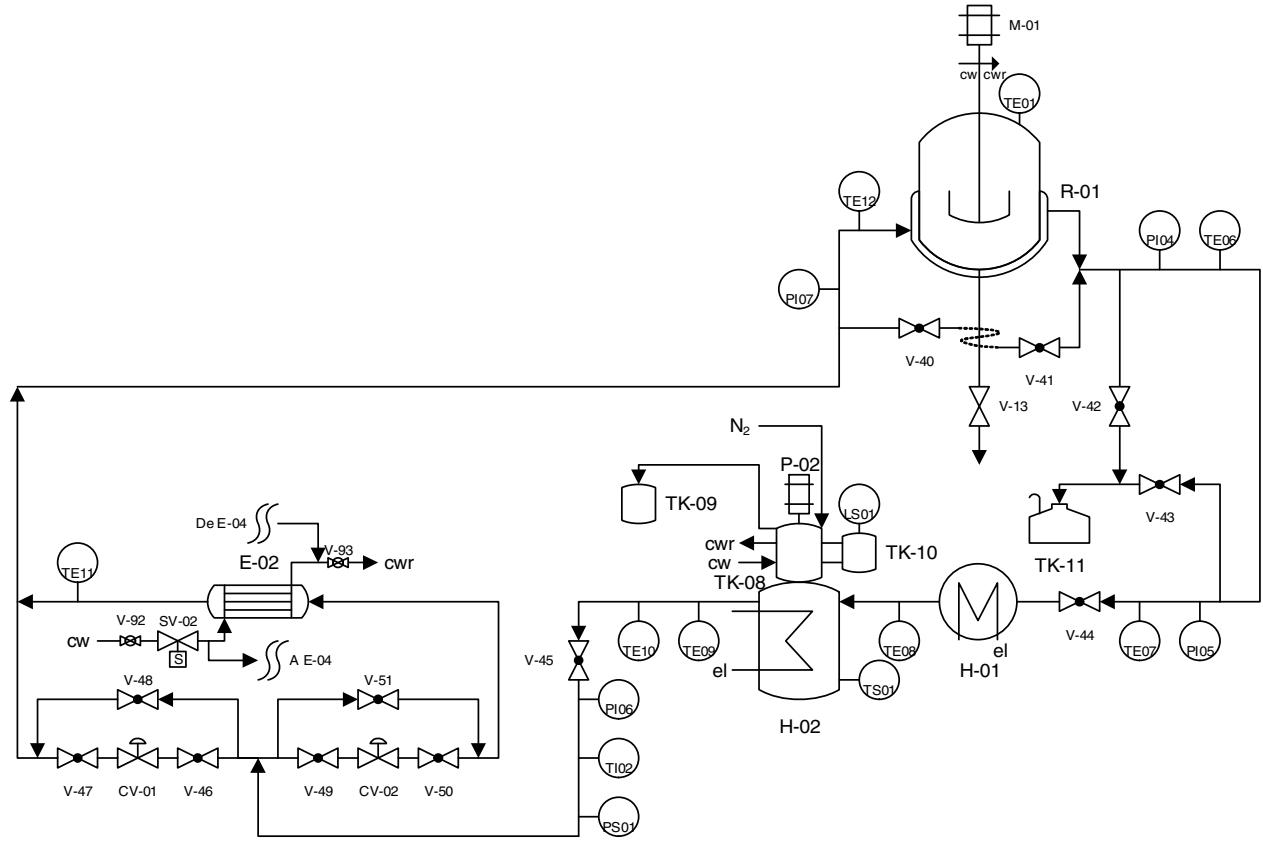
Figure 1. Process flow diagram of the case study

II. SYSTEM DESCRIPTION

The temperature control equipment studied consists of a thermal fluid circuit that supplies the required thermal energy to the reaction vessel or removes surplus heat according to the process needs. The final goal is to maintain the temperature inside the reaction vessel close to a reference value by manipulating some system devices (actuators). A process flow diagram (PFD) for this system is presented in Figure 1.

The thermal fluid is heated by a couple of electrical coil heaters (H-01, H-02) and pumped by P-02. The fluid then flows through pipeline towards the jacketed reaction vessel and back to the heaters. Along the way, there are several elements, such as valves and temperature/pressure sensors, which allow the measurement of some variables and the manipulation of the flow. After passing through the heaters, the fluid goes to the jacket through one of two paths: the former delivers it directly while the latter passes through a heat exchanger where the fluid can be cooled down if needed by means of cooling water. Thus, the action of heating the fluid is accomplished by turning on the electrical coils inside the heaters and cooling is reached by enabling the flow through the cooling water heat exchanger (E-02).

