Preprints of the 10th IFAC International Symposium on Dynamics and Control of Process Systems
The International Federation of Automatic Control
December 18-20, 2013. Mumbai, India

MPC Performance Monitoring of a Rigorously Simulated Industrial Process

Gabriele Pannocchia * **Michele Bottai** ** **Andrea De Luca** ***

* *Department of Civil and Industrial Engineering, University of Pisa, Pisa 56126, Italy (Corresponding author, e-mail: g.pannocchia@diccism.unipi.it).*
** *Novachem S.r.l., Sesto San Giovanni, Milan 20099, Italy*
*** *Solvay Specialty Polymers S.p.A, 15122 Spinetta Marengo (AL), Italy*

Abstract: We address in this paper the application of a recently proposed MPC performance monitoring method to a rigorously simulated industrial process. The methodology aims at detecting possible sources of suboptimal performance of linear offset-free MPC algorithms by analysis of the prediction error sequence, discriminating between the presence of plant/model mismatch and incorrect disturbance/state estimation, and proposing for each scenario an appropriate corrective action. We focus on the applicability of the method to large-scale industrial systems, which typically comprise a block structure, devising efficient and scalable diagnosis and correction procedures. We also discuss and support the application of this method when the controlled plant shows a mild nonlinear behavior mainly associated with operating point changes. A high-fidelity dynamic simulation model of a crude distillation unit was developed in UniSim® Design and used as representative test bench. Results show the efficacy of the method and indicate possible research directions for further improvements.

Keywords: Performance Monitoring, Modeling and Dynamic Simulation, Model Predictive Control, Systems Identification, Industrial Applications

1. INTRODUCTION

Model Predictive Control (MPC) received tremendous attention in the last decades both from academic and industrial communities due to the unique ability of handling multivariable constrained processes and achieving optimization goals (Rawlings and Mayne, 2009; Qin and Badgwell, 2003). In most continuous industrial processes a linear model is employed within the MPC algorithm (Qin and Badgwell, 2003), and such a model is typically obtained from plant data using systems identification methods (Zhu, 2001). Due to the fact that, in general, the actual process displays some nonlinear behavior (often mild around a given operating point) an effective system identification is a crucial step for the success of an MPC system (Qin and Badgwell, 2003; Darby and Nikolaou, 2012).

In order to cope with some unavoidable plant/model mismatch and/or the presence of unmeasured nonzero mean disturbances, all industrial MPC algorithms have some form of offset-free strategy. Most industrial implementations of such offset-free strategies can be cast within the framework of disturbance models (Muske and Badgwell, 2002; Pannocchia and Rawlings, 2003), in which the input-to-output model is augmented with (fictitious) integrating states referred to as disturbances, whose value is estimated from the output measurements and taken into account by the MPC in its predictions. This approach represents a simple form of model adaptation, which proves effective to remove steady-state offset under suitable conditions (Muske and Badgwell, 2002; Pannocchia and Rawlings, 2003), but the choice of disturbance model (i.e. model matrices and observer gain) is nonunique and can lead to significantly different closed-loop responses even for linear processes (Muske and Badgwell, 2002; Pannocchia, 2003; Pannocchia and Rawlings, 2003; Rajamani et al., 2009).

While many theoretical aspects of MPC (such as nominal stability, recursive feasibility, constraint satisfaction, etc.) can be regarded as relatively mature (Rawlings and Mayne, 2009), one of the aspects that currently draw the attention of both researchers and industrial practitioners is associated to devising effective control performance monitoring tools for MPC systems (Darby and Nikolaou, 2012; Shardt et al., 2012). While benchmarking of conventional (PID) controllers is purely "data-driven", the availability of a process model as in MPC allows for different opportunities. Many performance monitoring methods consider the prediction error, i.e. the difference between the actual process output and the model prediction, as a key performance indicator for MPC (Kesavan and Lee, 2001; Harrison and Qin, 2009; Zhao et al., 2010; Sun et al., 2013).

A common industrial situation arises when the MPC linear model no longer matches adequately the actual plant behavior due to a relevant change in plant operating conditions. This change is often determined by the MPC itself, which tends to drive the controlled plant around a new (typically more profitable) steady state. Other common sources of model incorrectness are associated to changes in raw materials, e.g. the crude oil in a topping distillation process, and unmeasured nonzero mean disturbances, e.g. changes in utilities conditions and other interacting variables. Because a new model identification step is costly, i.e. time consuming and production invasive (Qin and Badgwell, 2003), it is crucial to elaborate efficient and reliable methods for detecting plant/model mismatch and quantifying its influence onto the controller performance (Badwe et al., 2010; Jia et al., 2012). In this work we explore the potentials of the method recently proposed in (Pannocchia and De Luca, 2012; Pannocchia et al., 2013) within the context of a large-scale rigorously simulated industrial process, namely a crude

Copyright © 2013 IFAC

distillation unit (CDU). Due to the large number of inputs and outputs, the block structured nature of the process (i.e. blocks of outputs are affected by blocks of inputs, instead of all outputs being affected by all inputs), and the nonlinearity which mainly depends on changes of operating point, this represents a challenging and significant test of applicability.

2. MPC DESIGN AND PERFORMANCE MONITORING

We review the foundations of the performance monitoring strategy proposed in (Pannocchia et al., 2013), and support its extension to the context of the present application.

2.1 Linear offset-free MPC algorithm

The considered linear offset-free MPC algorithm has three main modules: an observer of the augmented state, a steady-state target calculator, and a dynamic optimizer. Due to space limitations, we only describe the observer because its variables directly appear in the performance monitoring method. The other two modules are described elsewhere [see e.g. (Pannocchia and Rawlings, 2003; Rawlings and Mayne, 2009)].

