Simultaneous Online Model Identification and Production Optimization Using Modifier Adaptation

José Matias, Vyacheslav Kungurtsev, Malcolm Egan

Simultaneous Online Model Identification and Production Optimization Using Modifier Adaptation ⋆

José Matias [a], Vyacheslav Kungurtsev [b], Malcolm Egan [c]

[a] *NTNU, Trondheim, Norway*

[b] *Czech Technical University in Prague, Prague, Czech Republic*

[c] *Univ Lyon, INSA Lyon, INRIA, Lyon, France*

Abstract

A key problem for many industrial processes is to limit exposure to system malfunction. However, it is often the case that control cost minimization is prioritized over model identification. Indeed, model identification is typically not considered in production optimization, which can lead to delayed awareness and alerting of malfunction. In this paper, we address the problem of simultaneous production optimization and system identification. We develop new algorithms based on modifier adaptation and reinforcement learning, which efficiently manage the tradeoff between cost minimization and identification. For two case studies based on a chemical reactor and subsea oil and gas exploration, we show that our algorithms yield control costs comparable to existing methods while yielding rapid identification of system degradation.

1 Introduction

In industrial processes, the dynamics can vary dramatically depending on the system degradation. For example, oil and gas flow during extraction from a subsea oil reservoir depends in a complex way on the degradation of a compressor [1]. A fundamental problem is therefore to optimize control inputs, accounting for uncertainty in the system behavior. At the same time, it is desirable to identify the state of the degradation in order to perform auxiliary tasks such as system maintenance.

In many cases, it is useful to approximate the set of potential degraded states as a finite set. In the context of oil and gas extraction, this finite set may correspond to a "healthy" state and one or more "degraded" states. In each case, the corresponding system behaviors are qualitatively different.

In practice, identification of degraded states is often carried out separately from control input optimization (e.g., [2]). Indeed, as the finite set of degraded states only yields approximate system behaviors, techniques such as modifier adaptation [3] are required in any case in order to account for plant-model mismatch in the set of system models. Moreover, the exploration required to find near-optimal control inputs complicates state identification.

Nevertheless, separating degraded state identification and production optimization has its own drawbacks. This is particularly clear when the degraded state corresponding, e.g., to a compressor, varies over time. As maintenance should be performed as soon as, for example, the compressor is not "healthy", performing identification after production optimization can lead to costly delays or conservative estimates of the time when the compressor transitions to a degraded state.

In this paper, we develop new algorithms for online production optimization and degraded state identification in a joint fashion. In particular, we consider the scenario where noisy observations of the steady state for the process dynamics are available. In this scenario, the general problem of control optimization and identification can be formulated as a partially observed Markov decision problem (POMDP). However, both observation and action spaces (corresponding to controls [1] and identified

⋆ Corresponding author José Matias.

V. Kungurtsev was supported by the OP VVV project CZ.02.1.01/0.0/0.0/16 019/0000765 "Research Center for Informatics". Jose Matias acknowledges financial support from the Norwegian Research Council, SUBPRO, grant number: 237893.

Email addresses: jose.o.a.matias@ntnu.no (José Matias), vyacheslav.kungurtsev@fel.cvut.cz (Vyacheslav Kungurtsev), malcom.egan@inria.fr (Malcolm Egan).

[1] Modifier Adaptation, introduced in Section 3, is a steady-state production optimization method, which allows the optimization on variables other than the system manipulated variables. For generalization purposes, these steady-state de-

degraded states) are uncountable, which means that obtaining solutions with guarantees is computationally intractable.

To resolve this challenge, we introduce a novel reduction in the action space by exploiting modifier adaptation. Instead of selecting a control from an uncountable set, our approach only requires optimization over a finite set of potential degraded states models. In order to improve model identification, we also introduce a regularization term consisting of a Jensen-Shannon divergence [4]. As such, depending on a weight trading off exploitation and exploration, the criterion for optimization of the control also accounts for the information that will be gained. Based on two case studies in the context of a chemical reactor and subsea oil and gas exploration, we show that our algorithms yield an economic performance comparable to existing methods [5] with improved identification of changes to degraded states.

2 Literature Review

2.1 Modifier Adaptation (MA)

Given a real-time optimization scheme with plant-model mismatch and noisy observations, modifier adaptation [3] provides a means of optimizing the input control and reaching plant optimality. MA iteratively refines the model using measurements and plant gradient estimates such that, under certain conditions, the computed inputs converge to a "true" cost-minimising value.

The need to estimate plant gradients is an important practical challenge of MA applications. Several variants have been proposed to deal with this issue. To name a few: [6] proposed a nested modifier-adaption scheme based on derivative-free algorithms that does not require plant gradient estimation; instead of correcting the model in all input directions, [7] suggested an MA variant that relies on directional derivatives; [8] and [9] trained Gaussian processes from past measurements for capturing plant-model mismatch, which avoided extra plant probing for gradient estimation.

Since computing accurate plant gradients in MA applications has been thoroughly investigated, we assume that we have a technique available to do so with reasonable accuracy. A second challenge in MA is that feasibility of the computed inputs is guaranteed only after convergence. [10] found sufficient conditions to ensure MA feasibility before convergence; however, the proposed approach requires the inference of the Lipschitz constants of the plant. As an alternative, [11] proposed that feasibility can be guaranteed by computing the MA iterations based on convex upper bounds of the plant. The drawback is the need to estimate second order information of the plant based on plant measurements. More recently, [12] used an aggregate model, which combined several models of the plant. This aggregate model provided an approximation for the convex upper bounds of the plant.

Along the same lines, [5,13] designed a MA variant for cases where several models of the plant are available but only one is used at a time. Here, the cost-minimising control input is computed while identifying the model that best fits the plant from an available set. Despite not focusing on guaranteeing feasibility, the method provided the framework for studies considering production optimization and system condition identification in a joint fashion.

2.2 Prognostics and health management: system condition identification

Maintenance has traditionally been performed only when the equipment breaks or the performance significantly decreases. In recent years, however, a shift towards prognostics and health management (PHM) is taking place [14]. In PHM, models are used for detecting whether an equipment has deviated from its normal operating condition and for predicting failures in advance. As a consequence, the focus shifts to mitigate or avoid faults altogether, not only react when a fault occurs [15].

A variety of PHM approaches can be found in the literature and they are classified according to the nature of the applied models [16]: model-based approaches are characterized by using a first principles model representation of the system (e.g. [17]); knowledge-based approaches are based on experience and heuristics, which are typically implemented as logic and IF/THEN commands (e.g. [18]); and data-driven approaches use statistical methods or machine learning for fault diagnosis, or to forecast the system degradation trends (e.g. [19–21]).

Independently of the chosen approach, PHM is currently carried out offline for assisting the the operator's decision making [22]. This presents a disadvantage due to the operator's limited reaction time, and the fact that his/her decision may not be optimal. Hence, initial efforts have been made to combine control and health prognostics with the development of fault-tolerant [23] and health-aware controllers [24]. The latter aims at providing means of optimizing the system economically while taking system degradation into account. However, the ability to identify the degraded state online is still a significant technical challenge.

2.3 Combining production optimization and system condition identification

If models for different degraded states are available (for example, for a healthy and degraded compressor), one alternative for system condition identification is to use sequential hypothesis testing [25]. It provides online algorithms to perform degraded state identification in the presence of noisy observations. Sequential hypothesis testing algorithms are obtained as solutions to a particular POMDP, where costs are defined in terms of the

cision variables are referred to, in this paper, as "controls".

probability of misidentification and the expected stopping time. As such, the control input and associated control costs are not considered.

Similarly, a number of other works have also considered variations on the identification problem. We point out in particular [26], where mixed integer nonlinear programming is used to find the best fit within a family of approximating functions. In the context of parameter estimation, Bayesian methods have been developed for function-valued observations (e.g., [27]), given that the structure of model equations is known. See [28], for applications to biological systems.

We remark that data-driven approaches have appeared recently in the literature in general. In one line of work, data is used in the identification process. Here, the input-output (IO) relationships of the system are identified based mainly on data. In [29], the authors used a neural network to capture the IO relationships of a complex industrial process, and then used the identified model to optimize the system. For speeding up and improving the network training step, the authors proposed a data dimensionality reduction algorithm based on t-distributed stochastic neighbor embedding methods. Despite presenting good results, pure data-based models like the neural network used in [29] can still benefit from using some process knowledge. For that end, [30] used available plant data to obtain a model structure based on graphlet representation taking into account specific process information such as the effect of raw material composition on product purity. Then, the authors applied the obtained model for optimizing production in ethylene plants. Our work is complementary in considering a situation wherein the control IO structure is known to be among one of a few set of possible models, a common but not universal scenario.

The joint identification and production optimization problem can also be formulated as a POMDP; albeit with an uncountable action space. For finite action spaces, there is a vast literature on optimal control of POMDPs beginning with the seminal work in [31]. Recently, there have also been investigations into optimal control of POMDPs with infinite action spaces [32]; however, generally applicable and tractable algorithms are not available in this case.