The temperature of the thermal fluid after the electric coil heaters is measured by the sensor TE-10 and, before entering the vessel's jacket, it is measured by TE-12. Sensor TE-06 is located at the jacket output, while TE-07 is placed before the heaters inlet. The controlled variable is the temperature of the

reacting mixture inside the vessel, measured using the sensor TE-01. The manipulated variables are heating and cooling actions of the thermal fluid, accomplished as described.

This equipment uses a SIEMENS S7-300 PLC for the monitoring and control of sensors and actuators and has a touchscreen panel for operator interaction. The data from temperature sensors is stored every minute, a value that is used as the sampling time for the controllers given that this is a relatively slow process and this time value yields in a suitable tracking of the process temperature behavior.

III. ARX MODELING

On an early stage of the system modeling, a phenomenological approach was proposed. However, EMPC controllers require simple models and, given that experimental datasets are available, polynomial models have been adjusted. The models used for the simulation of the system and the design of the controllers are now described. In general, the temperature of a body is an expression of the amount of thermal energy it contains. Thus, given that the thermal energy at any point of the system and at a given time depends on the energy it had at previous instants and the energy inputs or outputs affecting it, the temperature at any moment can be modeled as a function of the previous temperature(s) and the variables affecting its energy content. In this way, autoregressive exogenous –ARX– models can be used [7], obtaining their coefficients from experimental data.

ARX models are linear difference equations of the form

$$A(q) y(t) = B(q) u(t) + e(t), \quad (1)$$

where $A(q)$ and $B(q)$ are polynomials in the delay operator (q^{-1}) of order according to the desired order for the model (na, nb, nk). na is the order of the polynomial $A(q)$ and nb is the order of the polynomial $B(q)$. These determine the order of the delay operator in each case. nk is the input-output delay. This structure implies that the current output is predicted as a weighted sum of past output values and current and past input values [7],[8].

To describe the case study of this paper by using ARX models, a set of *control volumes*[1] was defined and, for each one, an ARX model was calculated. The corresponding coefficients were obtained from a set of experimental data using *System Identification Toolbox* in MATLAB [8]. A suitable set of models for the considered variables was produced. The control volumes are defined as follows (according to Figure 1):

- The reaction vessel, where the reaction mixture is contained and the reaction takes place. Its temperature, T_{01}, is influenced by the transfer from the thermal fluid and the loss to the environment. There can also be some reaction heat but, since this is a preliminary work, it is neglected here. This control volume will be referred to with the subscript R.

- The reaction vessel jacket, where the transfer between the thermal fluid and the reaction mixture takes place to either supply or remove the required energy. This control volume will be referred to with the subscript j and its boundaries are devices TE-12 and TE-06.

- The piping and equipment between the output of the vessel jacket and the output of the fluid heater H-02. This control volume is bound by devices TE-06 and TE-10 and will be referred to as p_1. Here, the thermal fluid is heated by using electrical coils and some loss to the environment also occurs.

- The piping and equipment between the output of the fluid heater H-02 and the inlet to the vessel jacket. This control volume is referred to as p_2 and is bound by TE-10 and TE-12. The thermal fluid cooler E-02, which removes energy from the thermal fluid when it is needed to cool down the vessel, is placed here. Some loss to the environment also occurs.

The measured temperatures are, accordingly, T_{01}, T_{06}, T_{10} and T_{12}, being T_{01} the output whose behavior is controlled. As said, the manipulated inputs are the heating and cooling actions of the thermal fluid. Heating is accomplished by turning on the electric heaters. These heaters have a fixed power and thus, for modeling purposes, a variable H can be assigned to represent heating and the regression from experimental datasets will give suitable coefficients. Accordingly, cooling, that is accomplished by diverting the fluid flow through E-02 by using CV-01 and CV-02, is represented by the variable cw. These valves operate as On/Off valves for the scope of this work and

flow variation will be subject to future work. The inputs H and cw would then be binary variables but, due to the simplicity and reduction of computational burden pursued in this work, they are just limited between 0 and 1 and fractional values would mean that either heating or cooling is enabled for a fraction of the time between samples.