Starting from a discrete-time linear time-invariant model of the controlled system, described in state-space form by the triple $(\hat{A},\hat{B},\hat{C})$, the design of an offset-free MPC algorithm is based on the following augmented system [see e.g. (Muske and Badgwell, 2002; Pannocchia and Rawlings, 2003)]:

$$\begin{bmatrix}\hat{x}\\ \hat{d}\end{bmatrix}^+ = \begin{bmatrix}\hat{A} & \hat{B}_d\\ 0 & I\end{bmatrix}\begin{bmatrix}\hat{x}\\ \hat{d}\end{bmatrix} + \begin{bmatrix}\hat{B}\\ 0\end{bmatrix} u \tag{1a}$$

$$\hat{y} = \begin{bmatrix}\hat{C} & \hat{C}_d\end{bmatrix}\begin{bmatrix}\hat{x}\\ \hat{d}\end{bmatrix}, \tag{1b}$$

in which $\hat{x}\in\mathbb{R}^{\hat{n}}$ is the model state, $\hat{d}\in\mathbb{R}^p$ is the (stepwise) model disturbance, $u\in\mathbb{R}^m$ is the (manipulated) input, $\hat{y}\in\mathbb{R}^p$ is the model output. We make the following assumptions.

Assumption 1. The *plant output*, $y\in\mathbb{R}^p$, is measured at each sampling time $k\in\mathbb{I}_{\geq 0}$. The pair $(\hat{A},\hat{B})$ is stabilizable, the pair $(\hat{A},\hat{C})$ is detectable, and rank $\left(\begin{bmatrix}\hat{A}-I & \hat{B}_d\\ \hat{C} & \hat{C}_d\end{bmatrix}\right) = \hat{n}+p$.

Given *plant* and *model* outputs, the prediction error is: $e \triangleq y - \hat{y} = y-(\hat{C}\hat{x}+\hat{C}_d\hat{d})$, where we used (1b). Given the prediction error e, we obtain a *filtered* estimate of the augmented state as:

$$\begin{bmatrix}\hat{x}_*\\ \hat{d}_*\end{bmatrix} \triangleq \begin{bmatrix}\hat{x}\\ \hat{d}\end{bmatrix} + \begin{bmatrix}\hat{L}_x\\ \hat{L}_d\end{bmatrix} e, \tag{2}$$

where $\hat{L}_x\in\mathbb{R}^{\hat{n}\times p}$, $\hat{L}_d\in\mathbb{R}^{p\times p}$. We remark that $\begin{bmatrix}\hat{x}(k)\\ \hat{d}(k)\end{bmatrix}$ is predicted at time $k-1$ using (1a), whereas $\begin{bmatrix}\hat{x}_*(k)\\ \hat{d}_*(k)\end{bmatrix}$ is computed at time k from (2), i.e. using $y(k)$.

2.2 Performance monitoring method: a basic review

We can combine prediction (1a) and filtering (2) steps into:

$$\begin{aligned}\hat{x}_a^+ &= \hat{A}_a\hat{x}_a + \hat{B}_a u + \hat{K}_a e\\ y &= \hat{C}_a\hat{x}_a + e,\end{aligned} \tag{3}$$

known as *predictor* form, in which

$$\hat{x}_a \triangleq \begin{bmatrix}\hat{x}\\ \hat{d}\end{bmatrix},\ \hat{A}_a \triangleq \begin{bmatrix}\hat{A} & \hat{B}_d\\ 0 & I\end{bmatrix},\ \hat{B}_a \triangleq \begin{bmatrix}\hat{B}\\ 0\end{bmatrix},\ \hat{C}_a \triangleq \begin{bmatrix}\hat{C} & \hat{C}_d\end{bmatrix},$$

$$\hat{K}_a \triangleq \begin{bmatrix}\hat{K}_x\\ \hat{K}_d\end{bmatrix} = \begin{bmatrix}\hat{A} & \hat{B}_d\\ 0 & I\end{bmatrix}\begin{bmatrix}\hat{L}_x\\ \hat{L}_d\end{bmatrix} = \begin{bmatrix}\hat{A}\hat{L}_x+\hat{B}_d\hat{L}_d\\ \hat{L}_d\end{bmatrix}. \tag{4}$$

We recall the following well-known result of systems theory.

Definition 2. A system in the form (3) is *algebraically equivalent* (AE) to a system in the same form with matrices (A_a,B_a,C_a,K_a) if there exists an invertible matrix T such that:

$$A_a = T\hat{A}_aT^{-1},\ B_a = T\hat{B}_a,\ C_a = \hat{C}_aT^{-1},\ K_a = T\hat{K}_a. \tag{5}$$

We remark that if two systems in form (3) are AE, they generate identical sequences of prediction error given the same sequences of inputs and outputs.

The method proposed in (Pannocchia et al., 2013) assumes that the *actual plant* evolves as follows:

$$\begin{bmatrix}x\\ d\end{bmatrix}^+ = \begin{bmatrix}A & B_d\\ 0 & I\end{bmatrix}\begin{bmatrix}x\\ d\end{bmatrix} + \begin{bmatrix}B\\ 0\end{bmatrix}u + \begin{bmatrix}K_x\\ K_d\end{bmatrix}v \tag{6a}$$

$$y = \begin{bmatrix}C & C_d\end{bmatrix}\begin{bmatrix}x\\ d\end{bmatrix} + v, \tag{6b}$$

in which $x\in\mathbb{R}^n$ is the *plant* state, $d\in\mathbb{R}^p$ is the *plant* disturbance, and $v\in\mathbb{R}^p$ is a zero-mean white noise. Let $x_a \triangleq \begin{bmatrix}x\\ d\end{bmatrix}$ and let (A_a,B_a,C_a,K_a) be defined as in (4) with $(\hat{A},\hat{B},\hat{C},\hat{B}_d,\hat{C}_d,\hat{K}_x,\hat{K}_d)$ replaced by (A,B,C,B_d,C_d,K_x,K_d). We can combine the plant (6) and the MPC model (3) to obtain a description of the closed-loop (plant and model) system:

