Use of Transient Measurements for Static Real-Time Optimization

Tafarel de Avila Ferreira * **Grégory François** **
Alejandro G. Marchetti * **Dominique Bonvin** *

* *Laboratoire d'Automatique, École Polytechnique Fédérale de Lausanne, CH-1015 Lausanne, Switzerland (tafarel.deavilaferreira@epfl.ch, alejandro.marchetti@epfl.ch, dominique.bonvin@epfl.ch).*
** *Institute for Materials and Processes, School of Engineering, The University of Edinburgh, Edinburgh EH93FB (gregory.francois@ed.ac.uk).*

Abstract: Modifier adaptation (MA) is a real-time optimization (RTO) method characterized by its ability to enforce plant optimality upon convergence despite the presence of model uncertainty. The approach is based on correcting the available model using gradient estimates computed at each iteration. MA uses steady-state measurements and solves a static optimization problem. Hence, after every input change, one typically waits for the plant to reach steady state before measurements are taken. With many iterations, this can make convergence to the plant optimum rather slow. Recently, an approach that uses *transient measurements* for steady-state MA has been proposed. This way, plant optimality can be reached in a single transient operation. This paper proposes to improve this approach by using a *dynamic model* to process transient measurements for gradient computations. The approach is illustrated through the simulated example of a CSTR. Furthermore, the proposed method is less dependent on the choice of the RTO period. The time needed to reach plant optimality is of the order of the plant settling time, whereas several transitions to steady state would have been necessary using the standard static MA scheme.

Keywords: Real-time optimization, Modifier adaptation, Plant-model mismatch, Gradient estimation, Transient measurements.

1. INTRODUCTION

Process optimization is very valuable for improving process performance, while meeting productivity, quality, safety and environmental objectives. Model-based methods rely extensively on the quality of the model at hand and thus suffer in the presence of plant-model mismatch. Furthermore, even with an accurate plant model, the presence of slow disturbances generally leads to a drift of the optimal operating conditions, and measurement-based adaptation is needed to maintain plant optimality. This constitutes the field of real-time optimization (RTO).
Among the various RTO methods, the class labeled modifier adaptation (MA) uses measurements to appropriately modify the cost and constraint functions of the optimization problem. These schemes, whose idea was put forward in the late 70s (Roberts, 1979; Roberts and Williams, 1981), have received renewed attention in the last decade (Brdyś and Tatjewski, 2005; Gao and Engell, 2005; Marchetti et al., 2009). MA is a model-based RTO method that repeatedly solves a modified model-based optimization problem. In contrast to the two-step approach that uses plant measurements to update the model parameters and the updated model to perform optimization (Marlin and Hrymak, 1997), MA constructs and updates correction terms that are added to the cost and constraint functions of the optimization problem. The main advantage of MA lies in its proven ability to converge to the plant optimum, even in the presence of structural plant-model mismatch, a case where the two-step approach will generally fail (Forbes and Marlin, 1996). Furthermore, MA does not need to assume that the set of active constraints is known, which is often the case for implicit RTO schemes that reformulate the optimization problem as a control problem (Skogestad, 2000; François et al., 2005). However, MA may require several iterations to steady state before reaching plant optimality. It is therefore of great interest to combine the ability of MA to converge to plant optimality without prior knowledge of the set of active constraints with the ability of implicit schemes to converge to the plant optimum in a single iteration to steady state.
A first step in that direction was made recently by François and Bonvin (2013), who proposed to implement *steady-state* MA in the *transient phase*, thereby attempting to reach optimality in a single transient operation to steady state. For this, the steady-state optimization problem is solved repeatedly online, with the steady-state modifiers being estimated from transient measurements. The latter are used as if they were steady-state measurements. The approach is consistent since transient measurements tend to steady-state measurements when the plant reaches steady state. The gradients are estimated from input and

output measurements using neighboring extremals under the assumption of parametric uncertainty (Gros et al., 2009). The nice MA property of reaching a KKT point of the plant upon convergence is preserved, provided the gradient estimates converge to their true values. However, the main disadvantage of this approach is that, during transient operation, the steady-state gradient estimates might be rather inaccurate, which might lead to oscillatory behavior and even prevent convergence.

This paper proposes to use *dynamic models* to process transient measurements and predict the corresponding steady-state constraint and output values. The contribution of this paper includes: (i) a simply way of using transient measurements to estimate the steady-state values of plant constraints and gradients, and (ii) an illustration that the use of dynamic models helps reach plant optimality.

The paper is organized as follows. The optimization problem and the static MA formulation are presented in Section 2. Section 3 describes the use of transient measurements to estimate steady-state values with the use of dynamic models. The implementation aspects are illustrated via a simulated CSTR in Section 4, while Section 5 concludes the paper.

2. RTO VIA MODIFIER ADAPTATION

2.1 Problem Formulation

The problem of optimizing the steady-state plant performance in the presence of constraints can be formulated mathematically as a nonlinear program (NLP):

$$\begin{aligned} \mathbf{u}_p^\star := \arg\min_{\mathbf{u}} \; & \phi_p\left(\mathbf{u}\right) \\ \text{s.t.} \quad & \mathbf{G}_p\left(\mathbf{u}\right) \leq \mathbf{0}\,, \end{aligned} \tag{2.1}$$

where $\mathbf{u}$ is the n_u-dimensional vector of inputs, $\mathbf{G}_p$ is the n_G-dimensional vector of plant constraints and ϕ_p is the scalar cost function at steady state. Here, the subscript $(\cdot)_p$ indicates a quantity related to the plant.