For the first control volume, the reaction vessel, the output variable is the temperature of the reaction mixture denoted by T_{01}. The energy input is given by the transfer from the thermal fluid passing through the jacket and the energy output is caused by the loss to the environment since the temperature of the thermal fluid through the jacket is higher than T_{01}. When T_{01} is higher, the heat transfer occurs in the opposite direction, cooling the mixture. However, the model is still valid since, in that case, the temperature difference is negative and so is this energy *input*. Both energy transfers are driven by the temperature difference between the measured temperature and the corresponding fluid –thermal fluid for the energy input and environment air for the energy output– and thereby these temperature differences are the model inputs.

The temperature difference for the heat transference phenomena taking place in the system is calculated as the *Log Mean Temperature Difference –LMTD–*, since its features are quite similar to those in a heat exchanger. The expression for $LMTD_i$, which accounts for the fact that the temperature difference along a heat exchanger is not uniform, is

$$LMTD_i = \frac{(T_{in} - T_\infty) - (T_{out} - T_\infty)}{\ln\left(\frac{T_{in} - T_\infty}{T_{out} - T_\infty}\right)}, \quad (2)$$

where i refers to the control volume for which this temperature difference is calculated, T_{in} is the temperature of the thermal fluid at the inlet of the control volume, T_{out} is the temperature of the thermal fluid at the outlet of the control volume and T_∞ is the temperature of the substance to/from which heat is transferred. This substance can be the reaction mixture, whose temperature is T_{01}, or the surrounding air, whose temperature is T_e. Nevertheless, for the particular case of the loss from the reaction mixture to the environment, the simple temperature difference is used since, in this case, both temperatures – mixture (T_{01}) and environment (T_e)– are uniform. Thus, $\Delta T_R = T_{01} - T_e$.

Therefore, for the first control volume, $y(t) = T_{01}(t)$ and $u(t) = \{LMTD_j(t), \Delta T_R(t)\}$, where

$$LMTD_j(t) = \frac{(T_{12}(t) - T_{01}(t)) - (T_{06}(t) - T_{01}(t))}{\ln\left(\frac{T_{12}(t) - T_{01}(t)}{T_{06}(t) - T_{01}(t)}\right)}, \quad (3)$$

yielding

$$A_R(q) T_{01}(t) = B_R(q)_{\{1\}} LMTD_j(t) + B_R(q)_{\{2\}} \Delta T_R(t). \quad (4)$$

The main difference between this one and the remaining control volumes is that those are *open* control volumes in the sense that there is a flow coming in and out its boundaries. As a result, the temperature of the incoming fluid is an additional input for the calculation of the model output, which is the outlet temperature.

For the reaction vessel jacket (control volume j), the model output is then the temperature of fluid leaving it (T_{06}),

[1]Arbitrary portions of the system, enclosed by declared boundaries, selected to study it.

measured by the sensor TE-06. The inputs are the temperature difference with the reaction mixture ($LMTD_j$), to account for the energy transferred to it, and the temperature difference with the surrounding air ($LMTD_e$) to account for the loss to the environment, in addition to the temperature of the fluid coming in, T_{12}. Thus, $y\left(t\right) = T_{06}\left(t\right)$ and $u\left(t\right) = \left\{T_{12}\left(t\right), LMTD_j\left(t\right), LMTD_e\left(t\right)\right\}$. Hence, the model for this control volume is then

$$
\begin{aligned}
A_j\left(q\right) T_{06}\left(t\right) = & B_j\left(q\right)_{\{1\}} T_{12}\left(t\right) \\
& + B_j\left(q\right)_{\{2\}} LMTD_j\left(t\right) \\
& + B_j\left(q\right)_{\{3\}} LMTD_e\left(t\right),
\end{aligned} \tag{5}
$$

where

$$
LMTD_e\left(t\right) = \frac{\left(T_{12}\left(t\right) - T_e\left(t\right)\right) - \left(T_{06}\left(t\right) - T_e\left(t\right)\right)}{\ln\left(\frac{T_{12}(t) - T_e(t)}{T_{06}(t) - T_e(t)}\right)}. \tag{6}
$$