$$\begin{aligned}\begin{bmatrix}x_a\\ \hat{x}_a\end{bmatrix}^+ &= \underbrace{\begin{bmatrix}A_a & 0\\ \hat{K}_aC_a & \hat{A}_a-\hat{K}_a\hat{C}_a\end{bmatrix}}_{\mathbf{A}}\begin{bmatrix}x_a\\ \hat{x}_a\end{bmatrix} + \underbrace{\begin{bmatrix}B_a\\ \hat{B}_a\end{bmatrix}}_{\mathbf{B}}u + \underbrace{\begin{bmatrix}K_a\\ \hat{K}_a\end{bmatrix}}_{\mathbf{K}}v\\ e &= \underbrace{\begin{bmatrix}C_a & -\hat{C}_a\end{bmatrix}}_{\mathbf{C}}\begin{bmatrix}x_a\\ \hat{x}_a\end{bmatrix} + v,\end{aligned} \tag{7}$$

in which the prediction error is regarded as the *output*. For convenience of exposition, the closed-loop system (7) can be written in (discrete) transfer function form as $(\tilde{e}(z),\tilde{u}(z),\tilde{v}(z)$ are, respectively, the $\mathscr{Z}$-transforms of $\{e(k)\},\{u(k)\},\{v(k)\}$):

$$\tilde{e}(z) = \hat{G}_u(z)\tilde{u}(z) + \hat{G}_v(z)\tilde{v}(z), \tag{8}$$

in which $\hat{G}_u(z)\triangleq \mathbf{C}(zI-\mathbf{A})^{-1}\mathbf{B}$ and $\hat{G}_v(z)\triangleq \mathbf{C}(zI-\mathbf{A})^{-1}\mathbf{K}+I$.

As shown in (Pannocchia et al., 2013) the orders of *minimal realizations* of $\hat{G}_u(z)$ and $\hat{G}_v(z)$ (denoted by $\mathscr{N}_u$ and $\mathscr{N}_v$, respectively) change in different scenarios of suboptimal performance (e.g. plant/model mismatch and incorrect disturbance model/observer). The three different scenarios of interest, noticing that the plant (6) is also in form (3), are:

- *S1. Correct model and disturbance model/observer.* Plant (6) and model (3) are AE.
- *S2. Correct model, incorrect disturbance model/observer.* Matrices (A,B,C) and $(\hat{A},\hat{B},\hat{C})$ are identical (up to a similarity transformation) but plant (6) and model (3) are not AE due to incorrect disturbance model/observer.
- *S3. Incorrect model.* Plant matrices (A,B,C) and model matrices $(\hat{A},\hat{B},\hat{C})$ are not identical (up to any similarity transformation). Thus, plant (6) and model (3) are not AE.

Notice that when the model matrices $(\hat{A},\hat{B},\hat{C})$ are incorrect, it is irrelevant to discuss whether the disturbance model/observer is correct or not because (3) cannot be AE to (6). The method is based on the following results (Pannocchia et al., 2013): (i) In *S1* the prediction error is white noise. (ii) In *S2* the prediction error is not white noise and $\mathscr{N}_u = 0$. (iii) In *S3* the prediction error is not white noise and $\mathscr{N}_u > 0$.

From a diagnosis perspective the method requires a whiteness test, to be applied to each component of the prediction error. To this aim we use the well-known test of Ljung and Box (1978), which computes a statistic from an estimate of the autocorrelation function and a threshold value (depending on the number of lags considered in the autocorrelation and on the confidence level). When this statistic is below the threshold value, i.e. the *whiteness ratio* is less than 1, the tested sequence can be regarded as white noise; otherwise it is regarded as nonwhite noise. The second diagnosis tool is a Subspace IDentification (SID) method applied to $\{u(k)\}$ as input and $\{e(k)\}$ as output to determine the order of a minimal realization of $\hat{G}_u(z)$. It is important to observe that the input u is correlated with the *output* e of the system (7), and therefore a SID method with proven closed-loop consistency should be used. Secondly, it is not necessary to compute $\mathcal{N}_u$ exactly but simply to detect if $\mathcal{N}_u = 0$ or $\mathcal{N}_u > 0$. In this work we use the PARSIM-K method (Pannocchia and Calosi, 2010), and devise a specific rule to establish if $\mathcal{N}_u = 0$ or $\mathcal{N}_u > 0$ from the ratio of the largest and smallest singular values of an appropriate SVD (Pannocchia and Calosi, 2010, Eq. 13).

In *S2* and *S3* different corrections are appropriate. In *S2* it is necessary to recompute the filter gain matrices $(\hat{L}_x, \hat{L}_d)$ or equivalently $(\hat{K}_x, \hat{K}_d)$. In fact, as proved by Rajamani et al. (2009), when two offset-free MPC models have the same matrices $(\hat{A}, \hat{B}, \hat{C})$ but different disturbance model matrices $(\hat{B}_{d1}, \hat{C}_{d1})$ and $(\hat{B}_{d2}, \hat{C}_{d2})$, respectively, a model as in (3) with matrices $(\hat{A}, \hat{B}, \hat{C}, \hat{B}_{d2}, \hat{C}_{d2}, \hat{K}_{x2}, \hat{K}_{d2})$ can be made AE to a model as in (3) with matrices $(\hat{A}, \hat{B}, \hat{C}, \hat{B}_{d1}, \hat{C}_{d1}, \hat{K}_{x1}, \hat{K}_{d1})$ by appropriate choices of $(\hat{K}_{x2}, \hat{K}_{d2})$. That is, in *S2* for any choice of disturbance model matrices $(\hat{B}_d, \hat{C}_d)$ respecting Assumption 1 there exists a pair $(\hat{K}_x, \hat{K}_d)$ that makes the MPC model (3) AE to the actual plant (6). In order to compute a suitable observer gain matrices $(\hat{K}_x, \hat{K}_d)$ we use the following approach. Given input and output sequences, $\{u(k)\}$ and $\{y(k)\}$, recalling (3) and (4), we solve the following nonlinear optimization problem:

$$\min_{\hat{K}_a, \hat{x}_a(0)} \sum_k \|y(k) - \hat{C}_a \hat{x}_a(k)\|^2 \qquad \text{s.t. (3).} \tag{9}$$

The nonlinear optimization problem (9) is nonconvex; thus, in general, only a *suboptimal* solution can be obtained numerically. However, most Sequential Quadratic Programming (SQP) algorithms can guarantee that the obtained solution has a cost no larger than that of the initial guess. Therefore, if we initialize the SQP algorithm with the current augmented observer matrix $\hat{K}_a$ we obtain a new observer that generates a prediction error sequence with no larger norm. In *S3*, instead, the model matrices $(\hat{A}, \hat{B}, \hat{C})$ should be re-identified from input/output data collected, in general, by a specific *identification campaign* (Zhu, 2001). After the new matrices $(\hat{A}, \hat{B}, \hat{C})$ are identified, the appropriate observer gain is computed from (9).