The necessary conditions of optimality (NCO) for the plant are:

$$\begin{aligned} \mathbf{G}_p\left(\mathbf{u}_p^\star\right) \leq \mathbf{0}, \quad \boldsymbol{\nu}_p^\star \geq \mathbf{0}, \quad {\boldsymbol{\nu}_p^\star}^T \mathbf{G}_p\left(\mathbf{u}_p^\star\right) = 0 \\ \nabla_{\mathbf{u}}\phi_p\left(\mathbf{u}_p^\star\right) + {\boldsymbol{\nu}_p^\star}^T \nabla_{\mathbf{u}}\mathbf{G}_p\left(\mathbf{u}_p^\star\right) = \mathbf{0}, \end{aligned} \tag{2.2}$$

where $\boldsymbol{\nu}_p$ is the n_G-dimensional vector of Lagrange multipliers. In practice, the functions ϕ_p and $\mathbf{G}_p$ are unknown. A steady-state model of the plant is used to construct the following model-based NLP:

$$\begin{aligned} \mathbf{u}^\star \; = \; & \arg\min_{\mathbf{u}} \; \phi\left(\mathbf{u},\, \boldsymbol{\theta}\right) \\ & \text{s.t.}\, \mathbf{G}\left(\mathbf{u},\, \boldsymbol{\theta}\right) \leq \mathbf{0}, \end{aligned} \tag{2.3}$$

where ϕ and $\mathbf{G}$ represent the models of the cost and constraint functions, respectively, and $\boldsymbol{\theta}$ the q-dimensional vector of uncertain model parameters. We will assume that ϕ and $\mathbf{G}$ are twice continuously differentiable.

2.2 Modifier Adaptation

In the presence of plant-model mismatch, the model optimum $\mathbf{u}^\star$ does not in general match the plant optimum $\mathbf{u}_p^\star$. MA uses plant measurements to iteratively modify the model-based optimization problem (2.3) in such a way that, upon convergence, the NCO of the *modified* problem match those of the plant. At the k^{th} iteration, the optimal inputs computed using the modified problem are applied to the plant, and the resulting values of the plant constraints and of the gradients of the plant cost and constraints at steady state are compared to the model predictions. Then, the following optimization problem is solved to determine the next inputs:

$$\mathbf{u}_{k+1}^\star \; := \; \arg\min_{\mathbf{u}} \; \phi_{m,k}(\mathbf{u},\, \boldsymbol{\theta}) \tag{2.4}$$

$$\text{s.t.} \;\; \mathbf{G}_{m,k}(\mathbf{u},\, \boldsymbol{\theta}) \leq \mathbf{0} \tag{2.5}$$

$$\text{with:} \;\; \phi_{m,k}(\mathbf{u},\, \boldsymbol{\theta}) := \phi(\mathbf{u},\, \boldsymbol{\theta}) + {\boldsymbol{\Lambda}_k^\phi}^T \mathbf{u} \tag{2.6}$$

$$\mathbf{G}_{m,k}(\mathbf{u},\, \boldsymbol{\theta}) := \mathbf{G}(\mathbf{u},\, \boldsymbol{\theta}) + \boldsymbol{\epsilon}_k^G + {\boldsymbol{\Lambda}_k^G}^T(\mathbf{u} - \mathbf{u}_k^\star) \tag{2.7}$$

$$\boldsymbol{\epsilon}_k^G := \mathbf{G}_p(\mathbf{u}_k^\star) - \mathbf{G}(\mathbf{u}_k^\star,\, \boldsymbol{\theta}) \tag{2.8}$$

$${\boldsymbol{\Lambda}_k^\phi}^T := \nabla_{\mathbf{u}}\phi_p\left(\mathbf{u}_k^\star\right) - \nabla_{\mathbf{u}}\phi\left(\mathbf{u}_k^\star,\, \boldsymbol{\theta}\right) \tag{2.9}$$

$${\boldsymbol{\Lambda}_k^G}^T := \nabla_{\mathbf{u}}\mathbf{G}_p\left(\mathbf{u}_k^\star\right) - \nabla_{\mathbf{u}}\mathbf{G}\left(\mathbf{u}_k^\star,\, \boldsymbol{\theta}\right), \tag{2.10}$$

where the n_G-dimensional vector $\boldsymbol{\epsilon}_k^G$ are the zeroth-order modifiers, and the n_u-dimensional vector $\boldsymbol{\Lambda}_k^\phi$ and the $(n_u \times n_G)$ matrix $\boldsymbol{\Lambda}_k^G$ represent the first-order modifiers.

Between the iterations k and $k+1$, the constant input values $\mathbf{u}_{k+1}^\star$ are applied until the plant reaches steady state, at which time new modifier terms are computed for the next RTO iteration. In practice, MA is implemented with exponential filtering of the modifiers, which (i) prevents too large modifications of the optimization problem between consecutive iterations, (ii) reduces the impact of measurement noise, and (iii) provides additional degrees of freedom (the gains of the exponential filter) for enforcing convergence (Marchetti et al., 2009). For the sake of simplicity, we will implicitly assume the presence of this filter.

