

Online model maintenance via output modifier adaptation

José Matias and Johannes Jäschke*

*Department of Chemical Engineering, Norwegian University of Science and Technology
(NTNU), Trondheim, Norway*

E-mail: johannes.jaschke@ntnu.no

Abstract

This paper describes a method for selecting and adapting the model structure online while running together with a real-time optimization algorithm using Output Modifier Adaptation. The method chooses, among several knowledge-based models, the model structure that is most consistent with the current process data. By allowing the model to change over time, its structure and complexity are able to adapt to the plant data. Competing models are compared based on the modifiers, which are also used by the Output Modifier Adaptation to drive the system to the optimal operating point. The approach is demonstrated on two case studies, a continuous stirred tank reactor and a gas lifted oil well network. In both cases, the best model structure is chosen among several candidates and the plant optimum is reached without constraint violations. The case studies indicate that, even with a significant amount of noise, the modifiers are good indicators for choosing among competing model structures.

Introduction

There are considerable challenges to be addressed when dealing with complex models with possible unknown interactions of its components. For example, in knowledge-driven models, physical-chemical relationships (like conservation of mass and reaction kinetics) are used for describing the inner workings of the process.¹ However, a variety of process models can be used for describing the same phenomenon and there is often not enough information to make proper choices during the model design phase.² Moreover, these choices should be made with a specific goal in mind, as each application requires models with specific characteristics.²

If the model is meant to be used in Real-time optimization (RTO) problems, for example, it should be able to predict the optimality conditions for the plant, namely the Karush-Kuhn-Tucker (KKT) conditions for a static optimization problem.³ Output Modifier Adaptation (MAy)³ is an RTO variant created to guarantee that the optimum computed by the RTO matches the actual plant optimum.

Taking advantage of the MAy scheme structure, this paper proposes a method that simultaneously adapts the model structure online and optimizes the process via Output Modifier Adaptation. Our method is developed to address the issue of dealing with unknown process features during the modeling phase but is also intended as a tool to automatically maintain the models during the process lifetime. Moreover, since the method is linked with MAy, it is especially suited for real-time optimization purposes.

The basic idea of the MAy scheme is to apply correction terms (modifiers) to the model in order to allow a KKT point for the model-based optimization problem to match the plant optimum upon convergence. At a given operating point u , the modifiers correct (and indicate) the differences between model predictions and plant measurements as well as the differences between the shape of the model and the shape of the plant, which is represented by the gradients.

Although the MAy is traditionally used for optimization, the modifiers contain valuable information about the relationship between a model and the plant. Therefore, their use

can be extended to aid the modeling process. They can be used for discriminating between different modeling or model updating decisions, which can be very helpful in cases that there are unknown features of the process that need to be modeled. For example, Hille and Budman⁴ indirectly use the modifiers for simultaneous model identification and optimization in cases where the model does not match the plant. Their method defines a model-update criterion online, which aims at estimating model parameters for satisfying the conditions of optimality while providing accurate predictions of the process outputs.

Motivated by the possibility of linking optimization with modeling using the modifiers, we propose to use the modifiers for discriminating between different knowledge-driven models. The basic concept of the proposed method is illustrated in Figure 1, and consists of four main steps: (1) First, the process model is divided into several blocks. Each block represents a given phenomenon or part of the process that needs to be described by a model. For example, a model block may be selected to be the pressure drop of flowing liquids in a pipe. (2) Then, several sub-model candidates are created for this block. They may be based on different assumptions that may be made when modeling the phenomenon. For example, different sub-model candidates for the pressure drop in a pipe may include (or exclude) effects of friction, hydrostatic pressure, turbulence, etc.. (3) As steady-state process data is obtained during the operation, the modifiers for all available sub-model combinations can be calculated. Using the modifiers, the effects of different sub-models in the complete process model can be compared. (4) By solving an optimization problem, the sub-models that best fit the process data are combined to generate the working model, which is used for optimizing the process. At each M_Ay iteration, steps (3) to (4) are repeated.

The proposed method is a step toward “fully automating” the modeling and model maintenance tasks. Instead of leaving the final choice of the model structure to the modeler, the method allows him/her to propose several models with different structures *a priori*. Then, the model structure is adjusted automatically based on the available operational data, which come in the form of the modifiers. Note that, since different sub-models can be used to rep-

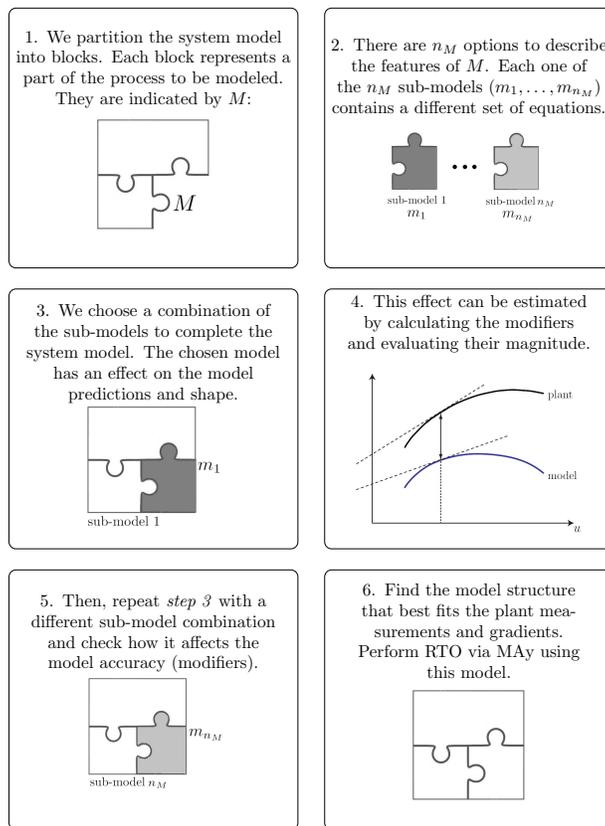


Figure 1: The basic concept of the structure updating algorithm. The differences between the plant and the original model are measured using the modifiers of the Output Modifier Adaptation (MAy) method.

resent the process, the complete model is able to describe a wider range of phenomena, mitigating the inherent uncertainties associated with the modeling step. Also, if none of the models is found to represent the process well, one may add new sub-models online during the operation.

The paper is organized as follows. First, we show how the modifiers can be used for choosing between different model structures and describe the proposed method. Then, the Model Structure Adaptation Method is introduced and assessed formally. The next two sections present the case studies, a continuous stirred-tank reactor and a gas lift oil well network. These sections also discuss the advantages and challenges of the proposed method. Finally, implementation issues are discussed and the paper is concluded.

Model structure adaptation

Main idea

MA methods incorporate plant information to the optimization problem through the modifiers. The classical MA method applies zeroth and first order modifiers (input-affine) for cost and constraint functions of the optimization problem in order to match the measurements and gradients of the plant.⁵ These modifiers are updated iteratively as new plant measurements become available. The modifiers enforce the optimum computed by the model-based optimization to match the plant optimum upon convergence.³ Marchetti et al.³ presented a second variant of the modifier adaptation method that consists of making modifications in the process model rather than in the optimization problem, referred as Output Modifier Adaptation. The adapted model is then used in the optimization problem. This second variant also enforces that the plant optimum is matched upon convergence. The modified model is defined as:

$$\mathbf{y}_{ad,k}(\mathbf{u}) := \mathbf{y}(\mathbf{u}) + \boldsymbol{\epsilon}_k + (\boldsymbol{\lambda}_k)^T(\mathbf{u} - \mathbf{u}_k) \quad (1)$$

where, $\mathbf{y} \in \mathcal{R}^{n_y}$ represents the original model and $\mathbf{y}_{ad}(\mathbf{u}) \in \mathcal{R}^{n_y}$ the modified (or adapted) model; $\mathbf{u} \in \mathcal{R}^{n_u}$ are the model inputs; and k is the subscript representing the k^{th} MAy iteration. Thus, \mathbf{u}_k are the model inputs implemented in the k^{th} MAy iteration. The zeroth-order modifiers, $\boldsymbol{\epsilon}_k \in \mathcal{R}^{n_y}$, correspond to bias terms between the predicted values of the model and the plant measurements, and the first order modifiers, $\boldsymbol{\lambda}_k \in \mathcal{R}^{n_u \times n_y}$, represent the difference between the plant gradients and the gradients predicted by the model at each iteration k . The modifiers are calculated by:

$$\boldsymbol{\epsilon}_k = \mathbf{y}_p(\mathbf{u}_k) - \mathbf{y}(\mathbf{u}_k), \quad \boldsymbol{\lambda}_k = \left(\frac{\partial \mathbf{y}_p}{\partial \mathbf{u}}(\mathbf{u}_k) - \frac{\partial \mathbf{y}}{\partial \mathbf{u}}(\mathbf{u}_k) \right)^T \quad (2)$$

where, \mathbf{y}_p are the plant measurements and $\frac{\partial \mathbf{y}_p}{\partial \mathbf{u}}(\mathbf{u}_k)$ the plant gradients, which need to be estimated.

Traditionally, the model modifiers are only used for modifying the real-time optimization problem, such that the model-based optimization problem converges to the plant optimum despite plant-model mismatch. However, in this work we propose to use them systematically as model quality indicators. We introduce the total model modifier as:

$$\psi := \|\boldsymbol{\epsilon}_k\|_F + \|\boldsymbol{\lambda}_k\|_F \quad (3)$$

in which, $\|\cdot\|_F$ indicates the Frobenius norm, which is a non-square matrix norm defined as the square root of the sum of the absolute squares of the matrix elements.⁶ The total model modifier quantifies how much a model was adapted. For example: if the zeroth order modifiers are close to zero but the first order modifiers are not, it is possible to infer that the model predicts the local values accurately but the model shape does not correspond to the plant in this specific region. Hence, when moving away from the current operating point, the model prediction will differ from the plant. Thus, any perturbation that drives the process away from the current operation point affects the model prediction capacity.

Based on the idea of the total model modifier, this paper proposes a new method for adapting the model structure online. The method finds the best model, among a pool of available models, that most accurately fits plant measurements and gradients while optimizing the system. In particular, among all possible model combinations, our approach is to select the model that requires least modification in order to match the plant optimum. That is, we select the best model that minimizes the modifiers. The new method basic steps are summarized in Algorithm 1.

Algorithm 1 Model Structure Adaptation method

- 1: Estimate plant gradients;
 - 2: Select a combination of models such that the total model modifier ψ is minimized;
 - 3: Use new model and its modifiers to compute optimal operating point via RTO using Output Modifier Adaptation, and implement the new operating;
 - 4: Repeat from 1.
-

Rather than determining a fixed model structure, the method allows the model to evolve

with the data obtained. By quantitatively defining which model structure represent the operation plant data poorly, the method provides valuable insights about the inner workings of the process. Such knowledge can be important for further model developments and model maintenance. For example, if the set of reactions chosen for the model is unable to represent the formation of one sub-product, new mechanisms can be suggested and included as new sub-models.

Relationship to other work

Our proposed method is similar in spirit with the simultaneous identification and optimization method proposed by Martinez Villegas et al.⁷ and Hille and Budman⁴. In their work, the authors also study on the synergy between identification and optimization via Modifier Adaptation. Despite the fact that the model adaptation step is not explicitly related to the first order modifiers, their parameter estimation objective function minimizes the error between the gradients of the model-based optimization problem and the plant, for both the cost and constraint. Note that, the objective of the method proposed in our paper and the one proposed by Martinez Villegas et al.⁷ and Hille and Budman⁴ are related, as they both determine the best model based on minimizing the value of the modifiers. However, in the latter, only the parameter values change, not the model structure.