We observe that in *S3* even the simpler choice of recomputing the observer gain can improve the MPC robustness to plant/model mismatch as discussed e.g. in (Pannocchia and Rawlings, 2003; Pannocchia, 2003), although it cannot make the MPC model (3) AE to the actual plant (6). The basic performance monitoring algorithm is described in Algorithm 1.

Algorithm 1. **Require:** Input, output and prediction error sequences $\{u(k)\}$, $\{y(k)\}$, $\{e(k)\}$ over a considered time period. MPC matrices $(\hat{A}, \hat{B}, \hat{C}, \hat{B}_d, \hat{C}_d, \hat{K}_x, \hat{K}_d)$.

1: For each component of $\{e(k)\}$, perform whiteness test
2: **if** $\{e(k)\}$ is "white" **then** {Optimal performance}
3:    No changes are necessary.
4: **else** {Suboptimal performance}
5:    Evaluate $\mathcal{N}_u$
6:    **if** $\mathcal{N}_u = 0$ **then** $\{(\hat{A}, \hat{B}, \hat{C})$ are correct.$\}$
7:       Evaluate $\hat{K}_a$ from (9).
8:    **else** $\{(\hat{A}, \hat{B}, \hat{C})$ are incorrect.$\}$
9:       Re-identify $(\hat{A}, \hat{B}, \hat{C})$. Choose $(\hat{B}_d, \hat{C}_d)$ satisfying Assumption 1. Compute $\hat{K}_a$ from (9).
10:    **end if**
11: **end if**

2.3 Performance monitoring method: extensions

Large-scale multivariable systems, as the example considered in this paper, are usually identified in a block form, i.e. from a subset of inputs to a subset of outputs. The i-th sub-model is:

$$\begin{aligned} \begin{bmatrix} \hat{x}_i \\ \hat{d}_i \end{bmatrix}^+ &= \begin{bmatrix} \hat{A}_i & \hat{B}_{di} \\ 0 & I \end{bmatrix} \begin{bmatrix} \hat{x}_i \\ \hat{d}_i \end{bmatrix} + \begin{bmatrix} \hat{B}_i \\ 0 \end{bmatrix} u + \begin{bmatrix} \hat{K}_{xi} \\ \hat{K}_{di} \end{bmatrix} e_i \\ y_i &= \begin{bmatrix} \hat{C}_i & \hat{C}_{di} \end{bmatrix} \begin{bmatrix} \hat{x}_i \\ \hat{d}_i \end{bmatrix} + e_i, \end{aligned} \tag{10}$$

in which $\hat{x}_i \in \mathbb{R}^{\hat{n}_i}$, $y_i \in \mathbb{R}^{p_i}$, $\hat{d}_i \in \mathbb{R}^{p_i}$, and $\sum_{i=1}^{P} p_i = p$. Notice that $\hat{B}_i$ may have some zero columns because not all components of the input vector u affect the subset of outputs y_i. The overall matrices $(\hat{A}, \hat{B}, \hat{C}, \hat{B}_d, \hat{C}_d, \hat{K}_x, \hat{K}_d)$ can be obtained by:

$$\hat{A} \triangleq \text{diag}\{\hat{A}_i\},\ \hat{B} \triangleq \begin{bmatrix} \hat{B}_1 \\ \vdots \\ \hat{B}_P \end{bmatrix},\ \hat{C} \triangleq \text{diag}\{\hat{C}_i\},\ \hat{B}_d \triangleq \text{diag}\{\hat{B}_{di}\},$$
$$\hat{C}_d \triangleq \text{diag}\{\hat{C}_{di}\},\ \hat{K}_x \triangleq \text{diag}\{\hat{K}_{xi}\},\ \hat{K}_d \triangleq \text{diag}\{\hat{K}_{di}\}. \tag{11}$$

The performance monitoring algorithm proposed in (Pannocchia and De Luca, 2012) can be specialized by treating each output and prediction error block (y_i and e_i), separately. In the next algorithm, $\mathcal{N}_u^i$ denotes the order of a minimal realization of $\hat{G}_u^i(z)$, transfer function from u to e_i, and:

$$\hat{A}_{ai} \triangleq \begin{bmatrix} \hat{A}_i & \hat{B}_{di} \\ 0 & I \end{bmatrix},\ \hat{B}_{ai} \triangleq \begin{bmatrix} \hat{B}_i \\ 0 \end{bmatrix},\ \hat{C}_{ai} \triangleq \begin{bmatrix} \hat{C}_i & \hat{C}_{di} \end{bmatrix},$$
$$\hat{K}_{ai} \triangleq \begin{bmatrix} \hat{K}_{xi} \\ \hat{K}_{di} \end{bmatrix},\ \hat{x}_{ai} \triangleq \begin{bmatrix} \hat{x}_i \\ \hat{d}_i \end{bmatrix}, \qquad i = 1, \ldots, P.$$

Algorithm 2. **Require:** Input, output and prediction error sequences $\{u(k)\}$, $\{y(k)\}$, $\{e(k)\}$ over a considered time period. MPC matrices $(\hat{A}, \hat{B}, \hat{C}, \hat{B}_d, \hat{C}_d, \hat{K}_x, \hat{K}_d)$ as in (11).