The remaining control volumes have similar characteristics, with the addition of the manipulated inputs of the system as inputs to the models: thermal fluid heating (H) and cooling (cw). Hence,

$$
\begin{aligned}
A_{p_1}\left(q\right) T_{10}\left(t\right) = & B_{p_1}\left(q\right)_{\{1\}} T_{06}\left(t\right) \\
& + B_{p_1}\left(q\right)_{\{2\}} H\left(t\right) \\
& + B_{p_1}\left(q\right)_{\{3\}} LMTD_{p_1}\left(t\right)
\end{aligned} \tag{7}
$$

and

$$
\begin{aligned}
A_{p_2}\left(q\right) T_{12}\left(t\right) = & B_{p_2}\left(q\right)_{\{1\}} T_{10}\left(t\right) \\
& + B_{p_2}\left(q\right)_{\{2\}} cw\left(t\right) \\
& + B_{p_2}\left(q\right)_{\{3\}} LMTD_{p_2}\left(t\right),
\end{aligned} \tag{8}
$$

where

$$
LMTD_{p_1}\left(t\right) = \frac{\left(T_{06}\left(t\right) - T_e\left(t\right)\right) - \left(T_{10}\left(t\right) - T_e\left(t\right)\right)}{\ln\left(\frac{T_{06}(t) - T_e(t)}{T_{10}(t) - T_e(t)}\right)} \tag{9}
$$

and

$$
LMTD_{p_2}\left(t\right) = \frac{\left(T_{10}\left(t\right) - T_e\left(t\right)\right) - \left(T_{12}\left(t\right) - T_e\left(t\right)\right)}{\ln\left(\frac{T_{10}(t) - T_e(t)}{T_{12}(t) - T_e(t)}\right)}. \tag{10}
$$

Experimental data were analyzed by using *System Identification Toolbox* in MATLAB to compute and validate a set of coefficients to describe the system behavior. These experimental data were obtained by manipulating the input variables of the system $\left(H, cw\right)$ and recording the output variables $\left(T_{01}, T_{06}, T_{10} \text{ and } T_{12}\right)$. Both thermal fluid heating and cooling were turned on and off sequentially to observe and record the resulting value of the temperature in the aforementioned locations. From this analysis, ARX models of order (1,2,1) were obtained for p_1 and p_2 whereas (1,1,1) was obtained for j and (1,3,1) for R. These are continuous models but they are going to be used for discrete control design. Thus, in the next section these continuous models are evaluated and compared at the sampling time instants (every minute).

Figure 2 shows the results of evaluating these models with the same inputs as used in the real system to obtain the experimental data, for the four control volumes. Dots are the measured values and solid lines show the model results. From
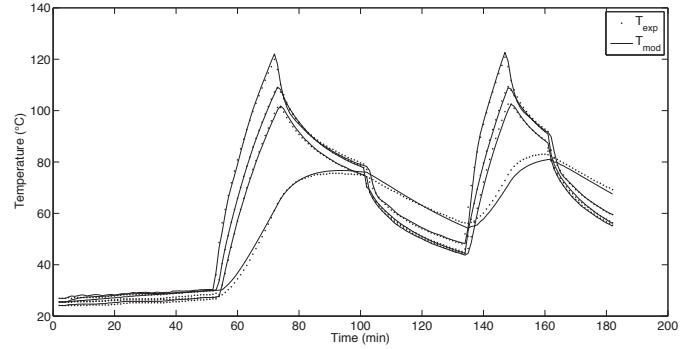

Figure 2. Experimental data and initial ARX model

the top to the bottom, the sets correspond to T_{01}, T_{06}, T_{10} and T_{12}, accordingly. Using the mean absolute percentage error, the fitting of the model to the experimental data is 96.6%, 98.6%, 97.8% and 97.7%, respectively. The fitting of this model is adequate to the purpose of this paper and this is the model used to simulate the system in controller evaluation in Section V.

IV. CONTROL-ORIENTED MODEL FACING THE CONTROL DESIGN

The model presented in Section III describes in a proper way the behavior of the system, but it poses some inconveniences to the on-line optimization required for an MPC controller design. This fact motivates the development of a simpler model to ease the aforementioned calculation. This simpler model follows the same structure than the previous one but substitutes the *Log Mean Temperature Difference* ($LMTD$), which mathematical expression is far from linear, for a simple temperature difference (ΔT) between the inlet temperature to the control volume and the temperature of the fluid to which energy is transferred (reaction mixture or environment air). The reason $LMTD$ is used for modeling a heat exchanger is that the temperature difference is not uniform along the exchanger. But, since in an ARX model this variation can be absorbed by the calculated coefficients, this change in temperature difference expressions can be suitably made.

