

Economic MPC with Modifier Adaptation using Transient Measurements

Erika Oliveira-Silva , Cesar de Prada , Daniel Montes , Daniel Navia

PII: S0098-1354(23)00074-1  
DOI: <https://doi.org/10.1016/j.compchemeng.2023.108205>  
Reference: CACE 108205



To appear in: *Computers and Chemical Engineering*

Received date: 25 October 2021  
Revised date: 26 October 2022  
Accepted date: 23 February 2023

Please cite this article as: Erika Oliveira-Silva , Cesar de Prada , Daniel Montes , Daniel Navia , Economic MPC with Modifier Adaptation using Transient Measurements, *Computers and Chemical Engineering* (2023), doi: <https://doi.org/10.1016/j.compchemeng.2023.108205>

This is a PDF file of an article that has undergone enhancements after acceptance, such as the addition of a cover page and metadata, and formatting for readability, but it is not yet the definitive version of record. This version will undergo additional copyediting, typesetting and review before it is published in its final form, but we are providing this version to give early visibility of the article. Please note that, during the production process, errors may be discovered which could affect the content, and all legal disclaimers that apply to the journal pertain.

**Highlights:**

- New methodology (TMA) aims to enhance the cost function of the process using transient data.
- eMPC+TMA is applied in simulation and in a real-time pilot plant.
- The methodology proposed increased the profit of both case studies significantly.

Journal Pre-proof

# Economic MPC with Modifier Adaptation using Transient Measurements

Erika Oliveira-Silva<sup>a,b,\*</sup> [\[ORCID\]](#), Cesar de Prada<sup>a,b</sup> [\[ORCID\]](#), Daniel Montes<sup>a</sup> [\[ORCID\]](#), Daniel Navia<sup>c</sup> [\[ORCID\]](#)

<sup>a</sup>*Department of Systems Engineering and Automatic Control, School of Industrial Engineering, University of Valladolid, Dr. Mergelina s/n, 47011, Valladolid, Spain*

<sup>b</sup>*Institute of Sustainable Processes, Dr. Mergelina s/n, 47011, Valladolid, Spain*

<sup>c</sup>*Dpto. Ingeniería Química y Ambiental, Universidad Técnica Federico Santa María, Avd. Vicuña Mackenna, Campus San Joaquín, Santiago, Chile*

*erika.oliveira@autom.uva.es, prada@autom.uva.es, danielalberto.montes.lopez@uva.es, daniel.navia@usm.cl*

## Abstract

This paper presents a method to estimate process dynamic gradients along the transient that combined with the idea of Modifier Adaptation (MA) improves the economic cost function of the examples presented. The gradient estimation method, called TMA, aims to reduce the large convergence time required to traditional MA in processes of slow dynamics. TMA is used with an economic predictive control with MA (eMPC+TMA) and was applied in two case studies: a simulation of the Williams-Otto reactor and a hybrid laboratory plant based on the Van de Vusse reactor. The results show that eMPC+TMA could reach the plant real steady-state optimum despite process-model mismatch, due to the inclusion of the effect of process dynamics in the TMA algorithm. Despite the estimation errors, the proposed methodology improved the profit of the experimental case study, with respect to the use of an eMPC with no modifiers, by about 20% for the unconstrained case, and by 130% in the constrained case.

**Keywords:** Real-time Optimization, Modifier Adaptation, Uncertainty, Transient Measurements, MPC, Hybrid Plant.

## 1 Introduction

Real-Time Optimization (RTO) is a largely used technique to increase the industry competitiveness, fulfilling quality, environmental, and security demands. RTO is formulated as an optimization problem that uses explicit process models to calculate the value of the decision variables with economic objectives. These values are used as setpoints for low-level controllers, commonly Model Predictive Controllers (MPC). The RTO problem generally consists in optimizing an economic cost function  $\phi$  with respect to the decision

variables or inputs  $\mathbf{u}$  (with lower and upper bounds,  $\mathbf{u}^L$  and  $\mathbf{u}^U$ , respectively) and constraints  $\mathbf{g}$ , using a steady-state model as in equation (1).

$$\begin{aligned} \min_{\mathbf{u}^L \leq \mathbf{u} \leq \mathbf{u}^U} & \phi(\mathbf{u}) \\ \text{s. t. } & \mathbf{g}(\mathbf{u}) \leq 0 \end{aligned} \quad (1)$$

In the case of parametric and/or structural model-process mismatch, there is no guarantee that equation (1) will find the correct value of the process optimum. To overcome this issue, the modifier adaptation methodology (MA) proposes the modification of equation (1) to add first and zeroth-order corrections to the cost function and constraints (called modifiers) so that the optimum of the process and the model are identical, resulting in problem (2) (Marchetti et al., 2009).

$$\begin{aligned} \min_{\mathbf{u}^L \leq \mathbf{u} \leq \mathbf{u}^U} & \phi_{MA}(\mathbf{u}) = \phi(\mathbf{u}) + \boldsymbol{\lambda}_k^T(\mathbf{u} - \mathbf{u}_{k-1}^*) \\ \text{s. t. } & \mathbf{g}_{MA}(\mathbf{u}) = \mathbf{g}(\mathbf{u}) + \boldsymbol{\gamma}_k^T(\mathbf{u} - \mathbf{u}_{k-1}^*) + \boldsymbol{\varepsilon}_k \leq 0 \end{aligned} \quad (2)$$

where  $\phi_{MA}$  is the modified cost function and  $\mathbf{u}_{k-1}^*$  are the values of the decision variables applied into the process in the current steady-state. In equation (2),  $\boldsymbol{\lambda}$ ,  $\boldsymbol{\gamma}$ , and  $\boldsymbol{\varepsilon}$  are the modifiers computed as in (3), where the subscript  $p$  stands for quantities calculated with information from the plant.

$$\begin{aligned} \boldsymbol{\lambda}_k^T &= \left. \frac{\partial \phi_p}{\partial \mathbf{u}} \right|_{\mathbf{u}_{k-1}^*} - \left. \frac{\partial \phi}{\partial \mathbf{u}} \right|_{\mathbf{u}_{k-1}^*} \\ \boldsymbol{\gamma}_k^T &= \left. \frac{\partial \mathbf{g}_p}{\partial \mathbf{u}} \right|_{\mathbf{u}_{k-1}^*} - \left. \frac{\partial \mathbf{g}}{\partial \mathbf{u}} \right|_{\mathbf{u}_{k-1}^*} \\ \boldsymbol{\varepsilon}_k &= \mathbf{g}_p(\mathbf{u}_{k-1}^*) - \mathbf{g}(\mathbf{u}_{k-1}^*) \end{aligned} \quad (3)$$

The RTO+MA problem (2) is a static optimization problem that uses a different model than MPC. To avoid compatibility problems and to have benefits from considering dynamics, it is logical to think of integrating these two layers. For example, the economic aim of the RTO+MA could be used as an objective function of an economic MPC (eMPC). This integration could permit the MA modifiers update at the same frequency of the controller in order to react quicker to the disturbances that could change the economic objective. So another point to have in mind refers to how the process gradients are estimated. In traditional MA, process gradients in (3) are estimated using steady-state measurements, requiring quite a few steady states for problem (2) to reach the optimum. Therefore, when dealing with slow dynamics processes, as often happens in practice, the time required for computing corrective actions with RTO+MA can be so large that

when the the algorithm converges, the process may have changed and the MA method becomes impractical. Hence, an estimation of process gradients using transient measurements can be a good alternative to decrease the convergence time.

Recent works have presented different formulations of the RTO+MA+MPC problem. Vaccari and Pannocchia (2016) designed an eMPC algorithm with MA where the target optimization problem has the correction term to enforce the necessary condition of optimality (NCO) of the problem. In this work, the gradients were considered known. After that, Pannocchia (2018) and Vaccari and Pannocchia (2018) estimated the plant gradients adding an identification stage in which an input-output plant model is estimated, from which they compute the process gradients. To guarantee the excitation needed, a random signal was included during a period of time, increasing the convergence time of the algorithm. Then, Hernández and Engell (2019) presented another formulation to an economic control based on MA where the corrections are made in the nominal dynamic plant model instead of the objective function. They identified linear dynamic models to approximate the true plant map and then calculate the plant gradients. As the estimated linearized model is only valid in the neighborhood of the current state, the sequence of optimal control moves is limited to this region. Faulwasser and Pannocchia (2019) used the output modifier adaptation and the eMPC from previous work without the terminal constraint. The authors emphasized the importance of estimating correct plant gradients. In this work, the gradients were considered known. Recently, Vaccari et al. (2020) presented a technique for direct estimation of the modifiers using steady-state perturbations and a Broyden update algorithm. After that, Oliveira-Silva et al. (2021) presented a dynamic optimizer with MA (DOMA) and an algorithm to estimate directly the modifiers, avoiding the estimation of the process gradients, using transient information, the Dynamic Modifier Estimation (DME). DME uses an optimization algorithm executed on-line to make the estimated modifiers converge to the ones in steady state required by MA. More recently, Vaccari et al. (2021) presented a extension of Vaccari et al. (2020) to compare two techniques to estimate either plant gradients or modifiers using steady-state measurements via Broyden's update and linear regression. The eMPC schemes were tested in simulation in two benchmark examples, stressing the fact that the use of transient measurements to estimate the process gradients can be an interesting approach to speed up convergence.

As can be seen in the papers mentioned, investigating how to estimate correctly plant gradients/modifiers to be used in a framework for RTO+MA+MPC is a continuing concern. Along this line, the major objective of this paper is to contribute to that research by proposing a new method to estimate dynamic process gradients using transient measurements that can be executed with the same sampling time of the MPC, aiming to reduce the convergence time of the method in processes with slow dynamics. To do so, in this new approach, intermediate gradients are estimated using the available transient measurements.

An antecedent of this method appears in (Navia et al., 2017; Rodríguez-Blanco et al., 2017), but it was formulated for an RTO context and does not take into account dynamic effects, so only the dependence of past decision variables was considered. Because of that, optimization and control were applied at different time scales.

In this paper, that approach was reformulated from a dynamic perspective and an eMPC+MA environment, and two applications, one in simulation and other in a pilot plant working in real-time were used to show its performance. We will denote the new gradient estimation method as TMA (Transient Modifier Adaptation), which will give rise to eMPC+TMA when using it in the eMPC+MA scheme.

The paper is organized as follows. The structure of the eMPC with Modifier Adaptation used in the paper is presented in Section 2. Then, Section 3 describes the algorithm to estimate gradients with transient data, as well as its fundamentals and implementation. To evaluate its performance, the eMPC+TMA is applied to two case studies: first in Section 4 to a simulated benchmark example of the Williams-Otto reactor and then in Section 5 in real-time to a laboratory plant based on the Van de Vusse reactor. Finally, some conclusions are presented in the last section.

## 2 eMPC+MA formulation

Some eMPC formulations are based on a two-step approach: first solve an economic problem in steady state and then use the solution as setpoints for the MPC. In our formulation instead, we use a single-step approach in which the economic target at the end of the prediction horizon is driving directly the dynamic operation of the controller. The eMPC+MA is formulated with a cost function that mixes the optimizer and the controller as in problem (4). That objective function is composed of three terms, denoted as A, B, and C. (A) corresponds to the economic objective function evaluated at the end of the prediction horizon. (B) represents an MA-type first-order correction of the objective function to reduce the difference of the gradient of the economic objective function between the process and the model, where  $\lambda_k$  is defined as in (3). (C) is a regularization term to smoothen the trajectories of the decision variables proposed by the dynamic optimization as it is usual in MPC. The modified constraints of the optimization problem are  $g_{MA}$ , and the dynamic model implemented in the eMPC is  $F$ .  $\hat{x}_k$  and  $v_k$  represent estimates of the current process states and disturbances respectively, while  $y_k^T$  and  $\varepsilon_k$  are MA-type first and zeroth-order modifiers of the model-based inequality constraints respectively, which are defined as in (3). Besides,  $t_k$  is the current sampling time,  $t_{pred}$  refers to the end of the prediction horizon of the controller,  $n_{pred}$  and  $n_u$  are the numbers of sampling times of the controller used for the prediction and control horizon respectively, while  $\bar{u}$  is the value of the decision variables at  $t_{pred}$ . For further information on this formulation please refer to Oliveira-Silva et al. (2021).