1: **for** $i = 1 \to P$ **do**
2:    Perform whiteness test on each component of $\{e_i(k)\}$.
3:    **if** $\{e_i(k)\}$ is "white" **then** {Optimal performance}
4:       No changes are necessary for i−th block.
5:    **else** {Suboptimal performance in i−th block}
6:       Evaluate $\mathcal{N}_u^i$.
7:       **if** $\mathcal{N}_u^i = 0$ **then** $\{(\hat{A}_i, \hat{B}_i, \hat{C}_i)$ are correct.$\}$
8:          Compute $\hat{K}_{ai}$ from (9) with $(\hat{A}_a, \hat{B}_a, \hat{C}_a, \hat{K}_a, \hat{x}_a, y)$ replaced by $(\hat{A}_{ai}, \hat{B}_{ai}, \hat{C}_{ai}, \hat{K}_{ai}, \hat{x}_{ai}, y_i)$.
9:       **else** $\{(\hat{A}_i, \hat{B}_i, \hat{C}_i)$ are incorrect.$\}$
10:          Re-identify $(\hat{A}_i, \hat{B}_i, \hat{C}_i)$. Choose $(\hat{B}_{di}, \hat{C}_{di})$ s.t. rank $\begin{bmatrix} \hat{A}_i - 1 & \hat{B}_{di} \\ \hat{C}_i & \hat{C}_{di} \end{bmatrix} = \hat{n}_i + p_i$. Compute $\hat{K}_{ai}$ as in Line 8.
11:       **end if**
12:    **end if**
13: **end for**

The second aspect that should be considered in industrial applications is that the true plant can be described by linear dynamics as in (6) only if the operating steady state remains constant. In general, we can write the plant dynamics as a nonlinear system:

$$x_a^+ = f(x_a, u, v) \tag{12a}$$
$$y = g(x_a) + v, \tag{12b}$$

in which $v \in \mathbb{R}^p$ is a zero-mean white noise and $f(\cdot)$ and $g(\cdot)$ are (obviously unknown) continuous nonlinear functions. For a given operating steady state, we can rewrite (12) as:

$$x_a^+ = A_a x_a + B_a u + K_a v + \varepsilon_x(x_a, u, v) \tag{13a}$$
$$y = C_x x_a + v + \varepsilon_y(x_a), \tag{13b}$$

in which

$$\varepsilon_x(x_a, u, v) \triangleq f(x_a, u, v) - A_a x_a + B_a u + K_a v \tag{14a}$$
$$\varepsilon_y(x_a) \triangleq g(x_a) - C_x x_a. \tag{14b}$$

By definition $\varepsilon_x(\cdot)$ and $\varepsilon_y(\cdot)$ vanish at the current equilibrium and are continuous. Thus, we can assume that they are small in a sufficiently small neighborhood of the current equilibrium. Combining (13) and the MPC model (3) we obtain:

$$\begin{bmatrix} x_a \\ \hat{x}_a \end{bmatrix}^+ = \underbrace{\begin{bmatrix} A_a & 0 \\ \hat{K}_a C_a & \hat{A}_a - \hat{K}_a \hat{C}_a \end{bmatrix}}_{\mathbf{A}} \begin{bmatrix} x_a \\ \hat{x}_a \end{bmatrix} + \underbrace{\begin{bmatrix} B_a \\ \hat{B}_a \end{bmatrix}}_{\mathbf{B}} u + \underbrace{\begin{bmatrix} K_a \\ \hat{K}_a \end{bmatrix}}_{\mathbf{K}} v + \begin{bmatrix} \varepsilon_x(\cdot) \\ \hat{K}_a \varepsilon_y(\cdot) \end{bmatrix}$$

$$e = \underbrace{\begin{bmatrix} C_a & -\hat{C}_a \end{bmatrix}}_{\mathbf{C}} \begin{bmatrix} x_a \\ \hat{x}_a \end{bmatrix} + v + \varepsilon_y(\cdot).$$

Comparing the above expression with (7), it follows that the main technical results of the proposed performance monitoring method discussed in Section 2.2 hold in the limit of $\varepsilon_x(\cdot), \varepsilon_y(\cdot) \to 0$.

3. CRUDE DISTILLATION UNIT: RIGOROUS DYNAMIC SIMULATION AND MPC DESIGN

We present in this section a simulated large-scale industrial process, specifically a crude distillation unit (CDU), as case study for the application of the proposed performance monitoring method.

3.1 Simulation model and base regulatory control details

The CDU rigorous simulation model was developed in (Bottai, 2012) using UniSim® Design, following two main steps: (i) steady state modeling; (ii) dynamic modeling. The first one involves building the process flow diagram (PFD) as interconnection of unit operations and solving mass, energy and equilibrium equations without accumulation terms to obtain a steady-state condition of the process. This is a necessary step for the transition to a dynamic model. In UniSim® Design every process equipment (valves, heaters, distillation columns, etc.) is modeled as one or more pressure node. Therefore, pressure gradients produce mass flows in the simulated PFD. High fidelity is ensured by using a small integration time (0.5 seconds), as well as considering the effect of static heads, equipment holdups, regulatory controllers, etc.

The PFD of the simulated CDU is represented in Figure 1. The crude fed to the plant was modeled using the Oil Characterization environment implemented in UniSim® Design. By providing conventional petroleum assay data it generates a series of discrete hypothetical components to represent components of a crude oil (and the corresponding properties). The following five products are obtained from the CDU:

- AN: naphtha, from the overhead three-phase separator;
- KERO: kerosene, from upper side stripper;
- LGO: light gas-oil, from middle side stripper;
- HGO: heavy gas-oil, from bottom side stripper;
- AR: atmospheric residue, from the bottom of the column.

Table 1. Quality specifications of the products.

Product	ASTM-D86 95% [°C]
AN	149
KERO	241
LGO	349.4
HGO	390.9

Table 2. MPC manipulated variables.

Controller	Tag	Description
TIC-101	MV1	Overhead tray temperature controller SP
FIC-201	MV2	KERO flow-rate controller SP
FIC-301	MV3	LGO flow-rate controller SP
FIC-401	MV4	HGO flow-rate controller SP
TIC-001	MV5	Furnace outlet temperature controller SP
FIC-001	MV6	Crude oil flow-rate controller SP
FIC-202	MV7	KERO stripping steam flow-rate controller SP
FIC-302	MV8	LGO stripping steam flow-rate controller SP
FIC-402	MV9	HGO stripping steam flow-rate controller SP
FIC-502	MV10	Main column steam flow-rate controller SP
Q-UPA	MV11	Upper pump-around duty
Q-MPA	MV12	Middle pump-around duty
Q-BPA	MV13	Bottom pump-around duty
PIC-101	MV14	Overhead pressure controller SP

For every column product we assigned values of ASTM-D86 95% vol. as quality specification (see Table 1). Base regulatory controllers were included in the UniSim® Design dynamic model, and they are represented in Figure 1.