In the context of nonlinear dynamic systems, active adaptive control addresses the identification of operating modes from which a control can then be computed [33]. These techniques typically require more fidelity in the models, whereas we are interested in cases with significant plant-model mismatch noise.

As mentioned, the closest work on the joint identification and control input optimization problem addressed in this paper appears in [5,13]. While modifier adaptation is exploited to find the optimal control, identification is based on a heuristic of choosing the least norm defined on the modifiers across the set of models. In contrast, our work formalizes the problem as a POMDP and provides procedures for obtaining an approximate solution.

3 Problem Formulation

Consider a plant with a steady-state map at time k, $\tilde{\boldsymbol{y}}_k : \mathbb{R}^n \mapsto \mathbb{R}^m$ in response to a control input $\boldsymbol{u} \in \mathbb{R}^n$. If the true plant steady-state as a function of the control input is perfectly known, then the main challenge is to optimize $\boldsymbol{u}$ in order to minimize economic costs. Formally, this corresponds to the following *steady-state* economic optimization problem at time k,

$$\begin{aligned} \boldsymbol{u}_k = \underset{\boldsymbol{u}\in\mathbb{U}\subseteq\mathbb{R}^n}{\operatorname{argmin}} \; & \phi(\boldsymbol{u}, \tilde{\boldsymbol{y}}_k(\boldsymbol{u})), \\ \text{s.t. } & \boldsymbol{g}(\boldsymbol{u}, \tilde{\boldsymbol{y}}_k(\boldsymbol{u})) \leq \boldsymbol{0}, \end{aligned} \tag{1}$$

where $\mathbb{U} \subseteq \mathbb{R}^n$ is a the input constraint set, $\phi : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}_{\leq 0}$ is an economic cost function, and $\boldsymbol{g} : \mathbb{R}^n \times \mathbb{R}^m$ is the constraint function.

In practice, the true steady-state $\tilde{\boldsymbol{y}}_k(\boldsymbol{u})$ at time k is not perfectly known and is instead approximated by a model m, yielding the steady state approximation $\boldsymbol{y}(\boldsymbol{u}; m)$. It is therefore necessary to perform model identification in addition to control input optimization. Moreover, given a control input $\boldsymbol{u}$, only noisy observations of $\tilde{\boldsymbol{y}}_k(\boldsymbol{u})$ and the gradient $\frac{\partial \tilde{\boldsymbol{y}}_k(u)}{\partial \boldsymbol{u}}$ are available. In particular, we assume that for a sequence of control inputs $\{\boldsymbol{u}_k\}_{k=1}^K$, the observations are given by

$$\begin{aligned} \boldsymbol{y}_k^p(\boldsymbol{u}_k) &= \tilde{\boldsymbol{y}}_k(\boldsymbol{u}_k) + \boldsymbol{z}_k, \; k = 0, \ldots, K, \\ \frac{\partial \boldsymbol{y}_k^p}{\partial \boldsymbol{u}}(\boldsymbol{u}_k) &= \frac{\partial \tilde{\boldsymbol{y}}_k}{\partial \boldsymbol{u}}(\boldsymbol{u}_k) + \boldsymbol{w}_k, \; k = 0, \ldots, K, \end{aligned} \tag{2}$$

where $\boldsymbol{z}_k \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{\Sigma})$ are independent Gaussian random vectors on $\mathbb{R}^m$ and $\boldsymbol{w}_k \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{\Sigma}_g)$ are independent Gaussian random matrices on $\mathbb{R}^{m\times n}$.

A natural framework for optimizing the control input given the true model is unknown is to view the system as a POMDP. Standard references on reinforcement learning (RL) are [34,35] and on POMDPs [36]. A POMDP formulation of the joint model identification and control optimization problem is defined as follows:

(i) *State Space:* The state space is a set of functions $\mathcal{V}$. In particular, the state at time k is the function $\tilde{\boldsymbol{y}}_k : \mathbb{U} \to \mathbb{R}^m$ in $\mathcal{V}$.
(ii) *Action Space:* The action space corresponds to the set of controls $\mathbb{U} \subset \mathbb{R}^n$ and the choice of estimated model $\hat{\boldsymbol{y}}_k \in \mathcal{V}$. The action at time k is denoted by $(\boldsymbol{u}_k, \hat{\boldsymbol{y}}_k) \in \mathbb{U} \times \mathcal{V}$.
(iii) *Transition Probability:* The transition probability is given by the kernel $\mathbb{P}(\tilde{\boldsymbol{y}}_k(\cdot) \in \mathcal{A} | \tilde{\boldsymbol{y}}_{k-1}, \boldsymbol{u}_k), \mathcal{A} \subset \mathcal{V}, \; k = 1, \ldots, n$.
(iv) *Observations:* The observations given a control input u_k, are given by Eq. (2).

(v) *Reward:* Let $\beta > 0$. The expected reward is given by

$$R_K = -\mathbb{E}\left[\sum_{k=0}^{K} \phi(\boldsymbol{u}_k, \hat{\boldsymbol{y}}_k(\boldsymbol{u}_k)) - \beta \mathbf{1}_{\{\tilde{\boldsymbol{y}}_k(\boldsymbol{u})=\hat{\boldsymbol{y}}_k(\boldsymbol{u}),\ \forall \boldsymbol{u}\in\mathbb{U}\}}\right]. \tag{3}$$

where β represents a reward for correct model identification. Note that the problem is a POMDP rather than a Markov decision process due to the uncertain observations; i.e., the true process dynamics, $\tilde{\boldsymbol{y}}_k$, are not available. Moreover, the action space is uncountable and the state space is a function space. Obtaining an optimal policy for this POMDP would also require estimation of a probability kernel on an uncountable space and the reward profile for every possible map from the observed state $\boldsymbol{y}^p_{k-1}$ to subsequent control $\boldsymbol{u}_k$. As such, solving this POMDP is intractable, and we seek approximate solutions.

In order to proceed, we assume that it is desirable to identify the best model m_k at each time k drawn from a discrete set $m_k \in \{1, \ldots, M\}$, and that the true model remains fixed over a sufficiently long period of time. Each model m_k corresponds to a different steady-state function $\tilde{\boldsymbol{y}}_k(\cdot; m_k)$. Although none of the models in $\{1, \ldots, M\}$ may correspond to the true plant steady-state, they may represent qualitatively different behavior; e.g., "healthy" or "degraded".

The problem is then to optimize the control inputs $\boldsymbol{u}_k$ in order to maximize the expected reward in Eq. (3), while exploiting the structure of the approximate models $\{1, \ldots, M\}$. In the following section, we overview a key technique, known as modifier adaptation, that will be used to account for imperfections in the models.

4 Modifier Adaptation and a Baseline Algorithm

A key technique in order to optimize controls with uncertain knowledge of the plant steady-state is modifier adaptation. It has also been observed in [5] that modifier adaptation can be utilized to perform model identification. In this section, we overview modifier adaptation and the approach in [5]. In Section 5, we will develop new model identification and control optimization algorithms based on modifier adaptation [2], for which [5] will form a baseline.

4.1 Modifier Adaptation

Consider the case where initially one has knowledge of an imperfect model m, corresponding to $\boldsymbol{y}(\cdot; m) : \mathbb{R}^n \to \mathbb{R}^m$ for the plant process. The idea of modifier adaptation is to apply correction terms to iteratively adapt the model based on observations of the plant output. In particular, at time k, define the modifiers $\boldsymbol{\epsilon}_k(m) \in \mathbb{R}^m$ and $\boldsymbol{\lambda}_k(m) \in \mathbb{R}^{m\times n}$ by

$$\begin{aligned} \boldsymbol{\epsilon}_k(m) &= \boldsymbol{y}^p_k(\boldsymbol{u}_k) - \boldsymbol{y}(\boldsymbol{u}_k; m), \\ \boldsymbol{\lambda}_k(m) &= \left(\frac{\partial \boldsymbol{y}^p_k}{\partial \boldsymbol{u}}(\boldsymbol{u}_k) - \frac{\partial \boldsymbol{y}}{\partial \boldsymbol{u}}(\boldsymbol{u}_k; m)\right) \end{aligned} \tag{4}$$

In Eq. (4), the zero-th order modifiers $\boldsymbol{\epsilon}_k(m)$ correspond to bias terms between the predicted values of the model and the plant measurements. The first order modifiers $\boldsymbol{\lambda}_k(m)$ represent the difference between the plant gradients and the gradients predicted by the model. Roughly speaking, the modifiers are a measure of the difference between the shape of the model and the plant process.