2.3 Gradient Estimation via Neighboring Extremals

Computing the values of $\boldsymbol{\epsilon}_k^G$ in Eq. (2.8) is relatively straightforward since the plant constraints are generally monitored. On the other hand, computing the plant gradients is much more demanding. Several gradient estimation methods used in the context of RTO are discussed in François et al. (2012). Here, we use the neighboring-extremal (NE) method that relies on input and output measurements (Gros et al., 2009). For this, we introduce the output equations

$$\mathbf{y} = \mathbf{H}\left(\mathbf{u},\, \boldsymbol{\theta}\right), \tag{2.11}$$

where $\mathbf{y}$ is the p-dimensional vector of outputs at steady state.

For $p \geq q$, a variational analysis of the cost (Gros et al., 2009) and of the constraints (François and Bonvin, 2013) gives:

$$\begin{aligned} \nabla_{\mathbf{u}}\phi\left(\mathbf{u},\, \boldsymbol{\theta}\right) = \nabla_{\mathbf{u}}\phi\left(\mathbf{u}^\star,\, \boldsymbol{\theta}\right) + \nabla^2_{\mathbf{u}\boldsymbol{\theta}}\phi\left(\nabla_{\boldsymbol{\theta}}\mathbf{H}\right)^+ \delta\mathbf{y} \\ + \left(\nabla^2_{\mathbf{u}\mathbf{u}}\phi - \nabla^2_{\mathbf{u}\boldsymbol{\theta}}\phi\left(\nabla_{\boldsymbol{\theta}}\mathbf{H}\right)^+ \nabla_{\mathbf{u}}\mathbf{H}\right)\delta\mathbf{u} \end{aligned} \tag{2.12}$$

$$\begin{aligned} \nabla_{\mathbf{u}}\mathbf{G}\left(\mathbf{u},\, \boldsymbol{\theta}\right) = \nabla_{\mathbf{u}}\mathbf{G}\left(\mathbf{u}^\star,\, \boldsymbol{\theta}\right) + \nabla^2_{\mathbf{u}\boldsymbol{\theta}}\mathbf{G}\left(\nabla_{\boldsymbol{\theta}}\mathbf{H}\right)^+ \delta\mathbf{y} \\ + \left(\nabla^2_{\mathbf{u}\mathbf{u}}\mathbf{G} - \nabla^2_{\mathbf{u}\boldsymbol{\theta}}\mathbf{G}\left(\nabla_{\boldsymbol{\theta}}\mathbf{H}\right)^+ \nabla_{\mathbf{u}}\mathbf{H}\right)\delta\mathbf{u}. \end{aligned} \tag{2.13}$$

where the deviation variables are $\delta\mathbf{y} := \mathbf{y} - \mathbf{y}_0^\star$ and $\delta\mathbf{u} := \mathbf{u} - \mathbf{u}_0^\star$, where $\mathbf{y}_0^\star = \mathbf{H}\left(\mathbf{u}_\mathbf{0}^\star, \boldsymbol{\theta}\right)$.
François and Bonvin (2013) proposed to replace the steady-state model predictions by steady-state output measurements in the estimation of steady-state gradients:

$$\widehat{\nabla_\mathbf{u}\phi_p}\left(\mathbf{u}\right) = \nabla_\mathbf{u}\phi\left(\mathbf{u}^\star, \boldsymbol{\theta}\right) + \nabla^2_{\mathbf{u}\boldsymbol{\theta}}\phi\left(\nabla_{\boldsymbol{\theta}}\mathbf{H}\right)^+ \delta\mathbf{y_p} + \left(\nabla^2_{\mathbf{uu}}\phi - \nabla^2_{\mathbf{u}\boldsymbol{\theta}}\phi\left(\nabla_{\boldsymbol{\theta}}\mathbf{H}\right)^+\nabla_\mathbf{u}\mathbf{H}\right)\delta\mathbf{u} \quad (2.14)$$

$$\widehat{\nabla_\mathbf{u}\mathbf{G}_p}\left(\mathbf{u}\right) = \nabla_\mathbf{u}\mathbf{G}\left(\mathbf{u}^\star, \boldsymbol{\theta}\right) + \nabla^2_{\mathbf{u}\boldsymbol{\theta}}\mathbf{G}\left(\nabla_{\boldsymbol{\theta}}\mathbf{H}\right)^+ \delta\mathbf{y_p} + \left(\nabla^2_{\mathbf{uu}}\mathbf{G} - \nabla^2_{\mathbf{u}\boldsymbol{\theta}}\mathbf{G}\left(\nabla_{\boldsymbol{\theta}}\mathbf{H}\right)^+\nabla_\mathbf{u}\mathbf{H}\right)\delta\mathbf{u}, \quad (2.15)$$

with $\delta\mathbf{y}_p := \mathbf{y}_p - \mathbf{y}_0^\star$, and where the notation $\widehat{(.)}$ indicates an estimated *steady-state* value.

3. MODIFIER ADAPTATION USING TRANSIENT MEASUREMENTS

François and Bonvin (2013) proposed to apply MA during transient operation. At time t_j, the optimal inputs $\mathbf{u}_{j+1}^\star$ are computed and applied to the plant until the next RTO execution at time $t_{j+1} = t_j + \tau_{RTO}$, where τ_{RTO} is the RTO period. The optimization problem for computing $\mathbf{u}_{j+1}^\star$ reads:

$$\mathbf{u}_{j+1}^\star := \arg\min_\mathbf{u} \phi_{m,j}(\mathbf{u}, \boldsymbol{\theta}) \quad (3.1)$$

$$\text{s.t. } \mathbf{G}_{m,j}(\mathbf{u}, \boldsymbol{\theta}) \leq \mathbf{0} \quad (3.2)$$