3.2 MPC objectives and design

The first step in an MPC design is the choice of the manipulated (MV) and controlled (CV) variables. As in most industrial applications, the MPC manipulates set-points of base controllers. With reference to Figure 1, set-points of the control loops highlighted in red were chosen as manipulated variables in the MPC design. In our case study we chose the 14 MVs listed in Table 2 because of their well known influence and relevance in a topping process. We chose 25 CVs divided into three main categories:

- 4 CVs (AN, KERO, LGO, HGO qualities, expressed in terms of ASTM-D86 95%) have set-points;
- 5 CVs (flow ratios) have ranges to respect;
- 16 CVs (valve openings) have saturation constraints.

In order to apply the MPC algorithm developed in MATLAB to the rigorously simulated CDU model in UniSim® Design environment, it was necessary to create a connection interface for the data exchange between the two software environments. At each sampling time (1 minute), the MPC algorithm in MATLAB reads the CVs from UniSim® Design model and computes the MVs which are passed to the UniSim® Design model as set-points. Then MATLAB sends a command to UniSim® Design to integrate the model for 1 minute. Using the connection interface it was also possible to collect plant data for systems identification.

Due to the large number of CVs and MVs, it was useful to identify and arrange the state-space matrices $(\hat{A}, \hat{B}, \hat{C})$ in blocks, each describing the dynamic response of a group of CVs for variations of a group of MVs as discussed in Section 2.3.

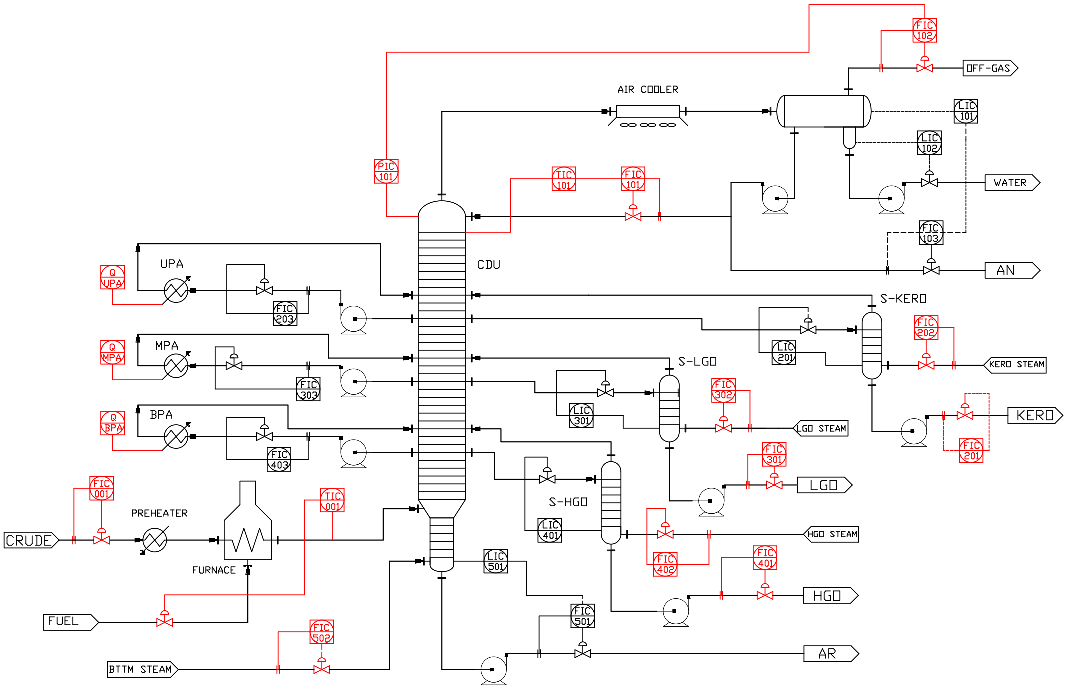

Fig. 1. Flow sheet of the rigorously simulated CDU.