Given an initial model m, the updated or modified model after the k-th observation arising from a control $\boldsymbol{u}_k$ is then given by

$$\boldsymbol{y}_{ad,k}(\boldsymbol{u}; m) := \boldsymbol{y}(\boldsymbol{u}; m) + \boldsymbol{\epsilon}_k + \boldsymbol{\lambda}_k(\boldsymbol{u} - \boldsymbol{u}_k). \tag{5}$$

The control input is then optimized by solving

$$\begin{aligned} \boldsymbol{u}^{MA}_{k+1}(m) = \underset{\boldsymbol{u}\in\mathbb{U}\subseteq\mathbb{R}^n}{\operatorname{argmin}}\ & \phi(\boldsymbol{u}, \boldsymbol{y}_{ad,k}(\boldsymbol{u}; m)), \\ \text{s.t. } & \boldsymbol{g}(\boldsymbol{u}, \boldsymbol{y}_{ad,k}(\boldsymbol{u}; m)) \leq \mathbf{0}. \end{aligned} \tag{6}$$

If the model satisfies certain regularity conditions near the vicinity of the plant optimum, modifier adaptation guarantees that u_∞ matches the plant optimum [3].

4.2 A Baseline Algorithm

Although modifier adaptation is traditionally used for control input optimization, the modifiers contain *valuable information about the relationship between a model and the plant.* This observation has been exploited in [5,13] in order to perform model identification. In particular, the modifiers can be used for discriminating between different models, which can be helpful in cases that there are unknown features of the process that need to be modeled. As such, the modifiers can be used as model quality indicators.

To this end, for a given initial model m, define the total modifier as

$$\psi_k(m) := \|\boldsymbol{\epsilon}_k(m)\|_F + \|\boldsymbol{\lambda}_k(m)\|_F \tag{7}$$

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. As such, the total modifier $\psi_k(m)$ quantifies how much

[2] Note that we use a variant of the traditional modifier adaptation approach called output modifier adaptation. However, we use the name modifier adaptation for simplicity.

a model was adapted. As steady-state process data—corresponding to noisy measurements of $\tilde{\boldsymbol{y}}_k$ and gradient estimates $\frac{\partial \tilde{\boldsymbol{y}}_k}{\partial \boldsymbol{u}}$—is obtained during operation, an approximation of the plant steady-state can be obtained by comparing $\psi_k(m)$ for each model. In particular, a smaller value of $\psi_k(m)$ can indicate that the model is a better approximation of the plant for production optimization purposes.

Adopting this strategy yields the existing approach to joint identification and control [5], where the following discrete optimization problem is solved:

$$\begin{aligned} m_k^\star = \underset{m\in\{1,\dots,M\}}{\operatorname{argmin}} \quad & \psi_k(m) := \|\boldsymbol{\epsilon}_k(m)\|_F + \|\boldsymbol{\lambda}_k(m)\|_F \\ \text{s.t. } & \boldsymbol{\epsilon}_k(m) = \boldsymbol{y}_k^p(\boldsymbol{u}_k) - \boldsymbol{y}(\boldsymbol{u}_k; m), \\ & \boldsymbol{\lambda}_k(m) = \left(\frac{\partial \boldsymbol{y}_k^p(\boldsymbol{u}_k)}{\partial \boldsymbol{u}} - \frac{\partial \boldsymbol{y}(\boldsymbol{u}_k; m)}{\partial \boldsymbol{u}}\right) \end{aligned} \tag{8}$$

At time k, the control input is then obtained by applying modifier adaptation using the model identified in Eq. (8). The complete algorithm in [5] is summarized in Alg. 1 as follows.

Algorithm 1. Joint Identification and Control via the Total Modifier [5]

1: Measure $\boldsymbol{y}_k^p(\boldsymbol{u}_k)$ and estimate $\frac{\partial \boldsymbol{y}_k^p(\boldsymbol{u}_k)}{\partial \boldsymbol{u}}$ via the observations as in Eq. (2).
2: Select a model $m^* \in \{1,\dots,M\}$ such that the total modifier $\psi_k(m)$ is minimized via Eq. (8).
3: Use the model m^* and corresponding modifiers $\boldsymbol{\epsilon}_k(m^*)$, $\boldsymbol{\lambda}_k(m^*)$ to compute a new optimal operating point $\boldsymbol{u}_{k+1}^{MA}$ via the modifier adaptation update in Eq. (6).
4: Repeat from 1.

5 Proposed Algorithms

A drawback of the baseline Alg. 1 is that only information from a single observation is used to identify the model. On the other hand, modifier adaptation requires a number of iterations, yielding multiple observations, which can improve model identification accuracy. In this section, we develop an alternative approach which exploits the information available from all modifier adaptation iterations.

In our approach, we first consider a new POMDP which reduces the action space of the POMDP in Sec. 3 by exploiting modifier adaptation:

(i) *State Space:* The state at time k is $m_k^* \in \{1,\dots,M\}$, the set of models.
(ii) *Action Space:* The action space at time k is indexed by the set of models $\{1,\dots,M\}$, where action $m \in \{1,\dots,M\}$ corresponds to the modifier adaptation iteration in Eq. (5) and Eq. (6) based on model m, leading to a control $\boldsymbol{u}_k^{MA}(m)$.
(iii) *Transition Probability:* The transition probability is given by the kernel $\mathbb{P}(m_k|m_{k-1}, u_k), m_k \in \{1,\dots,M\},\ k = 0,\dots,n$.
(iv) *Observations:* The observations given a control input u_k, are given by Eq. (2).
(v) *Reward:* Let $\beta > 0$. The expected reward is given by

$$\begin{aligned} & R_K \\ & = -\mathbb{E}\left[\sum_{k=0}^{K} \phi(\boldsymbol{u}_k, \boldsymbol{y}_k(\boldsymbol{u}_k; m_k^*) - \beta \mathbf{1}_{m_k = m_k^*}\right]. \end{aligned} \tag{9}$$

Observe that instead of selecting any control in $\mathbb{U}$, the control is restricted to the finite set defined by the modifier adaptation iterations corresponding to each model $m_k \in \{1,\dots,M\}$. Moreover, the state space is approximated by the behavior modeled by each of the models $m_k \in \{1,\dots,M\}$.

Estimation of the transition probability requires additional information about the degradation processes in the system. As such, we adopt a myopic approach, where the control and model identification do not account for future states of the system or observations. Nevertheless, the POMDP structure of the problem implies a useful statistic summarizing all available information is given by the belief [31], which is a probability distribution over the set of models $m \in \{1,\dots,M\}$. In particular, the belief at time k is given by

$$\rho_k(m) = \frac{p(f_k|m, \boldsymbol{u}_k)\rho_{k-1}(m)}{\sum_{\overline{m}\in\{1,\dots,M\}} p(f_k|\overline{m}, \boldsymbol{u}_k)\rho_{k-1}(\overline{m})}, \quad k > 0 \tag{10}$$

where $f_k = (\boldsymbol{y}_k^p(\boldsymbol{u}_k), \frac{\partial \boldsymbol{y}_k^p}{\partial \boldsymbol{u}})$ is the plant observation available at time k and $p(f_k|m, \boldsymbol{u}_k)$ is the probability density of the observations under model m and control $\boldsymbol{u}_k$. Note that $p(f_k|m, \boldsymbol{u}_k)$ is a Gaussian density under the observation model in Eq. (2). The belief at time $k = 0$, denoted by ρ_0, corresponds to a prior belief.

In the remainder of this section, we develop two algorithms to perform joint production optimization and model identification by incorporating the belief.

5.1 Expected Cost Minimization

At time k, noisy plant outputs and gradients for the inputs $\boldsymbol{u}_0,\dots,\boldsymbol{u}_k$ have been observed. As such, M modified models $\boldsymbol{y}_{ad,k}(\boldsymbol{u}; m)$ defined in Eq. (5) can be computed. In addition, the belief ρ_k is also available. Thus, the model estimate m_k^* and the control input $\boldsymbol{u}_{k+1}$ can exploit this information.

In particular, it is desirable to minimize the expected cost under the belief ρ_k, given by

$$V_k = \sum_{m=1}^{M} \rho_k(m)\phi(\boldsymbol{u}_{k+1}^{MA}, \boldsymbol{y}_{ad,k}(\boldsymbol{u}_{k+1}^{MA}; m)), \tag{11}$$

where $\boldsymbol{u}_{k+1}^{MA}$ is given in Eq. (6). The estimated model and control input are then given by

$$\begin{aligned} m_k^* &= \arg \min_{m \in \{1,\dots,M\}} V_k \\ \boldsymbol{u}_{k+1} &= \boldsymbol{u}_{k+1}^{MA}(m_k^*). \end{aligned} \tag{12}$$

For implementing this algorithm, we replace Steps 2 and 3 of Alg. 1 by Eq. (12).