$$\text{with: } \phi_{m,j}(\mathbf{u}, \boldsymbol{\theta}) := \phi(\mathbf{u}, \boldsymbol{\theta}) + \widehat{\boldsymbol{\Lambda}}_j^{\phi^T}\mathbf{u} \quad (3.3)$$

$$\mathbf{G}_{m,j}(\mathbf{u}, \boldsymbol{\theta}) := \mathbf{G}(\mathbf{u}, \boldsymbol{\theta}) + \widehat{\boldsymbol{\epsilon}}_j^G + \widehat{\boldsymbol{\Lambda}}_j^{G^T}\left(\mathbf{u} - \mathbf{u}_j^\star\right) \quad (3.4)$$

$$\widehat{\boldsymbol{\epsilon}}_j^G := \widehat{\mathbf{G}}_p\left(\mathbf{u}_j^\star\right) - \mathbf{G}\left(\mathbf{u}_j^\star, \boldsymbol{\theta}\right) \quad (3.5)$$

$$\widehat{\boldsymbol{\Lambda}}_j^{\phi^T} := \widehat{\nabla_\mathbf{u}\phi_p}\left(\mathbf{u}_j^\star\right) - \nabla_\mathbf{u}\phi\left(\mathbf{u}_j^\star, \boldsymbol{\theta}\right) \quad (3.6)$$

$$\widehat{\boldsymbol{\Lambda}}_j^{G^T} := \widehat{\nabla_\mathbf{u}\mathbf{G}_p}\left(\mathbf{u}_j^\star\right) - \nabla_\mathbf{u}\mathbf{G}\left(\mathbf{u}_j^\star, \boldsymbol{\theta}\right). \quad (3.7)$$

Here, the modifiers have to be estimated as the *steady-state* values corresponding to the current inputs $\mathbf{u}_j^\star$. The conditions ensuring that the operating point reached upon convergence is optimal for the plant are given next.

Theorem 1. Consider the MA problem that uses transient measurements with the inputs computed iteratively as the solution to the optimization problem (3.1)-(3.7). If the plant reaches steady state and the estimates $\widehat{\mathbf{G}}_p\left(\mathbf{u}_j^\star\right)$, $\widehat{\nabla_\mathbf{u}\phi_p}\left(\mathbf{u}_j^\star\right)$ and $\widehat{\nabla_\mathbf{u}\mathbf{G}_p}\left(\mathbf{u}_j^\star\right)$ converge to their true values, then the plant will satisfy the NCO (2.2).

Proof 1
The proof can be found in François and Bonvin (2013).

It is assumed that the steady-state plant constraints $\widehat{\mathbf{G}}_p\left(\mathbf{u}_j^\star\right)$ can be estimated from online measurements. Furthermore, the gradients $\widehat{\nabla_\mathbf{u}\phi_p}\left(\mathbf{u}_j^\star\right)$ and $\widehat{\nabla_\mathbf{u}\mathbf{G}_p}\left(\mathbf{u}_j^\star\right)$ are inferred from the steady-state values $\delta\widehat{\mathbf{y}}_p(\mathbf{u}_j^\star) := \widehat{\mathbf{y}}_p\left(\mathbf{u}_j^\star\right) - \mathbf{y}_0^\star$ using Eqs. (2.14) and (2.15). We describe next two ways of estimating $\widehat{\mathbf{G}}_p\left(\mathbf{u}_j^\star\right)$ and $\widehat{\mathbf{y}}_p(\mathbf{u}_j^\star)$.

3.1 Method A: Steady-State Values from Transient Measurements

François and Bonvin (2013) proposed to simply use the transient measurements at time t_j as estimates of the steady-state values corresponding to the current inputs $\mathbf{u}_j^\star$, that is:

$$\widehat{\mathbf{G}}_p\left(\mathbf{u}_j^\star\right) = \mathbf{G}_p\left(t_j\right) \quad (3.8)$$

$$\widehat{\mathbf{y}}_p\left(\mathbf{u}_j^\star\right) = \mathbf{y}_p\left(t_j\right). \quad (3.9)$$

Remark 1. Note that, for $j \to \infty$, $\widehat{\mathbf{G}}_p(\mathbf{u}_\infty^\star) = \mathbf{G}_p(t_\infty)$ and $\widehat{\mathbf{y}}_p(\mathbf{u}_\infty^\star) = \mathbf{y}_p(t_\infty)$, which leads to $\widehat{\nabla_\mathbf{u}\phi_p}\left(\mathbf{u}_\infty^\star\right) = \nabla_\mathbf{u}\phi_p\left(\mathbf{u}_\infty^\star\right)$ and $\widehat{\nabla_\mathbf{u}\mathbf{G}_p}\left(\mathbf{u}_\infty^\star\right) = \nabla_\mathbf{u}\mathbf{G}_p\left(\mathbf{u}_\infty^\star\right)$ if the gradient expressions (2.14) -(2.15) are accurate. Hence, if the constraint and output measurements are accurate, $\widehat{\mathbf{G}}_p\left(\mathbf{u}_j^\star\right)$, $\widehat{\nabla_\mathbf{u}\phi_p}\left(\mathbf{u}_j^\star\right)$ and $\widehat{\nabla_\mathbf{u}\mathbf{G}_p}\left(\mathbf{u}_j^\star\right)$ will converge to their true values, thus meeting the assumptions in Theorem 1.