CVs/MVs	TIC101SP	TIC001SP	PIC101SP	QBPASP	FIC001SP	QMPASP	QUPASP	FIC301SP	FIC201SP	FIC401SP	FIC502SP	FIC402SP	FIC202SP	FIC302SP
ASTM AN	1,597	-0,084	-24,683	0,168	-0,018	0,104	0	0	0	0	0	0	0	0
TIC001OP	-0,206	0,376	-0,839	0,017	0,098	-0,050	0	0	0	0	0	0	0	0
FIC103OP	0,716	-0,076	-2,702	0,138	0,036	0,165	0	0	0	0	0	0	0	0
Reflux Ratio	-0,028	0,020	0,167	-0,035	0,001	-0,041	0	0	0	0	0	0	0	0
ASTM KERO	1,341	-0,109	-10,770	0,433	-0,072	0,021	0,021	0,083	0,386	0,109	0,147	0	0	0
LIC102OP	-0,039	-0,138	-38,332	-0,059	-0,020	0,061	-0,084	0,134	0,005	0,133	5,626	0	0	0
ASTM HGO	3,241	-2,068	-31,815	-0,438	-0,086	0,114	-1,325	1,428	1,329	2,961	1,449	0	0	0
ASTM LGO	3,428	-1,131	-12,750	1,272	-0,216	-0,769	-0,414	1,056	1,036	0,968	0,232	0	0	0
FIC501OP	-0,324	-0,019	0,599	-0,088	0,093	0,081	-0,058	-0,156	-0,176	-0,250	-0,332	0	0	0
STEAM/AR	0,516	-0,145	-7,108	0,055	-0,087	-0,013	-0,016	0,151	0,228	0,437	3,911	0	0	0
FIC102OP	-1,133	0,165	-168,640	-1,965	0,086	-1,544	-2,339	0,521	0,080	0,493	1,299	0	0	0
FIC101OP	-0,272	0,516	-4,531	-1,066	0,059	-1,061	-1,207	0,009	0,031	0,064	0,267	0	0	0
LIC301OP	0	0	0	0	0	0	0,001	0,273	-0,007	-0,001	-0,041	0	0	0
FIC301OP	0	0	0	0	0	0	0,025	0,276	-0,002	-0,001	-0,026	0	0	0
FIC201OP	0	0	0	0	0	0	0,028	-0,003	0,323	-0,003	-0,015	0	0	0
LIC201OP	0	0	0	0	0	0	-0,012	-0,010	0,340	-0,003	-0,051	0	0	0
LIC401OP	0	0	0	0	0	0	0,019	-0,018	-0,026	1,952	-0,103	0	0	0
FIC401OP	0	0	0	0	0	0	0,035	0,000	-0,003	2,596	-0,040	0	0	0
FIC501OP	0	0	0	0	0	0	-0,098	-0,019	-0,019	-0,003	6,001	0	0	0
STEAM/HGO	0	0	0	0	0	0	-0,060	0,004	0,015	-0,748	-0,060	36,772	-7,106	0,389
STEAM/KERO	0	0	0	0	0	0	0,000	0,000	-0,004	0,000	0,000	-0,001	6,795	0,002
STEAM/LGO	0	0	0	0	0	0	0,001	-0,007	0,000	0,000	0,004	0,013	0,209	5,610
FIC402OP	0	0	0	0	0	0	0	0	0	0	0	27,043	0,051	0,143
FIC202OP	0	0	0	0	0	0	0	0	0	0	0	0,022	241,980	0,063
FIC302OP	0	0	0	0	0	0	0	0	0	0	0	0,125	-0,136	108,180
MVs ref.	129	338	1,95	15	860	8	3	177,2	147,3	27,7	13	0,575	0,085	0,23
UnitS	°C	°C	atm	Mcal	ton/hr	Mcal	Mcal	ton/hr	ton/hr	ton/hr	ton/hr	ton/hr	ton/hr	ton/hr
GBN STEP	1	5	0,05	1	5	1	1	3	5	2	1	0,1	0,01	0,1

Fig. 2. Identified steady-state matrix.

Systems identification was carried out in three steps: (i) Generalized Binary Noise signals [GBN, see e.g. (Zhu, 2001)] were generated in MATLAB for each MV and commanded to the CDU model to collect CVs; (ii) from the collected MVs and CVs data a state-space model was obtained for each block using the PARSIM-K method (Pannocchia and Calosi, 2010); (iii) the overall state-space model as in (4) was assembled. The steady-state gain matrix is reported in Figure 2, in which the various blocks (5 CV blocks and 3 MV blocks) and the overall sparsity of the model are highlighted. The model has $\hat{n} = 80$ states, and the MPC uses $\hat{B}_d = 0$, $\hat{C}_d = I$, $\hat{L}_x = 0$, $\hat{L}_d = I$.

4. RESULTS AND DISCUSSION

We considered the case in which the MPC algorithm uses a linear model that was obtained from identification around a steady-state that is no longer up-to-date. This is a frequent situation in industrial MPC applications because in the design phase the MPC model is typically obtained around the current operating point, but then after closed-loop implementation, the MPC tends to move the plant towards a different (generally, more profitable) steady state.

We summarize in Table 3 the most significant performance results. In reference conditions, the performance monitoring method detects that the prediction error is nonwhite in all variables, as highlighted in Table 3 by a whiteness ratio significantly higher than 1 for all product quality CVs (but also for the other CVs not shown in Table 3). The test on the order of $\hat{G}_u(z)$ reveals that the model is incorrect, i.e. $\mathcal{N}_u > 0$, and recommends re-identification of the model matrices $(\hat{A}, \hat{B}, \hat{C})$ followed by recalculation of the observer gain. As cheaper and non-invasive remedy action, if we simply recompute the observer gain from (9), we obtain an appreciable decrease of the whiteness ratio in all product quality CVs. Most importantly, the closed-loop cost function decreases by 36%. As ultimate solution suggested by the proposed performance monitoring method, we collected identification data around the new current steady state, proceeding in a similar way as in the first identification required for initial MPC commissioning, and obtained new model matrices $(\hat{A}, \hat{B}, \hat{C})$ in which $\hat{n} = 72$. When this correction is implemented the whiteness ratio of each output is reduced significantly, although it never approaches 1 because of the unavoidable mismatch between the linear MPC model and the actual nonlinear behavior of the plant. Most importantly, the overall closed-loop cost function is reduced by 80% with respect to reference conditions.

In order to appreciate the benefits of the corrective action, we report in Figure 3 the closed-loop behavior of the four product qualities, in reference conditions and after use of the newly identified model and observer, during a sequence of set-point changes in naphtha and kerosene quality specifications. We clearly notice that the use of a new model (and observer) grants tremendous improvements in the closed-loop behavior.

Table 3. Whiteness ratio for the four product qualities and closed-loop cost function.

Case	Whiteness ratio				Closed-loop
	AN	KERO	LGO	HGO	cost
Reference	1087	3541	3304	3641	$4.27 \cdot 10^5$
New observer	164.5	848.2	2187	2183	$2.73 \cdot 10^5$
New model and observer	577.5	1023	756.1	905.7	$8.27 \cdot 10^4$