In some sense, our proposed method may also be related to robust optimization schemes, like scenario-based optimization.⁸⁻¹⁰ Scenario-based approaches account for model uncertainty by proposing different scenarios, which can be thought of being represented by different models (multi-model RTO). The RTO problem is solved to satisfy all the scenarios (model combinations). Despite being robust, the computed solution can be conservative. While the optimum found by the scenario-based RTO will be feasible for all models, our approach selects the scenario, in this case model structure, that best fits the data, and uses it for optimizing the process. This provides information that is useful to model and understand the process, and to maintain the model over the plant lifetime.

Detailed description of the method

The initial step of our method consists of partitioning the complete system model in blocks, as previously shown in Figure 1, where each block encompasses a given phenomena happening in the process. Then, several sub-models are proposed for each one of these blocks. Note that the method also works if the entire model of interest is unknown. In this case, several sub-models can be proposed for all the phenomena happening in the process and their interconnections are determined by the method. Next, the total model modifier (Equation (3)) is used to indicate which combination of sub-models best represents the plant behavior. Hence, the available process knowledge is used for selecting and building the sub-models while the missing information, which is the interconnections of the sub-models, is determined by process data.

However, as the size of the model increases, the number of blocks, their sub-models and the possible combination among them increase exponentially. Thus, an exhaustive exploration of the sub-models arrangements as proposed in Algorithm 1 can easily become intractable. In this paper, we propose to formulate the model selection problem as an optimization problem using a generalized disjunctive programming approach¹¹ with Boolean variables to indicate the sub-model combinations. Our formulation is based on the systematic framework of superstructure optimization proposed by Yeomans and Grossmann¹².

Let n_b different blocks M compose the complete model, which is represented by $Y := M_1 \cup M_2 \cup \dots \cup M_{n_b}$. Each one of the blocks can be described by several sub-models, for example the b^{th} block is characterized by $M_b := \{m_{b,1} \vee m_{b,2} \vee \dots \vee m_{b,n_{M_b}}\}$. This “set” represents the n_{M_b} different sub-models (m_b) that can be assigned to block M_b . Clearly, only one of the available sub-models can be selected for a given block at the k^{th} MAy iteration.

The model selection algorithm is then written as the following optimization problem:

$$\begin{aligned}
\mathbf{z}_{k+1}^* &= \arg \min_{\mathbf{z}} \psi := \|\boldsymbol{\epsilon}_k\|_F + \|\boldsymbol{\lambda}_k\|_F \\
\text{s.t. } \boldsymbol{\epsilon}_k &= \mathbf{y}_p(\mathbf{u}_k) - \mathbf{y}(\mathbf{u}_k, M_1, \dots, M_{n_b}), \\
(\boldsymbol{\lambda}_k)^T &= \left(\frac{\partial \mathbf{y}_p}{\partial \mathbf{u}}(\mathbf{u}_k) - \frac{\partial \mathbf{y}}{\partial \mathbf{u}}(\mathbf{u}_k, M_1, \dots, M_{n_b}) \right) \\
&\left[\bigvee_{j=1, \dots, n_{M_1}} \left(\begin{array}{c} \text{if } z_{1,j} = 1 \\ M_1 := m_{1,j}(\mathbf{u}_k) \end{array} \right) \right] \wedge \dots \wedge \left[\bigvee_{j=1, \dots, n_{M_{n_b}}} \left(\begin{array}{c} \text{if } z_{n_b,j} = 1 \\ M_{n_b} := m_{n_b,j}(\mathbf{u}_k) \end{array} \right) \right] \\
&\sum_{j=1}^{n_{M_b}} z_{b,j} = 1 \quad \text{where } b = 1, \dots, n_b \quad \text{and } z_{b,j} \in \{0, 1\}
\end{aligned} \tag{4}$$

All the variables in Equation (4) have been already defined except for z_{bj} and \mathbf{z} . The variables z_{bj} are Boolean variables, which indicate that the j^{th} sub-model of the set M_b is assigned (or not) to the model, and \mathbf{z} is the vector of all z_{bj} . Once the optimization is formulated as above, it can be solved with any mixed-integer nonlinear programming solver.

Remark. In Equation (4), the modifiers are updated using only local information at u_k . Such strategy may be sensitive to noise, affecting the model selection algorithm convergence. In order to include past information in the updating strategy, one may compute the modifiers using a first-order filter:⁵

$$\begin{aligned}
\boldsymbol{\epsilon}_k &= (I - K_\epsilon)\boldsymbol{\epsilon}_{k-1} + K_\epsilon(\mathbf{y}_p(\mathbf{u}_k) - \mathbf{y}(\mathbf{u}_k, M_1, \dots, M_{n_b})) \\
\boldsymbol{\lambda}_k &= (I - K_\lambda)\boldsymbol{\lambda}_{k-1} + K_\lambda \left(\frac{\partial \mathbf{y}_p}{\partial \mathbf{u}}(\mathbf{u}_k) - \frac{\partial \mathbf{y}}{\partial \mathbf{u}}(\mathbf{u}_k, M_1, \dots, M_{n_b}) \right)^T
\end{aligned} \tag{5}$$

in which, K_ϵ and K_λ are the filter gain matrices, which are square matrices with values between $[0, 1)$ on the main diagonal and zeros elsewhere, and I is the identity matrix of appropriate dimensions.

Output modifier adaptation

Before introducing the flowsheet of the complete algorithm for simultaneous model structure adaptation and economic optimization, the Output Modifier Adaptation method is briefly presented. Like in standard RTO applications, MAy aims at minimizing an economic-like cost function ϕ while respecting the operation constraints G . As mentioned earlier, the main difference between them lies in the fact that, in MAy, the model is corrected using modifiers instead of re-estimating the model parameters at each optimization iteration. The main consequence is that the modified model meets the optimality conditions at the plant optimum. The RTO problem to find the optimal plant inputs via MAy is:

$$\begin{aligned} \mathbf{u}_{k+1}^* &= \arg \min_{\mathbf{u}} \phi(\mathbf{u}, \mathbf{y}_{ad,k}(\mathbf{u})) \\ \text{s.t. } G &:= \{g_i(\mathbf{u}, \mathbf{y}_{ad,k}(\mathbf{u})) \leq 0, \quad i = 1, \dots, n_g\} \end{aligned} \quad (6)$$

where, $\phi : \mathcal{R}^{n_u} \times \mathcal{R}^{n_y} \mapsto \mathcal{R}$ is the cost function that normally represents some economic criterion; $g_i : \mathcal{R}^{n_u} \times \mathcal{R}^{n_y} \mapsto \mathcal{R}$, $i = 1, \dots, n_g$ are the problem constraints. G is the set of all constraints, which includes lower and upper bounds for measurements as well as operation and safety constraints. The adapted model $\mathbf{y}_{ad,k}(\mathbf{u})$ is computed according to Equation (1) and the modifiers are calculated using either local information (Equation (2)) or a first order filter (Equation (5)).

Regardless of the chosen strategy, the plant measurements \mathbf{y}_p are obtained directly, and the plant gradients $\frac{\partial \mathbf{y}_p}{\partial \mathbf{u}}(\mathbf{u}_k)$ are estimated. However, obtaining reliable gradient estimates with noisy measurements of \mathbf{y}_p can be a challenging task. This is a topic of ongoing research, and several methods for estimating plant gradients are available in the literature, but there is no consensus on the best method for MA applications.⁵ For a more detailed review of gradient methods for MA, please refer to Marchetti et al.⁵

After obtaining the appropriate adapted model, the modified economic problem is solved and \mathbf{u}_{k+1}^* computed. Next, an input filter, $\mathbf{u}_{k+1} = \mathbf{u}_k + K_u(\mathbf{u}_{k+1}^* - \mathbf{u}_k)$, is also used to

mitigate the effect of noise and errors in the gradient predictions. Small values of the filter gain represent a more conservative updating strategy. Finally, \mathbf{u}_{k+1} is implemented in the plant and the MAY iteration is finished.

Summary of the model adaptation method using MAY

Given Equations (4) and (6), the complete algorithm for simultaneous model structure adaptation and economic optimization can be outlined. It is presented in Figure 2.

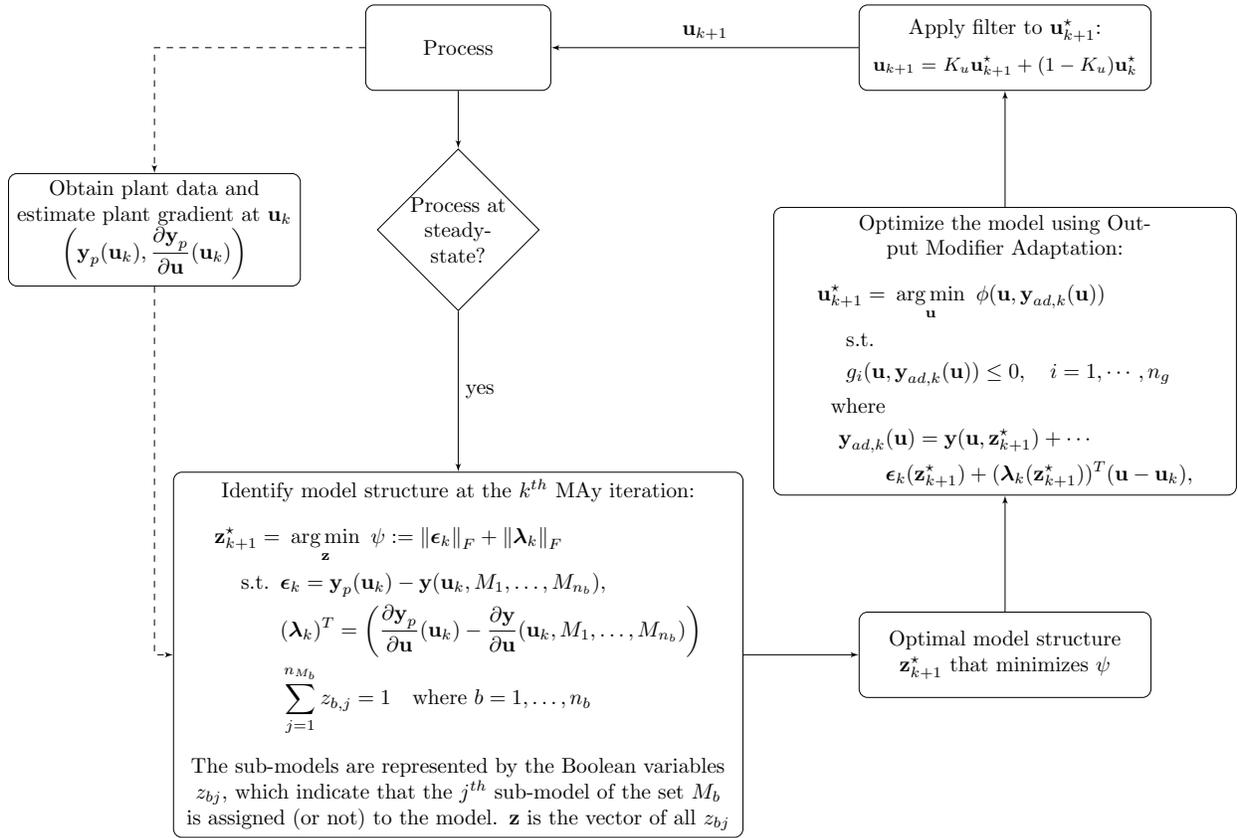


Figure 2: Flowsheet of complete algorithm for simultaneous model structure adaptation and economic optimization. The model structure adaptation is nested in the optimization cycle.

Starting with a given set of measurements \mathbf{y}_p and the gradients $\frac{\partial \mathbf{y}_p}{\partial \mathbf{u}}(\mathbf{u}_k)$, the model structure that minimizes the total modifier is found. The optimal model and the corresponding

modifier values are passed to the MAy-RTO, which computes the input value for the next iteration. The input is filtered before being passed to the process, where it results in an updated steady state with corresponding plant measurements and plant gradients that can be used in the next iteration.

In the following sections, the proposed method is implemented in two case studies in order to illustrate its steps, the use of modifiers as model performance indicators, and the effects of the method on the economic optimization.