The orders of the polynomials obtained are (1,1,1) for p_1, p_2 and j; and (1,3,1) for R. This model was validated as well and the results obtained were quite alike those in Figure 2. By doing the substitution of $LMTD$ by ΔT, the model becomes linear and it can be represented in the form [7]

$$
x\left(k+1\right) = Ax\left(k\right) + Bu\left(k\right) + f, \tag{11}
$$

where $x \in \mathbb{R}^8$ is the state vector defined as

$$
\begin{aligned}
x = & \left[T_{01}(k), T_{01}(k-1), T_{01}(k-2), T_{06}(k), T_{10}(k),\right. \\
& \left. T_{12}(k), T_{12}(k-1), T_{12}(k-2)\right]^T,
\end{aligned}
$$

besides, $u \in \mathcal{U} \subseteq \mathbb{R}^2$, where $\mathcal{U} \triangleq \left[0,1\right]^2$ is the input vector defined as $u = \left[H(k), cw(k)\right]^T$ and f is an affine factor given by CT_e. Moreover, the outputs are: $T_{01}\left(k\right) = x\left(k\right)_{\{1\}}$, $T_{06}\left(k\right) = x\left(k\right)_{\{4\}}$, $T_{10}\left(k\right) = x\left(k\right)_{\{5\}}$ and $T_{12}\left(k\right) = x\left(k\right)_{\{6\}}$.
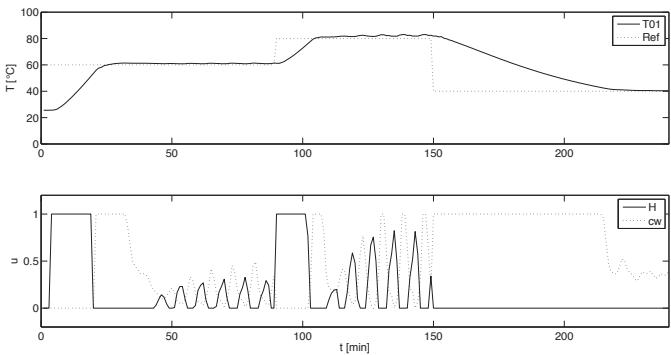
Figure 3. Performance of preliminary on-line MPC

Figure 4. Performance of EMPC controller

V. EMPC CONTROLLER DESIGN

Using the models described in Sections III and IV, some preliminary MPC controllers were developed, solving the on-line optimization problems using the `fmincon` function of MATLAB, TOMLAB optimization package for MATLAB [9] and the Multi-Parametric Toolbox 3 (MPT3) [10]. An hypothetical trajectory was established for the reference temperature to T_{01}: starting from room temperature (25°C), it is then fixed at 60 °C during 90 minutes, then it shifts to 80 °C for 60 minutes and finally goes down to 40 °C. This trajectory replicates the one of a typical batch reaction in which the reactants are heated up for mixing, then they are further heated for the reaction to take place at steady temperature and finally they are cooled down for discharge. The results and analysis for those controllers are presented in [11].

The objective function that collects the control objectives for the case study has the form

$$J(k) = \gamma_1 \sum_{k=1}^{H_p} (T_{01}(k) - T_{01}^r(k))^2 + \gamma_2 \sum_{k=1}^{H_p} (u_1(k)) + \gamma_3 \sum_{k=1}^{H_p} (u_2(k)),$$
(12)

where H_p denotes the prediction horizon, T_{01}^r is the reference trajectory and γ_i, for $i = \{1, 2, 3\}$, are the weighting factors that prioritize each term withing the multi-objective cost function (12). For all these preliminary controllers, $H_p = 10$ and $\gamma_1 = \gamma_2 = \gamma_3 = 1$. All the calculations were performed on MATLAB R2013b (8.2.0.701) using a Windows 8 PC with Intel CORE i5 3337U, 6 GB RAM. The elapsed time for this controllers was 936.89 s (`fmincon`), 126.54 s (TOMLAB) and 180.5 s (MPT3). The results for the last of this controllers are shown in Figure 3.