3.2 Method B: Steady-State Values from Transient Measurements and Dynamic Models

We assume the availability of dynamic models that are consistent with the steady-state models used for optimization, and we introduce the superscript $(.)^{dyn}$ to denote the variables resulting from integration of the dynamic model:

$$\dot{\mathbf{x}}^{dyn}(t) = \mathbf{F}\left(\mathbf{x}^{dyn}(t), \mathbf{u}(t), \boldsymbol{\theta}\right) \quad \mathbf{x}^{dyn}(0) = \mathbf{x}_0 \quad (3.10)$$

$$\mathbf{y}^{dyn}(t) = \mathbf{h}\left(\mathbf{x}^{dyn}(t), \mathbf{u}(t), \boldsymbol{\theta}\right) \quad (3.11)$$

$$\mathbf{G}^{dyn}(t) = \mathbf{g}\left(\mathbf{y}^{dyn}(t), \mathbf{u}(t), \boldsymbol{\theta}\right). \quad (3.12)$$

Consistency with the steady-state model means that, if $\mathbf{u}(t) \to \mathbf{u}_\infty$ when $t \to \infty$, then $\mathbf{G}^{dyn}(t) \to \mathbf{G}(\mathbf{u}_\infty, \boldsymbol{\theta})$ and $\mathbf{y}^{dyn}(t) \to \mathbf{y}_\infty = \mathbf{H}(\mathbf{u}_\infty, \boldsymbol{\theta})$, that is, the dynamic and steady-state models predict the same steady-state values of the constraints and outputs for any input value.

In this work, instead of taking the current measured values of the plant constraints and outputs as the steady-state estimates, as was done in Eqs (3.8)-(3.9), we propose to account for dynamic effects (at least those predicted by the model) by estimating the plant steady-state quantities as the steady-state values predicted by the model corrected with an additive bias term that is computed as the difference between the measured and predicted quantities at time t_j:

$$\widehat{\mathbf{G}}_p(\mathbf{u}_j^\star) := \mathbf{G}(\mathbf{u}_j^\star, \boldsymbol{\theta}) + \left(\mathbf{G}_p(t_j) - \mathbf{G}^{dyn}(t_j)\right), \quad (3.13)$$

$$\widehat{\mathbf{y}}_p(\mathbf{u}_j^\star) := \mathbf{y}(\mathbf{u}_j^\star, \boldsymbol{\theta}) + \left(\mathbf{y}_p(t_j) - \mathbf{y}^{dyn}(t_j)\right). \quad (3.14)$$

$\widehat{\mathbf{G}}_p(\mathbf{u}_j^\star)$ in Eq. (3.13) is then used in Eq. (3.5) to estimate the zeroth-order modifiers $\widehat{\boldsymbol{\epsilon}}_j^G$. Similarly, we propose to use $\widehat{\mathbf{y}}_p(\mathbf{u}_j^\star)$ in Eq. (3.14), rather than simply $\mathbf{y}_p(t_j)$, to estimate the steady-state plant gradients via Eqs. (2.14) and (2.15).

Remark 2. It is easily seen that the use of dynamic models still allows meeting the assumptions in Theorem 1, in particular $\widehat{\mathbf{G}}_p(\mathbf{u}_\infty^\star) = \mathbf{G}_p(t_\infty)$ and $\widehat{\mathbf{y}}_p(\mathbf{u}_\infty^\star) = \mathbf{y}_p(t_\infty)$. Indeed, from the assumption that the dynamic and the steady-state models are consistent, it follows that, upon convergence to $\mathbf{u}_\infty^\star$ when $j \to \infty$, $\mathbf{G}(t_\infty) = \mathbf{G}(\mathbf{u}_\infty^\star, \boldsymbol{\theta})$.

Using this last equality in Eq. (3.13) for $j \to \infty$ gives $\widehat{\mathbf{G}}_p(\mathbf{u}^\star_\infty) = \mathbf{G}_p(t_\infty)$, which leads to $\widehat{\mathbf{G}}_p(\mathbf{u}^\star_\infty) = \mathbf{G}_p(\mathbf{u}^\star_\infty)$. Similarly, $\widehat{\phi}_p(\mathbf{u}^\star_\infty) = \phi_p(\mathbf{u}^\star_\infty)$ and $\widehat{\mathbf{y}}_p(\mathbf{u}^\star_\infty) = \mathbf{y}_p(\mathbf{u}^\star_\infty)$, that is, the estimates (3.13)-(3.14) converge to their true values.

Remark 3. If the dynamic and the steady-state models are consistent, the contribution of Method B over Method A is a dynamic adjustment that accounts for the modeled dynamic effects, which vanish when steady state is reached upon convergence.

4. ILLUSTRATIVE CASE STUDY

4.1 Problem Formulation

We consider the well-known Van de Vusse reaction system (Van de Vusse, 1964). In a CSTR, cyclopentadiene (A) is converted into the product cyclopentanol (B). Dicyclopentadiene (D) is also produced from A and cyclopentadiol (C) is produced through the undesired conversion of the desired product B:

$$A \xrightarrow{k_1} B \xrightarrow{k_2} C$$
$$2A \xrightarrow{k_3} D.$$

The dynamic reactor model reads:

$$\dot{c}_A = -r_1 - r_3 + (c_{A,in} - c_A)d \tag{4.1}$$
$$\dot{c}_B = r_1 - r_2 - c_B d \tag{4.2}$$
$$\dot{c}_C = r_2 - c_C d \tag{4.3}$$
$$\dot{c}_D = \frac{1}{2} r_3 - c_D d \tag{4.4}$$
$$\dot{T} = (T_{in} - T)d + \frac{k_w A_R}{\rho c_p V_R}(T_J - T) - \frac{1}{\rho c_p}(r_1 \Delta H_1 + r_2 \Delta H_2 + r_3 \Delta H_3) \tag{4.5}$$
$$r_1 = k_1 c_A, \quad r_2 = k_2 c_B, \quad r_3 = k_3 {c_A}^2, \tag{4.6}$$