Case study 1: Continuous stirred-tank reactor

Overall process description

This section introduces a simple example to illustrate how the proposed method works. For this purpose, a continuous stirred-tank reactor with perfect control is chosen. To simplify the model, it is assumed that the reactor is isothermal and that heat can be instantaneously removed without cost. The system is based on the reactor presented in François and Bonvin¹³.

There are four components in the system A , B , C and D and the process goal is to maximize the concentration of C in the reactor outlet. Thus, in Equation (6), the economic optimization objective can be formulated as $\phi := C_C$ and the set of operational constraints G is empty. Regarding the process, the reactor is fed with two streams, F_1 and F_2 , whose composition is partially unknown. Only the concentration of A is known in F_1 and the concentration of B in F_2 , the concentration of other components is not measured at the inlets. On the other hand, the concentration of all components except A is measured at the reactor outlet. The set of reactions taking place in the system is also initially unknown.

Given the process information above, three possible process scenarios are proposed (and there is no prior evidence of which one is correct):

1. There are two reactions taking place, $A + B \rightarrow C$ and $2B \rightarrow D$. In addition, the

- process is fed with two streams, one containing pure A and the other pure B ;
- For the second scenario, the reaction set is different, only one reaction is taking place, $A + 3B \rightarrow C + D$. The process is still fed with two streams containing only pure components;
 - In the third scenario, there is also only one reaction taking place in the system, $A + B \rightarrow C$. However, instead of feeding the process with pure A and B , both streams are contaminated with a certain amount of D .

A process flowsheet presenting the unknown process features/scenarios as blocks is shown in Figure 3.

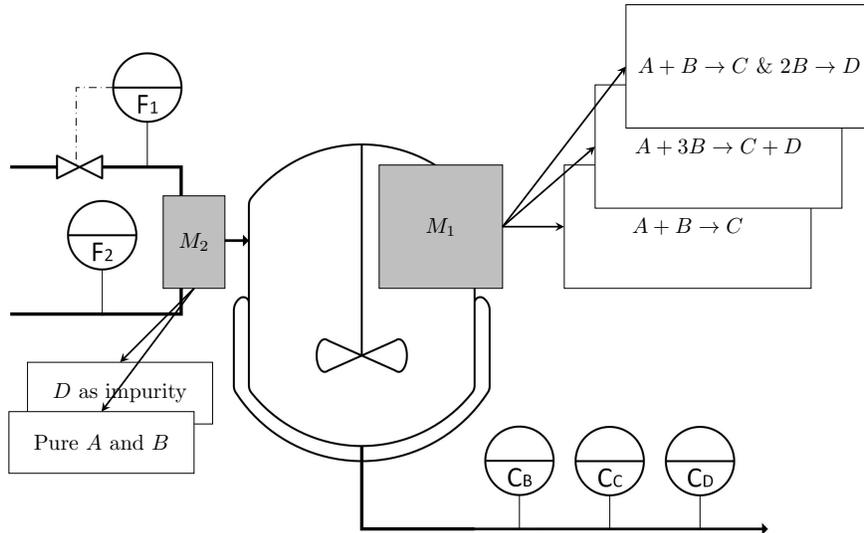


Figure 3: Flowsheet of the continuous stirred-tank reactor. The process scenarios that might occur are listed showing the possible model options for each block

Note that, the three scenarios have a significant impact on the unit optimization strategy. Due to the simplifying model assumptions, the reaction rates can be controlled only by manipulating the reactor residence time. This is achieved by changing the system input F_1 , which is the stream containing either pure A or mostly A with D as impurity, because F_2 is given and fixed at a nominal value during the simulations. Thus, the reaction sets of scenarios (1) and (2) provide different situation for optimization. In scenario (1), larger residence times

increase the production of D instead of C , which is the product of interest, for the same concentration of A . Such condition happens due to the order of the reactions. On the other hand, in scenario (2), there are no reactions competing for the reactants. Moreover, due to the reaction rate equation, increasing the residence time affects the production of both C and D equally. In the third scenario, D is not produced by the reaction set but comes as impurity in the feed streams. Therefore, changing the feed flow, impacts the concentration of D leaving the system but do not impact the trade-off between forming C and D .

Model description

The complete system can be described by the following steady-state model:

$$\begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} \frac{F_1 C_{A,in}}{V} \\ \frac{F_2 C_{B,in}}{V} \\ 0 \\ M_2 \end{bmatrix} - \frac{(F_1 + F_2)}{V} \begin{bmatrix} C_A \\ C_B \\ C_C \\ C_D \end{bmatrix} + M_1 \quad (7)$$

where, C_i is the concentration of the component i leaving the reactor. $C_{A,in}$ and $C_{B,in}$ are the inlet compositions of A and B . V is the reactor volume. F_1 and F_2 are the process inlet streams. F_1 is the manipulated variable of the reactor system and F_2 is kept constant during the simulations. The complete model contains four states, $x = [C_A, C_B, C_C, C_D]^T$, but only three are measured, $y = [C_B, C_C, C_D]^T$. Given the scenarios, two different blocks, M_1 and M_2 , can be identified in the complete system model as shown in Equation (7) and Figure 3. A block M_1 that encompasses the reaction set and a block M_2 that represents the purity of the feed stream. According to the nomenclature used in Equation (4), they are represented

by:

$$M_1 := \left\{ \left(m_{11} : \begin{bmatrix} -1 & 0 \\ -1 & -2 \\ 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} k_1 C_A C_B \\ k_2 C_B^2 \end{bmatrix} \right) \vee \left(m_{12} : \begin{bmatrix} -1 \\ -3 \\ 1 \\ 1 \end{bmatrix} \begin{bmatrix} k_3 C_A C_B^3 \end{bmatrix} \right) \vee \left(m_{13} : \begin{bmatrix} -1 \\ -1 \\ 1 \\ 0 \end{bmatrix} \begin{bmatrix} k_1 C_A C_B \end{bmatrix} \right) \right\} \quad (8)$$

where, k_1 , k_2 and k_3 are the rate constants of the chemical reactions. Note that, the stoichiometry matrix changes for each reaction set. Also, the notation $m : [\dots]$ means that sub-model m can be defined by the equations $[\dots]$. For the second block, the following set is defined:

$$M_2 := \left\{ \left(m_{21} : 0 \right) \vee \left(m_{22} : \frac{(F_1 + F_2)C_D}{V} \right) \right\} \quad (9)$$

Depending on the model structure, F_1 contains pure A or mostly A and D as impurity. In turn, F_2 contains pure B or mostly B and D as impurity. Note that, with 3 options for M_1 and 2 for M_2 , there are 6 different model structures available. However, there exists a correlation between sets M_1 and M_2 . If the third option m_{13} is chosen for M_1 , it means that D is not a reaction product but a impurity in the feed streams. For this reason, choosing m_{13} directly implies that m_{22} must be chosen to explain the presence of D in the system. On the other hand, if m_{11} or m_{12} are chosen, according to the initial model assumptions showed in the beginning of this Section, D is exclusively formed as a reaction product. In this case, only m_{21} can be chosen. Hence, there are only 3 model structures available, which are shown in Table 1.

Table 1: Model structure adaptation: Available model structures for case study 1. CM_1 is the model that represents the plant correctly.

Complete model	Block 1 (M_1) - Reaction set	Block 2 (M_2) - Feed stream	Model structure
CM_1	$A + B \rightarrow C$ & $2B \rightarrow D$	Pure A and B	$(m_{11} \wedge m_{21})$
CM_2	$A + 3B \rightarrow C + D$	Pure A and B	$(m_{12} \wedge m_{21})$
CM_3	$A + B \rightarrow C$	D as impurity	$(m_{13} \wedge m_{22})$

Simulation set-up

The proposed method is simulated in the MATLAB R2018a programming environment (Mathworks Inc., Natick, MA, USA) using the CasADi v3.4.5 extension¹⁴ for algorithmic differentiation. The NLP problem is solved using IPOPT version 3.12.2.¹⁵ The MINLP problem is solved using the genetic algorithm solver embedded in MATLAB.¹⁶

In order to represent the plant, the model CM_1 of the previous section is used. Thus, the actual reaction set is $A + B \rightarrow C$ and $2B \rightarrow D$ and the plant is fed with pure A and pure B . No noise is added to the model outputs in the simulations. Hence, the solution of the model that represents the plant is directly used to estimate plant gradients and calculate the modifiers. The initial condition and the parameters used in the simulations are shown in Table 2 and the MAy tuning parameters in Table 3.

Table 2: Initial condition and parameters for the CSTR simulation

Description	Symbol	Value	Unit
Initial concentration of A	$C_{A,0}$	0.7385	[mol/L]
Initial concentration of B	$C_{B,0}$	0.0231	[mol/L]
Initial concentration of C	$C_{C,0}$	0.4922	[mol/L]
Initial concentration of D	$C_{D,0}$	0.0308	[mol/L]
Feed concentration of A	$C_{A,in}$	2	[mol/L]
Feed concentration of B	$C_{B,in}$	1.5	[mol/L]
F_1 initial flow rate	$F_{1,0}$	8	[L/min]
F_2 flow rate	F_2	5	[L/min]
Rate constants of reaction 1	k_1	0.75	[L/(mol min)]
Rate constants of reaction 2	k_2	1.5	[L/(mol min)]
Rate constants of reaction 3	k_3	5	[L ³ /(mol ³ min)]
Reactor volume	V	500	[L]

The modifier implementation strategy should be carefully determined because it can compromise the MAy method convergence.⁵ Instead of implementing the full modifiers as in Equation (2), first-order exponential filters in Equation (5) are used in the simulations for avoiding overaggressive corrections to the modifiers that might destabilize the system.¹⁷ The modifier filter values as well as the input filter value, which was introduced earlier, were

found by trial-and-error. Additionally, the algorithm to estimate the plant gradients ($\frac{\partial \mathbf{y}_p}{\partial \mathbf{u}}$) need to be determined. In this case study, a finite-difference approximation (FDA), which is the simplest SS method to estimate steady-state gradients,¹³ is used. The basic idea of this method is to apply small perturbation (Δh) around the current operating point to estimate the plant gradients. However, there is a trade-off regarding the perturbation size. If Δh decreases, the errors relative to noise measurements increase. In turn, if it increases, the derivative approximation error increases¹⁸. The value of Δh is chosen in a trial-and-error process and it is also shown in Table 3.

Table 3: Parameters of economic optimization via MAy and gradient estimation method for the CSTR simulation.

Description	Symbol	Value
Input filter	K_u	0.4
Zeroth order modifier	K_ϵ	0.9
First order modifier	K_λ	0.9
Perturbation step size	Δh	0.01

Results

The CSTR presented in the previous section is optimized using Output Modifier Adaptation while the model structure is adapted online. Since both use the information provided by the modifiers, the behavior of ϵ and λ is analyzed in addition to the set-points \mathbf{u}_k computed by the MAy method (using Equation (6) and the input filter). Clearly, the new method should not affect the MAy performance (i.e. the plant optimum should be reached while the model structure changes). Therefore, the results of the optimization run are analyzed first.

The RTO inputs computed by MAy and the perturbation points for estimating the plant gradient are plotted in Figure 4 together with the production of C as a function of the process input, F_1 . Different markers are used for identifying which model structure is chosen along the optimization run. $(m_{11} \wedge m_{21})$ is represented by a diamond, $(m_{12} \wedge m_{21})$ by a square and $(m_{13} \wedge m_{22})$ by a triangle. The asterisks represent the perturbation points of the FDA

method for estimating plant gradients.

Clearly, the optimization scheme reaches the plant optimum. The model structure algorithm is initialized with the assumption that there is only one reaction taking place ($A + 3B \rightarrow C + D$) and both feed streams are pure. Since each one of the three proposed scenarios has a significant impact on operation of the unit and there is no noise in the simulations, the proposed method quickly identifies the correct model structure.