$$\begin{aligned}
 & \min_{\substack{\Delta \mathbf{u}_{k+i} \\ i=0,1,\dots,n_u-1}} \underbrace{\phi(t_{pred})}_{(A)} + \underbrace{\lambda_k^T(\bar{\mathbf{u}} - \mathbf{u}_{k-1})}_{(B)} + \underbrace{\sum_{i=0}^{n_u-1} \Delta \mathbf{u}_{k+i}^T \mathbf{Q} \Delta \mathbf{u}_{k+i}}_{(C)} \\
 & s. t. \mathbf{g}_{MA} = \mathbf{g}(\bar{\mathbf{u}}) + \boldsymbol{\gamma}_k^T(\bar{\mathbf{u}} - \mathbf{u}_{k-1}) + \boldsymbol{\varepsilon}_k \leq \mathbf{0} \\
 & \quad \mathbf{F}(\dot{\mathbf{x}}, \mathbf{x}, \mathbf{u}, \mathbf{v}_k) = \mathbf{0}, \forall t \in [t_k, t_{pred}] \\
 & \quad \mathbf{u}_{k+i} = \mathbf{u}_{k+i-1} + \Delta \mathbf{u}_{k+i}, \quad i = 0, 1, \dots, n_{pred} - 1 \\
 & \quad \mathbf{u}^L \leq \mathbf{u}_{k+i} \leq \mathbf{u}^U, \quad i = 0, 1, \dots, n_{pred} - 1 \\
 & \quad \mathbf{u}(t) = \mathbf{u}_{k+i}, \quad t \in [t_{k+i}, t_{k+i+1}], \quad i = 0, 1, \dots, n_{pred} - 1 \\
 & \quad \Delta \mathbf{u}_{k+i} = 0, \quad i = n_u, \dots, n_{pred} - 1 \\
 & \quad \mathbf{x}(t_k) = \hat{\mathbf{x}}_k \quad \forall t \in [t_k, t_{k+n_u}]
 \end{aligned} \tag{4}$$

The eMPC+MA controller is implemented in an iterative way using a receding horizon strategy, which implies that problem (4) must be solved at every sampling time of the controller. Before solving (4) it is necessary to estimate the current states, the disturbances, and the first and zeroth-order modifiers. The states and the disturbances can be calculated using a moving horizon estimator (MHE) (Rawlings et al., 2019), while to calculate the value of the modifiers it is necessary to estimate first the process derivatives. These estimations must be done with the available data, measured in previous sampling times. Figure 1 summarizes the implementation of the eMPC+MA algorithm, where  $\mathbf{y}_{p_{k-i}}$  represents the vector of dependent variables measured at past instants  $k-i$  for  $i = 1 \dots n_e$ , being  $n_e$  the number of previous instants needed for the estimation. The modifiers estimated from equation (3) can be filtered to improve the convergence of the algorithm as in traditional MA (Marchetti et al., 2009).

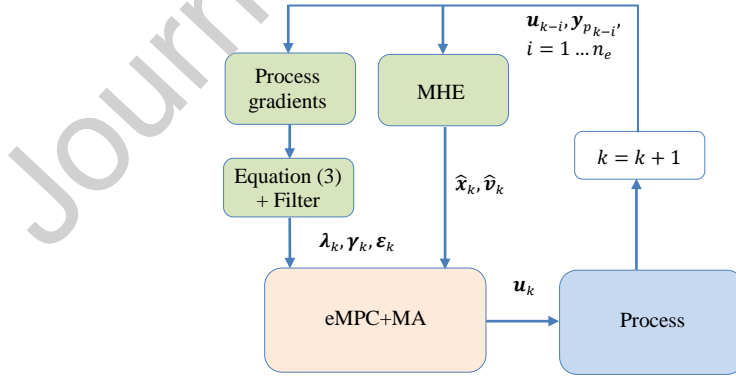


Figure 1: eMPC+MA architecture.

MHE is a dynamic optimization problem to find the best state trajectory to minimize the errors between model and measurements caused by unknown disturbances that affect the process. In this paper, the MHE has been chosen because its optimization problem is similar to a MPC problem, besides the fact that it allows the estimation of process disturbances and, if necessary, it can include constraints. The equations of the MHE problem used are presented in Appendix A.

### 3 Modifier Adaptation using transient measurements for the estimation of process gradients (TMA)

When proposing a method for estimating on-line process derivatives, the overall performance of the algorithms must be taken into account. From the point of view of the precision of the algorithm, an accurate estimation of static process gradients, which implies collecting information of the process in these conditions, allows a precise detection of the real optimum of the process. However, the time spent in the estimation is an important issue, as this affects the time operating in sub-optimal conditions. It can even make the implementation of MA impractical for processes with large settling times, if process gradients are estimated with steady-state data. Then, the use of transient measurements seems to be an interesting alternative to reduce the time needed to reach the convergence for all the RTO+MA+MPC approaches already mentioned in the introduction of this work.

For generalization purposes, let us define the variable  $\mathbf{z}_{p_k}$  as the set of process variables for which we need to estimate the derivatives i.e.,  $\mathbf{z}_{p_k}^T := [\boldsymbol{\phi}_{p_k}, \mathbf{g}_{p_k}^T]$ . The dynamical behaviour of a system from a certain time instant can be approximated as a contribution of the free dynamic evolution of the process and the effect of recent input moves from steady state. The approximation is correct for linear systems and may contain a certain error for nonlinear ones.

#### 3.1 On-line predictions of $\mathbf{z}_{p_k}$

We are interested in developing an expression for the value of  $\mathbf{z}_p$  at time  $k$  using plant information up to time  $k - 1$ . The contribution of what can be called the free dynamic evolution of a variable of a process can be computed from data using the values of the variable  $\mathbf{z}_p$  available at time  $k - 1$  and previous time instants, e.g. using a polynomial approximation that goes through the previous  $k - n$  values of  $\mathbf{z}_p$  as in Figure 2. This polynomial can then be used to extrapolate the values of the variable at time  $k$ . For



extrapolation, values of the time derivatives of  $\mathbf{z}_{p_{k-1}}$  are required, and we can perform first, second or higher-order extrapolations with these derivatives.

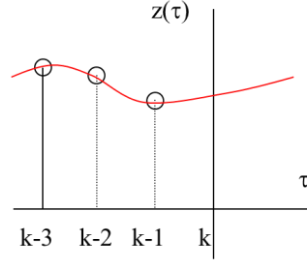


Figure 2: Polynomial approximation of a function.

In the context of numerical methods to solve time-dependent systems of differential equations (ODE), Backward Differentiation Formulas give approximations to a derivative of a dependent variable  $\mathbf{z}(t)$  at a time  $t_k$ , in terms of the value of  $\mathbf{z}_k$  and earlier values (Bank et al., 1993). In addition, Nordsieck (1962) derived a procedure based on a polynomial expression of  $n^{th}$  order to estimate the time derivatives for numerical integration methods of ODE systems. Combining these two approaches, one can implement an estimator for the time derivative of process variables using, for instance, a third-degree Nordsieck vector using past data, as presented in equation (5).

$$\begin{bmatrix} \mathbf{z}_{p_{k-1}} \\ \left. \frac{d\mathbf{z}_p}{dt} \right|_{k-1} \Delta t_{k-1} \\ \left. \frac{d^2\mathbf{z}_p}{dt^2} \right|_{k-1} \frac{\Delta t_{k-1}^2}{2} \\ \left. \frac{d^3\mathbf{z}_p}{dt^3} \right|_{k-1} \frac{\Delta t_{k-1}^3}{6} \end{bmatrix} = \frac{1}{6} \begin{bmatrix} 6 & 0 & 0 & 0 \\ 11 & -18 & 9 & -2 \\ 6 & -15 & 12 & -3 \\ 1 & -3 & 3 & -1 \end{bmatrix} \begin{bmatrix} \mathbf{z}_{p_{k-1}} \\ \mathbf{z}_{p_{k-2}} \\ \mathbf{z}_{p_{k-3}} \\ \mathbf{z}_{p_{k-4}} \end{bmatrix} \quad (5)$$

For example, a first-order estimation of the free evolution of  $\mathbf{z}_{p_k}$  could be computed using (5) as in Equation (6).

$$\mathbf{z}_{p_{k,free}} \approx \mathbf{z}_{p_{k-1}} + \left. \frac{d\mathbf{z}_p}{dt} \right|_{k-1} \Delta t = \mathbf{z}_{p_{k-1}} + \frac{1}{6} (11\mathbf{z}_{p_{k-1}} - 18\mathbf{z}_{p_{k-2}} + 9\mathbf{z}_{p_{k-3}} - 2\mathbf{z}_{p_{k-4}}) \quad (6)$$

Notice that equation (5) also calculates higher-order terms with respect to time that can be used in higher-order extrapolations.

Obviously, equation (6) reflects the influence of almost all past control actions on the evolution of the system except the effect of the recent control actions  $\Delta \mathbf{u}_{k-1}$  performed at time  $k-1$  that also impacts on the value of  $\mathbf{z}_{p_k}$ . Notice that, if there is a delay  $d$ , this comment extends to  $\Delta \mathbf{u}_{k-d}$ . So, in order to predict correctly the value of  $\mathbf{z}_{p_k}$ , we have to consider the additional effects of the changes of these control variables on  $\mathbf{z}_{p_k}$ . The corresponding changes on  $\mathbf{z}_{p_k}$ ,  $\Delta \mathbf{z}_{p_k,forced}$ , have to be estimated as if the plant were at steady state at the moment when the changes in  $\Delta \mathbf{u}_{k-1}$  took place, to make them independent of the evolution due to its past history as reflected in (6). Approximations of different orders for this component  $\Delta \mathbf{z}_{p_k,forced}$  can be obtained by means of the expansion (7), where it is clear that the  $\Delta \mathbf{z}_{p_k,forced}$  depends only on the most recent actions and not on past history:

$$\Delta \mathbf{z}_{p_k,forced} \approx \left. \frac{\partial \mathbf{z}_p}{\partial \mathbf{u}_{k-1}} \right|_{k-1} \Delta \mathbf{u}_{k-1} + \frac{1}{2} \Delta \mathbf{u}_{k-1}^T \left. \frac{\partial^2 \mathbf{z}_p}{\partial \mathbf{u}_{k-1}^2} \right|_{k-1} \Delta \mathbf{u}_{k-1} + \dots \quad (7)$$

Here  $\Delta \mathbf{u}_{k-i}$  refers to  $\Delta \mathbf{u}_{k-i} := \mathbf{u}_{k-i} - \mathbf{u}_{k-i-1}$ , and  $\left. \frac{\partial \mathbf{z}_p}{\partial \mathbf{u}_{k-i}} \right|_{k-i}$  stands for the partial derivative of  $\mathbf{z}_p$  with respect to the decision variables applied at  $t_{k-i}$ , estimated at time instant  $t_{k-i}$  starting from steady state.

Joining together (6) and (7), “free response” and “forced response”, we can write an expression for the prediction of  $\mathbf{z}_{p_k}$  based on plant data such as the equation (8):

$$\Delta \mathbf{z}_{p_k} = \mathbf{z}_{p_k} - \mathbf{z}_{p_{k-1}} = \left. \frac{\partial \mathbf{z}_p}{\partial \mathbf{u}_{k-1}} \right|_{k-1} \Delta \mathbf{u}_{k-1} + \frac{1}{2} \Delta \mathbf{u}_{k-1}^T \left. \frac{\partial^2 \mathbf{z}_p}{\partial \mathbf{u}_{k-1}^2} \right|_{k-1} \Delta \mathbf{u}_{k-1} + \left. \frac{d\mathbf{z}_p}{dt} \right|_{k-1} \Delta t + R_n \quad (8)$$

where  $\Delta \mathbf{z}_{p_k}$  represents the current change in the process variables with respect to the ones measured in the previous sampling time.  $R_n$  is a noise term that reflects the different errors that result from the polynomial approximations and for adding (6) and (7) as the system is normally nonlinear.

The interest of (8) resides in the fact that we have process data and the (unknown) process gradients related to the ones necessary for MA in the same expression. However, it is important to remark that the derivatives estimated from equation (8) are not the steady-state process gradients, but dynamic ones, that change at every time step. The derivatives in (8) describe the effect of a change in  $\mathbf{u}$  on  $\mathbf{z}_p$  from steady state at a certain time instant during the transient.