5.2 Jensen-Shannon Divergence Regularization

While the expected cost minimization algorithm exploits current beliefs on the state of the system, it does not incorporate the information that may be gained from exploring other models. One measure of this information, which has been used in optimizing the choice of sensors in sensor networks [4], is the Jensen-Shannon divergence defined as

$$\begin{aligned} &JS(\rho_k, f(\cdot|1,\boldsymbol{u}_k),\dots,f(\cdot|M,\boldsymbol{u}_k)) \\ &= \sum_{m=1}^{M} \rho_k(m) D\left(f(\cdot|m,\boldsymbol{u}_k) || \sum_{m=1}^{M} \rho_k(m) f(\cdot|m,\boldsymbol{u}_k) \right), \end{aligned} \tag{13}$$

where $D(\cdot||\cdot)$ is the Kullback-Leiber divergence, which for two probability density functions q_1, q_2 on $\mathbb{R}^l$ is given by

$$D(q_1||q_2) = \int_{\mathbb{R}^l} \int_{\mathbb{R}^l} q_1(\mathbf{x}) \log \frac{q_1(\mathbf{x})}{q_2(\mathbf{x})} \mathrm{d}\mathbf{x}. \tag{14}$$

Intuitively, the Jensen-Shannon divergence is a measure of the distance between each of the probability density functions $f(\cdot|m,\boldsymbol{u}_k)$. As such, selecting a control that maximizes the Jensen-Shannon divergence will yield observations that better distinguish differences between each model. This can be interpreted as a form of exploration.

Accounting for the Jensen-Shannon divergence, it is desirable to minimize a regularized expected cost under the belief ρ_k, given by

$$\begin{aligned} V_k^{JS}(\boldsymbol{u},\alpha) = \alpha &\sum_{m=1}^{M} \rho_k(m) \phi(\boldsymbol{u}, \boldsymbol{y}_{ad,k}(\boldsymbol{u};m)) \\ &- (1-\alpha) JS(\rho_k, f(\cdot|1,\boldsymbol{u}),\dots,f(\cdot|M,\boldsymbol{u})), \end{aligned} \tag{15}$$

where $0 \leq \alpha \leq 1$ is a weight which yields a trade-off between exploitation (i.e., cost minimization) and exploration (i.e., Jensen-Shannon divergence maximization). We remark that the numerical computation of the Jensen-Shannon divergence requires the evaluation of a Kullback-Leibler divergence between a Gaussian probability density function and a Gaussian mixture. In order to perform this computation, the approximations developed in [37, Eq. (2)] can be exploited.

In order to obtain the control $\boldsymbol{u}_{k+1}$, let

$$\begin{aligned} &\boldsymbol{u}_{k+1}(\alpha) \\ &= \arg\min_{\boldsymbol{u} \in \mathbb{U}} V_k^{JS}(\boldsymbol{u},\alpha), \\ &\quad \text{s.t. } \boldsymbol{g}(\boldsymbol{u}, \boldsymbol{y}_{ad,k}(\boldsymbol{u};m)) \leq 0,\ m = 1,\dots,M. \end{aligned}$$

The control input is then given by

$$\boldsymbol{u}_{k+1} = \arg\min_{\alpha \in [0,1]} \boldsymbol{u}_{k+1}(\alpha). \tag{16}$$

Note that since $\phi(\cdot,\cdot)$ is non-positive, it follows that the α corresponding to $\boldsymbol{u}_{k+1}$ in general satisfies $\alpha > 0$. As such, the algorithm emphasizes exploration when the cost ϕ is low.

On the other hand, the estimated model can be obtained via

$$m_k^* = \arg \max_{m \in \{1,\dots,M\}} \rho_k(m). \tag{17}$$

For implementing this algorithm, we replace Steps 2 and 3 of Alg. 1 by Eq. (16) and (17), respectively.

5.3 Remark on formal guarantees and model convergence

The main advantage of combining modifier adaptation with model selection in the proposed algorithm lies in the MA capacity to find the optimal inputs of the real process upon convergence. MA can deal with structural plant-model mismatch if the modifiers can be measured or correctly estimated, and the model is adequate (i.e. it is locally convex in the vicinity of the plant optimal inputs [3]). The proposed algorithms will maintain this interesting MA property if all models in the available set are adequate in this sense. Since the plant optimum is unknown beforehand, this could only be guaranteed a priori by enforcing that all models are convex over the whole feasible input space or by convexifying them (see, e.g. [12]). Despite no proof of guaranteed convergence, the case studies in the following section illustrate the benefits of our algorithms when an appropriated model set is chosen.

Considering the underlying framework of a POMDP is intractable, we cannot provide formal theoretical guarantees. This presents the theoretical possibility of the model updates ending up in a limit cycle. As discussed in [13], unless the models in the available set are distinguishable (i.e. can be uniquely identified given a set of plant measurements and gradient estimation estimates), the model selection conclusions drawn at iteration k can be different than those of iteration $k+1$. Especially if only local information is used. This problem is partially

addressed by the belief update (Eq. (10)), which "averages out" local disturbances and noise realizations from the model structure selection, and in general we expect to see more model chattering in classical MA than the algorithms presented here, as such. Note that, for the specific application of this paper of identification of the system degradation state, the models are quite distinct as the dynamics of a degraded and healthy plant are significantly separated, so model chattering is highly unlikely.

6 Numerical Results

The three algorithms are applied to two case studies, a simplified continuous stirred-tankreactor (CSTR) and a Subsea Gas Compression Station forsubsea oil wells. They are implemented according to the block diagrams in Fig. 1. The codes for both case studies are available at `https://github.com/JoseMatias14/SimultaneousIdentificationOptimization`.

6.1 Case Study 1 - simplified continuous stirred-tank reactor (CSTR)

Here, the approaches of Section 5 are implemented on a CSTR, where reactants A and B generate product C. The case study is based on [13].

A simplified flowsheet of the process is shown in Fig. 2. The reactor is fed with a stream containing only A (F_1) and a second stream F_2. The composition of the latter is partially unknown. It is either purely B, or mostly B with some impurity D. The flowrate F_1 can be freely manipulated while the F_2 flowrate is determined by the upstream process. The outlet concentrations of B, C and D are measured. The reaction set is unknown *a priori*.

Given this modelling uncertainty scenario, the following model candidates are considered:

M_1) Reaction set: $\mathrm{A} + \mathrm{B} \xrightarrow{k_1} \mathrm{C}$ and $2\,\mathrm{B} \xrightarrow{k_2} \mathrm{D}$
F_2 contains only B.

M_2) Reaction set: $\mathrm{A} + \mathrm{B} \xrightarrow{k_3} 2\,\mathrm{C}$ and $\mathrm{C} + 2\,\mathrm{B} \xrightarrow{k_4} \mathrm{D}$
F_2 contains only B

M_3) Reaction set: $\mathrm{A} + \mathrm{B} \xrightarrow{k_1} \mathrm{C}$
D enters the system as an impurity in F_2.

For illustration purposes, we increase the number of rival model structures by assuming that different parameter values represent different models. The available structures are shown in Table 1.

Table 1
Available model structures

	M_{11}	M_{12}	M_{13}	M_{21}	M_{22}	M_{23}	M_{31}	M_{32}	M_{33}
k_1	0.75	0.72	0.75	-	-	-	0.13	0.14	0.12
k_2	1.20	1.20	1.50	-	-	-	-	-	-
k_3	-	-	-	0.25	0.20	0.28	-	-	-
k_4	-	-	-	0.17	0.20	0.20	-	-	-

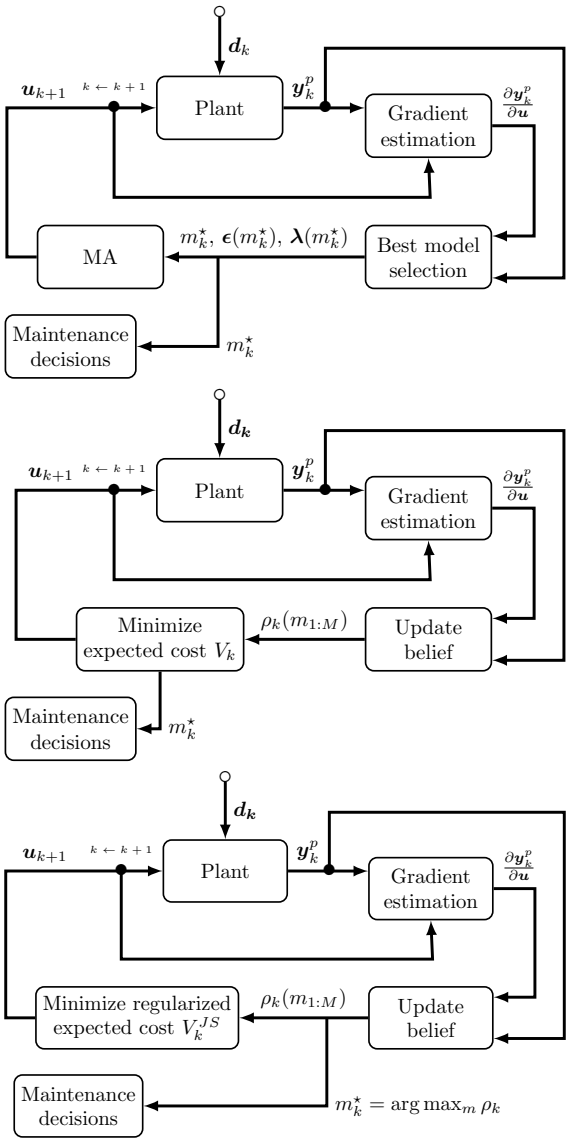

Fig. 1. Block diagrams from top to bottom: Baseline approach, Expected Cost Minimization (ECM), and Jensen-Shannon Divergence Regularization (JSDR). The plant is affected by disturbances $\boldsymbol{d}_k$that can affect both the optimal input values as well as the system degradation condition.