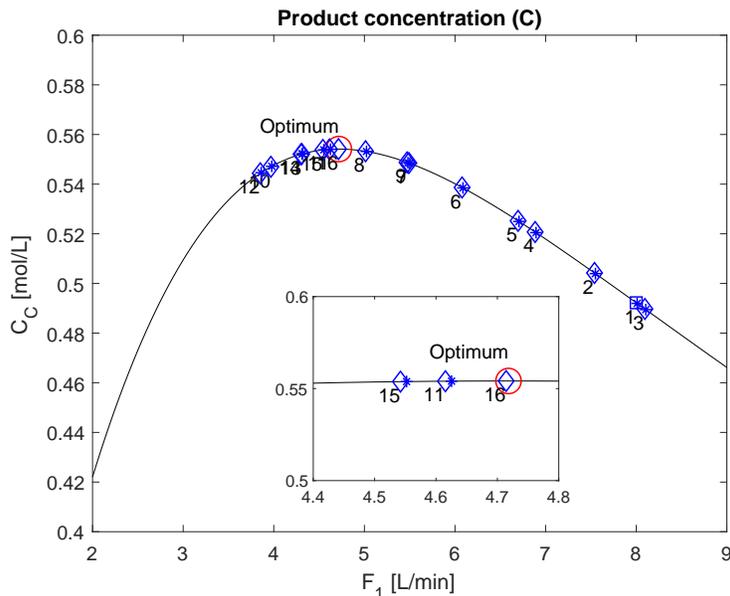
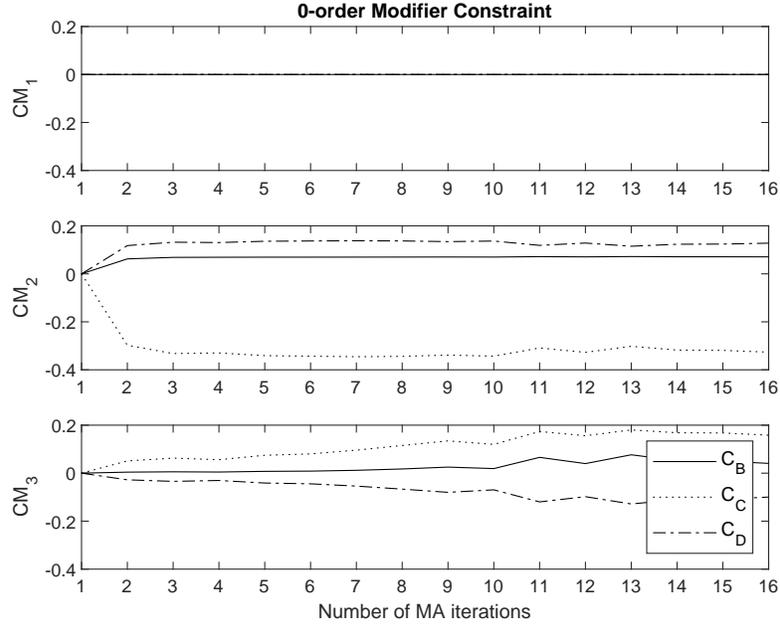


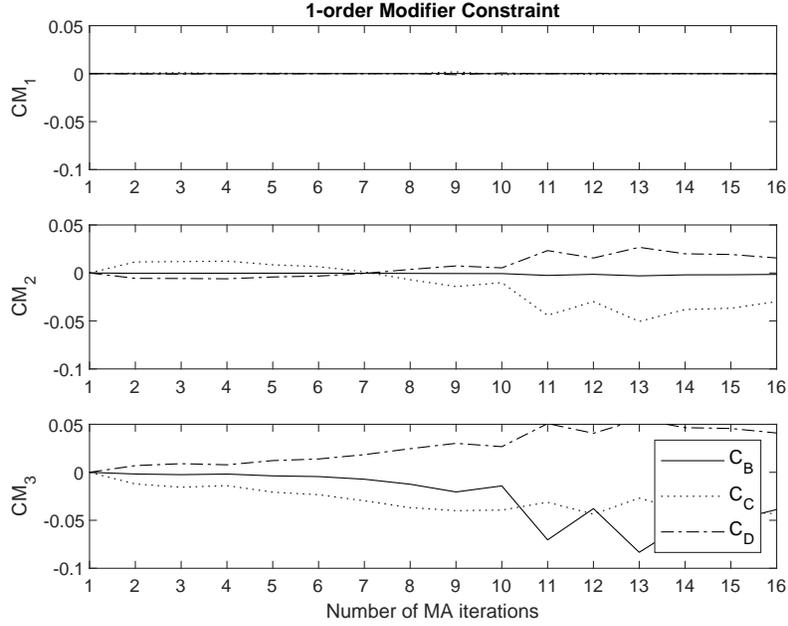
Figure 4: Optimization run. The black line represents the production of C as a function of the process input. The blue markers represent the RTO inputs calculated by the MAY method (using Equation (6)). Each marker type represents a different model structure. The asterisks are the perturbation points for estimating the plant gradient via FDA. The plant optimum is indicated by a red circle. An inset showing the region near the optimum is also plotted.

Next, the behavior of the modifiers is analyzed in Figure 5. Since CM_1 represents the plant perfectly and there is no noise, the modifiers are approximately zero in every MAY iteration. Note that, due to small errors in the gradient estimation, there is a minor deviation in λ around the ninth iteration. On the other hand, for model structures CM_2 and CM_3 , the modifiers deviate from zero to a greater extent, taken into account that the mea-

surement order of magnitude is around 0.1. Given the modifier profiles, the incorrect set of equations can be easily identified. For example, CM_3 does not propose any mechanism for the generating D and the consumption of B is underestimated. Thus, the modifiers related to the measurement of C_B and C_D present a larger deviation from 0. However, it uses the a correct reaction mechanisms for predicting C_C . Therefore, the modifiers regarding C_C are closer to zero in comparison to CM_2 , which uses an incorrect mechanism for predicting the production of C .



(a) Zero order modifier - ϵ



(b) First order modifier - λ

Figure 5: Evolution of the modifiers during the optimization executions. For each measurement $y = [C_B, C_C, C_D]^T$, a modifier ϵ and λ can be computed. The modifiers are calculated for all model structures. The y -axis is labeled to identify each structure: CM_1 is $(m_{11} \wedge m_{21})$, CM_2 is $(m_{12} \wedge m_{21})$ and CM_3 is $(m_{13} \wedge m_{22})$.

In conclusion, this first case study shows the behavior of the modifiers in a situation close to ideal, in which the process measurements do not contain noise. This simplifying hypothesis is taken in order to show that the modifiers possess significant information in relation to the model structure, illustrating their capability to be used as indicators that the model does not match the plant. Therefore, it is assumed that they are good candidates to be used as a model performance indicator. The second case studies applies the proposed method for selecting the model structure online in a more complex process, a gas lifted oil well network, with the presence of measurement noise.

Case study 2: Gas lifted oil well network

Overall process description

In offshore production of oil and gas, the economic performance is affected by the reservoir pressure. In cases that the pressure is not high enough to economically lift the fluids from the reservoir to the surface, artificial methods are required. One of the most commonly applied methods is gas lifting,¹⁹ in which gas is injected at the well bottom reducing fluid mixture density and decreasing the pressure at the bottom of the well. As a result, the inflow from the reservoir increases. On the other hand, larger gas lift flowrates also result on an increase of frictional pressure drop, which reduces the reservoir outflow. Depending on the operational point, the frictional drop effects overcome the hydrostatic pressure reduction, which may decrease the well production.

Therefore, there is a trade-off between gas lift flowrate and well production, which is intensified in cases of well networks. In the latter case, allocation of limited resources among wells, like gas availability, and production capacity constraints of the downstream facilities affect the optimal well network production. This scenario shows potential for RTO applications in gas lifted oil well networks, which has been explored by an increasing number of papers.^{8,19} A simplified flowsheet of the two-well network used in this case study is shown

in Figure 6.

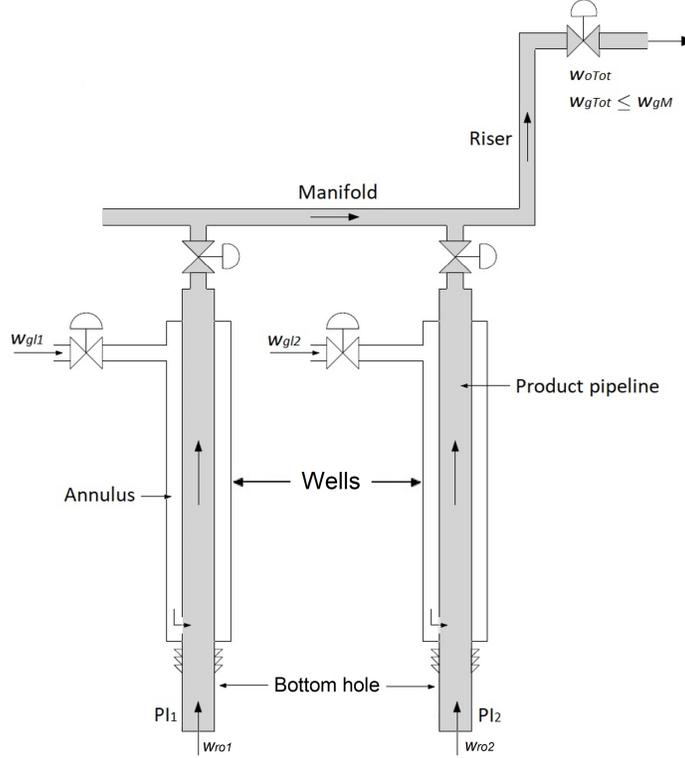


Figure 6: Network containing two gas lifted wells.⁸ The system consists of *wells* with an *annulus*, a void between the actual product pipeline and the external tubing where gas is injected to increase the flow from reservoir, as well as a *riser*, a pipeline system in which the gas/oil mixture is transported from the wells to the surface. The system also includes a *manifold* that connects the wells to the main pipeline.

The optimal operation of the system is achieved by maximizing oil production while minimizing the cost related to compression of the gas for artificial lifting. The optimization problem (Equation 6) considers processing capacity constraints and constraints on the maximum gas lift flowrates. For a 2 well network, it becomes:

$$\begin{aligned}
 & \max_{\mathbf{u}=[w_{gl,1}, w_{gl,2}]^T} \phi := w_{oTot}^2 - 0.5 \sum_{i=1}^2 w_{gl,i}^2 \\
 & \text{s. t.} \quad G := \left\{ \begin{array}{l} g_1 : w_{gTot} - w_{gM} \leq 0 \\ g_2 : w_{gl,1} - w_{glM} \leq 0 \\ g_3 : w_{gl,2} - w_{glM} \leq 0 \end{array} \right\} \quad (10)
 \end{aligned}$$

in which, w_{oTot} and w_{gTot} are the total oil and gas production of the well network, respectively; $w_{gl,i}$ is the gas lift flow rate of well i ; w_{gM} is the maximum gas processing capacity of the downstream processes; and w_{glM} is the maximum gas lift flowrate. The steady-state problem has two decision variables, $\mathbf{u} = [w_{gl,1}, w_{gl,2}]^T$: the mass flow rate of gas lift in each well. The total oil and gas production of the well are calculated using the input-output mapping, $[w_{oTot}, w_{gTot}]^T = \mathbf{y}(w_{gl,1}, w_{gl,2})$, which represents the steady-state solution of the gas lifted well network nonlinear model. This mapping is adapted by the Output Modifier Adaptation method (i.e. $\mathbf{y} \mapsto \mathbf{y}_{ad}$) like in Equation (1).