The problem now is to estimate the unknown process gradients. The terms of equation (8) can be truncated including terms up to a certain order, as in equation (9), where we considered, for instance, that the delay is one sampling period, using a second-order approximation with respect to the decision variables and a first-order one w.r.t. time.

$$\Delta \mathbf{z}_{p_k} = \left. \frac{\partial \mathbf{z}_p}{\partial \mathbf{u}_{k-1}} \right|_{k-1} \Delta \mathbf{u}_{k-1} + \frac{1}{2} \Delta \mathbf{u}_{k-1}^T \left. \frac{\partial^2 \mathbf{z}_p}{\partial \mathbf{u}_{k-1}^2} \right|_{k-1} \Delta \mathbf{u}_{k-1} + \left. \frac{d\mathbf{z}_p}{dt} \right|_{k-1} \Delta t_{k-1} + R_n \quad (9)$$

After that, one can rearrange the terms of equation (8), defining a vector  $\boldsymbol{\varphi}_{k-1}$  with known variables (measured or estimated), and the vector  $\hat{\boldsymbol{\theta}}_{k-1}$  with the coefficients to be estimated, as equation (10) shows.

$$\begin{aligned} \Delta \mathbf{z}_{p_k} &= \hat{\boldsymbol{\theta}}_{k-1}^T \boldsymbol{\varphi}_{k-1} + R_n \\ \hat{\boldsymbol{\theta}}_{k-1} &:= \left[ \left. \frac{\partial \mathbf{z}_p}{\partial \mathbf{u}_{k-1}} \right|_{k-1} \quad \left. \frac{\partial^2 \mathbf{z}_p}{\partial \mathbf{u}_{k-1}^2} \right|_{k-1} \quad \hat{\eta} \right] \\ \boldsymbol{\varphi}_{k-1}^T &:= \left[ \Delta \mathbf{u}_{k-1}^T \quad \frac{1}{2} \Delta \mathbf{u}_{k-1} \Delta \mathbf{u}_{k-1}^T \quad \left. \frac{d\mathbf{z}_p}{dt} \right|_{k-1} \Delta t_{k-1} \right] \end{aligned} \quad (10)$$

Here, vector  $\hat{\boldsymbol{\theta}}_{k-1}$  contains the unknown values to be estimated, where the gradients of the process variables with respect to the decision variables appear in the first positions of the vector. Vector  $\boldsymbol{\varphi}_{k-1}^T$  contains values of past moves known at time  $k$  or that can be computed with information up to time  $k$  as the time derivatives. To consider the estimation error of the time derivatives, an additional coefficient  $\hat{\eta}$  has been defined. As equation (10) has the form of a typical model used in parameter identification problems, current data of the measured  $\Delta \mathbf{z}_{p_k}$  and  $\boldsymbol{\varphi}_{k-1}^T$  can be used to estimate the process derivatives by solving a recursive identification problem.

During the transient, the estimated gradients are not the required gradients for MA, so that, over this time, the controller will only implement partial corrections. These partial corrections made during the transient will improve the overall performance of the real process in comparison with a standard RTO.

### 3.2 Identification algorithm

As we mentioned before, equation (10) has the form of a typical linear model used in parameter identification problems, so that we can use a recursive identification algorithm to estimate the unknown vector  $\theta$  containing the process derivatives.

Notice that  $\theta$  is a time-varying vector, as it contains the dynamic gradients, so that the recursive identification should take into account this circumstance. Also, notice that only the identified components of  $\theta$  corresponding to the first-order gradients  $\partial \mathbf{z}_p / \partial \mathbf{u}$  are required for the MA corrections. As the iterations run, if the identification algorithm performs well, and the process converges to a steady state, the identified gradients will be closer and closer to the static ones required by the MA corrections so that the system should be able to converge to the real process optimum.

Of course, the success of this scheme is linked to the ability of the identification algorithm to converge to the time-varying parameters, which depends for each particular application on both the identification method chosen and the excitation of the control signals.

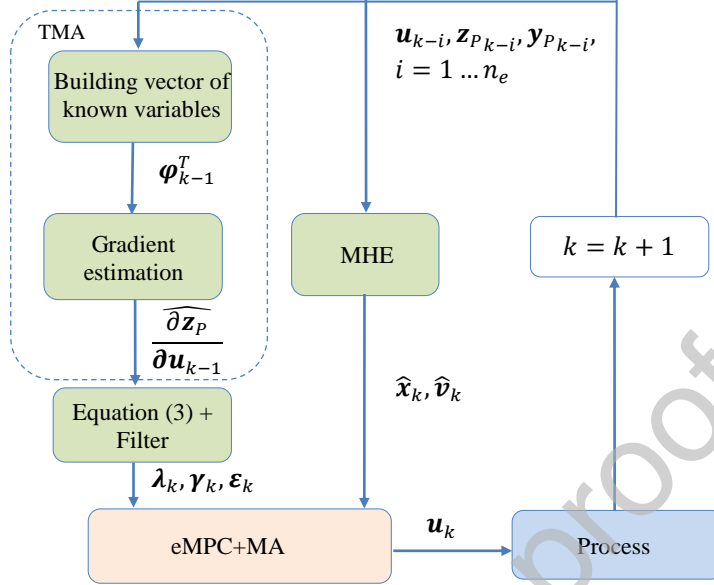
In this work, the normalized least mean square algorithm (NLMS) has been used as identification algorithm. NLMS is formulated in the parameter space, using as target to minimize directly the module of the distance,  $V$ , from the current estimate of  $\theta$  to the real value of the parameters  $\theta^*$  as in equation (11). The NLMS algorithm has been chosen because it is easy to implement, and it is computationally less expensive than other recursive methods (Isermann and Münchhof, 2011). Furthermore, the gain of the estimator is different from zero, so it can be applied to time-varying problems (Goodwin and Sin, 1984). In equation (12),  $\sigma$  is a small positive constant to prevent the numerical difficulties associated with a denominator close to zero, and  $\mu$  is a gain constant that must be between 0 and 2 in order to decrease the difference between the current estimate of  $\theta$  and the real one (Richalet, 1991).

$$V = \|\theta - \theta^*\| \quad (11)$$

$$\hat{\theta}_k = \hat{\theta}_{k-1} + \mu \frac{[\Delta \mathbf{z}_{p_{k-1}} - \boldsymbol{\varphi}_k^T \hat{\theta}_{k-1}]}{\sigma + \|\boldsymbol{\varphi}_k^T\|^2}, \quad 0 < \mu < 2 \quad (12)$$

NLMS is globally exponentially convergent to the real  $\theta$  if some conditions are fulfilled (Goodwin and Sin, 1984).

After estimating the process gradients, equation (3) is applied to finally calculate the modifiers. Figure 3 describes the two-step identification procedure proposed (TMA), coupled with the eMPC+MA



problem. In Figure 3 and hereafter, the hat over the variables stands for estimated quantities.

Figure 3: eMPC+TMA architecture.

#### 4 Williams-Otto case study

As the first case study, the Williams-Otto reactor benchmark example has been used to evaluate the performance of the proposed eMPC+TMA approach (Williams and Otto, 1960). The process consists of a continuous stirred tank reactor (CSTR) where the reactants A and B are combined to generate four species C, E, G, P in three different reactions.

Figure 4 represents the schematic of the simulated reactor operating at temperature  $T$ . In Figure 4,  $F_A$  and  $F_B$  represent the volumetric flow of the influents containing pure A and B at molar concentrations given by  $X_{A0}$  and  $X_{B0}$  respectively;  $F_R$  represents the volumetric flow of the effluent of the reactor with molar concentrations of the components denoted by  $X_i, i \in \{A, B, C, E, G, P\}$ . The system has two degrees of freedom ( $F_B$  and  $T$ ) that can be manipulated by a supervisory layer, while  $F_A$  can be considered as a measured disturbance. As the outlet of the reactor is located at the top of the vessel, the total volume of the reaction ( $V_R$ ) can be considered constant. In addition, the mixer allows assuming that the system behaves as an ideal

CSTR. It is also assumed that the measurements of the molar concentrations of C and B in the effluent are unavailable, while the rest of the variables are measured.

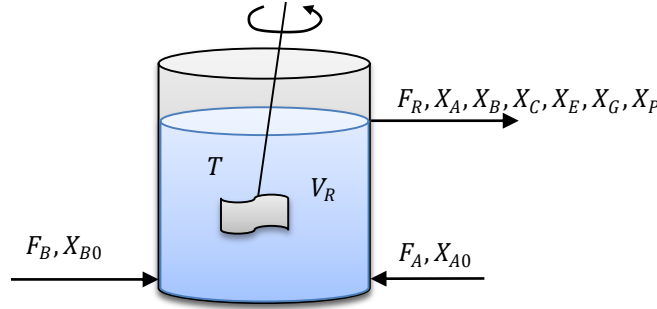


Figure 4: Schematic of the Williams-Otto reactor.

The process-modeling mismatch has been emulated using a simplified dynamic model to solve the model-based optimization in the controller. The simplified model considers only two reactions and neglects the existence of component C, which implies a structural mismatch. In addition, parametric uncertainty has been incorporated using different values for the model parameters and real process model parameters. The model that emulates the real process and the simplified one are based on mass balances for each component and the elemental kinetics with Arrhenius temperature dependence. More details of both models can be found in Appendix B.

To evaluate the performance of the proposed algorithm, two problems have been considered. The first problem is presented in equation (13), and it consists of optimizing an economic objective function with no inequality constraints in the dependent variables, which implies that only the modifiers of the objective function are used. The second problem, defined as equation (14), includes constraints on the molar concentration of A and G in the effluent. The upper bound for  $X_A$  and  $X_G$  have been chosen such that these constraints are active in the optimum of the process. This implies that the modifiers of these constraints are needed for the convergence of the algorithm.

$$\max_{\substack{\Delta F_{B,k+i}, \Delta T_{k+i} \\ i=0 \dots n_u-1}} \left\{ \bar{F}_R (\bar{X}_P p_P + \bar{X}_E p_E) - F_A X_{A0} p_A - \bar{F}_B X_{B0} p_B + \lambda_{1,k} (\bar{F}_B - F_{B,k-1}) \right. \\ \left. + \lambda_{2,k} (\bar{T} - T_{k-1}) + \beta_1 \sum_{i=0}^{n_u-1} (\Delta F_{B,k+i})^2 + \beta_2 \sum_{i=0}^{n_u-1} (\Delta T_{k+i})^2 \right\}$$

s.t. nonlinear simplified dynamic model plus disturbances from MHE,  $\forall t \in [t_k, t_{pred}]$

$$\begin{aligned} F_B(t) &= F_{B,k+i}, & t \in [t_{k+i}, t_{k+i+1}], & i = 0, \dots, n_{pred} - 1 \\ T(t) &= T_{k+i}, & t \in [t_{k+i}, t_{k+i+1}], & i = 0, \dots, n_{pred} - 1 \\ F_{B,k+i} &= F_{B,k+i-1} + \Delta F_{B,k+i}, & i = 0, \dots, n_{pred} - 1 \\ T_{k+i} &= T_{k+i-1} + \Delta T_{k+i}, & i = 0, \dots, n_{pred} - 1 \\ \Delta F_{B,k+i} &= 0, & i = n_u, \dots, n_{pred} - 1 \\ \Delta T_{k+i} &= 0, & i = n_u, \dots, n_{pred} - 1 \\ 180 \frac{L}{min} &\leq F_{B,k+i} \leq 360 \frac{L}{min}, & i = 0 \dots n_{pred} - 1 \\ 75^\circ\text{C} &\leq T_{k+i} \leq 100^\circ\text{C}, & i = 0 \dots n_{pred} - 1 \end{aligned} \quad (13)$$

$$\max_{\substack{\Delta F_{B,k+i}, \Delta T_{k+i} \\ i=0 \dots n_u-1}} \left\{ \bar{F}_R (\bar{X}_P p_P + \bar{X}_E p_E) - F_A X_{A0} p_A - \bar{F}_B X_{B0} p_B + \lambda_{1,k} (\bar{F}_B - F_{B,k-1}) \right. \\ \left. + \lambda_{2,k} (\bar{T} - T_{k-1}) + \beta_1 \sum_{i=0}^{n_u-1} (\Delta F_{B,k+i})^2 + \beta_2 \sum_{i=0}^{n_u-1} (\Delta T_{k+i})^2 \right\} \quad (14)$$

s.t. nonlinear simplified dynamic model and same constraints from (13)

$$\begin{aligned} X_A(t_{pred}) + \gamma_{1,k} (\bar{F}_B - F_{B,k-1}) + \gamma_{2,k} (\bar{T} - T_{k-1}) + \varepsilon_{1,k} &\leq 0.8 \\ X_G(t_{pred}) + \gamma_{3,k} (\bar{F}_B - F_{B,k-1}) + \gamma_{4,k} (\bar{T} - T_{k-1}) + \varepsilon_{2,k} &\leq 0.35 \end{aligned}$$

#### 4.1 Results

In both problems, the initial states and the disturbances in each iteration  $k$  of the controller were provided by the MHE module. To evaluate the effect of the proposed eMPC+TMA algorithm, each problem was solved with three strategies: (a) estimating the process gradients using the TMA, (b) estimating the process gradients neglecting the time derivative in equation (8) that it is equivalent to the gradient estimation strategy proposed by Rodríguez-Blanco et al., (2017), and (c) solving the eMPC without modifiers

(equivalent to setting the value of the modifiers to zero). In addition, another experiment for problem (13) was conducted; including an additive Gaussian noise in the measurements of the molar concentrations of the effluent. This experiment was only solved with strategy (a), as the objective was to evaluate the effect of measurement noise in the behavior of the TMA.