6.1.1 Simulation setup

The operational goal is to maximize the concentration of C at the reactor outlet. In order to achieve the best performance, we can manipulate the inlet flow of reactant F_1. The reactor temperature is considered constant in the simulation and we do not consider any operational constraints.

The plant is represented by the structure M_{13}. i.e. inlet F_2 contains only B, and the true reaction set is $\mathrm{A} + \mathrm{B} \xrightarrow{k_1} \mathrm{C}$ and $2\,\mathrm{B} \xrightarrow{k_2} \mathrm{D}$ with $k_1 = 0.75$ [$\mathrm{L}^3/(\mathrm{mol}^3$ min)] and $k_2 = 0.75$ [$\mathrm{L}^3/(\mathrm{mol}^3$ min)]. We consider that the system measurements (outlet concentration of B, C

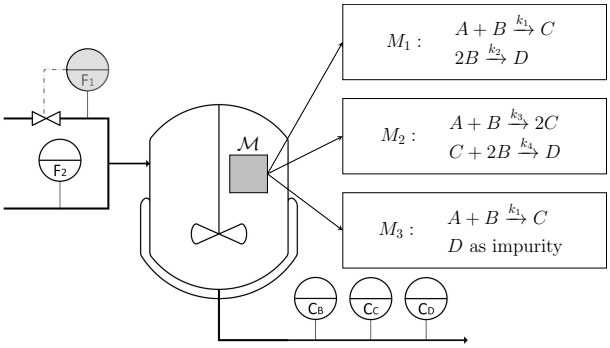

Fig. 2. CSTR, where reactants A and B generate product C. There are three available kinetic models for the reaction.

Table 2
Initial condition and parameters for the CSTR simulation.

Description	Symbol	Value	Unit
Plant initial condition	$x_0 = \begin{bmatrix} C_{A,0} \\ C_{B,0} \\ C_{C,0} \\ C_{A,0} \end{bmatrix}$	$\begin{bmatrix} 0.7385 \\ 0.0231 \\ 0.4922 \\ 0.0308 \end{bmatrix}$	[mol/L]
F_1 initial flow rate	$F_{1,0}$	8	[L/min]
Feed concentration of A	$C_{A,in}$	2	[mol/L]
Feed concentration of B	$C_{B,in}$	1.5	[mol/L]
F_2 flow rate	F_2	5	[L/min]

and D) are corrupted by Gaussian noise with mean 0 and variance 0.025, which leads to very noisy measurement since the concentrations are in the range of 0.05 mol/L to 0.5 mol/L. In turn, we assume that we cannot measure the gradients directly, and use plant experiments with central finite differences for estimating them. For further details on the system model, please refer to [13]. The initial condition and additional model parameters are summarized in Table 2.

The simulations are carried out in the MATLAB R2019b programming environment (Mathworks Inc., Natick, MA, USA) using the CasADi v3.4.5 extension [38] for algorithmic differentiation. The NLP problem is solved using IPOPT version 3.12.2 [39].

6.1.2 Results

For maximizing the production of C while identifying the best available model structure, we implement:

(1) the approach of [5] described in Alg. 1
(2) the Expected Cost Minimization (ECM) algorithm
(3) Jensen-Shannon Divergence Regularization (JSDR)

The results are presented in Fig. 3, 4, and 5, respectively. All figures are set up in the same format. The top plot shows the model structure with the highest probability of being the "true" one. Then, we show the probability associated with each model structure of Table 1. Next, we present the value of the control F_1 computed by the approach. In this plot, we also show the true optimum values as a dotted line. For the ECM and JSDR, we also

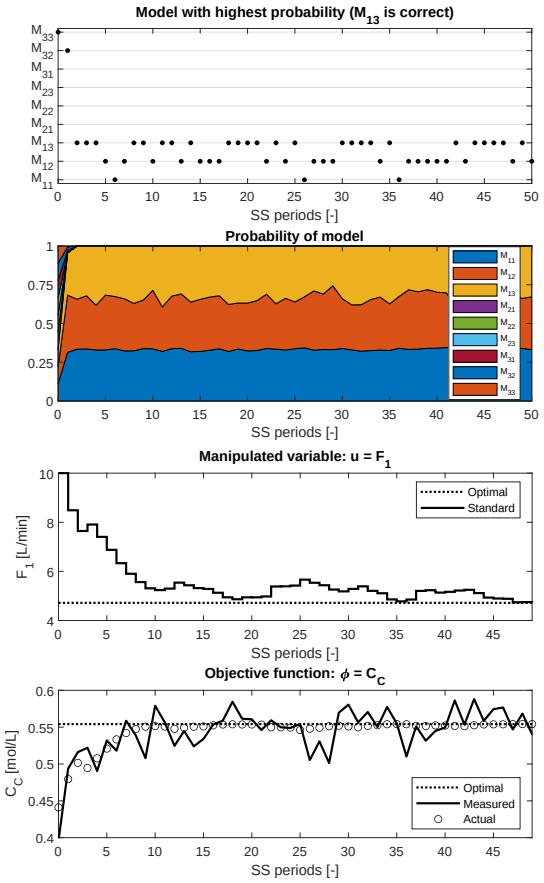

Fig. 3. Results of the approach of [5] (see Alg. 1) applied to Case Study 1.

plot the baseline solution for comparison. The bottom plot shows the measured concentration of C as a full line and the true value (without noise) as circles. We also indicate the optimal concentration of C as a dotted line.

Baseline approach: We see that Alg. 1 easily discards the model structures M_2 and M_3. However, the remaining models M_{11}, M_{12} and M_{13} are indistinguishable with this approach. The causes are: (i) they yield similar predictions of $\boldsymbol{y}$ and $\frac{\partial \boldsymbol{y}}{\partial u}$ for the same value of u; and (ii) we only use local noisy information to identify the model. Regarding the production optimization, we note that F_1 oscillates due to the noisy measurement, but it converges to the plant optimum even when the algorithm chooses an incorrect model. This result is a direct consequence of using Modifier Adaptation for optimizing the plant. If we used a production optimization method that could not cope with plant-model mismatch, the system would very likely converge to a sub-optimal point. We see that, between iterations $20 - 35$ and $37 - 45$, the trajectory deviates from the optimal point due to noisy measurements that affect the gradient estimation. Nonetheless, the optimization strategy is able to bring the plant back to the optimum in both cases.

Expected Cost Minimization: for implementing this

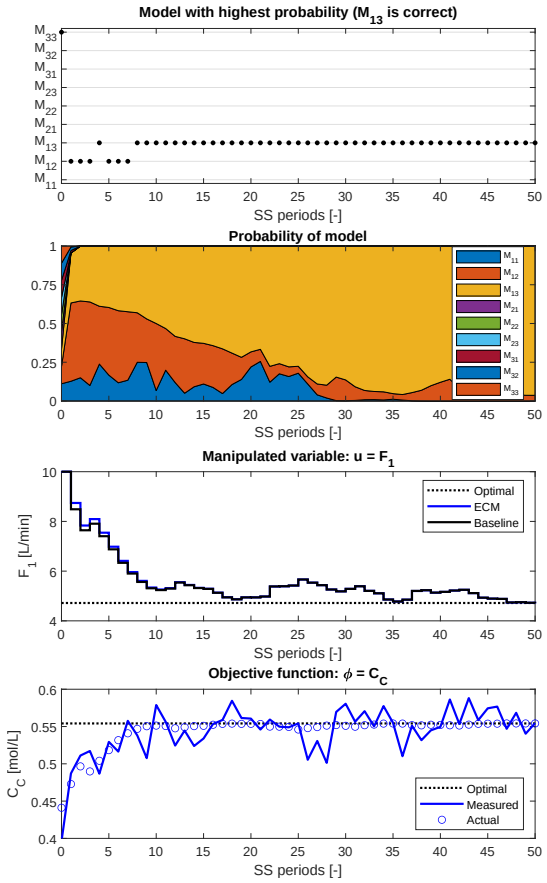

Fig. 4. Results of the Expected Cost Minimization applied to Case Study 1.