Model description

The two well nonlinear process model is based on Krishnamoorthy et al.⁸. A detailed description can be found in the original paper and also in the Supporting Information Section of this paper. The network is modeled as a nonlinear system of equations, which includes mass balances for oil and gas and relations for calculating density of the fluid mixture, flow through valves and pressure drop along the well and riser section. The main modeling assumptions are: constant temperatures, ideal gas behavior, and simple linear relations to calculate the reservoir outlet flows. The pressures along the wells and riser are the measured variable of the process and the inputs are the mass flow rate of gas lift in each well. Since both Output Modifier Adaptation and the method for selecting the model structure online are steady-state methods, only the steady-state behavior is considered. No dynamics are included in the case study.

To implement the proposed method, three blocks are considered in the process model. In Figure 7, the blocks are highlighted in the process flowsheet. The first block is related to the pressure loss due to friction along the riser/manifold:

$$M_1 := \left\{ \left(m_{1,1} : \begin{bmatrix} \text{pressure drop} \\ \text{is negligible} \end{bmatrix} \right) \vee \left(m_{1,2} : \begin{bmatrix} \text{pressure drop calculated by} \\ \text{Darcy-Weisbach equation} \end{bmatrix} \right) \right\}$$

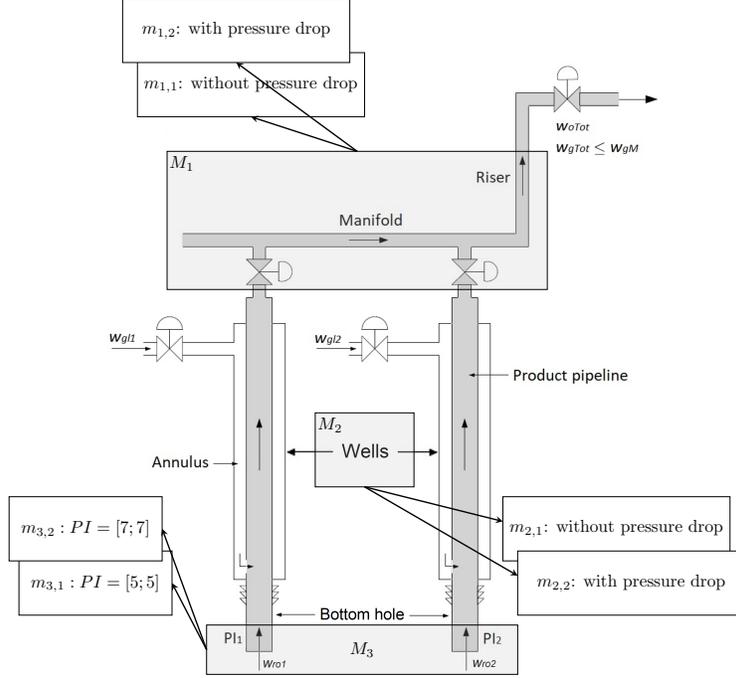


Figure 7: Flowsheet of the network containing two gas lifted wells, showing the model features or blocks that are used in the model structure adaptation algorithm

The second block is related to the pressure loss due to friction along the along the wells:

$$M_2 := \left\{ \left(m_{2,1} : \begin{bmatrix} \text{pressure drop} \\ \text{is negligible} \end{bmatrix} \right) \vee \left(m_{2,2} : \begin{bmatrix} \text{pressure drop calculated by} \\ \text{Darcy-Weisbach equation} \end{bmatrix} \right) \right\}$$

The third block is related to the reservoir model. M_3 contains two different sub-models for calculating the oil flowrate leaving the reservoir, which is associated with the productivity index (PI). The reservoir is represented by $w_{ro,i} = PI(i)(p_r - p_{bh,i})$ where $i = 1, 2$ (one model for each well). This linear equation relates the reservoir oil outlet, w_{ro} , with the difference between well bottom hole, p_{bh} , and reservoir, p_r , pressures. For each sub-model, different values of PI are used:

$$M_3 := \left\{ \left(m_{3,1} : PI = \begin{bmatrix} 5 \\ 5 \end{bmatrix} \right) \vee \left(m_{3,2} : PI = \begin{bmatrix} 7 \\ 7 \end{bmatrix} \right) \right\}$$

Note that with only 3 blocks with two sub-models each, 2^3 different block combina-

tions are possible. Thus, the number of available sub-model combinations can increase to a great extent with the number of blocks and sub-models, which highlights the importance of elaborating the model structure decision as an optimization problem (Equation (4)).

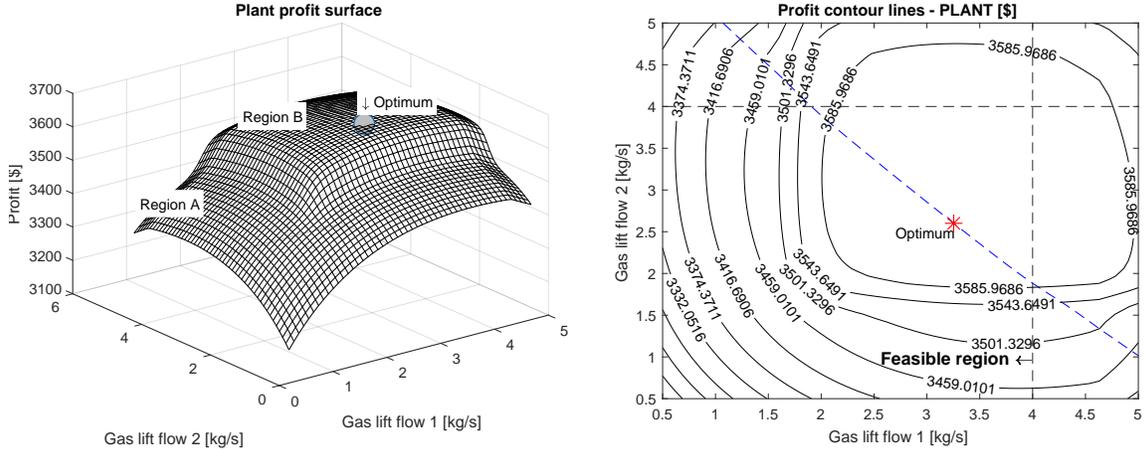
Simulation set-up

In order to test the ability of the method to track changes in the process, the plant behavior is modeled differently depending on the current operating point. As shown in Figure 8, the feasible input region is divided in Regions *A* and *B*. In Region *A*, pressure loss due to friction is taken into account in both wells and in the riser. In turn, Region *B* does not consider frictional pressure drop (i.e. the pressure difference between the top and bottom is related only with the hydrostatic pressure exerted by the liquid column). Therefore, two different plant models are developed, one for Region *A* and one for Region *B*. For assuring continuity, the transition between the different pressure drop models is calculated by a 2-D sigmoidal function of the form:

$$\omega(\mathbf{u}) = \left(\frac{1}{1 + e^{-10w_{gl,1}+17.5}} \right) \left(\frac{1}{1 + e^{-10w_{gl,2}+17.5}} \right) \quad (11)$$

where, $w_{gl,1}$ and $w_{gl,2}$ are the mass flow rate of gas lift in the well 1 and 2, respectively. For more details, please refer to the Supporting Information Section. The surface and contour plot of the plant profit are shown in Figure 8. Note that, the system optimum point is in Region *B*.

Given this set-up, a single model is not able to predict the plant behavior in both regions. There is an inherent plant-model mismatch problem in this system. Clearly, by applying MAy, the plant optimum is reached even with an incorrect model. However, as discussed previously, there exists an opportunity to extract more information of the modifiers and use them to identify the characteristics of the operating region and update the model structure in order to match the plant conditions.



(a) Plant surface. Regions *A* and *B* as well as plant optimum are highlighted.

(b) Profit contour lines for the plant, the feasible region is indicated with an arrow. Dashed blue line is the maximum gas production constraint and dashed black lines are the maximum gas lift flow rate for each well. The plant optimum is also indicated.

Figure 8: Plant profit surface

The case study simulations are carried out using the same solvers and software as in the previous case study. However, in order to estimate plant gradient for calculating the first order modifiers, the method proposed by Gao et al.²⁰ is applied. It uses current and past operating points to compute local quadratic approximations of input to output mapping functions. The quadratic approximation captures information from the plant curvature and it has the potential to decrease the influence of noise.²⁰ After computing the quadratic functions, the gradients are easily obtained by derivation. Matias et al.²¹ have previously applied this method for estimating the gradients of gas lifted oil well networks. The results indicated that it was able to provide good approximations for the plant and estimate plant gradients within an adequate precision. The tuning parameters for the gradient estimation method as well as the values of the model parameter are not shown here for the sake of brevity. They can be found in the Supporting Information Section and in Matias et al.²¹.

Finally, the following assumptions are made for simulating the case study:

1. Output Modifier Adaptation is used for optimizing the gas lifted oil well network;

2. All sub-models proposed in the previous section are used. Thus, the algorithm has 8 different model options to choose based on the modifiers;
3. The measurements of all the pressures in the system are available, however they are affected by noise. The noisy measurements have 1% noise and are calculated as $y_{noise} = y_{model}(1 + 0.01r)$, in which r is drawn from a standard normal distribution. The noise is neither correlated in time nor between measurements.

Results

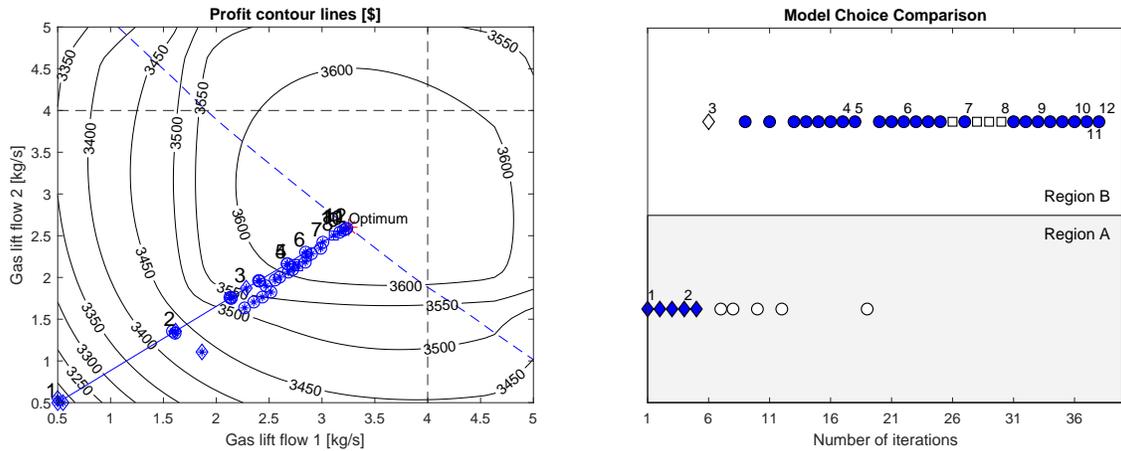
Optimization

The results of the optimization run are shown in Figure 9, they are plotted together with the plant contour lines. In addition to the optimization, the model structure adaptation algorithm is also implemented. Along the optimization path, markers are used for indicating which model is chosen. The correspondence of markers and models is shown in Table 4.

Table 4: Model structure adaptation - marker and model correspondence. Complete model 8 (CM_8) is the correct model for plant region A and complete model 2 (CM_2) for region B

Complete model	Symbol	Block 1 - Well frictional pressure drop	Block 2 - Riser frictional pressure drop	Block 3 - Reservoir parameters (PI)
CM_1	\triangleleft	without	without	[7; 7]
CM_2	\bigcirc	without	without	[5; 5]
CM_3	\star	without	with	[7; 7]
CM_4	∇	without	with	[5; 5]
CM_5	\triangle	with	without	[7; 7]
CM_6	\square	with	without	[5; 5]
CM_7	\triangleright	with	with	[7; 7]
CM_8	\diamond	with	with	[5; 5]

Figure 9a shows that the plant optimum is reached without any constraint violations despite the presence of noise. Note that, besides the optimization operating points, there are several plant probing points, which are used by the gradient estimation method (26 probing operating points *vs.* 12 MAy optimization operating points). The model structure selection algorithm also takes advantage of the plant probing points and calculates the model structure



(a) Profit contour lines for the plant. The dashed blue line is the maximum gas production constraint and dashed black lines are the maximum gas lift flow rate for each well. The plant optimum is also indicated.