The simulation of the reactor and the dynamic optimization problem in the MHE and the economic controller have been formulated in a continuous-time domain in MATLAB. The controller was implemented using the *nlmpc* object. Both the MHE and the controller were solved using the sequential quadratic algorithm, available in the *fmincon* NLP solver. On average, the whole problem (MHE + controller) took 6 seconds to be solved in each sampling time of the supervisory layer, in a PC under Windows10, Intel Core i5 processor running at 3.2GHz and 16GB of memory.

For problems (13) and (14), the supervisory layer was executed every 2 minutes. The control and prediction horizons were chosen as  $n_u = 3$  and  $n_{pred} = 30$ , respectively. The MA filter was set as  $K = 0.7$  and the move suppression parameter for both decision variables was  $\beta_1 = \beta_2 = 0.5$ . The NLMS parameters were fixed to  $\mu = 0.1$  and  $\sigma = 1e - 4$ . The first 8 minutes were used to collect the data needed for the identification algorithm. The controller started at minute 10.

#### 4.1.1 Unconstrained Problem

Figure 5 shows the trajectories of the objective function and the decision variables using the three solution strategies tested in problem (13). It can be seen that each algorithm reaches a different steady state, depending on the method to estimate the value of the modifiers.



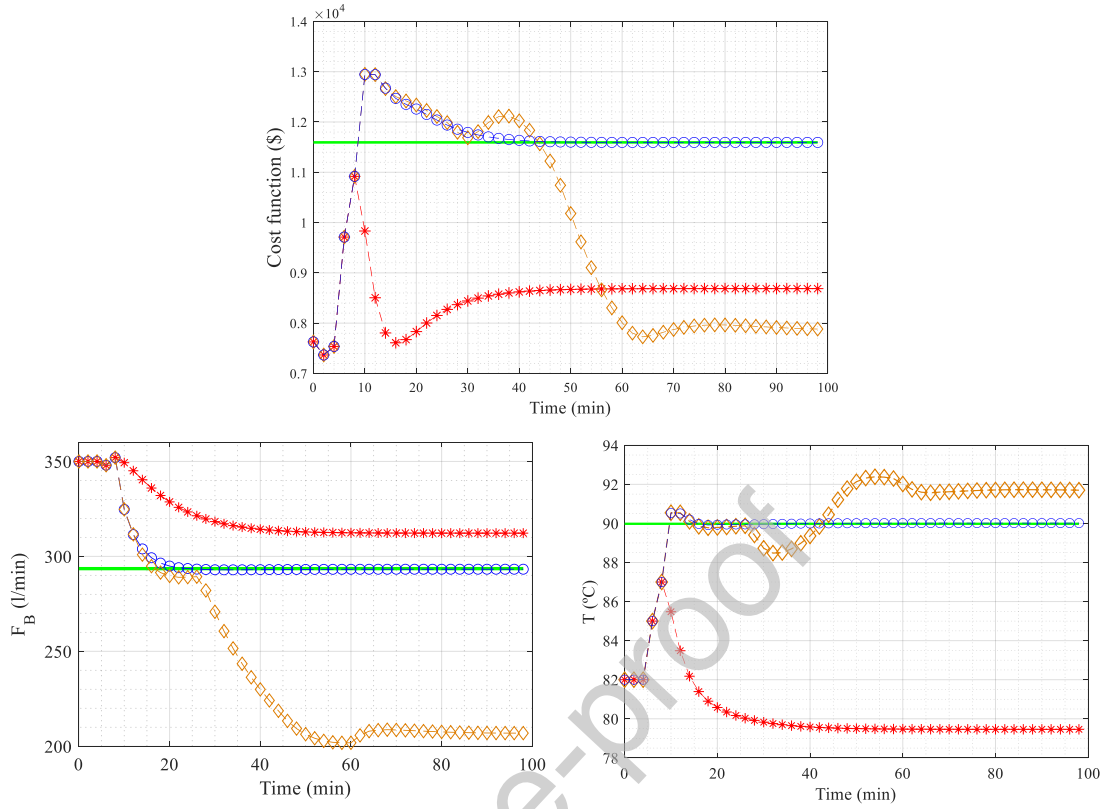


Figure 5: Evolution of cost function and manipulated variables for problem (13). (—) Plant optimum, (---o---) eMPC+TMA, (---□---) eMPC+MA without time derivatives, (---□---) eMPC.

For the case of the eMPC, as no correction is implemented in the objective function, the controller drives the process into the optimum of the model ( $F_B = 307.2$  L/min and  $T = 79.8^\circ\text{C}$ ), which is identical to the expected results of RTO without the modifiers to correct the structural uncertainty. These decision variables applied to the simplified stationary model without MHE corrections (the same dynamic model where the derivatives are set to zero) would result in a profit of  $\sim 11259$  €. However, as can be seen in Figure 5, these values applied to the real process achieve a much lower profit ( $\sim 8686$ €). Meanwhile, for the case of the eMPC+TMA algorithm, the controller detects the real optimum of the process and drives the system to a profit of  $\sim 11593$  €, a value far greater than the profit obtained without MA. Also, notice that in the eMPC+TMA case the closed-loop system stabilizes around 50 min, which is, approximately, twice the time of the reactor open-loop response time (30 min), and similar to the stabilization time of the eMPC. As the system converges on the expected value, it can be assumed that the TMA allows a correct calculation of the process gradients. This implies that the first-order correction of the objective function agrees in finding the

stationary condition of the process. To evaluate if this is a consequence of the inclusion of the time derivative in the identification step, this result can be compared with the trajectory given by the eMPC+MA where the supervisory layer fails to detect the optimum of the process despite the inclusion of  $\lambda_k$  in the objective function, since the gradient estimation step does not consider the effect of time. It can be noticed that before  $t \approx 25$  min the decision variables proposed by the controller are closer to the optimum of the process, but after this time the supervisory layer suggests a different steady state condition that is kept until the end of the experiment. The failure in detecting the optimum for the eMPC+MA can be explained considering that in the interval from 20 to 25 min the changes in the decision variables become smaller than the previous ones; however, the cost function is affected by the process dynamics. This implies that the effect of time in the objective function emerges as relatively more important than the effect of the decision variables, a condition that produces a miscalculation in the gradient of the process due to not including the time derivative in this case.

Figure 6 presents the evolution of the eMPC+TMA with measurement noise in the molar concentrations. Note that the algorithm is capable of finding the optimum of the process, stabilizing the system in about 60 minutes, which is 20% larger than the noise-free example. This difference can be attributed to the oscillations that the temperature presents before minute 25, and because of the additional time that the controller needs to bring the value of  $F_B$  closer to the optimum. This is a consequence of the effect that noise has in the MHE and in the gradient estimation step, probably related to the linear model assumed in time dependence. The loss of optimality associated with the extra time needed to converge, due to the additive noise of a standard deviation of 5% with respect to the expected span of the measurement, seems reasonable, especially considering that the process optimum was detected despite the measurement noise.

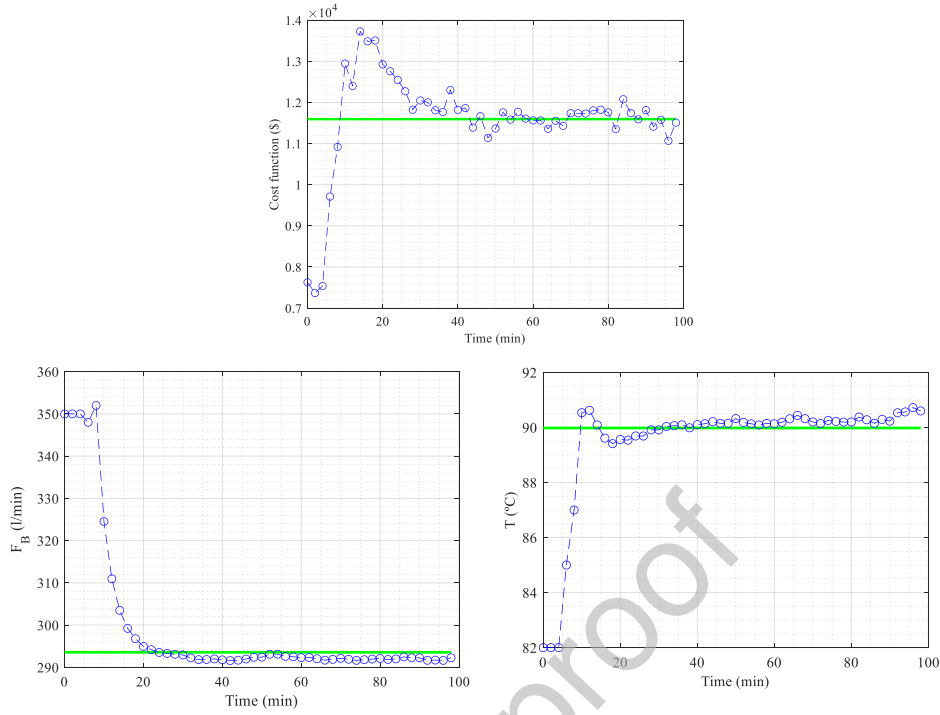


Figure 6: Evolution of cost function and manipulated variables for problem (13) with measurement noise. (—) Plant optimum, (---o---) eMPC+TMA.

#### 4.1.2 Constrained Problem

The evolution of the cost function, molar fractions of A and G, and decision variables for problem (14) are presented in Figure 7 for the three strategies tested. Similar behavior to the one observed for the unconstrained case can be seen, i.e., the value of the modifiers determines the steady state reached. As in this case the NCO of the process implies that the inequality constraints on  $X_A$  and  $X_G$  are active, the accuracy of  $\lambda$ ,  $\gamma$ , and  $\varepsilon$  is relevant to detect the optimum of the process. The eMPC converges to a point that is the model optimum at steady state and is not the process optimum, as the eMPC problem does not consider first and zeroth-order corrections in the model. The effect of using these corrections can be seen in the trajectory of the eMPC+TMA. Notice that this controller is capable of detecting the optimum of the process and converges to this value, which implies an increase in the profit of the process of almost 20% as a result of the correct estimation of the modifiers using the TMA strategy. Also, notice that both constraints are satisfied after a while, even though  $X_A$  starts at an unfeasible value. The evolution of the process using eMPC+MA shows that the algorithm cannot converge to the process optimum, although the modifiers have been included. As

discussed for the unconstrained case, this difference can be attributed to the effect of the miscalculation of the modifiers because the value of the time derivatives is neglected in this process gradient estimation procedure. This miscalculation becomes more critical when the system tends to stabilize, as can be observed in the interval from  $t = 20$  to  $t = 40$  min. In this interval, the decision variables remain practically constant while the cost function and the constrained variables are still changing in time because of the inertia of the process. These findings support the need to include the dynamics in the process gradient estimation step to calculate the modifiers used in the eMPC+MA formulation. An additional consequence of not including the dynamics in the calculation of the modifiers is the increase in the settling time of the controller because of the changes in the value of the proposed steady state.