Then, an EMPC controller has been designed and simulated looking forward to obtaining a controller suitable to direct calculation by the existing PLC on the subject equipment. To attain that, the model used in the prediction stage of the MPC was further simplified by reducing its order on the previous section to (1,1,1) for all control volumes. Doing so, it resulted in a model of the same form as (11) but with $x \in \mathbb{R}^4$ defined as $x = [T_{01}(k), T_{06}(k), T_{10}(k), T_{12}(k)]^T$. The quality of the prediction slightly decreased but results are still similar to those in Figure 2 and this reduction in accuracy is mitigated by the feedback of the system measurements or simulation.

By using MPT3 for MATLAB [10], an EMPC is obtained for the (1,1,1)-order ARX model. To do this, the optimiza-
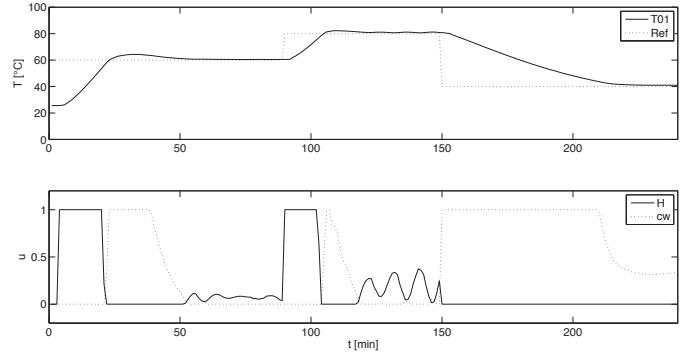
tion problem is reformulated as a MP-QP –Multi Parametric Quadratic Programming– problem where the state space is divided into several regions when solving and for each one an explicit control law is formulated, as follows [12],[13]:

$$u = F_i x + g_i \ \text{if} \ A_i x \le b_i,$$
(13)

where F, g, A and b are matrices of suitable dimensions given by the statement of the multi parametric optimization problem according to the formulation presented by [12],[13].

In addition, reference tracking can be enabled also by setting the reference as an additional parameter. It is then included as an additional element in the state vector x, resulting in $x = [T_{01}(k), T_{06}(k), T_{10}(k), T_{12}(k), T_{01}^{\text{ref}}(k)]^T$. The solution for this problem, with $H_p = 5$ to prevent the problem size from growing further, is composed of 38 regions. With a set of matrices F, g, A and b for each region. These are not further detailed here because of space limitations. The results of using it to control the system are shown in Figure 4.

Despite the reference shifts are not previewed by the controller, both controllers achieve an acceptable reference tracking considering the nature of the system. The controlled variable (T_{01}) reaches the reference quickly and with a small overshoot. However, they show a small steady-state error that may be caused by the mismatch between simulation and control models. A summary of the key performance indexes (KPI) used to evaluate the performance of the closed loop with every designed controller, is presented in Table I. As expected, computational time is much less for the EMPC than for the MPC. Besides, EMPC shows a better general performance than on-line MPC. As for the objective function (12) with $\gamma_i = 1$, EMPC yields a smaller sum of squared error, with softer changes and avoiding excessive heating and the subsequent need for additional cooling. This also reduces oscillation, as observed after time 100 minutes in Figure 3.

Table II shows some transient time performance indicators for the first and second steps in Figures 3 and 4. Rise times are quite similar since they are limited by the heating and cooling power of the equipment, not by the controller, as it can be noticed on the fact that both controllers keep heating (H) on until short before reaching the target temperature for the first time. Overshoot is higher for EMPC because it uses a shorter H_p in order to keep the number of regions in the MP-QP solution reasonably *low* looking for its implementation on the equipment, which makes it underestimate the error in

Table I.   KPI Comparison

KPI	MPC	EMPC
Computational time [s]	180.5	0.9
Total cost function	55338	52047
Sum of squared error	55175	51905
Sum of H	42.9	39.5
Sum of cw	120.6	102.6

Table II.   Performance indicators for each step

Step/Controller	Rise Time [min]	Overshoot [%]	Settling Time [min]
1st / MPC	13.3	2.3	23.1
1st / EMPC	13.2	7.1	46.5
2nd / MPC	9.7	2.4	—
2nd / EMPC	10.1	2.8	48.5

overshooting in comparison to the on-line MPC controller with a longer H_p. In addition, this higher overshoot makes the settling time longer. Moreover, due to the oscillations in H and cw caused by the MPC, T_{01} does not settle within the 5% band in the second step when using the on-line MPC, as seen in Figure 3. In general, it can be said that EMPC has a better overall performance than MPC, under the conditions of this paper.