(b) Model structure selected along the optimization run. Both MAy operating points and plant probing points for gradient estimation are shown. However, only the optimization operating points are numerated (in correspondence with Figure 9a). Full markers indicate that the correct model was chosen.

Figure 9: Models structure adaptation during optimization. Marker and model structure correspondence is shown in Table 4

in these points. Figure 9b shows which models are chosen during the economic optimization run. Even with measurement noise, the method is able to choose the right model for the region.

Between the 3th and 4th MAy iterations, there are several plant probing points for the gradient estimation in the plant transition region (between Regions A and B, which is still labeled as Region A for simplicity). Despite the fact that there is no correct model for this region, the method chooses the model with the smallest modifiers (i.e. minimum ψ). Thus, it chooses the model that best represents the plant according to the selected criterion. This result indicates that the method is able to cope with the situation where there is no “correct” model available, which is the case in practical applications.

Finally, after the 20th iteration, the operation does not leave Region B. However, the algorithm chooses the “wrong” model in 4 iterations. It chooses the model CM_6 , which have the correct reservoir model but the incorrect pressure drop model for the wells. As the

effects of the pressure drop are less significant, noise has a larger effect on the discrimination of models. Moreover, as the pressures along the well network are coupled, the effects of the pressure drop models are more dispersed among the measurements and are more prone to be affected by noise. Nevertheless, the method is able to choose the correct models in most iterations.

Modifiers as a model performance criterion

In Figures 10 and 11, the behavior of the total modifier for different models along the optimization run is presented. The idea is to justify the use of the modifiers as model structure indicators. Instead of analyzing the profiles for all models proposed in Table 4, only four are chosen: CM_2 and CM_8 , which are the correct models for plant regions B and A , respectively; CM_6 , which is the most frequent incorrect model chosen by the method. Note that, the only difference between models CM_6 and CM_8 is the computation of the frictional pressure drop at the riser. In turn, the difference between CM_6 and CM_2 lies in the computation of frictional pressure drop at the wells; Finally, CM_7 is included in the analysis because it has the incorrect reservoir model.

Figure 10 analyzes the contribution of the modifiers for the objective function of the model structure adaptation method, Equation (3). Since the objective function is posed as $\psi := \|\epsilon_k\|_F + \|\lambda_k\|_F$, the contribution of ϵ is calculated by $\psi_\epsilon = \|\epsilon_k\|_F$ and the contribution of λ by $\psi_\lambda = \|\lambda_k\|_F$. However, instead of plotting the absolute value of ψ_ϵ and ψ_λ , the difference between ψ for a given model and for the true model in the specific region of the iteration is shown. The values are presented for each one of the four models. Furthermore, they are related only to the 12 MAy iterations (i.e. optimization operational points but not the plant probing points). The plant regions are also indicated.

Even without knowledge of the plant-model mismatch source, the incorrect hypothesis of the reservoir model can be easily identified using the modifiers. For example, since CM_7 has the wrong reservoir model, the model predictions present significant deviations from the

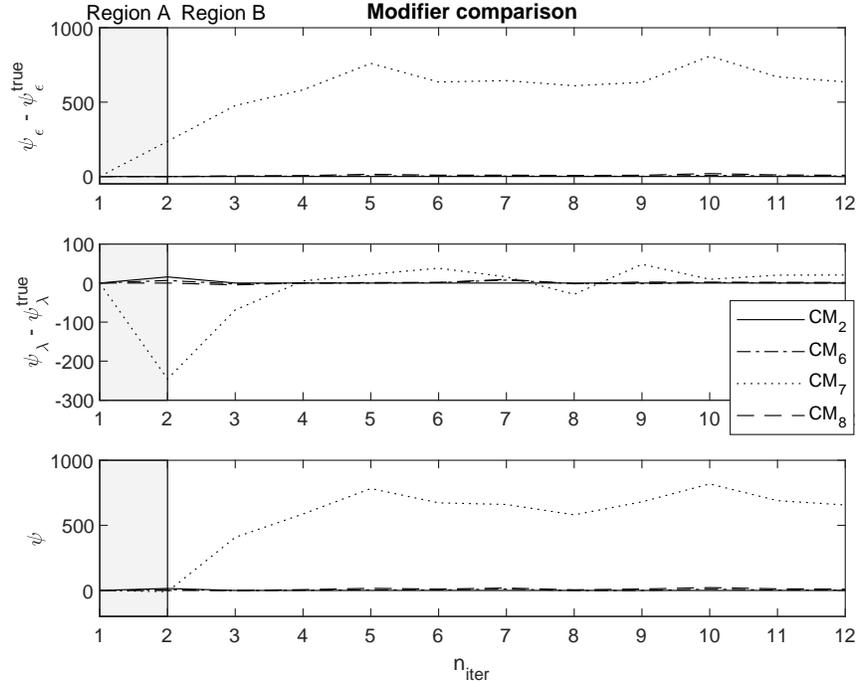


Figure 10: Behavior of the modifiers. Only four models are used for the comparison. CM_2 is the correct model for plant region B , CM_8 for region A . CM_7 has the wrong reservoir model and CM_6 is the most frequent incorrect model chosen by the method.

plant measurements, which is reflected in ψ_ϵ . In turn, the differences in ψ_λ are not so explicit. Note that, as a consequence of noise, $\psi_{\lambda,CM1} < \psi_\lambda^{\text{true}}$ in some iterations. Process noise has a great influence on the gradient estimate accuracy such that it directly affects the quality of information of the first order modifier. However, since both modifiers have a contribution to the problem objective function ψ , the method still avoids to choose models with wrong reservoir model. They are never chosen by the proposed method.

On the other hand, the effects of the pressure loss on the different models are not so distinct (CM_2 , CM_6 and CM_8 models have the correct reservoir model but different pressure loss models). In order to illustrate how the method chooses among these models, the same information shown in Figure 10 is plotted in Figure 11. However, CM_7 is excluded from the analysis.

Similarly to Figure 10, the effects of the incorrect pressure loss model are more explicit in the zeroth order modifier in Region B . On the other hand, in Region A , the contribution of

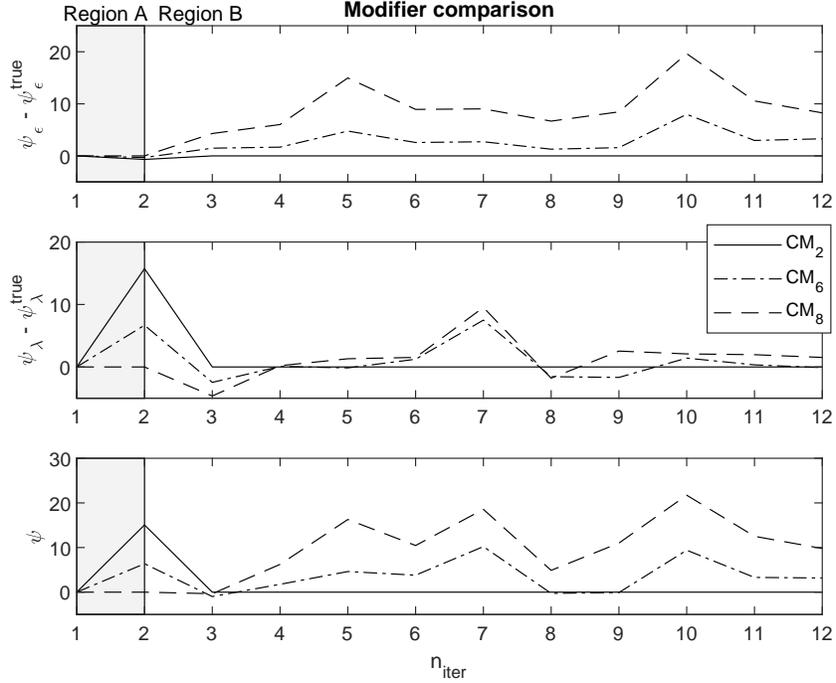


Figure 11: Behavior of the modifiers in the MAY iteration points. In contrast to Figure 10, only models with the correct reservoir model (CM_2 , CM_6 and CM_8) are shown.

the first order modifier to the objective function is more significant. The objective function of the model structure adaptation method is formulated in order to capture this balance between the influence of the zeroth and the first order modifiers. However, any norm-like function of the modifiers or any model selection criterion (like, R-square, Akaike information criterion (AIC), Bayes information criterion (BIC), etc.) could be used as objective function (Equation (3)).

Comparison of the total model modifier with commonly used model selection criterion

Among the model selection criterion, R-square is one of the most commonly used criterion to differ between competing models.²² It measures the goodness of fitness for the model, i.e. the ability of the model to represent the variability of data by comparing the variance of the model predictions with the total variance of the data.²² However, the R-square measures can

have poor performance for nonlinear models.²³ Other popular model selection criteria are the Akaike information criterion (AIC) and the Bayes information criterion (BIC). Both criteria have different theoretical backgrounds, however they are very similar in practice. They measure the trade-off between model fit, which is represented by the likelihood function, and complexity of the model.²⁴ The main difference between AIC and BIC is the penalty for model complexity.²²

In order to compare the performance of the total model modifier with commonly used model selection criterion (AIC and R-square in this case), the following experiment is performed: All the operating points (probing and optimization) shown in Figure 9 that are in Region B are analyzed. The residual (the difference between the model prediction and the “plant” measured values) for every operating point is calculated for each one of the 8 models in Table 4. Next, R-square and AIC are computed for every model.

Calculating R-square is straightforward. For calculating AIC, it is assumed that the residuals are independent, normally distributed random vectors and that the number of parameters for every model is 1. The first assumption is based on the fact that the noise added to the plant is drawn from a normal distribution. The second, on the fact that the model parameters are not estimated but a nominal set of parameters is used. Hence, there is no extra penalty regarding the model complexity. The values of AIC and R-square are shown in Table 5.

Table 5: Model comparison using common model selection criterion. Since data of Region B is used, CM_2 , which is the correct model, has the largest R-square value and the lowest AIC.

Complete model	R^2	$AIC (\times 10^3)$
CM_1	-1.9629	1.1398
CM_2	0.8545	1.0046
CM_3	-2.1764	1.1420
CM_4	0.8290	1.0282
CM_5	-2.0642	1.1405
CM_6	0.8426	1.0068
CM_7	-2.2792	1.1426
CM_8	0.8134	1.0305

After calculating the model comparison criteria for all models, the ψ value for all 8 models is plotted in Figure 12. The value of the objective function is shown for every iteration in Region B, which were already shown in Figure 9. However, instead of plotting the absolute value, the models were organized qualitatively from the smallest ψ to the largest. The information is organized in this manner in order to facilitate the comparison with the data in Table 5.

By comparing the results of Table 5 and Figure 12, it is possible to conclude that the models with the wrong reservoir model (CM_1 , CM_3 , CM_5 and CM_7) have poor prediction capacity. The R-square criterion, for example, is less than zero, which indicates that the mean predicts the data better than the model in consideration. (The R-square results confirm the previous conclusion drawn from Figure 10 regarding the behavior of ψ_ϵ). Regarding CM_2 and CM_6 , the AIC and R-square values are similar indicating that the models have comparable prediction capacity in relation to the sample, which was already observed in Figure 9b, where the wrong model, CM_6 , is chosen instead of the correct model, CM_2 , in Region B.

Since CM_2 and CM_6 have a comparable performance, the noise has a major influence on differentiating both models. Depending on the noise level, both models can be considered the same regarding the comparison criteria applied in the case study (ψ , AIC and R-square).