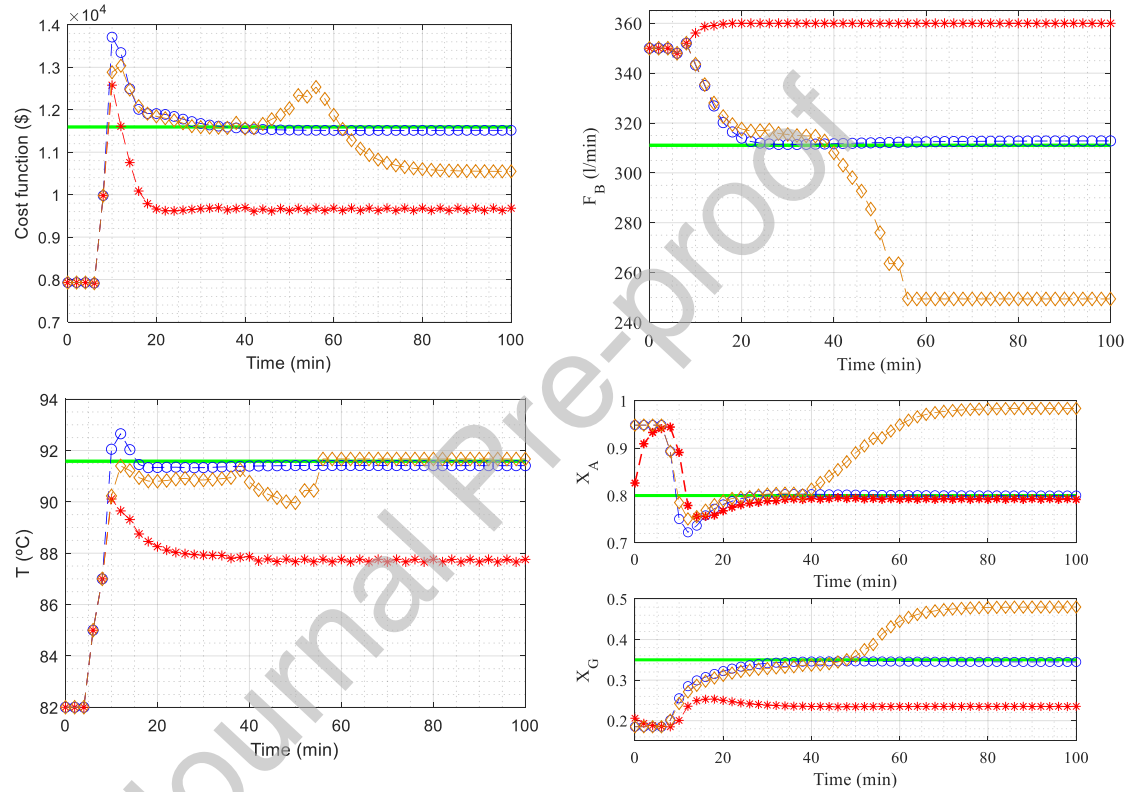


Figure 7: Evolution of cost function and manipulated variables for problem (14). (—) Plant optimum, (---o---) eMPC+TMA, (---□---) eMPC+MA without time derivatives, (---◇---) eMPC.

## 5 Hybrid laboratory plant case study

The second case study for evaluating the performance of the TMA methodology is a system implemented in a hybrid laboratory plant (Figure 8) that emulates the behavior of a CSTR with the Van de Vusse reactions (Van de Vusse, 1964). Hybrid plants are real processes with some simulated components. They have been used in metallurgical processes as an alternative to validate the performance of supervisory algorithms in more realistic situations than the ones given by computational experiments (Bergh and Yianatos, 2014; Navia et al., 2019, 2016). The concept of hybrid plants is based on the fact that the essential phenomena occurring in a real process can be divided into two aspects: fluid dynamics and physicochemical mechanisms. Fluid dynamics can be emulated using an experimental set-up and a fluid with similar characteristics to the original one, while physicochemical mechanisms can be emulated using a first-principles model that uses the experimental measurements to close the degrees of freedom. This configuration allows defining two kinds of decision variables: experimental and virtual. Experimental decision variables are associated with the variables that can be manipulated in the actual experimental set-up, such as flows, pressure, or liquid level. Virtual decision variables can be any variable that affects the output of the process, such as the simulated properties of the raw materials. The use of hybrid plants permits studying the effect of experimental errors in the behavior of supervisory algorithms, knowing part of the modeling mismatch and also facilitates the maintenance of pilot plants in a lab.

The schematic of the experimental set-up is presented in Figure 8 (Montes et al., 2021). It consists of a CSTR exothermic reactor with four temperature sensors (TT-01 to 04) and two flowmeters (FT-01 and 02). The final control elements (experimental decision variables) are the peristaltic reagent pump (P-101) and the coolant valve (V-101). The pump P-102 allows emulating the reactants leaving the reactor by overflow. The power of two heating coils located at the bottom of the vessel can be manipulated by a power amplifier in J-101. The system uses water as the working fluid and the chemical reactions are simulated using a chemical model executed in real time and with process data.

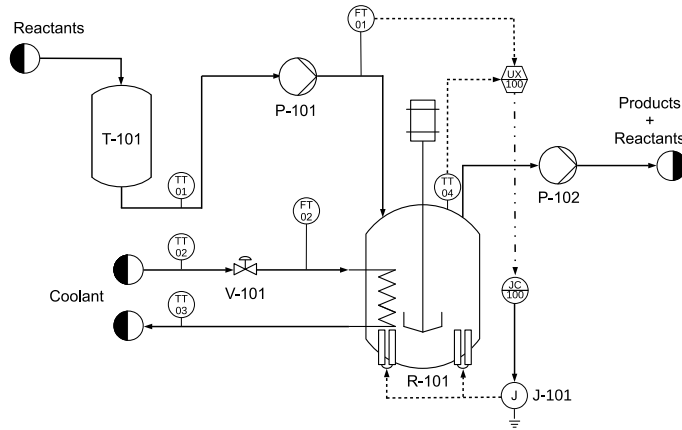
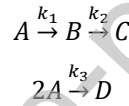


Figure 8: Laboratory plant P&amp;ID.

The chemical model, integrated in the process as component UX-100, has been obtained from the description of the system. The process consists of the three Van de Vusse reactions, where component B is the desired product:



Assuming that the physical properties of water are constant within the range of the operating conditions and the vessel is perfectly agitated, a first-principles model of the reactions based on mass balances for each component can be derived. Equations (15) to (18) represent the dynamic mass balances for each compound, whereas equations (19) to (21) describe the respective reaction rates assuming elemental kinetics and Arrhenius dependence with temperature. Equation (22) computes the heat that is applied to the reactor by means of the amplifier J-101 in real-time.

$$V \frac{dC_A}{dt} = q(C_{A0} - C_A) + V(-r_1 - 2r_3) \quad (15)$$

$$V \frac{dC_B}{dt} = -qC_B + V(r_1 - r_2) \quad (16)$$

$$V \frac{dC_C}{dt} = -qC_C + Vr_2 \quad (17)$$

$$V \frac{dC_D}{dt} = -qC_D + Vr_3 \quad (18)$$

$$r_1 = k_{10} e^{-\frac{E_1}{RT}} C_A \quad (19)$$

$$r_2 = k_{20} e^{-\frac{E_2}{RT}} C_B \quad (20)$$

$$r_3 = k_{30} e^{-\frac{E_3}{RT}} C_A^2 \quad (21)$$

$$P = \frac{1000}{60} V \sum_{j=1}^3 -r_j \Delta H_j \quad (22)$$

where  $C_i$  represents the molar concentration of component  $i$  in the effluent,  $q$  is the total volumetric flow,  $C_{A0}$  is the molar concentration of component A in the influent and  $V$  is the reaction volume. For the kinetic expressions  $r_j$ ,  $k_{j0}$  and  $E_j$  are the reaction rate, exponential constant, and activation energy of reaction  $j$  respectively, while  $T$  is the temperature in the reactor. Equation (22) defines the conversion between the heat generated by reactions ( $\Delta H_j$  is the heat of the reaction  $j$ ) to the heat power applied by the electrical resistances denoted as  $P$ .

To clarify the relationship between the experimental and the simulated variables, Table 1 summarizes how the value of each variable is obtained. Note that as  $E_j$ ,  $k_{j0}$  and  $\Delta H_j$  are parameters assumed to be known so that the model of the process can be simulated.

Table 1: Description of the source of the variables for the hybrid process.

Variable	Source
$C_i, i \in \{A, B, C, D\}$	Calculated from the chemical model
$r_j, j = \{1, 2, 3\}$	Calculated from the chemical model
$P$	Calculated from the chemical model
$q$	Measured from FT-01
$T$	Measured from TT-04
$V$	Defined as the volume of the vessel
$C_{A0}$	Defined by the user (fixed value)

The value of the heat of reaction calculated from the chemical model is used as a set-point of J-101, which allows emulating the effect of the exothermic reactions in the experimental set-up.

As internal model of the eMPC+TMA and MHE, a simplified nonlinear dynamic model that neglects the existence of reaction 3 and component D has been implemented in the supervisory layer (eMPC+TMA and MHE), which defines a structural uncertainty in the available model. The experimental system has two decision variables: the flowrate of the reactants (manipulated with P-101), and the flowrate of

the coolant (manipulated with V-101). The model is based on mass and energy balances and a description of how the heat is transferred from the vessel to the coolant. Equations (23) to (30) summarize the model.

$$V \frac{dC_A}{dt} = q(C_{A0} - C_A) + V(-r_1) \quad (23)$$

$$V \frac{dC_B}{dt} = -qC_B + V(r_1 - r_2) \quad (24)$$

$$V \frac{dC_C}{dt} = -qC_C + Vr_2 \quad (25)$$

$$r_1 = k_{10} e^{-\frac{E_1}{RT}} C_A \quad (26)$$

$$r_2 = k_{20} e^{-\frac{E_2}{RT}} C_B \quad (27)$$

$$\rho C_p V \frac{dT_{out}}{dt} = q\rho C_p(T_{in} - T_{out}) - Q - V \sum_{j=1}^2 r_j \Delta H_j \quad (28)$$

$$\rho C_p V_c \frac{dT_{c,out}}{dt} = F_R \rho C_p (T_{c,in} - T_{c,out}) + Q \quad (29)$$

$$Q = \alpha_1 F_R^{\alpha_2} (T_{out} - T_{c,out}) \quad (30)$$

where  $T$  and  $T_c$  are the temperatures of the reactant and coolant streams respectively. The subindex *in* and *out* indicates if it is the temperature of the inlet or outlet stream.  $C_p$  and  $\rho$  are the heat capacity and density of the coolant.  $Q$  is the heat transferred,  $F_R$  is the coolant flowrate and  $V_c$  is the volume of the cooling coil. Note that equation (30) approximates Newton's law of cooling, where the overall heat transfer coefficient has been defined as a function of the coolant flowrate with parameters  $\alpha_1$  and  $\alpha_2$ .

The value of the parameters used in this case study is presented in Table 2. In Table 2 and hereafter, "Chemical model" refers to the model used to emulate the physicochemical mechanisms in the hybrid plant (equations (15) to (22)), while "eMPC model" is used for the model implemented in the supervisory layer (equations (23) to (30)).



Table 2: Value of the parameters used in the case study.

Parameter	Chemical Model	eMPC Model	Parameter	Chemical Model	eMPC Model
$k_{10}$ ( $\text{min}^{-1}$ )	$9.95 \times 10^{10}$	$6.40 \times 10^9$	$\Delta H_3$ ( $\text{kJ} \cdot \text{mol}^{-1}$ )	-66.54	-
$k_{20}$ ( $\text{min}^{-1}$ )	$9.99 \times 10^{11}$	$4.84 \times 10^{10}$	$V$ (L)	11.5	11.5
$k_{30}$ ( $\text{L} \cdot \text{mol}^{-1} \cdot \text{min}^{-1}$ )	$9.99 \times 10^{12}$	-	$V_c$ (L)	-	1.0
$E_1$ ( $\text{kJ} \cdot \text{mol}^{-1}$ )	59.0	52.1	$\rho$ ( $\text{kg} \cdot \text{L}^{-1}$ )	-	1.0
$E_2$ ( $\text{kJ} \cdot \text{mol}^{-1}$ )	77.62	70.0	$C_p$ ( $\text{kJ} \cdot \text{kg}^{-1} \cdot ^\circ\text{C}^{-1}$ )	-	4.18
$E_3$ ( $\text{kJ} \cdot \text{mol}^{-1}$ )	71.11	-	$C_{A0}$ ( $\text{mol} \cdot \text{L}^{-1}$ )	5.0	5.0
$\Delta H_1$ ( $\text{kJ} \cdot \text{mol}^{-1}$ )	-21.22	-48.75	$\alpha_1$	-	1.8
$\Delta H_2$ ( $\text{kJ} \cdot \text{mol}^{-1}$ )	-2.68	-34.50	$\alpha_2$	-	0.8

Note that the uncertainty of the model-based optimization can be considered structural because both reaction 3 and component D have been disregarded and the model of the heat transfer is a simplification. Additionally, there is also parametric uncertainty as shown in Table 2.