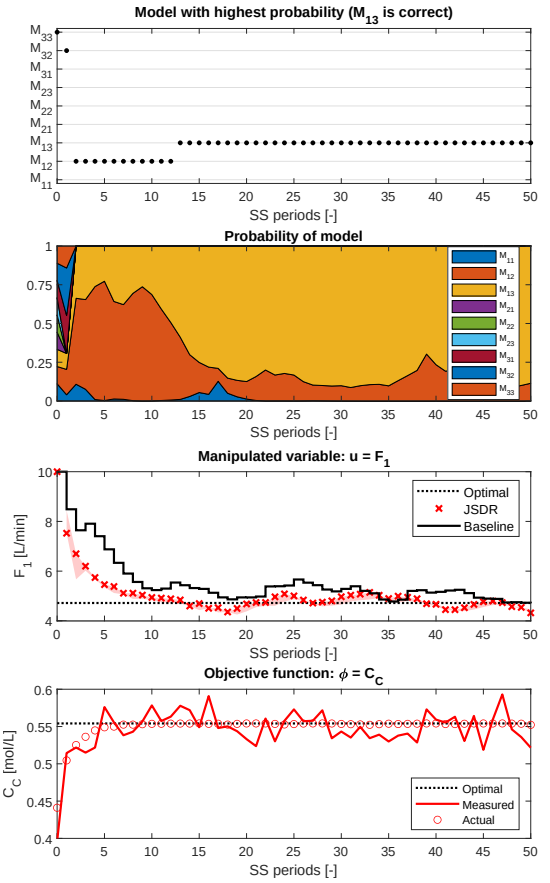

Fig. 5. Results of the Jensen-Shannon Divergence Regularization approach applied to Case Study 1.

approach, we need to compute the belief vector ρ_k for the Bayesian update in Eq. (10). In order to do so, we need to determine the probability density function of the estimates $p(f_k|m, u_k)$, which can be challenging in real implementation. Here, we assume they follow a Gaussian distribution. To calculate the covariance of both measurements and gradients estimates, we record data from 100 genuine independent observations. For ρ_0, we used a non-informative prior $\rho_0^{M_{11}} = \ldots = \rho_0^{M_{33}} = 1/9$.

By using a Bayesian updating strategy, we can properly identify the best model structure after a certain number of iterations (top plot of Fig. 4). ECM alternates between structures M_{11}, M_{12}, and M_{13} in the beginning of the simulation. This is due to a lack of information about the underlying process. Regarding the production optimization results (third plot), the computed F_1 does not deviate significantly from the baseline case. The main reason is that we "adapt" the selected model based on the plant information using the modifiers according to Eq. (5). Hence, the chosen model structure contribution is mostly limited to the second and higher order information. Since there is little mismatch among the structures M_{11}, M_{12}, and M_{13}, the optimal values of F_1 computed using these structures are, thus, similar.

Jensen-Shannon Divergence Regularization: We start the analysis of the performance of this approach by considering the third plot. The value of F_1 is computed based on a distribution over α, which is represented by the red region. The upper bound corresponds to $\alpha = 1$, favoring an optimization step (see Eq. (16)). In turn, the lower bound is $\alpha = 0$, which aims to solely optimize the information for selecting the model structure. We see that the first input is chosen in a region where the model predictions differ the most ($\alpha \to 0$). Consequently, after iteration 2, we discard all models with the wrong structure from the available set (probability $\to 0$ in the second plot). Next, JSDR favors the "production optimization steps" ($\alpha \to 1$). The reason is that the most likely models, M_{12} and M_{13}, are very similar. Thus, exploring the plant to discriminate between them is not advantageous. Note that, in this case study, the first input moves are very aggressive in order to differentiate between the model structures, which incidentally improves the optimal value tracking. However, this cannot be generalized.

6.1.3 Comparison between strategies

By using the ECM and the JSDR, we can significantly improve the model selection without affecting the pro-

duction optimization performance. This is despite the presence of noisy measurements and with almost equivalent model candidates in the available set.

The distinguishability of the models in the available set influences the identification step. The available model structures should be associated with a specific phenomenon and/or assumption (e.g. different reaction sets). Thus, it is expected that with a given sequence of measurements $\boldsymbol{y}_p$ and gradient estimates $\frac{\partial \boldsymbol{y}_p}{\partial \boldsymbol{u}}$, a unique model structure can be selected. In practice, however, there is a possibility of model chattering occurring due to noisy measurements and gradient estimation errors (see Fig. 3). This highlights the importance of exploring the feasible input space in order to improve the selection between rival model structures, which is done by ECM and JSDR approaches.

We remark that even for ECM and JSDR, the appearance of model chattering may arise when there is not a single model associated with a much stronger belief; i.e., a standout model that best fits recent observations. This only arises when the models are close near the control maximising the expected cost (which is approximately found via MA based on any of the models). However, in the applications we are considering, this scenario should not arise. As a consequence, while we cannot rule out model chattering, we believe that it is not a critical issue and highlight that significant chattering did not appear in any of the case studies that were considered.

One challenge for implementing both approaches is determining the probability density function for the Bayesian update. If a Gaussian probability density function is assumed, the challenge becomes calculating the gradient estimate covariance. Obtaining data from genuine independent experiments can be difficult in practice. Note, however, that it is not necessary to use the gradients for model discrimination, but it improves the distinguishability between competing model structures [13].

Finally, solving Eq. (16) online using a large scale model when several structures are available is not a trivial task. If the Gaussian approximation cannot be applied, the resulting expression for the Kullback Leibler divergence can become intractable for NLP solvers such as IPOPT. In this case, different optimization methods can be used, such as Bayesian optimization, Monte Carlo Optimization, or Genetic algorithms, which, however, suffer from slower computational performance to obtain a solution.

6.2 Case Study 2 - Subsea Gas Compression Station for subsea oil wells

Here, we show a possible application of the proposed approaches to equipment condition monitoring. Subsea boosting stations are important in production fields where the reservoir natural pressure is not enough to drive the extracted fluids from the seafloor to the sea level facilities. Additionally, by installing compression stations, the production can be transported over greater distances, and separation steps can be moved onshore. A simplified flowsheet is shown in Fig. 6.

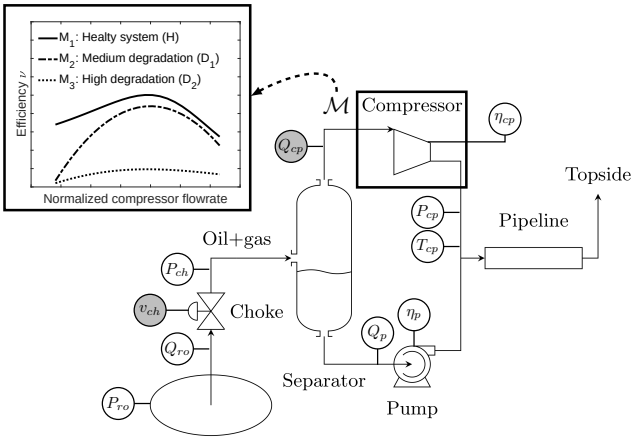

Fig. 6. Subsea Gas Compression Station. The controls and measurements are indicated. Adapted from [1].

There are different configurations of boosting stations. The case study deals with a single-phase compression/pumping, where the oil and gas mixture is divided in a separator. Then, a pump is used for increasing the liquid pressure, and a wet gas compressor for the gas pressure. A wet-gas compressor is necessary because of the liquid carry-over due to separation inefficiency (gas-volume-fraction from 0.95 to 1). After the boosting phase, the gas and liquid are recombined and sent to the top facilities through a pipeline. A detailed system description is given in [1].

The modelling uncertainty considered in this case study is related to the compressor degradation. We assume that its efficiency ν decreases with time. Since the compressor is complex with lots of moving parts, it is prone to break faster than the other system components and it is defined as the system's critical piece of equipment. The compressor states are modeled by one "healthy state" (H) and two "degraded states" (D_1 and D_2), in which the second one represents a heavily damaged compressor and the first an intermediate stage of degradation. The resulting three efficiency curves of the compressor are shown Fig. 6.

We assume ν remains unchanged for given periods and, between them, the efficiency degrades linearly like in Fig. 7. For example, we compute ν_{plant} as:

$$\nu_{plant} = (1-d)\nu_H + d\nu_{D_1} \tag{18}$$

where $d = (k-50)/(100-50)$ and k is the SS iteration number). Note that in this intermediary state, there is no "true" model in the available set.

6.2.1 Simulation set-up

We want to maximize the ratio between compressor flowrate and power ($\phi = Q_{comp}/P_{comp}$). Since it repre-

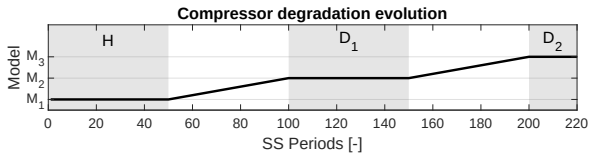

Fig. 7. Compressor degradation.

sents the transported volume per energy unit, this function is an indirect indicator of the compressor efficiency. The economic optimization problem also considers an operating constraint on the compressor outlet pressure ($P_{cp} \geq 10\,\text{MPa}$) for flow assurance.

The system has two manipulated variables (reservoir choke opening v_{ch} and normalized compressor flowrate Q_{cp}). They are indicated as gray circles in Fig. 6. The measured variables are also indicated in the figure. They are summarized in Table 3. The model representing the plant can be found in [1].

Table 3
Initial condition and parameters for the Subsea Gas Compression Station simulation.