where c_X denotes the concentration of Species $X \in \{A, B, C, D\}$, d is the dilution rate, $c_{A,in}$ is the inlet concentration of A, k_w is the coefficient of heat transfer with the cooling jacket, A_R denotes the cooling-jacket surface, V_R is the reactor volume, c_p is the heat capacity, T_{in} is the inlet temperature of the feed A, T_J is the jacket temperature, T is the reactor temperature, and r_i is reaction rate of the ith reaction, $i = 1, 2, 3$, with the reaction enthalpy ΔH_i. The associated reaction rate constant k_i, pre-exponential factor k_{i_0} and normalized activation energy E_i follow the Arrhenius law:

$$k_i = k_{i_0} exp\left(-\frac{E_i}{T + 273.15}\right), \quad i = 1, 2, 3. \tag{4.7}$$

The measured output variables are $\mathbf{y} = [c_A \ c_B \ c_C \ c_D \ T]^T$. The objective is to maximize the production of B at steady state by manipulating the inputs d and T_J. The optimization problem, which considers bounds on d and T_J, is formulated mathematically as follows:

Table 1. Model and plant parameters (uncertain parameters in bold)

	Model	Plant	Unit
$c_{A,in}$	4.6×10^3	$\mathbf{5.1 \times 10^3}$	$mol.m^{-3}$
T_{in}	98	**104.9**	$°C$
k_{10}	1.544×10^{12}	$\mathbf{1.287 \times 10^{12}}$	h^{-1}
k_{20}	1.544×10^{12}	$\mathbf{1.287 \times 10^{12}}$	h^{-1}
k_{30}	7.234×10^6	$\mathbf{9.043 \times 10^6}$	$m^3.mol^{-1}.h^{-1}$
E_1	9.758×10^3	9.758×10^3	K
E_2	9.758×10^3	9.758×10^3	K
E_3	8.560×10^3	8.560×10^3	K
ΔH_1	4.2	4.2	$kJ.mol^{-1}$
ΔH_2	-11	-11	$kJ.mol^{-1}$
ΔH_3	-62.78	$\mathbf{-41.85}$	$kJ.mol^{-1}$
$\frac{k_w A_R}{\rho c_p V_R}$	30.828	30.828	h^{-1}
$\frac{1}{\rho c_p}$	3.522×10^{-4}	3.522×10^{-4}	$m^3.K.kJ^{-1}$

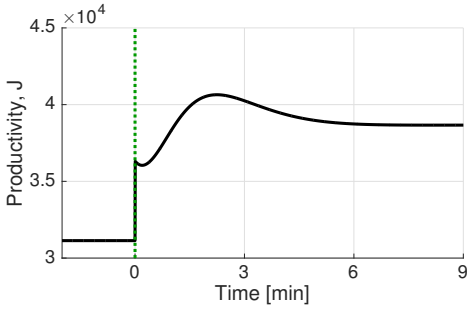

Figure 1. *Open-loop response of the plant productivity to steps in the inputs.*

$$\max_{\overline{d}, \overline{T}_J} \ J := \overline{c}_B \overline{d} \tag{4.8}$$
s.t. Eqs (4.1)-(4.7) at steady state
$$d_{min} \le \overline{d} \le d_{max} \tag{4.9}$$
$$T_{J,min} \le \overline{T}_J \le T_{J,max}, \tag{4.10}$$

where $\overline{(.)}$ denotes a steady-state value, $d_{min} = 3\,h^{-1}$, $d_{max} = 35\,h^{-1}$, $T_{J,min} = 50°C$ and $T_{J,max} = 200°C$.

Table 1 summarizes the model and plant parameters. Uncertainty affects the values of the inlet concentration of Species A, the inlet temperature, the kinetic parameters of the three reactions, and the enthalpy of the third reaction. The upper bound on the dilution rate is active at the plant optimum.

The input steps $d = 30 \to 35\ h^{-1}$ and $T_J = 120 \to 144°C$ are applied to the reactor to characterize the open-loop plant response (Figure 1). It can be seen that the plant takes about 8 min to reach steady state, and the non-minimum phase behaviour can be observed by the initial decrease in the productivity.

4.2 MA Implementation using Transient Measurements

MA is implemented using transient measurements and either Method A or Method B to estimate the steady-state quantities $\widehat{\mathbf{G}}_p(\mathbf{u}^\star_j)$ and $\widehat{\mathbf{y}}_p(\mathbf{u}^\star_j)$. The operation is started at the conservative operating point $\mathbf{u} = [30 \ 120]^T$. The objective is to steer the plant to its *unknown* optimum by applying MA using transient measurements.