The objective of the supervisory layer is to maximize the economic benefit, which can be defined as the income related to selling the desired product B minus the costs of the reactants and coolant, manipulating  $q$  and  $F_R$ . Analogous to section 4, two dynamic optimization problems have been considered for this case study. The first problem consists in the optimization of the benefit with no inequality constraints in the dependent variables, implying that only the modifiers of the cost function are needed as presented in equation (31). The second problem includes an additional inequality constraint for the reactor temperature. For this constrained problem we have used the output MA (MAy) formulation to solve the optimization problem. This MAy formulation includes the modifiers in the output variables as presented in equation (32). According to Papasavvas et al. (2019), this formulation improves the convergence of the MA-type algorithms. In both problems, their formulations follow the structure presented in problem (4), while the value of the prices and costs are given in Table 3.

$$\max_{\substack{\Delta q_{k+i}, \Delta F_{R_{k+i}} \\ i=0 \dots n_u-1}} \left\{ \bar{q}(C_B p_B - C_{A0} p_A) - \bar{F}_R p_R + \lambda_{1,k}(\bar{q} - q_{k-1}) + \lambda_{2,k}(\bar{F}_R - F_{R_{k-1}}) \right. \\ \left. + \beta_1 \sum_{i=0}^{n_u-1} (\Delta q_{k+i})^2 + \beta_2 \sum_{i=0}^{n_u-1} (\Delta F_{R_{k+i}})^2 \right\}$$

s.t. Nonlinear model (23) to (30) plus disturbances from MHE,  $\forall t \in [t_k, t_{pred}]$

$$\begin{aligned} q(t) &= q_{k+i}, & t \in [t_{k+i}, t_{k+i+1}], & i = 0, \dots, n_{pred} - 1 \\ F_R(t) &= F_{R_{k+i}}, & t \in [t_{k+i}, t_{k+i+1}], & i = 0, \dots, n_{pred} - 1 \\ q_{k+i} &= q_{k+i-1} + \Delta q_{k+i}, & i = 0, \dots, n_{pred} - 1 \\ F_{R_{k+i}} &= F_{R_{k+i-1}} + \Delta F_{R_{k+i}}, & i = 0, \dots, n_{pred} - 1 \\ \Delta q_{k+i} &= 0, & i = n_u, \dots, n_{pred} - 1 \\ \Delta F_{R_{k+i}} &= 0, & i = n_u, \dots, n_{pred} - 1 \\ 0.3 \frac{L}{\min} &\leq q_{k+i} \leq 1.2 \frac{L}{\min}, & i = 0 \dots n_{pred} - 1 \\ 6 \frac{L}{\min} &\leq F_{R_{k+i}} \leq 15 \frac{L}{\min}, & i = 0 \dots n_{pred} - 1 \end{aligned} \quad (31)$$

$$\max_{\substack{\Delta q_{k+i}, \Delta F_{R_{k+i}} \\ i=0 \dots n_u-1}} \left\{ \bar{q}(C_{B,MA} p_B - C_{A0} p_A) - \bar{F}_R p_R \right. \\ \left. + \beta_1 \sum_{i=0}^{n_u-1} (\Delta q_{k+i})^2 + \beta_2 \sum_{i=0}^{n_u-1} (\Delta F_{R_{k+i}})^2 \right\} \quad (32)$$

s.t. Same constraints from problem (31)

$$\begin{aligned} C_{B,MA} &= \bar{C}_B + \lambda_{1,k}(\bar{q} - q_{k-1}) + \lambda_{2,k}(\bar{F}_R - F_{R_{k-1}}) + \varepsilon_{1,k} \\ T_{R,MA} &= \bar{T}_R + \gamma_{1,k}(\bar{q} - q_{k-1}) + \gamma_{2,k}(\bar{F}_R - F_{R_{k-1}}) + \varepsilon_{2,k} \\ T_{R,MA} &\leq 38^\circ\text{C} \end{aligned}$$

Table 3: Cost and prices for the case study.

Parameter	Description	Value
$p_B$	Price of component B	18.0 € · mol <sup>-1</sup>
$p_A$	Cost of component A	0.2 € · mol <sup>-1</sup>
$p_R$	Cost of coolant	3.0 € · L <sup>-1</sup>

## 5.1 Results

To evaluate the effect of the proposed algorithm, each problem was solved with two strategies: (a) estimating the process gradients using the TMA and (b) solving the eMPC without modifiers. For both problems, the simulation of the equations of the hybrid process (the chemical model) was developed with EcosimPro (EA Int., 2020) and encapsulated as an OPC-DA server generated in the same environment.

### 5.1.1 Unconstrained Problem

For problem (31), the control and the prediction horizons were defined as  $n_u = 3$  and  $n_{pred} = 60$  respectively, and the move suppression parameter was fixed to  $\beta_1 = \beta_2 = 0.5$ . The parameters for the NLMS algorithm were set to  $\mu = 1.6$  and  $\sigma = 1e - 4$ . The dynamic optimization problem was solved using the simultaneous approach, discretizing the dynamic model using the orthogonal collocation method. The resulting algebraic problem was implemented in Pyomo (Hart et al., 2017, 2011), using IPOPT as an NLP solver. The supervisory layer started at time  $t = 3$  min and was executed every 30s, with an average computation time of 0.5s in a PC under Windows10, Intel Core i5-4460 processor at 3.2GHz and 8GB of RAM.

Figure 9 shows the evolution of the objective function and the decision variables of two experiments performed using the eMPC+TMA (left), and the eMPC with no correction (right) under similar conditions. The two lines that appear in each figure of  $q$  and  $F_R$  correspond to: the dashed line is the set point (decision variable) of a PID governing that variable (i.e., is the decision variable generated by the controller) and the continuous line is the current value of the variable. It can be noticed that both algorithms converge on different steady states. Table 4 summarizes the 99% confidence intervals for the mean of the profit and the decision variables at the end of the experiments for both algorithms.

Table 4: 99% confidence intervals of the mean of the convergence point for both algorithms in problem (31).

Variable	eMPC+TMA	eMPC
<i>Profit</i> (€/min)	[32.1, 33.5]	[26.7, 28.4]
$q$ (L/min)	[1.18, 1.20]	[1.19, 1.20]
$F_R$ (L/min)	[5.8, 6.2]	[10.8, 11.3]

From the point of view of the profit, the inclusion of the modifiers represents a significant improvement in the economic results of the process of 19% on average. Table 4 shows that the confidence interval of the decision variables obtained with the eMPC is very close to the optimum of the model, which is consistent with the results obtained for the case study of section 4. The convergence point of the eMPC+TMA supports the hypothesis that the inclusion of the modifiers allows improving the performance of the process, correcting the optimum computed with a model with structural modeling mismatch. However, it is not possible to evaluate if the steady state reached by the eMPC+TMA is the process optimum because a “perfect model” is never available when dealing with real processes. Nevertheless, analyzing the results, the prices, and the projected consequence of the modeling mismatch some comments can be made. From the economic point of view, as the price of B is three orders of magnitude larger than the price of A, and two orders of magnitude larger than the price of the coolant, it is expected that the process optimum result increases the production of component B as much as possible. So the flow of  $q$  is set to its upper bound, condition reached for both algorithms (no statistical differences are observed for  $q$  in Table 4). As  $q$  is fixed in the optimum, the remaining degree of freedom ( $F_R$ ) must be selected to maximise the concentration of B. Assuming unlimited cooling capacity, this is equivalent to finding the value of  $T_R$  that maximizes  $C_B$  using equations (15) to (21) for the process, and equations (23) to (27) for the model, both in steady state. Unlike the original problem, the cooling assumption removes the experimental uncertainty and a comparison can be made by solving this surrogate steady-state optimization problem with fixed  $q = 1.2 \text{ L min}^{-1}$ , and using the parameters given in Table 2. The solution shows that the process optimal temperature is  $8^\circ\text{C}$  larger than the model optimum. This difference in temperature between process and model implies that the correct process optimum should reduce the use of coolant, as is observed in Figure 9 where  $F_R$  is close to its lower bound at the end of the experiment.

From the point of view of the process dynamic behavior, as in both cases the system reaches a steady state, the inclusion of the modifiers does not affect the stability of the controller for this case study. However, from the evolution of the objective function, it can be noticed that the closed-loop system stabilizes

in around 60 minutes for the eMPC+TMA, which is twice the stabilization time of the eMPC. This could be explained by comparing the transient behavior of  $F_R$  for both algorithms. Notice that the eMPC detects a target value that remains constant during the whole experiment, unlike the eMPC+TMA that changes three times the target value of  $F_R$ : (1) for  $t < 8$  min the system behaves in a similar way than the eMPC, looking for the optimum of the model; (2) from  $t = 10$  to  $t = 32$  min the controller proposes a decrease in  $F_R$  which improves the objective function; and (3) from  $t = 35$  min ahead, where the expected optimum of the process is identified. The observed changes in the target value of  $F_R$  are similar to the behavior of the decision variables from section 4 when the estimation of the process gradients neglects the effect of time (denoted as eMPC+MA in Figure 5 and Figure 7). Probably due to the presence of measurement noise in the experimental variables, the assumption of linear dependence on time in the TMA is not enough to detect the correct value during the transient. However, as the identification algorithm implemented in the TMA (NLMS) is recursive, the increase of its accuracy as more data is available produces successive changes in the decision variable, improving the objective function as a consequence of including the time dependency, which makes the algorithm converge to the expected value. Regarding the value of  $q$ , model and process coincide, and the eMPC+TMA is continuously driving the process to the right value, independently of the supposed estimation error of the experimental gradient.

An alternative to avoid the undesired behavior observed in  $F_R$  and reduce the convergence time of the algorithm can be adding additional dependencies on time in the TMA, using higher-order Taylor expansions. This can be an interesting future research direction.

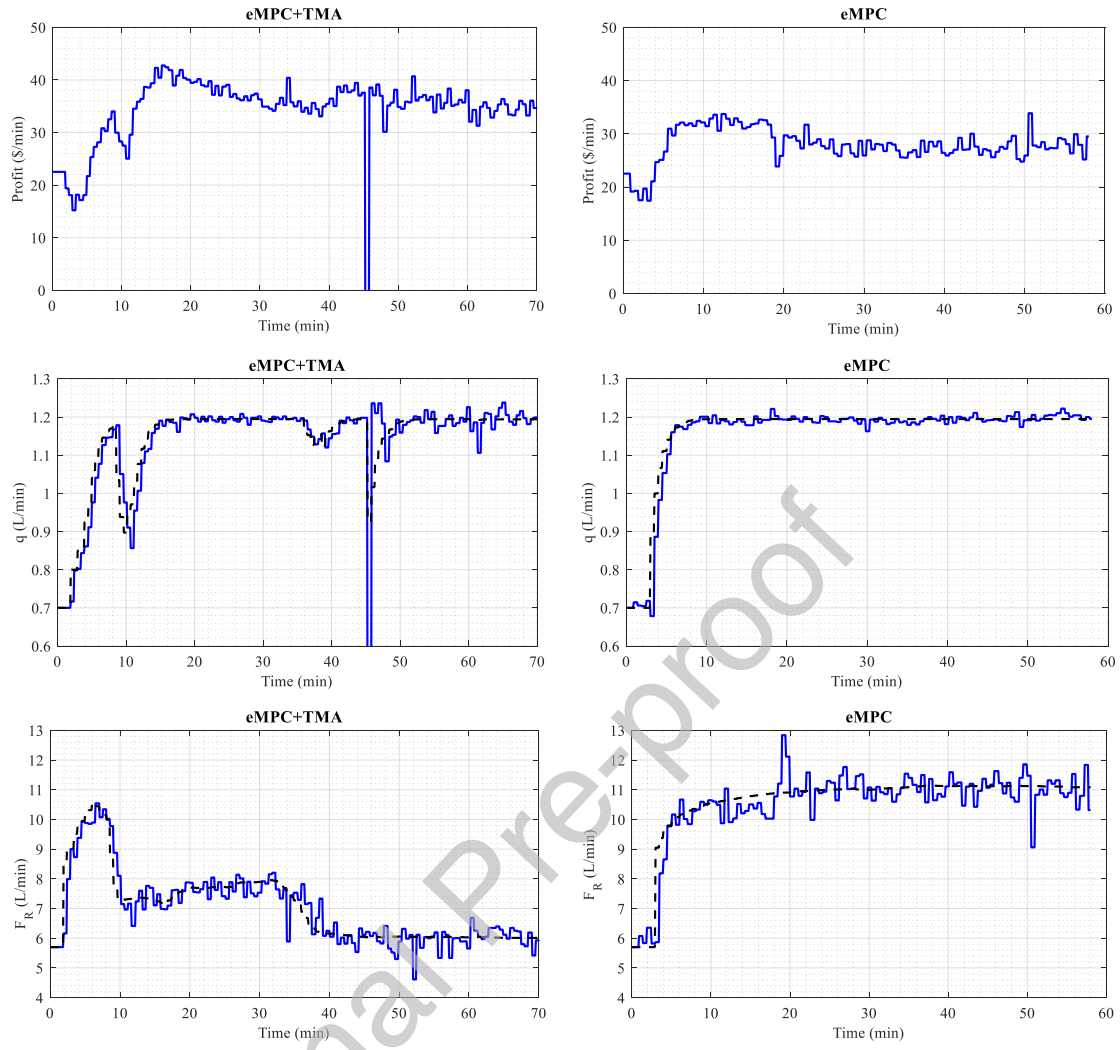


Figure 9: Cost function and manipulated variables over time for problem (31). (--) Proposed decision variables, (—) Measured decision variables.