VI.   Conclusions

This paper has presented and discussed the temperature control of a pilot-scale batch reaction system by means of the design of an explicit model predictive control (EMPC). A (1,1,1)-order ARX model has been developed to ease the calculations aiming at the implementation of the final controller on a existing PLC-based control system. An EMPC was obtained using MPT3 with this simplest model; it is composed of 38 regions and the results of using it to control the simulated system were good, though the reference shifts are no previewed by the controller. The controlled variable reaches the reference quickly and with a small overshoot. The changes on manipulated variables are softer than those of on–line optimization solutions. Some comparisons have been also performed with a traditional MPC computed online in order to see the potential differences in the system performance when a predictive-like control is computed on-line and off-line. Under the conditions of this study, the overall performance of the EMPC resulted better than that of the MPC.

The simple computation of control laws (only matrix sums and products), along with the still manageable number of regions and size of the matrices involved, makes it possible to implement the EMPC on the intended equipment, which is the subject of ongoing work. At the time of this publication, the EMPC described in section V had already been coded for the PLC on the real system using SIEMENS SCL programming language and its performance was about to be tested.

References

[1] A. Sanz, A. Cardete, R. Lucio, R. Martinez, S. M. noz, D. Ruiz, C. Ruiz, O. Gerbi, and J. Papon, "Model-based predictive control increases batch reactor production," *Hydrocarbon Processing*, no. 5, pp. 61–65, 2005.

[2] P. Jarupintusophon, M. L. Lann, M. Cabassud, and G. Casamatia, "Reallistic Model-based Predictive and Adaptative Control of Batch Reactor," *Computers & Chemical Engineering*, vol. 18, pp. S445–S449, 1994.

[3] Y. Ho, F. Mjalli, and H. Yeoh, "Multivariable adaptive predictive model based bontrol of a biodiesel transesterification reactor," *Journal of Applied Sciences*, vol. 10, no. 12, pp. 1019–1027, 2010.

[4] S. Lucia, T. Finkler, D. Basak, and S. Engell, "A new robust NMPC scheme and its application to a semi-batch reactor example," in *Proceedings of the 8th IFAC Symposium on Advanced Control of Chemical Processes*, Furama, Riverfront, July 10–13 2012, pp. 69–74.

[5] M. Bakosova, J. Oravec, and K. Matejickova, "Model predictive control-based robust stabilization of a chemical reactor," *Chemical Papers*, vol. 67, no. 9, pp. 1146–1156, 2013.

[6] V. Agarwal, M. Gupta, U. Gupta, and R. Saraswat, "A Model Predictive Controller Using Multiple Linear Models for Continuous Stirred Tank Reactor (CSTR) and its Implementation Issue," in *Proceedings of the 4th International Conference on Communication Systems and Network Technologies*, Bhopal, India, 7-9 April 2014.

[7] L. Ljung, *System Identification: Theory for the User*, 2nd ed.   Upper Saddle River, N.J.: PTR Prentice Hall, 1999.

[8] ——, "System identification toolbox for use with MATLAB : user's guide," 1995.

[9] K. Holmström, "Tomlab – an environment for solving optimization problems in matlab," in *Proceedings for the Nordic MATLAB Conference '97*, 1997, pp. 27–28.

[10] M. Herceg, M. Kvasnica, C. Jones, and M. Morari, "Multi-Parametric Toolbox 3.0," in *Proceedings of the European Control Conference*, Zürich, Switzerland, July 17–19 2013, pp. 502–510.

[11] J. Sanchez and C. Ocampo-Martinez, "Temperature modelling and model predictive control of a pilot scale batch reaction system," Institut de Robòtica i Informàtica Industrial (IRI), Tech. Rep. IRI-TR-03-14, September 2014.

[12] A. Bemporad, M. Morari, V. Dua, and E. Pistikopoulos, "The Explicit Linear Quadratic Regulator for Constrained Systems," *Automatica*, vol. 38, no. 1, pp. 3–20, 2002.

[13] A. Bemporad, F. Borrelli, and M. Morari, "Model Predictive Control Based on Linear Programming - The Explicit Solution," *IEEE Transactions on Automatic Control*, vol. 47, no. 12, pp. 1974–1985, 2002.