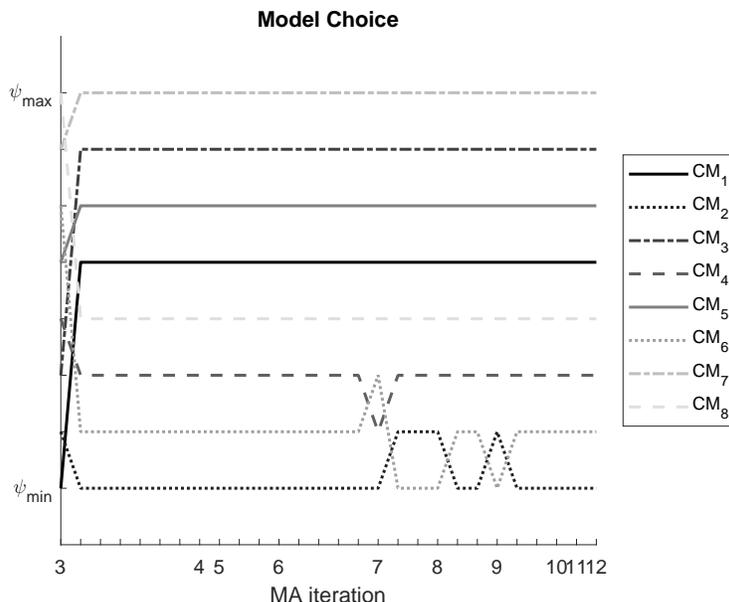


Figure 12: Behavior of the ψ in Region A for all 8 models. The models are qualitatively arranged from the smallest value of ϕ to the largest.

Hence, based on available data, it is not possible to discriminate between these models. This reflects the fact that there is no “true” model to a real process plant, and that all models are just descriptions for what is happening in the plant.

In order to justify the choice of ψ as a model selection criterion, the form of ψ in Equation (3) is used. It is easier to assess the effectiveness of the selected model structure for optimization purposes because ψ indicates how the model structure affects the constraint prediction and model shape (zeroth and first order modifiers). A second advantage of ψ is that there is no extra assumption about the residuals distribution as for calculating AIC.

The results of this case study already support the paper fundamental hypothesis that the modifiers are good candidates to be used as a model performance indicators and that they can be used to compose the objective function of the model structure adaptation method. Moreover, the findings of this case study confirm that the proposed method is able to determine the best model structure (i.e. more consistent with plant data) online while optimizing the process to the actual plant optimum using MAy.

Discussion

The two case studies demonstrate the most important aspects from the proposed method and show how it can be applied. In this section, we discuss further issues pertaining our method. In particular, these are: (1) convergence of the RTO scheme to the plant optimum; (2) convergence to the “correct” model; (3) model chattering between RTO iterations; (4) effect of scaling of the outputs; and (5) model parameter estimation.

Convergence of the RTO scheme to the plant optimum

Regarding the convergence to the plant economic optimum, the proposed method takes advantage of the MAy convergence properties in the case of structural plant-model mismatch. If the available models are adequate in a MAy sense,³ namely the reduced Hessian of the objective function is positive definite in the vicinity of the plant optimum, the economic optimization problem converges to the plant optimum even if the “correct” model is not available.

Note that, it does not matter if different models are selected between RTO-MAy iterates, because all models are modified to have the same predictions and gradients. Hence, the step computed by the RTO-MAy (Equation (6)) is the same, regardless of which model is active. However, to safeguard the procedure, one may include steps to check the model adequacy before implementing the MAy-RTO solution in the plant.

By checking the convexity of the Hessian, one can also limit the possible model choices at each iteration, i.e. including the adequacy check as a constraint to the model structure adaptation problem. Moreover, the second order information can be included directly in the problem objective function as in Marchetti et al.²⁵. In this case, the second order modifiers (upper bounds for the plant Hessian) enforce that the modified model is a strictly convex upper bound for the plant, guaranteeing convergence of the MAy scheme. The drawback of the latter feature is the difficulty in obtaining second order information of the plant.

Instead of relying on Hessian analysis or obtaining second order information, one may fall back by simply including a quadratic model, which is always adequate in a MAy sense, in the available model set. A similar approach is followed by Gao et al.²⁰, where the authors propose to switch between the modified nonlinear model and a quadratic model approximation obtained from data for computing the RTO iteration. The method by Gao et al.²⁰ is also described in details in the Supporting Information. Despite ensuring global convergence to the plant optimum, including a quadratic model in the available model set may decrease the RTO-MAy convergence speed due to the fact that the accuracy of the second order information of the model-based optimization problem results in faster convergence rates than model-free schemes Gao et al.²⁶.

Convergence to the “correct” model

Convergence to the “optimal model structure” is not straightforward. This is a much more complicated question because in reality there is no “true” model. We provide a brief investigation of this issue on a simple toy example. Figures 13 and 14 illustrate the case where the “true” model is not included in the model set. Here, two models $y_{m,1}$ and $y_{m,2}$ are used for describing the plant. They are visually compared in Figure 13. Since both models are wrong, the choice of the correct model is not obvious. Instinctively, by analyzing the plot, one may argue that is better to choose model 2, if the current operating point is on the right-hand side region of the plot, and model 1 otherwise. Such choice is also reflected by the total modifier criterion applied to the toy problem (Figure 14).

In addition to the toy example, we have already seen in the gas-lift case-study how the algorithm behaves if there is no correct model available. In Figure 9, between the 3th and 4th MAy iterations, the model selection algorithm chooses a model based on the total modifier even though the operation points lie on the transition region between Regions *A* and *B* (i.e. there is no correct model for this region in the available model set). In this case, it is difficult to give a physical interpretation for what our algorithm is doing. It simply selects the model

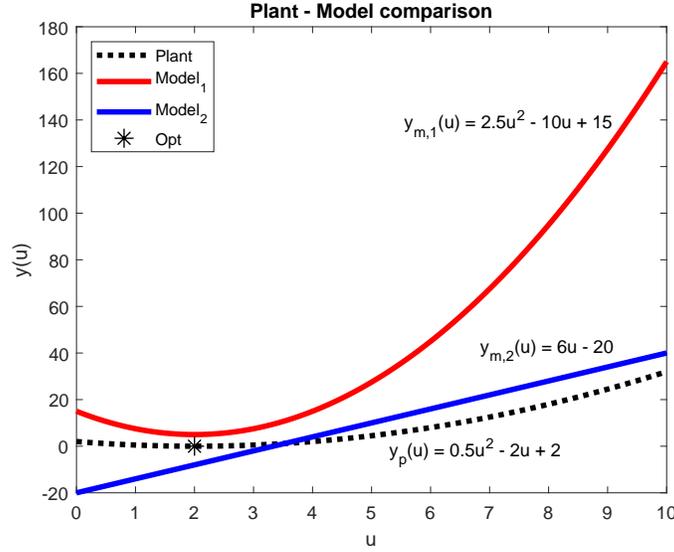


Figure 13: Models $y_{m,1}$ and $y_{m,2}$ are compared to the plant, y_p . The plant optimum is indicated with a star. Both models are different from the plant. Note that, in the right-hand side region, both the slope and the predicted value of model 2 are close to the plant, while model 1 is a better approximation of the plant near the optimum.

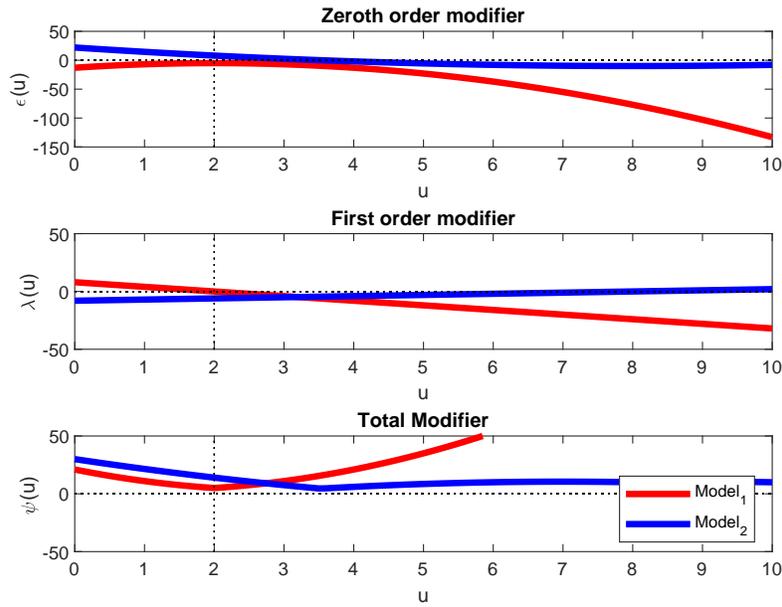


Figure 14: The models shown in Figure 13 are compared using the total modifier criterion (Equation 3). The dashed lines are used for indicating the zero line, where there is no difference between plant and model according to the total modifier criterion, and the plant optimum (at $u_p^* = 2$). Even in the case that there is no “correct model”, the total modifier criterion is able to indicate a “good enough” model for the region.

that results in the smallest total modifier value (our criterion for a “good” model).

In order to further explore how the absence of a correct model affects the convergence of the model selection method in the gas-lift case-study, we change the simulation set-up presented in the **Case Study 2** description. All conditions are kept the same except that Model 2 in Table 4 is excluded from the available model set. That is, there is no correct model for Region B, which contains the process optimum. In this new set-up, the method still drives the plant optimum. However, the model structure adaptation method chooses Model 6 (symbol: \square) in most of the iterations located in Region B as shown in Figure 15.

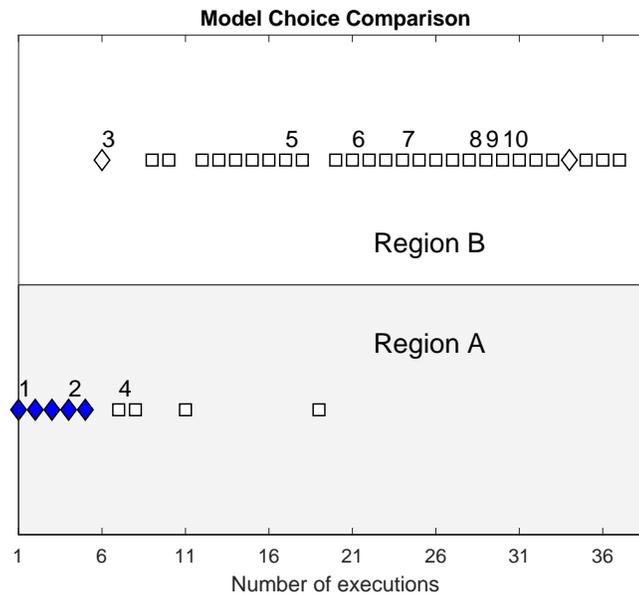


Figure 15: Model structure selected along the optimization run in the case that the correct model for Region B is excluded from the available model set. The figure is plotted exactly as Figure 9b and the model structure follows the same marker/model correspondence introduced in Table 4. Model 6 (symbol: \square) is the most frequent model chosen by the method. In the previous section, it was shown that this model has a comparable prediction capacity to the “true” plant model.

As argued previously in Table 5 and Figure 12, Model 6 has a similar prediction capacity to Model 2, which is the “true” plant model, according to the applied model comparison indicators (ψ , R^2 and AIC). Therefore, given that no correct plant model is available, the

method chooses a model that is “good enough” for the region.

What exactly is meant by “good enough” is highly connected to the application, and the issue of “modeling for a goal”² plays an important role. For the same process, different models may be used for control, simulation, optimization, etc. Since our method deals with process optimization on the background, we wish to chose a good enough model that predicts the optimality conditions for the optimization problem, i.e. the Karush–KuhnTucker (KKT) conditions for the static optimization problem. Therefore, we believe that using the smallest total modifier value as an indicator of a “good” model is a good choice despite the fact that it has no physical interpretation.