For Method B, the first iteration uses the model steady-state values as initial states for simulating the dynamic

model. The subsequent iterations use as initial states the final values of the previous iteration.
We apply an exponential filter in the adaptation of the cost-gradient modifiers in order to avoid abrupt input changes,

$$\widehat{\mathbf{\Lambda}}_j^{\phi^T} = \left(1 - K_f\right) \widehat{\mathbf{\Lambda}}_{j-1}^{\phi^T} + K_f \left(\widehat{\nabla_{\mathbf{u}} \phi_p} \left(\mathbf{u}_j^\star\right) - \nabla_{\mathbf{u}} \phi \left(\mathbf{u}_j^\star, \boldsymbol{\theta}\right)\right). \quad (4.11)$$

Note that only the cost-gradient modifiers are filtered as there are no *uncertain* inequality constraints in this problem.
The following four scenarios are considered next:

- **Scenario 1**: $K_f = 0.5$ and $\tau_{RTO} = 60$ sec.
- **Scenario 2**: $K_f = 0.5$ and $\tau_{RTO} = 30$ sec.
- **Scenario 3**: $K_f = 0.75$ and $\tau_{RTO} = 20$ sec.
- **Scenario 4**: $K_f = 0.2$ and $\tau_{RTO} = 20$ sec.

The performances obtained with Methods A and B are compared for each of the four scenarios. Note that, in all cases, the dilution rate reaches its upper bound in the first RTO execution.

Scenario 1. Figure 2 shows that the performances of both methods are fairly similar, with a slight advantage to Method B. Note that both methods converge to the plant optimum despite significant parametric mismatch. The reactor reaches optimality in about 9 min, which is slightly more than the reactor settling time.

Scenario 2. Figure 3 shows that, with a more frequent repetition of the optimization, the degradation due to the uncompensated dynamic effects increases. The oscillations exhibited by Method A are due to poor approximations of the cost gradients in the initial part of the transient. With the use of dynamic models (Method B), the plant converges faster and without much oscillation.

Scenario 3. Figure 4 shows that reducing τ_{RTO} and the amount of filtering are quite detrimental to Method A. In contrast, Method B handles the situation very nicely.

Scenario 4. Figure 5 shows that the performance of Method A can be significantly improved with more filtering, even with small values of the RTO period. However, Method B still performs better than Method A.

Method B clearly outperforms Method A. While decreasing τ_{RTO} reduces the time to converge to the plant optimum with Method B, it increases oscillations with Method A. When the oscillations become too large, the jacket-temperature input saturates, which results in a limit cycle (Fig. 4).

Table 2 compares the convergence of all scenarios and methods. One sees that Method B converges faster than Method A. An interesting feature is the fact that Method B appears to be rather insensitive to the choices of K_f and τ_{RTO}, which is not the case of Method A. This very nice property can be related to the use of dynamic models.
Finally, note that, with Method B, there is no real need for decreasing τ_{RTO} any further, since the system has its own dynamics and cannot settle to steady state much faster. Here, the reactor does not simply settle to steady state, it settles to the *optimal* steady state!

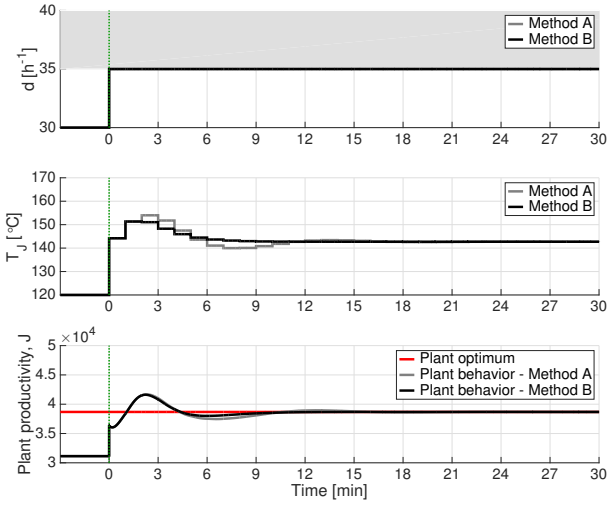

Figure 2. *Scenario 1: Dilution rate, jacket temperature, and plant productivity vs. time.*

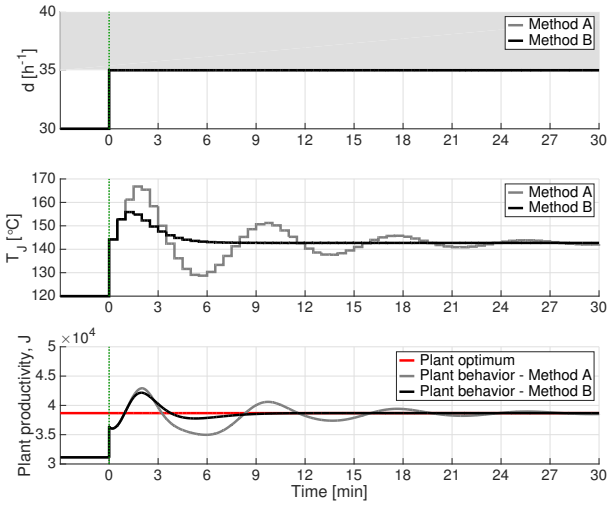

Figure 3. *Scenario 2: Dilution rate, jacket temperature, and plant productivity vs. time.*

Table 2. Convergence times for different scenarios and methods

Scenario	Method	K_f	τ_{RTO} [sec]	Convergence time [min]
1	A	0.5	60	12
	B	0.5	60	10
2	A	0.5	30	>30
	B	0.5	30	10
3	A	0.75	20	-
	B	0.75	20	9
4	A	0.2	20	15
	B	0.2	20	10