### 5.1.2 Constrained Problem

For this example, an additional constraint in the upper bound of the temperature of the reactor  $T_R \leq 38^\circ\text{C}$  has been introduced to modify the expected optima of the unconstrained case, where its final value measured was  $T_R = 39.5^\circ\text{C}$ . The MAy (output MA) approach has been implemented in the modified problem (32), so the first and zeroth-order modifiers have been calculated for  $\mathbf{C}_B$  and  $T_R$ , which are the dependent variables that explicitly appear in the objective function and the inequality constraint. The first-order modifiers for  $\mathbf{C}_B$  ( $\lambda_1$  and  $\lambda_2$ ) and for  $T_R$  ( $\gamma_1$  and  $\gamma_2$ ) were calculated using their respective process gradients estimated by the TMA approach. This implies that in this example  $\mathbf{z}_p := [\mathbf{C}_B, T_R]$  for the TMA.

The control and the prediction horizons were defined as  $n_u = 3$  and  $n_{pred} = 60$  respectively. The suppression parameters were fixed to  $\beta_1 = 0.004$  and  $\beta_2 = 0.002$ . The parameters for the NLMS algorithm were set to  $\mu = 0.05$  and  $\sigma = 1e - 4$ . The rest of the specifications are the same as for the unconstrained case.

The evolution of the objective function, reactor temperature (constrained variable), and decision variables are presented in Figure 10 for the eMPC+TMA (left), and the eMPC (right).

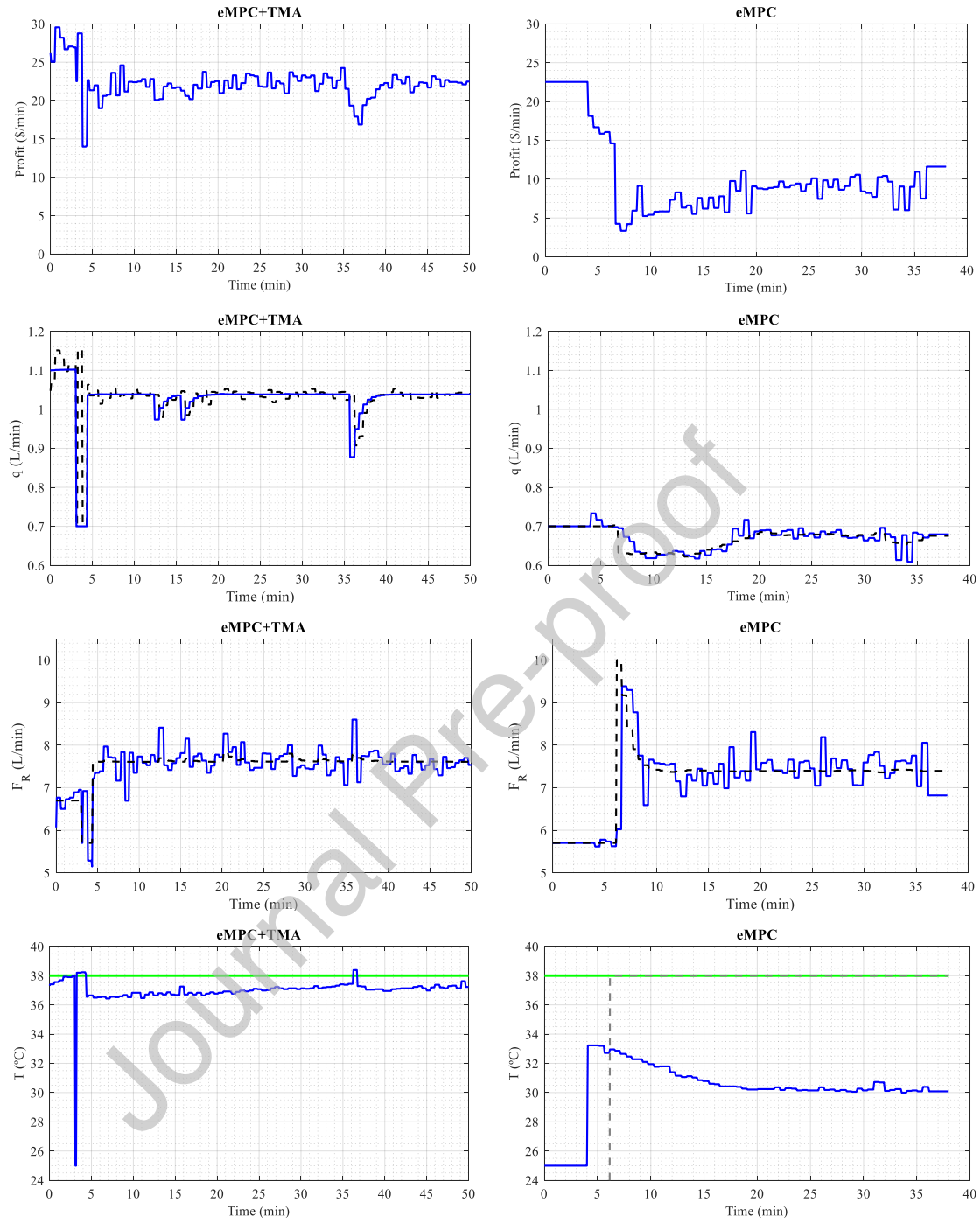


Figure 10: Cost function, constrained variable, and manipulated variables over time for problem (32). (--) Proposed decision variables, (—) Measured decision variables, (—) Upper bound for  $T_R$ .



Figure 10 shows a similar behavior as the previous examples, where both strategies converge to different steady states, implying that the value of the modifiers affects the economic target of the controller. The 99% confidence intervals for the objective function, reactor temperature, and decision variables obtained with both strategies are summarized in Table 5.

Table 5: 99% confidence intervals of the mean of the convergence point for both algorithms for problem (32).

Variable	eMPC+TMA	eMPC
<i>Profit</i> (€/min)	[21.0, 22.0]	[8.6 , 9.9]
$T$ (°C)	[37.2, 37.4]	[30.1, 30.2]
$q$ (L/min)	[1.02, 1.04]	[0.67 , 0.68]
$F_R$ (L/min)	[7.5, 7.8]	[7.3 , 7.6]

Table 5 shows that the profits obtained with both methods are significantly different with a 99% of confidence. This difference implies that the use of the eMPC+TMA in this process reports an increase of 130% on average in the economic benefit of the process compared to the eMPC results. As the upper bound of the temperature is smaller than its final value for the unconstrained case, it is expected that the algorithm proposes a final condition such that this constraint is active, increasing the flowrate of coolant with respect to previous results, keeping the value of  $q$  in its upper bound. The confidence intervals of  $F_R$  from Table 4 and Table 5 shows that the eMPC+TMA proposes a significant increase of 27% with respect to the unconstrained example. However, Table 5 and Figure 10 shows that the algorithm is not capable of detecting the active constraint of  $T_R$ , even though its value is close to the upper bound. The “experimental optimality gap” observed could be produced by a constant error in the estimation of the first-order modifiers for  $T_R$  as a consequence of the experimental noise, which can also explain the decrease in the final value of 13% in  $q$  with respect to its upper bound.

Concerning the eMPC results, notice that the evolution of  $T_R$ , and the confidence interval at steady state show that the modeling mismatch causes an overestimation of  $T_R$  (this also can be noted from the results of the eMPC for the unconstrained case). This overestimation is translated into the decrease in 50% in the proposed value of  $q$  with respect to the unconstrained solution of the eMPC. Finally, as commented before, the estimated value of the modifiers  $\gamma_1$ ,  $\gamma_2$  in the eMPC+TMA case might have persistent errors, causing the constant gap in the NCO of the modified problem. Because of that, one can conclude that the overestimation of  $T_R$  can be reduced by TMA, but it is still preserved.

## 6 Conclusions

The main objective of this work was to present and test a practical methodology that applies the concepts of modifier adaptation approach to improve the long-term economic performance of a process. The methodology includes changes in the value of the decision variables during the dynamic transitions, updating the expected value of the steady-state process optimum, using the available measurements to estimate the process derivatives. To do this, we have proposed the use of a single layer economic nonlinear predictive controller, modified with first and zeroth-order correctors. The states and disturbances were estimated with a moving horizon estimator. The first-order modifiers were calculated using an estimation of the process gradient obtained from an approximated model of the dependent variables that explicitly considers the dependence on time and the decision variables. To calculate the process derivatives, we have proposed a two-step identification algorithm named TMA. The TMA uses the available measurements from the process to estimate the time derivative using Nordsieck's vector, followed by the estimation of the process dynamic derivatives with respect to the decision variables using a recursive normalized least square algorithm. Even though the algorithm does not estimate the gradients in steady state but at intermediate point, the results obtained were better than the standard eMPC. Actually considering good initial values for the estimation, the result could also achieve the true optimum of the plant. The modified dynamic real-time optimization with the identification algorithm proposed is called eMPC+TMA.

The eMPC+TMA has been tested in two case studies with simulation and experimental components, solving in both cases optimization problems with and without uncertain inequality constraints. The simulation case study has been implemented in the Williams-Otto reactor evaluating the effect of including the modifiers, and the effect of including the time dependence in the estimation of the process gradients. In addition, the effect of measurement noise in the TMA was evaluated including an additive white noise. The experimental case study consists of a hybrid plant that emulates the Van der Vusse reactor, where we have evaluated the effect of including the modifiers.

The results allow us to conclude that the use of modifiers in a dynamic economic optimization problem permits improving the economic performance of an uncertain system, detecting the steady-state optimum of the process during the dynamic transition provided the process gradients are correctly calculated. It can also be concluded that, to have a better estimation of the dynamic process gradients from the transient measurements using the proposed approximated model of the dependent variables, it is necessary to include not only the decision variables but also the dependence on time. Besides, it is important to guarantee that the process has the necessary excitation so the identification method can estimate properly the gradients.

Finally, it is important to remark that the inclusion of the modifiers calculated with the TMA has increased the profit of the process significantly for both experimental examples, reducing the optimality gap related to the process-modeling mismatch.

## Acknowledgments

This work was supported by JCyL under programs CLU 2017-09 and UIC 233, as well as by project InCO4In, from Spanish AEI under project PGC2018-099312-B-C31, both with FEDER funds. The authors thank the European Social Fund and the “Consejería de Educación de la Junta de Castilla y León”. The third author thanks Banco Santander and Universidad de Valladolid for financing his doctoral studies through the 2020 call for pre-doctoral contracts.

## Appendix A. MHE

MHE uses the dynamic optimization problem (A.1) to estimate values for the states  $\mathbf{x}$  and disturbances  $\mathbf{v}$ . The problem is executed at every sample time  $k$ , considering a past horizon  $t \in [t_{k-n_e}, t_k]$ , where  $n_e$  is the MHE past horizon. The MHE problem objective function (A.1a) contains three terms. The first one aims to minimize the difference between the output computed by the model  $\mathbf{y}$  using (A.1b-c) and its measurement  $\mathbf{y}_p$ . The second one penalizes the distance of the current decision variable  $\mathbf{x}_{k-n_e}$  and the value estimated at the previous sample time  $k - n_e$  ( $\hat{\mathbf{x}}_{k-n_e}$ ). Finally, the last one, minimizes the magnitude of the estimated disturbances  $\mathbf{v}_{k-i}$ . The parameters  $\mathbf{Q}_x$ ,  $\mathbf{Q}_v$  and  $\mathbf{Q}_y$  are positive definite matrices, with weighting and normalization factors. The MHE problem also includes the inequality constraints (A.1d) to avoid non-desirable values of the process variables and it bounds the disturbances in the allowed range (A.1f).

$$\min_{\substack{\mathbf{x}_{k-n_e}, \mathbf{v}_{k-i} \\ i=1, \dots, n_e}} \sum_{i=0}^{n_e-1} \Delta \mathbf{y}_{k-i}^T \mathbf{Q}_y \Delta \mathbf{y}_{k-i} + \Delta \mathbf{x}_{k-n_e}^T \mathbf{Q}_x \Delta \mathbf{x}_{k-n_e} + \sum_{i=1}^{n_e} \mathbf{v}_{k-i}^T \mathbf{Q}_v \mathbf{v}_{k-i} \quad (\text{A.1a})$$

$$\text{s.t.} \quad \mathbf{f}(\dot{\mathbf{x}}, \mathbf{x}, \mathbf{u}, \mathbf{v}) = \mathbf{0}, \forall t \in [t_{k-n_e}, t_k], \quad \mathbf{x}(t_{k-n_e}) = \mathbf{x}_{k-n_e} \quad (\text{A.1b})$$

$$\mathbf{h}(\mathbf{x}, \mathbf{u}, \mathbf{y}, \mathbf{v}) = \mathbf{0}, \forall t \in [t_{k-n_e}, t_k] \quad (\text{A.1c})$$

$$\mathbf{g}(\mathbf{u}, \mathbf{y}) \leq \mathbf{0}, \forall t \in [t_{k-n_e}, t_k] \quad (\text{A.1d})$$

$$\mathbf{u}(t) = \mathbf{u}_{k-i}, \quad \mathbf{v}(t) = \mathbf{v}_{k-i}, \quad t \in [t_{k-i}, t_{k-i+1}], \quad i = 1, \dots, n_e \quad (\text{A.1e})$$

$$\mathbf{v}^L \leq \mathbf{v}_{k-i} \leq \mathbf{v}^U, \quad i = 1 \dots n_e \quad (\text{A.1f})$$

$$\Delta \mathbf{y}_{k-i} = \mathbf{y}_{k-i} - \mathbf{y}_{P,k-i}, \quad i = 0, \dots, n_e - 1 \quad (\text{A.1g})$$

$$\Delta \mathbf{x}_{k-n_e} = \mathbf{x}_{k-n_e} - \hat{\mathbf{x}}_{k-n_e} \quad (\text{A.1h})$$