Model chattering between RTO iterations

Although the modifiers are good indicators for differentiating between optimization models, they contain only local information (at u_k). Since, they are only locally valid, the model selection algorithm may chatter between RTO iterations depending on the noise level in the measurements. For example, if two models have comparable prediction capacity, like Models 2 and 6 of the gas-lift case study, nothing guarantees that the conclusions drawn from the total modifier indicator at iteration k is the same as at the next iteration.

Including past data to the model structure adaptation problem can prevent model chattering between RTO iterations. Since data from previous operating points is used, the effect of local noisy information on the problem is reduced. However, since the plant is expected to be affected by time-varying disturbances, it is interesting to consider only recent data for choosing the model structure. For example, the modifiers can be updated using a first-order exponential filter (Equation (5)) in order to mitigate the effect of noise and avoid excessive correction of the modifiers, like in Marchetti et al.³. Such strategy is adopted in the case studies of this paper.

If desired, an extra step may be included in Algorithm 1 for avoiding chattering: Using the current plant information (at iteration k), the total modifier ψ is computed for the optimal

model structure selected at the current iteration (z_k^*) and for the model structure selected at the previous iteration z_{k-1}^* . If the norm of the difference is lower than a given threshold τ , the model structure is not updated (i.e. $z_k \leftarrow z_{k-1}^*$) chosen at the previous iteration is used. Clearly, this *ad-hoc* strategy may lead to a suboptimal performance in terms of model selection, but avoids chattering between the models if the level of noise (and/or error in the gradient estimation) increases. On the other hand, the RTO convergence is not affected.

Algorithm 2 Model Structure Adaptation method - extended

- 1: Estimate plant gradients;
 - 2: Select a combination of models such that the total model modifier ψ is minimized;
 - 3: Use new model and its modifiers to compute optimal operating point via RTO using Output Modifier Adaptation, and implement the new operating;
 - 4: If $\|\psi(z_k^*) - \psi(z_{k-1}^*)\| \leq \tau$, then $z_k \leftarrow z_{k-1}^*$;
 - 5: Repeat from 1.
-

Effect of scaling the outputs

The objective function of the model structure adaptation problem, Equation (2) is scaling dependent. Therefore, if one of the outputs dominates others in magnitude, the algorithm will force the minimization of the modifiers related to this output instead of the others, which may affect the convergence to the correct model.

In order to have the same basis of comparison, the measurements need to be scaled before calculating the modifiers if they have different order of magnitudes. Several strategies can be applied, like standardization, mean normalization, min-max normalization, etc.. However, since the models used in RTO-MAy are usually nonlinear, formulating a general strategy for scaling is difficult. This decision is normally based on engineering aspects, relying on process knowledge. Clearly, if the modifiers are scaled, the adapted model (Equation (1)) needs to be adjusted properly to consider the scaling strategy.

In addition, weighting factors may be introduced to change the emphasis on finding a model structure that fits the bias (zeroth-order modifier) or gradient (first-order modifier).

This, however, must be done on additional engineering insight and depends on the case at hand and the reliability of the available measurements and gradient estimates.

Model parameter estimation

Finally, the fifth issue is related to estimating model parameters. The model structure adaptation problem can be extended by adapting the parameters while selecting the model structure, similarly to Hille and Budman⁴. The model parameters for all available sub-model combinations can be adapted to correct for errors in the predicted cost-function and gradients and then the model structure adaptation is carried out to select the best adapted structure. A second, more interesting alternative, is to merge the parameter adaptation step and the model selection into a single optimization problem. Although unnecessary for guaranteeing MAy convergence to the plant optimum, this extension can reduce the number of times that the model structure needs to be changed during the algorithm execution.

Optimizing for optimal parameter values leads to MINLP with many more decision variables and further issues regarding the parameter estimation problem (i.e. identifiability issues related to non-unique parameters in addition to possibly non-unique models). In order to clearly present the main idea of our approach, we decided to not include the parameter estimation problem with its associated complications in this paper. Instead, this will be investigated in future work.

Conclusion

This paper proposes a new method that combines Output Modifier Adaptation (MAy), an RTO variant, with an online model structure adaptation method. Without deriving a fixed model for the system, the novel approach is able to determine the best model structure (i.e. more consistent with plant data) online by combining different sub-models while optimizing the process to the actual plant optimum.

The method is based on the following idea: The system model is divided in blocks. Each of these blocks encompasses a certain phenomenon described by the model. Next, several sub-models containing first principle equations are proposed to describe the given phenomenon. With the operation, information comes from the plant and the competing sub-models are compared using the modifiers of MAy as criterion. Therefore, the novel method uses a population of models to describe the inner workings of the process and, by using the modifiers, is able to determine the overall model structure that best fits the plant data.

This approach is appealing in cases that some aspects of the process are not well known or there is not enough information to make a priori assumptions for the model structure (i.e. model uncertainties are mitigated by proposing several knowledge-based models for the same phenomenon). Due to the general validity of these knowledge-based relations, the model is capable of predicting the behavior of the system to a large range of process conditions. In addition, while all available knowledge of the process is used for building the sub-models, the missing information (representing interconnections of the sub-models) is determined fitting the model to process data. The combination of process knowledge with online data can be attractive and rewarding. Additionally, after determining a first “generation” of process block candidates, the proposed method allows the addition of more blocks, increasing the number of possible combinations.

The novel method is demonstrated in a continuous stirred tank reactor and in a gas lifted oil well network. This first case study is designed to illustrate the behavior of the modifiers in a situation without noise in order to avoid errors in the gradient estimation. The idea is to show that the modifiers contain significant information in relation to the model structure and prediction capacity. Hence, they are good candidates for indicating model performance.

The second case study is used for showing the benefits of the method in a more complex process, a gas lifted oil well network. In this case, the plant presents two different behaviors (depending on the operation region) and 8 possible models are proposed. Among this popu-

lation of models, there are models valid for one of the plant behaviors, models that describe the plant behavior partially and models that completely fail to predict it. The simulation results show that, in few iterations, the algorithm is able to determine among the available models, the one that best describes the plant.

This novel approach is the basis of a development towards online model structure improvement, unifying process knowledge (represented by the model blocks) and data (by the modifiers). Due to the proposed framework, the model is allowed to evolve over time, new blocks can be added (or deleted) during the operation. The most important contribution of this work is that, by allowing that the model evolve over time while optimizing the system, the method can be a starting point for changing the paradigm that the model structure is an immutable entity.

Acknowledgement

The authors would like to thank M.Sc. Dinesh Krishnamoorthy for providing the process models used in this work. The authors acknowledge Financial Support from the Norwegian research council /Intpart, SUBPRO

Supporting Information Available

The Supporting Information contains the model equations and the description of the method proposed by Gao et al.²⁰ to estimate plant gradients. This information is available free of charge via the Internet at <http://pubs.acs.org/>.

References

- (1) Bequette, B. W. *Process control: modeling, design, and simulation*; Prentice Hall Professional, 2003.

- (2) Bonvin, D.; Georgakis, C.; Pantelides, C.; Barolo, M.; Grover, M.; Rodrigues, D.; Schneider, R.; Dochain, D. Linking models and experiments. *Industrial & Engineering Chemistry Research* **2016**, *55*, 6891–6903.
- (3) Marchetti, A.; Chachuat, B.; Bonvin, D. Modifier-adaptation methodology for real-time optimization. *Industrial & engineering chemistry research* **2009**, *48*, 6022–6033.
- (4) Hille, R.; Budman, H. M. Simultaneous identification and optimization of biochemical processes under model-plant mismatch using output uncertainty bounds. *Computers & Chemical Engineering* **2018**, *113*, 125–138.
- (5) Marchetti, A. G.; François, G.; Faulwasser, T.; Bonvin, D. Modifier Adaptation for Real-Time Optimization Methods and Applications. *Processes* **2016**, *4*, 55.
- (6) Golub, G. H.; Van Loan, C. F. *Matrix computations*; JHU Press, 2012; Vol. 3.
- (7) Martinez Villegas, R.; Budman, H.; Elkamel, A. Identification of dynamic metabolic flux balance models based on parametric sensitivity analysis. *Industrial & Engineering Chemistry Research* **2017**, *56*, 1911–1919.
- (8) Krishnamoorthy, D.; Foss, B.; Skogestad, S. Real-time optimization under uncertainty applied to a gas lifted well network. *Processes* **2016**, *4*, 52.
- (9) Hülse, E. O.; Camponogara, E. Robust formulations for production optimization of satellite oil wells. *Engineering Optimization* **2017**, *49*, 846–863.
- (10) Krishnamoorthy, D.; Suwartadi, E.; Foss, B.; Skogestad, S.; Jäschke, J. Improving scenario decomposition for multistage mpc using a sensitivity-based path-following algorithm. *IEEE control systems letters* **2018**, *2*, 581–586.
- (11) Raman, R.; Grossmann, I. E. Modelling and computational techniques for logic based integer programming. *Computers & Chemical Engineering* **1994**, *18*, 563–578.

- (12) Yeomans, H.; Grossmann, I. E. A systematic modeling framework of superstructure optimization in process synthesis. *Computers and Chemical Engineering* **1999**, *23*, 709–731.
- (13) François, G.; Bonvin, D. Use of Transient Measurements for the Optimization of Steady-state Performance via Modifier Adaptation. *Industrial & Engineering Chemistry Research* **2013**, *53*, 5148–5159.
- (14) Andersson, J. A general-purpose software framework for dynamic optimization. Ph.D. thesis, PhD thesis, Arenberg Doctoral School, KU Leuven, Department of Electrical Engineering (ESAT/SCD) and Optimization in Engineering Center, Kasteelpark Arenberg 10, 3001-Heverlee, Belgium, 2013.
- (15) Wächter, A.; Biegler, L. T. On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming. *Mathematical programming* **2006**, *106*, 25–57.
- (16) Conn, A. R.; Gould, N. I.; Toint, P. A globally convergent augmented Lagrangian algorithm for optimization with general constraints and simple bounds. *SIAM Journal on Numerical Analysis* **1991**, *28*, 545–572.
- (17) Chachuat, B.; Marchetti, A.; Bonvin, D. Process optimization via constraints adaptation. *Journal of Process Control* **2008**, *18*, 244–257.
- (18) Brekelmans, R.; Driessen, L.; Hamers, H.; Hertog, D. D. Gradient Estimation Schemes for Noisy Functions. *Journal of Optimization Theory and Applications* **2005**, *126*, 529–551.
- (19) Peixoto, A. J.; Pereira-Dias, D.; Xaud, A. F.; Secchi, A. R. Modelling and extremum seeking control of gas lifted oil wells. *IFAC-PapersOnLine* **2015**, *48*, 21–26.

- (20) Gao, W.; Wenzel, S.; Engell, S. A Reliable Modifier-Adaptation Strategy for Real-Time Optimization. *Computers & Chemical Engineering* **2016**, *91*, 318–328.
- (21) Matias, J. O.; Le Roux, G. A.; Jäschke, J. Modifier adaptation for real-time optimization of a gas lifted well network. *IFAC-PapersOnLine* **2018**, *51*, 31–36.
- (22) Busemeyer, J. R.; Diederich, A. *Neuroeconomics (Second Edition)*; Elsevier, 2014; pp 49–61.
- (23) Cameron, A. C.; Windmeijer, F. A. G. An R-squared measure of goodness of fit for some common nonlinear regression models. *Journal of econometrics* **1997**, *77*, 329–342.
- (24) Brink, A.; Westerlund, T. The joint problem of model structure determination and parameter estimation in quantitative IR spectroscopy. *Chemometrics and intelligent laboratory systems* **1995**, *29*, 29–36.
- (25) Marchetti, A. G.; Faulwasser, T.; Bonvin, D. A feasible-side globally convergent modifier-adaptation scheme. *Journal of Process Control* **2017**, *54*, 38–46.
- (26) Gao, W.; Wenzel, S.; Engell, S. Integration of gradient adaptation and quadratic approximation in real-time optimization. 2015 34th Chinese Control Conference (CCC). 2015; pp 2780–2785.

Graphical TOC Entry