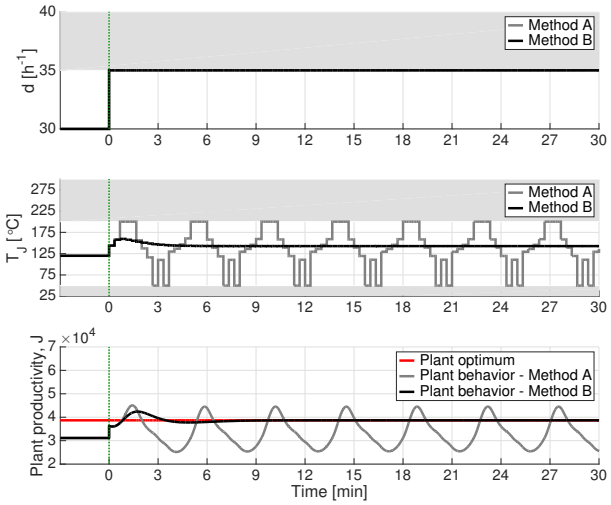

Figure 4. *Scenario 3: Dilution rate, jacket temperature, and plant productivity vs. time.*

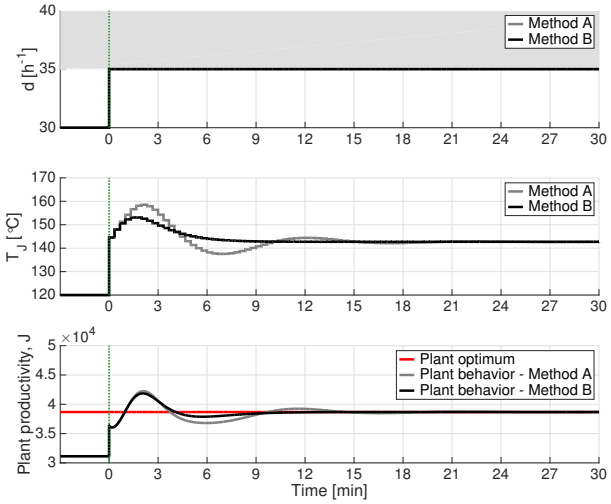

Figure 5. *Scenario 4: Dilution rate, jacket temperature, and plant productivity vs. time.*

5. CONCLUSIONS

This paper has proposed a significant improvement for a recent MA scheme that uses transient measurements for static optimization. The improvement consists in processing these transient measurements through dynamic models, thereby improving the estimation of plant constraints and gradients at steady state. This way, static MA can be implemented during transient operation, with the objective to reach the plant optimum in a single transition to steady state. In a way, this is similar to what is done with *implicit* RTO methods, such as extremum-seeking control (Ariyur and Krstic, 2003), self-optimizing control (Skogestad, 2000) and NCO-tracking (François et al., 2005). However, MA that uses transient information has the advantage of being an *explicit* RTO method, which repeatedly solves the optimization problem and is therefore able to deal with changing active constraints.

The approach has been illustrated by means of the real-time optimization of the Van de Vusse CSTR reactor. This reaction system is known to exhibit non-minimum phase behavior, which makes the use of the original MA scheme with transient measurements more difficult. Simulations have shown that the new approach makes it possible to converge to the plant optimum within the settling time of the reactor, while being much less sensitive than the previous approach to the choice of the RTO period and the filter gains applied to the gradient modifiers.

REFERENCES

Ariyur, K. and Krstic, M. (2003). *Real-Time Optimization by Extremum-Seeking Control*. John Wiley, New York.

Brdyś, M. and Tatjewski, P. (2005). *Iterative Algorithms for Multilayer Optimizing Control*. Imperial College Press, London UK.

Forbes, J.F. and Marlin, T. (1996). Design cost: A systematic approach totechnology selection for model-based real-time optimization systems. *Comp. Chem. Eng.*, 20, 717–734.

François, G. and Bonvin, D. (2013). Use of transient measurements for the optimization of steady-state performance via modifier adaptation. *Ind. Eng. Chem. Res.*, 53(13), 5148–5159.

François, G., Srinivasan, B., and Bonvin, D. (2005). Use of measurements for enforcing the necessary conditions of optimality in the presence of constraints and uncertainty. *J. Process Contr.*, 15(6), 701–712.

François, G., Srinivasan, B., and Bonvin, D. (2012). Comparison of six implicit real-time optimization schemes. *J. Européen des Systèmes Automatisés*, 46(2-3), 291–305.

Gao, W. and Engell, S. (2005). Iterative set-point optimization of batch chromatography. *Comp. and Chem. Engng.*, 29(6), 1401–1409.

Gros, S., Srinivasan, B., and Bonvin, D. (2009). Optimizing control based on output feedback. *Comput. Chem. Engng.*, 33(1), 191–198.

Marchetti, A., Chachuat, B., and Bonvin, D. (2009). Modifier-adaptation methodology for real-time optimization. *Ind. Eng. Chem. Res.*, 48, 6022–6033.

Marlin, T.E. and Hrymak, A.N. (1997). Real-time operations optimization of continuous processes. In *AIChE Symposium Series - CPC-V*, volume 93, 156–164.

Roberts, P.D. (1979). An algorithm for steady-state system optimization and parameter estimation. *Int. J. Systems Sci.*, 10, 719–734.

Roberts, P.D. and Williams, T.W.C. (1981). On an algorithm for combined system optimization and parameter estimation. *Automatica*, 17, 199–209.

Skogestad, S. (2000). Plantwide control: The search for the self-optimizing control structure. *J. Process Contr.*, 10, 487–507.

Van de Vusse, J.G. (1964). Plug-flow type reactor versus tank reactor. *Chemical Engineering Science*, 19, 994–996.