The solution of problem (A.1) gives  $\mathbf{x}_{k-n_e}^*$  and  $\mathbf{v}_{k-i}^*, i = 1 \dots n_e$ . To obtain  $\hat{\mathbf{x}}_k$  the model equations (A.2) must be integrated over  $t \in [t_{k-n_e}, t_k]$  starting from  $\mathbf{x}_{k-n_e}^*$ , using the estimated disturbances  $\mathbf{v}_{k-i}^*, i = 1, \dots, n_e$ , and applying  $\mathbf{u}_{k-i}$ .

$$\mathbf{f}(\dot{\mathbf{x}}, \mathbf{x}, \mathbf{u}, \mathbf{v}^*) = 0, \quad \forall t \in [t_{k-n_e}, t_k], \quad \mathbf{x}(t_{k-n_e}) = \mathbf{x}_{k-n_e}^* \quad (\text{A.2})$$

The states calculated at time  $t_k$  from (A.2) are defined as the initial value for the MPC problem as in Equation (A.3).

$$\hat{\mathbf{x}}_k = \mathbf{x}(t_k) \quad (\text{A.3})$$

For more detailed information about the implementation of the MHE algorithm please refer to Oliveira-Silva et al. (2021).

## Appendix B. Williams-Otto model equations

The full model that emulates the real process and the simplified model used in the supervisory layer (eMPC+MHE) of the Williams-Otto reactor in Section 4 are presented in the Table B.1. The parameters used in each model are in Table B.2 and the prices and costs used in the economic problem are in Table B.3. For more detailed information about the models please refer to Oliveira-Silva et al. (2021).

Table B.1: Equations of the models presented in Section 0.

	Full Model	Simplified Model
Kinetics	$r_1 = k_1 X_A X_B$	$r_1 = k_1 X_A X_B^2$
	$r_2 = k_2 X_B X_C$	$r_2 = k_2 X_A X_B X_P$

	$r_3 = k_3 X_C X_P$	-
	$k_i = k_{i,0} e^{-(E_{ai}/T)}, i = 1,2,3$	$k_i = k_{i,0} e^{-(E_{ai}/T)}, i = 1,2$
CSTR mass balance	$V_R \frac{dX_A}{dt} = F_A X_{A0} - F_R X_A - V_R r_1$	$V_R \frac{dX_A}{dt} = F_A X_{A0} - F_R X_A - V_R r_1 - V_R r_2$
	$V_R \frac{dX_B}{dt} = F_B X_{B0} - F_R X_B - V_R r_1 - V_R r_2$	$V_R \frac{dX_B}{dt} = F_B X_{B0} - F_R X_B - 2V_R r_1 - V_R r_2$
	$V_R \frac{dX_C}{dt} = -F_R X_C + V_R r_1 - V_R r_2 - V_R r_3$	-
	$V_R \frac{dX_E}{dt} = -F_R X_E + V_R r_2$	$V_R \frac{dX_E}{dt} = -F_R X_E + V_R r_1$
	$V_R \frac{dX_G}{dt} = -F_R X_G + V_R r_3$	$V_R \frac{dX_G}{dt} = -F_R X_G + V_R r_2$
	$V_R \frac{dX_P}{dt} = -F_R X_P + V_R r_2 - V_R r_3$	$V_R \frac{dX_P}{dt} = -F_R X_P + V_R r_1 - V_R r_2$
	$F_R = F_A + F_B$	Same

Table B.2: Parameters of Williams-Otto reactor models.

Parameter	Full model	Simplified Model	Unit
$k_{1,0}$	$9.9594 \times 10^6$	$1.3134 \times 10^8$	$l/(mol \min)$
$k_{2,0}$	$8.66124 \times 10^9$	$2.586 \times 10^{13}$	$l/(mol \min)$
$k_{3,0}$	$1.6047 \times 10^{13}$	--	$l/(mol \min)$
$E_{a1}$	6666.7	8077.6	K
$E_{a2}$	8333.3	12438.5	K
$E_{a3}$	11111	--	K
$X_{A0}$	10	10	$mol/l$
$X_{B0}$	10	10	$mol/l$
$V_R$	2105	2105	$l$
$F_A$	112.35	112.35	$l/min$

Table B.3: Cost of the reactants and price of products.

Reactant or product	Value
$p_A$	7.623 €/mol

$p_B$	11.434 €/mol
$p_p$	114.338 €/mol
$p_E$	5.184 €/mol

## References

- Bank, R., Graham, R.L., Stoer, J., Varga, R., Yserentant, H., 1993. Solving Ordinary Differential Equations I, Math2, Springer Series in Computational Mathematics. Springer Berlin Heidelberg, Berlin, Heidelberg. <https://doi.org/10.1007/978-3-540-78862-1>
- Bergh, L.G., Yianatos, J.B., 2014. Engineers training in automation of flotation processes, IFAC Proceedings Volumes (IFAC-PapersOnline). IFAC. <https://doi.org/10.3182/20140824-6-za-1003.01536>
- EA Int., 2020. EcosimPro 6.2 [WWW Document]. User Man. URL <https://www.ecosimpro.com/>
- Faulwasser, T., Pannocchia, G., 2019. Toward a Unifying Framework Blending Real-Time Optimization and Economic Model Predictive Control. Ind. Eng. Chem. Res. 58, 13583–13598. <https://doi.org/10.1021/acs.iecr.9b00782>
- Goodwin, G.C., Sin, K.S., 1984. Adaptive filtering prediction and control, Dover Publications. Englewood Cliffs, New York.
- Hart, W.E., Laird, C.D., Watson, J.-P., Woodruff, D.L., Hackebeil, G.A., Nicholson, B.L., Siirola, J.D., 2017. Pyomo — Optimization Modeling in Python, Springer Optimization and Its Applications. Springer International Publishing, Cham. <https://doi.org/10.1007/978-3-319-58821-6>
- Hart, W.E., Watson, J.P., Woodruff, D.L., 2011. Pyomo: Modeling and solving mathematical programs in Python. Math. Program. Comput. 3, 219–260. <https://doi.org/10.1007/s12532-011-0026-8>
- Hernández, R., Engell, S., 2019. Economics optimizing control with model mismatch based on modifier adaptation. IFAC-PapersOnLine 52, 46–51. <https://doi.org/10.1016/j.ifacol.2019.06.035>
- Isermann, R., Münchhof, M., 2011. Identification of Dynamic Systems: An Introduction with Applications, Identification of Dynamic Systems. [https://doi.org/10.1007/978-3-540-78879-9\\_9](https://doi.org/10.1007/978-3-540-78879-9_9)
- Marchetti, A., Chachuat, B., Bonvin, D., 2009. Modifier-Adaptation Methodology for Real-Time Optimization. Ind. Eng. Chem. Res. 48, 6022–6033. <https://doi.org/10.1021/ie801352x>
- Montes, D., Zamarreño, J.M., Pitarch, J.L., Oliveira da Silva, E., Prada, C. De, 2021. Implementación de capas superiores de la pirámide de automatización en una planta piloto híbrida, in: XLII JORNADAS DE AUTOMÁTICA: LIBRO DE ACTAS. Servizo de Publicacións da UDC, pp. 403–410. <https://doi.org/10.17979/spudc.9788497498043.403>
- Navia, D., Puen, A., Bergh, L., Rodriguez-Blanco, T., Sarabia, D., de Prada, C., 2017. Modifier-Adaptation

- Based on Transient Measurements Applied to a Laboratory-Scale Flotation Column. *Comput. Aided Chem. Eng.* 40, 1729–1734. <https://doi.org/10.1016/B978-0-444-63965-3.50290-7>
- Navia, D., Puen, A., Quintanilla, P., Briceño, L., Bergh, L., 2019. On dealing with measured disturbances in the modifier adaptation method for real-time optimization. *Comput. Chem. Eng.* 128, 141–163. <https://doi.org/10.1016/j.compchemeng.2019.06.004>
- Navia, D., Villegas, D., Cornejo, I., de Prada, C., 2016. Real-time optimization for a laboratory-scale flotation column. *Comput. Chem. Eng.* 86, 62–74. <https://doi.org/10.1016/j.compchemeng.2015.12.006>
- Nordsieck, A., 1962. On Numerical Integration of Ordinary Differential Equations. *Math. Comput.* 16, 22–49.
- Oliveira-Silva, E., de Prada, C., Navia, D., 2021. Dynamic optimization integrating modifier adaptation using transient measurements. *Comput. Chem. Eng.* 149, 107282. <https://doi.org/10.1016/j.compchemeng.2021.107282>
- Pannocchia, G., 2018. An economic MPC formulation with offset-free asymptotic performance. *IFAC-PapersOnLine* 51, 393–398. <https://doi.org/10.1016/j.ifacol.2018.09.332>
- Papasavvas, A., de Avila Ferreira, T., Marchetti, A.G., Bonvin, D., 2019. Analysis of output modifier adaptation for real-time optimization. *Comput. Chem. Eng.* 121, 285–293. <https://doi.org/10.1016/j.compchemeng.2018.09.028>
- Rawlings, J.B., Mayne, D.Q., Diehl, M.M., 2019. *Model predictive control: Theory, Computation, and Design*, 2nd ed, Studies in Systems, Decision and Control. Nob Hill Publishing, Santa Barbara.
- Richalet, J., 1991. *Pratique de l'identification*. Editions Hermes, Paris.
- Rodríguez-Blanco, T., Sarabia, D., Pitarch, J.L., de Prada, C., 2017. Modifier Adaptation methodology based on transient and static measurements for RTO to cope with structural uncertainty. *Comput. Chem. Eng.* 106, 480–500. <https://doi.org/10.1016/j.compchemeng.2017.07.001>
- Vaccari, M., Bonvin, D., Pelagagge, F., Pannocchia, G., 2021. Offset-Free Economic MPC Based on Modifier Adaptation : Investigation of Several Gradient-Estimation Techniques 1–24.
- Vaccari, M., Pannocchia, G., 2018. Implementation of an economic MPC with robustly optimal steady-state behavior. *IFAC-PapersOnLine* 51, 92–97. <https://doi.org/10.1016/j.ifacol.2018.10.180>
- Vaccari, M., Pannocchia, G., 2016. A Modifier-Adaptation Strategy towards Offset-Free Economic MPC. *Processes* 5, 2. <https://doi.org/10.3390/pr5010002>
- Vaccari, M., Pelagagge, F., Bonvin, D., Pannocchia, G., 2020. Estimation technique for offset-free economic MPC based on modifier adaptation. *IFAC-PapersOnLine* 53, 11251–11256. <https://doi.org/10.1016/j.ifacol.2020.12.357>
- Van de Vusse, J.G., 1964. Plug-flow type reactor versus tank reactor. *Chem. Eng. Sci.* 19, 994–996.

[https://doi.org/10.1016/0009-2509\(64\)85109-5](https://doi.org/10.1016/0009-2509(64)85109-5)

Williams, T.J., Otto, R.E., 1960. A generalized chemical processing model for the investigation of computer control. Trans. Am. Inst. Electr. Eng. Part I Commun. Electron. 79, 458–473.  
<https://doi.org/10.1109/tce.1960.6367296>

#### Declaration of interests

☒ The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

☐ The authors declare the following financial interests/personal relationships which may be considered as potential competing interests:

#### CRediT author statement

**Erika Oliveira-Silva:** Methodology, Software, Investigation, Visualization, Writing-Original draft preparation, Writing - review & editing. **Cesar de Prada:** Conceptualization, Methodology, Supervision, Funding acquisition, Project Administration, Writing - review & editing. **Daniel Montes:** Methodology, Software, Investigation, Visualization, Writing-Original draft preparation. **Daniel Navia:** Supervision, Writing - Review & Editing.