Symbol	Description
P_{ro}	Reservoir pressure
Q_{ro}	Reservoir volumetric flow
P_{ch}	Choke outlet pressure
P_{cp}	Compressor outlet pressure
T_{cp}	Compressor outlet temperature
η_{cp}	Compressor power flow rate
Q_p	Pump inlet volumetric flowrate
η_p	Pump power

We assume that the measurement noise comes from a normal distribution with zero mean and standard deviation of 0.5 % of the current value for the meas. For estimating the system gradients, we use plant experiments with central finite differences as in Case Study with $h = 0.05$.

We simulate the system for 220 steady-state (SS) cycles, in which the compressor efficiency changes from $\nu_H \rightarrow \nu_{D_1} \rightarrow \nu_{D_2}$ as shown in Fig. 7. The system is affected by regular process disturbances. The reservoir pressure P_{ro} oscillates between 11 MPa and 12 MPa every 50 SS periods as shown in Fig. 8. We use the same software and solvers as in Case Study 1.

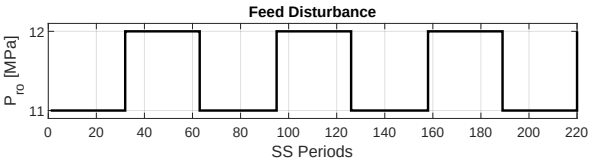

Fig. 8. Reservoir pressure disturbance.

6.2.2 Results

The main goal is to use the proposed approaches to monitor the compressor health. In a subsea system, performing maintenance is costly as it requires specialized lifting vessels, specific weather conditions, and available spare modules. Moreover, maintenance stops lead to long production halts and large production losses. Therefore, unplanned equipment breakdown is very undesirable in this systems.

In this case study, we assume that a maintenance stop is already scheduled at the 220^{th} SS period. However, if the heavily degraded compressor state is *identified for* 10 *consecutive SS periods*, the production is shut down to avoid the wearing out of the compressor. Operating with a degraded compressor can cause critical and permanent damage to its bearings and shaft, degrading these parts beyond repair. Therefore, it is desirable that the approaches do not misidentify the compressor state, which can cause a premature production shutdown and significantly affect the system profitability.

In Fig. 9, we compare the performance of the three approaches. The three top plots show, respectively, the model choice of the baseline method, the model probability cumulative distribution function (CDF) for the Expected Cost Minimization and for the Jensen-Shannon Divergence Regularization approaches. Note that, after the 10^{th} consecutive identification of state D_2, the production is shut down. In the CDF plots, the shut down is indicated by all the model probabilities going to zero. We see that the ECM is superior to the baseline case regarding the medium degradation (D_1) model identification. Instead of chattering between the models, it is able to differentiate H and D_1 with probability $\rightarrow 1$. In turn, both the baseline and ECM are very conservative for the identification of the heavily degraded model (D_2) when compared to the JSDR approach.

Such improvement on the model structure identification comes with an associated penalization to the economic performance. By analyzing the trajectory of the manipulated variable v_{ch} (plots 4 to 6), we see that the profiles of the three approaches are similar, except near the 140^{th} SS period. Here, the JSDR deviates from the others because the method "forces" the exploration of the feasible space, which pays off in the long run since the method does not misidentify model D_2. Note that we show the profiles of only one of the manipulated variables. Due to the nature of the optimization problem, the compressor outlet pressure constraint is always active. Thus, the other manipulated variable, Q_{cp}, is used for constraint tracking and kept practically constant during the simulation.

In the JSDR plot, we also show a red region indicating the optimal v_{ch} when $\alpha = 0.05$ (upper bound) and 0.95 (lower bound). After iteration 160, this region widens. This indicates that there is a considerable advantage, in terms of model identification, if we probe the plant

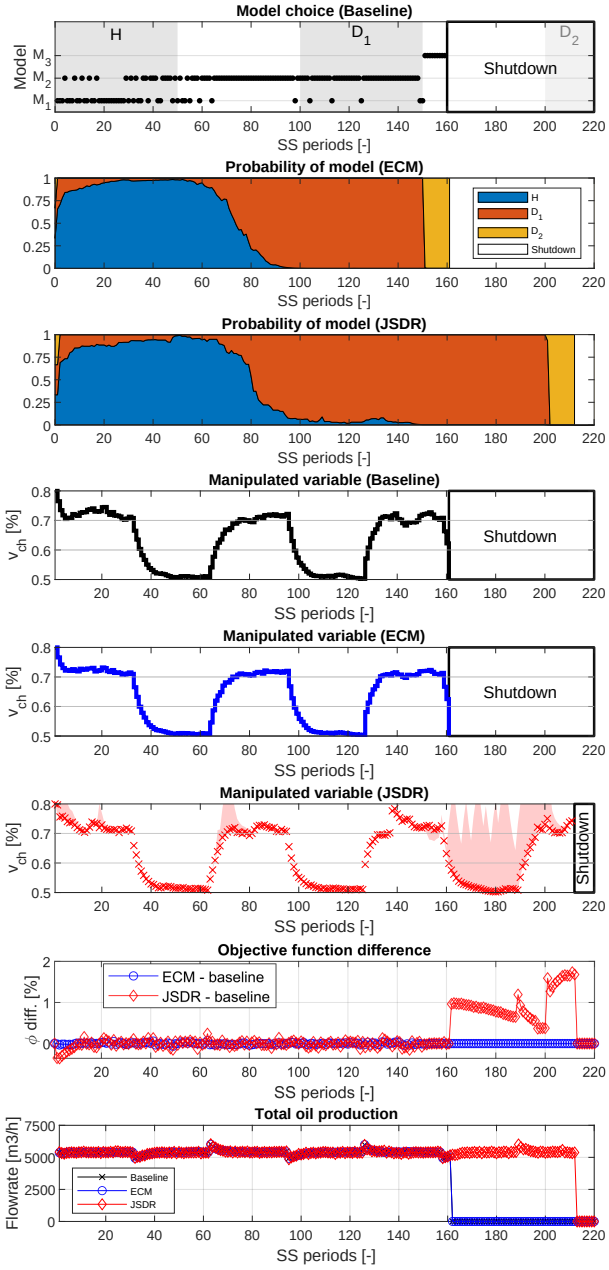

Fig. 9. Results of Baseline algorithm of [5], Expected Cost Minimization, the Jensen-Shannon Divergence Regularization approaches applied to Case Study 2.

towards the lower bound ($\alpha \rightarrow 1$). In other words, it is not expected that the model D_1 and D_2 predictions diverge significantly if we take only "production optimization steps". Although the JSDR approach still favors "production optimization steps", staying near the region lower bound ($\alpha \rightarrow 1$), the inputs are still informative enough to yield accurate model identification. In turn, the other approaches cannot capture this trade-off and indicate the need of a shut down.

Plot 7 in Fig. 9 shows the economic function difference, which is calculated as $\phi - \phi_{baseline}$. We see that the ECM underperforms in the beginning, due to initial exploration. On the other hand, the economic performance of the ECM is nearly identical to the baseline case, which corroborates the findings of the previous case study. Next, the bottom plot shows the total oil production of the well. We clearly see that the premature shut down of the plant, caused by erroneous model identification, leads to a loss of an opportunity to extract oil from the well and significantly decreases the system's profitability.

6.2.3 Practical implementation challenges

For successfully implementing the proposed algorithms in practice, two different challenges need to be faced. The first is related to gradient estimation and the second one to selecting the degradation models in the available set.

As mentioned in Section 4.1, estimating the plant gradients is difficult and the need to infer them from plant measurements is one of the greatest pitfalls for implementing MA in practice [11]. In spite of this issue, MA has been successfully implemented on industrially relevant systems such as an experimental solid-oxide fuel-cell stack [40], Leaching process [11], and a Flotation column [41]. The results in these papers show that estimating plant gradients in practice is an active field of research, and significant efforts have been made to overcome this problem. Specifically for the proposed algorithms, including plant gradient information avoids issues such like non-uniqueness of model structures. However, this information is not strictly necessary for identifying the model structures [13].

Regarding the subsea equipment degradation models, the biggest challenge is the lack of real-life time-to-failure data to obtain the models in the first place [42]. Usually lab data is available, but the results do not fit to realistic operating conditions. However, PHM applied to subsea processes has received much attention due to more stringent operating requirements and the competition of renewable energy sources (see [22] for a detailed review). This means that our algorithms can take advantage of this trend and contribute to improve subsea operations sustainability, taking into account both economic and safety aspects. Clearly, the proposed algorithms are not restricted to subsea processing. Industries, where PHM is better established (e.g. aviation and energy [43]) can also profit from the algorithms.

7 Conclusion

In recent years there has been an increased attention to the possibility of using Reinforcement Learning techniques to aid data-driven optimal control. Of course the mathematical link between Bellman dynamic programming and reinforcement learning is clear [44], however, typically a true dynamic programming solution is intractable. Optimal control and reinforcement learning had thus diverged in approximating and solving noisy

models using a combination of optimization and filtering (optimal control) or supervised learning and approximated value and policy iterations (reinforcement learning). Conceptually, bringing these two disciplines together exhibits strong motivating intuition [45].