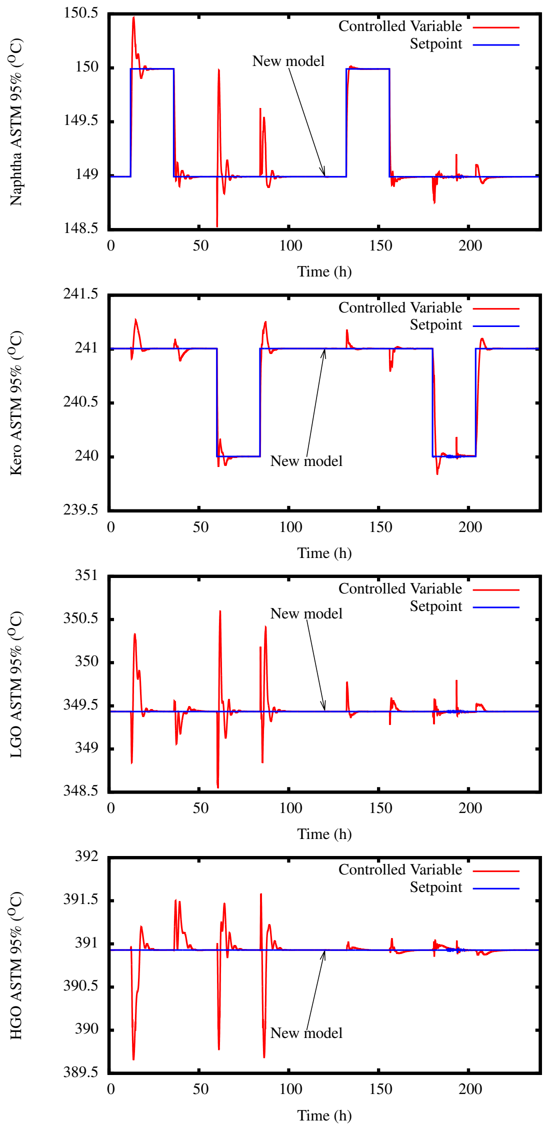

Fig. 3. Behavior of AN, KERO, LGO and HGO ASTM-D86 95% in reference conditions (up to time 120 h) and after use of the re-identified MPC model (after time 120 h)

5. CONCLUSIONS

We discussed in this paper the application of the performance monitoring method proposed in (Pannocchia et al., 2013) for linear offset-free MPC algorithms to a rigorously simulated Crude Distillation Unit. Such a process was chosen due to its high relevance within the process industries and, most importantly, because it represents a challenging test bench of a large-scale industrial process. The required extensions to large-scale block structured systems were discussed. Then the CDU process was rigorously modeled using UniSim® Design, and an interface between the controlled process and the MPC algorithm developed in MATLAB was devised. The performance monitoring method revealed correctly the presence of plant/model mismatch and suggested re-identification of the MPC model. After implementation of the new model the closed-loop performance improved by 80%. A less effective, but significantly less invasive, corrective action is the computation of a new observer gain, which can prove useful in all situations in which systems identification cannot be scheduled in a reasonable timeframe, e.g. due production constraints. Future research will be devoted to quantifying the tradeoff between these two corrective actions, hence providing the control engineers with further information regarding the strong or tolerable necessity of re-identifying the MPC model.

REFERENCES

Badwe, A., Gudi, R.D., Patwardhan, R.S., Shah, S.L., and Patwardhan, S.C. (2010). Quantifying the impact of model-plant mismatch on controller performance. *J. Proc. Contr.*, 20, 408–425.

Bottai, M. (2012). *Application of a monitoring and diagnosis system of predictive controllers*. Master's thesis, Chemical Engineering, University of Pisa, Available (in Italian) at http://etd.adm.unipi.it/theses/available/etd-06232012-114823/.

Darby, M.L. and Nikolaou, M. (2012). MPC: Current practice and challenges. *Contr. Eng. Pract.*, 20, 328–342.

Harrison, C.A. and Qin, S.J. (2009). Discriminating between disturbance and process model mismatch. *J. Proc. Contr.*, 19, 1610–1616.

Jia, G., Zhanga, K., and Zhu, Y. (2012). A method of MPC model error detection. *J. Proc. Contr.*, 22, 635–642.

Kesavan, P. and Lee, J.H. (2001). A set based approach to detection and isolation of faults in multivariable systems. *Comp. Chem. Eng.*, 25, 925–940.

Ljung, G.M. and Box, G.E.P. (1978). On a measure of lack of fit in time series models. *Biometrika*, 65, 297–303.

Muske, K.R. and Badgwell, T.A. (2002). Disturbance modeling for offset-free linear model predictive control. *J. Proc. Contr.*, 12, 617–632.

Pannocchia, G. (2003). Robust disturbance modeling for model predictive control with application to multivariable ill-conditioned processes. *J. Proc. Cont.*, 13, 693–701.

Pannocchia, G. and Calosi, M. (2010). A predictor form PARSIMonious algorithm for closed-loop subspace identification. *J. Proc. Contr.*, 20, 517–524.

Pannocchia, G. and De Luca, A. (2012). Performance degradation diagnosis and remedies in offset-free MPC. In *Proceedings of the 2012 Amer. Contr. Conf., Montreal, Canada*, 429–434.

Pannocchia, G., De Luca, A., and Bottai, M. (2013). Prediction error based performance monitoring, degradation diagnosis and remedies in offset-free MPC: theory and applications. *Asian J. of Contr.*, In press.

Pannocchia, G. and Rawlings, J.B. (2003). Disturbance models for offset-free model predictive control. *AIChE J.*, 49, 426–437.

Qin, S.J. and Badgwell, T.A. (2003). A survey of industrial model predictive control technology. *Contr. Eng. Pract.*, 11, 733–764.

Rajamani, M.R., Rawlings, J.B., and Qin, S.J. (2009). Achieving state estimation equivalence for misassigned disturbances in offset-free model predictive control. *AIChE J.*, 55, 396–407.

Rawlings, J.B. and Mayne, D.Q. (2009). *Model Predictive Control: Theory and Design*. Nob Hill Publishing, Madison, WI.

Shardt, Y., Zhao, Y., Qi, F., Lee, K., Yu, X., Huang, B., and Shah, S. (2012). Determining the state of a process control system: Current trends and future challenges. *Canad. J. Chem. Engin.*, 90, 217–245.

Sun, Z., Qin, S.J., Singhal, A., and Megan, L. (2013). Performance monitoring of model-predictive controllers via model residual assessment. *J. Proc. Contr.*, 23, 473–482.

Zhao, Y., Chu, J., Su, H., and Huang, B. (2010). Multi-step prediction error approach for controller performance monitoring. *Contr. Eng. Pract.*, 18, 1–12.

Zhu, Y. (2001). *Multivariable System Identification for Process Control*. Elsevier Science Ltd., Kidlington, UK.