In this work, we have developed new algorithms for joint production optimization and model identification.By exploiting modifier adaptation within a reinforcement learning framework, our approach reduces an otherwise uncountable action space to practical procedures amenable to repeated online computation. While existing work in [5,13] also proposed methods based on modifier adaptation, the resulting algorithm did not fully exploit all available information from all controls. As such, illustrated in two case studies based on a chemical reactor and subsea oil and gas exploration, our algorithms yield similar control cost while substantially improving the accuracy of model identification. As costs associated with system degradation, such as shutdown, have a large impact on process viability, we expect our methods may significantly reduce operational risk.

References

[1] A. Verheyleweghen and J. Jäschke. Framework for combined diagnostics, prognostics and optimal operation of a subsea gas compression system. *IFAC-PapersOnLine*, 50(1):15916–15921, 2017.

[2] M. N. Kashid, A. Renken, and L. Kiwi-Minsker. Gas–liquid and liquid–liquid mass transfer in microstructured reactors. *Chemical Engineering Science*, 66(17):3876–3897, 2011.

[3] A. Marchetti, B. Chachuat, and D. Bonvin. Modifier-adaptation methodology for real-time optimization. *Industrial & engineering chemistry research*, 48(13):6022–6033, 2009.

[4] M. Naghshvar. *Active learning and hypothesis testing*. PhD thesis, UC San Diego, 2013.

[5] J. Matias and J. Jäschke. Online model maintenance via output modifier adaptation. *Industrial & Engineering Chemistry Research*, 58(30):13750–13766, 2019.

[6] G. Gutiérrez D. Navia and C. de Prada. Nested modifier-adaptation for rto in the otto williams reactor. *IFAC Proceedings Volumes*, 46(32):123–128, 2013.

[7] S. Costello, G. François, and D. Bonvin. A directional modifier-adaptation algorithm for real-time optimization. *Journal of Process Control*, 39:64–76, 2016.

[8] T. de Avila Ferreira, H. A. Shukla, T. Faulwasser, C. N. Jones, and D. Bonvin. Real-time optimization of uncertain process systems via modifier adaptation and gaussian processes. In *2018 European Control Conference (ECC)*, pages 465–470. IEEE, 2018.

[9] E.A. del Rio Chanona et al. Real-time optimization meets Bayesian optimization and derivative-free optimization: a tale of modifier adaptation. *Computers & Chemical Engineering*, 147, 2021.

[10] G. A. Bunin, G. François, and D. Bonvin. Sufficient conditions for feasibility and optimality of real-time optimization schemes-i. theoretical foundations. *arXiv preprint arXiv:1308.2620*, 2013.

[11] A. G. Marchetti, T. Faulwasser, and D. Bonvin. A feasible-side globally convergent modifier-adaptation scheme. *Journal of Process Control*, 54:38–46, 2017.

[12] J. Speakman, A. Papasavvas, and G. François. A robust modifier adaptation method via hessian augmentation using model uncertainties. *Journal of Process Control*, 99:28–40, 2021.

[13] J. Matias and J. Jäschke. Online model maintenance in real-time optimization methods. *Computers & Chemical Engineering*, page 107141, 2020.

[14] B. Sun, S. Zeng, R. Kang, and M. G. Pecht. Benefits and challenges of system prognostics. *IEEE Transactions on reliability*, 61(2):323–335, 2012.

[15] Y. Peng, M. Dong, and M. J. Zuo. Current status of machine prognostics in condition-based maintenance: a review. *The International Journal of Advanced Manufacturing Technology*, 50(1-4):297–313, 2010.

[16] G. Vachtsevanos, F. Lewis, M. Roemer, A. Hess, and B. Wu. *Intelligent fault diagnosis and prognosis for engineering systems*, volume 13. Wiley Online Library, 2006.

[17] J. P. P. Gomes, B. P. Leão, W. O. L. Vianna, R. K. H. Galvão, and T. Yoneyama. Failure prognostics of a hydraulic pump using kalman filter. In *Annual Conference of the PHM Society*, volume 4, 2012.

[18] T. Biagetti and E. Sciubba. Automatic diagnostics and prognostics of energy conversion processes via knowledge-based systems. *Energy*, 29(12-15):2553–2572, 2004.

[19] J. H. Jahren, J. Matias, and J. Jäschke. Data-driven modelling of choke valve erosion using data simulated from a first principles model. In *Computer Aided Chemical Engineering*, volume 50, pages 773–778. Elsevier, 2021.

[20] X. Liu, J. Matias, J. Jäschke, and J. Vatn. Gibbs sampler for noisy transformed gamma process: Inference and remaining useful life estimation. *Reliability Engineering & System Safety*, page 108084, 2021.

[21] Y. Han, W. Qi, N. Ding, and Z. Geng. Short-time wavelet entropy integrating improved lstm for fault diagnosis of modular multilevel converter. *IEEE Transactions on Cybernetics*, 2021.

[22] A. Verheyleweghen. Control degrees of freedom for optimal operation and extending remaining useful life. 2020.

[23] Y. Zhang and J. Jiang. Bibliographical review on reconfigurable fault-tolerant control systems. *Annual reviews in control*, 32(2):229–252, 2008.

[24] J. Matias, J. Ågotnes, and J. Jäschke. Health-aware advanced control applied to a gas-lifted oil well network. *IFAC-PapersOnLine*, 53(3):301–306, 2020.

[25] M. Naghshvar, T. Javidi, et al. Active sequential hypothesis testing. *The Annals of Statistics*, 41(6):2703–2738, 2013.

[26] A. Cozad and N. V. Sahinidis. A global minlp approach to symbolic regression. *Mathematical Programming*, 170(1):97–119, 2018.

[27] T. J. Sullivan. *Introduction to uncertainty quantification*, volume 63. Springer, 2015.

[28] D. Swigon. 2.1 ensemble modeling of biological systems. *Mathematics and life sciences*, 1, 2012.

[29] Y. Han, S. Liu, D. Cong, Z. Geng, J. Fan, J. Gao, and T. Pan. Resource optimization model using novel extreme learning machine with t-distributed stochastic neighbor embedding: Application to complex industrial processes. *Energy*, 225:120255, 2021.

[30] Z. Wang, Y. Han, C. Li, Z. Geng, and J. Fan. Input-output networks considering graphlet-based analysis for production optimization: Application in ethylene plants. *Journal of Cleaner Production*, 278:123955, 2021.

[31] K. J. Åström. Optimal control of markov processes with incomplete state information. *Journal of Mathematical Analysis and Applications*, 10(1):174–205, 1965.

[32] B. Alt, M. Schultheis, and H. Koeppl. Pomdps in continuous time and discrete spaces. *Advances in Neural Information Processing Systems*, 33, 2020.

[33] E. Arcari, L. Hewing, M. Schlichting, and M. Zeilinger. Dual stochastic MPC for systems with parametric and structural uncertainty. In *Learning for Dynamics and Control*, 2020.

[34] R. S. Sutton and A. G. Barto. *Reinforcement learning: An introduction*. MIT press, 2018.

[35] D. P. Bertsekas and J. N. Tsitsiklis. *Neuro-dynamic programming*. Athena Scientific, 1996.

[36] V. Krishnamurthy. *Partially observed Markov decision processes*. Cambridge University Press, 2016.

[37] J. R. Hershey and P. Olsen. Approximating the kullback leibler divergence between gaussian mixture models. In *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 1988.

[38] J. A. E. Andersson, J. Gillis, G. Horn, J. B. Rawlings, and M. Diehl. CasADi – A software framework for nonlinear optimization and optimal control. *Mathematical Programming Computation*, 11(1):1–36, 2019.

[39] A. Wächter and L. T. Biegler. On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming. *Mathematical programming*, 106(1):25–57, 2006.

[40] A. G. Marchetti, T. A. Ferreira, S. Costello, and D. Bonvin. Modifier adaptation as a feedback control scheme. *Industrial & Engineering Chemistry Research*, 59(6):2261–2274, 2020.

[41] D. Navia, D. Villegas, I. Cornejo, and C. de Prada. Real-time optimization for a laboratory-scale flotation column. *Computers & chemical engineering*, 86:62–74, 2016.

[42] P. Vaidya. Prognosis-subsea oil and gas industry. In *Annual Conference of the PHM Society*, volume 2, 2010.

[43] J. D. Friedemann, A. Varma, P. Bonissone, and N. Iyer. Subsea condition monitoring: A path to increased availability and increased recovery. In *Intelligent Energy Conference and Exhibition*. OnePetro, 2008.

[44] D. P. Bertsekas. *Reinforcement learning and optimal control*. Athena Scientific Belmont, MA, 2019.

[45] W. B. Powell. From reinforcement learning to optimal control: A unified framework for sequential decisions. pages 29–74, 2021.