Optimal switching of MPC cost function for changing active constraints

Lucas Ferreira Bernardino[a], Sigurd Skogestad[a]

[a]*Department of Chemical Engineering, Norwegian University of Science and Technology, Sem Sælands vei 4, Kjemiblokk 5, 101B, Trondheim, 7491, Trøndelag, Norway*

Abstract

Model predictive control (MPC) allows for dealing with multivariable interactions and dynamic satisfaction of constraints. Most commonly, standard MPC has a cost function that aims at keeping selected controlled variables at constant setpoints. This work considers systems where the *steady-state* active constraints change during operation, which is not optimally covered by conventional MPC which uses fixed controlled variables. We propose a simple framework that detects the constraint changes and updates the controlled variables accordingly. The unconstrained controlled variables are chosen to minimize an economic cost. In this paper, the nullspace method for self-optimizing control is used to estimate the steady-state cost gradient based on a static combination of measurements. This estimated gradient is also used for detecting the current set of active constraints, which in particular allows for giving up constraints that were previously active. The proposed framework, here referred to as "region-based MPC", is shown to be optimal for linear constrained systems with a quadratic economic cost function, and it allows for good economic performance in nonlinear systems in a neighborhood of the considered design points.

Keywords: self-optimizing control, model predictive control, optimal operation

1. Introduction

Model predictive control (MPC) denotes a class of control strategies based on the online optimization of the predicted dynamic trajectory of the system [1]. It is a valuable tool for process control, being able to deal with multivariable interactions and constraint satisfaction. In practice, MPC is usually implemented as a supervisory control layer above the plant regulatory layer, where stability is assessed, and is subordinate to a real-time optimization (RTO) layer, which updates plant operation based on economics, as presented in Figure 1. It is possible to combine the RTO and control (MPC) layers into one; this is commonly known as Economic MPC [2] and is not considered in this paper.

The steady-state economic optimization of the plant, solved at the RTO layer, can be defined as the following constrained optimization problem:

$$\begin{aligned} \min_{u} \quad & J^{ec}(u,d) \\ \text{s.t.} \quad & g(u,d) \leq 0 \end{aligned} \tag{1}$$

where $u \in \mathbb{R}^{n_u}$ is the vector of inputs or manipulated variables (MVs), $d \in \mathbb{R}^{n_d}$ is the vector of disturbances, J^{ec} is the scalar economic cost function, and $g(u,d) \in \mathbb{R}^{n_g}$ is the vector of inequality constraints. Note that the model equations and correspondent states have been formally eliminated from the formulation. The set of active constraints $\mathcal{A}$ is defined for the optimal solution u^* as the set for which $g_i(u^*,d) = 0$ with $i \in \mathcal{A}$.

Solving the problem in (1) results in the optimal plant inputs u^*, but as shown in Figure 1 the RTO layer implements the optimization results in the form of setpoint updates CV^{sp} to the

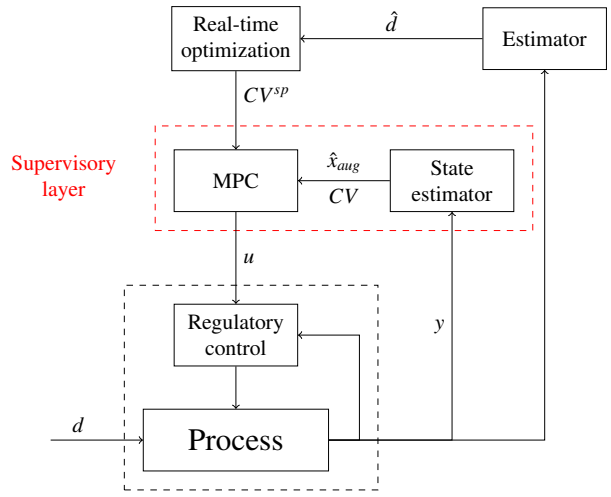

Figure 1: Typical hierarchical control structure with standard setpoint-tracking MPC in the supervisory layer. The cost function for the RTO layer is J^{ec} and the cost function for the MPC layer is J^{MPC}. With no RTO layer (and thus constant setpoints CV^{sp}), this structure is not economically optimal when there are changes in the active constraints. For smaller applications, the state estimator may be used also as the RTO estimator.

MPC layer. We refer to this type of implementation as setpoint-tracking MPC, or standard MPC. As discussed in more detail later, see Equation (5), standard MPC uses a cost function of the form:

$$J^{MPC} = \sum_{k=1}^{N} \|CV_k - CV^{sp}\|_Q^2 + \|\Delta u_k\|_R^2$$

where the first term penalizes setpoint deviations and the last term penalizes dynamic input changes.

Standard MPC has two main elements: a state estimator and an open-loop moving horizon optimizer (which is often referred to simply as MPC). The state estimator ensures feedback, correcting the internal model according to the measurements, and the MPC uses that information to calculate the input sequence that drives the internal model to the desired operating point. The MPC problem can accommodate constraint satisfaction, either as direct constraints in the optimization problem or through the use of penalty terms. Additionally, one may consider a target calculation block, which ensures that the setpoint that the MPC tracks is feasible at steady state [1].

In most standard MPC implementations, the CVs are selected based on process intuition, and not in a systematic manner. In this context, self-optimizing control (SOC) provides useful tools for systematic selection of CVs, having optimal steady-state operation as the main goal [3, 4]. This gives the controlled variables:

$$CV = Hy$$

where y denotes the available process measurements (including selected inputs and measured disturbances) and H is a selection or combination matrix. Most SOC approaches for CV selection assume that the steady-state active constraint set $\mathcal{A}$ is constant [4].

Graciano et al. [5] implemented MPC using nominal self-optimizing CVs, that is, with the nominally active constraints. This can reject disturbances in fast timescales and minimize the nominal economic loss without the intervention of the RTO layer, at the same time avoiding violation of constraints. This is relatively simple to implement, but it cannot be regarded as self-optimizing in a broad sense, because the optimal approach is to use different self-optimizing CVs for each set of active steady-state constraints [6].

This work proposes a framework for self-optimizing control under changing active constraints, which we label "region-based MPC", see Figure 2. Here, the self-optimizing CVs tracked by MPC are a function of the detected active constraint set. The constraint switching is based on the work of Woodward et al. [7]. In three case studies, we show that standard MPC with a single (nominal) set of CVs leads to economic loss when there are changes in active constraints during operation, and we show that the proposed MPC framework attains steady-state optimal operation if the design conditions of SOC are met, and near-optimal operation in a broader sense.

The rest of the paper is organized as follows. In Section 2 we present some basic notions of MPC implementation. In Section 3 we describe the control structure proposed in this work,

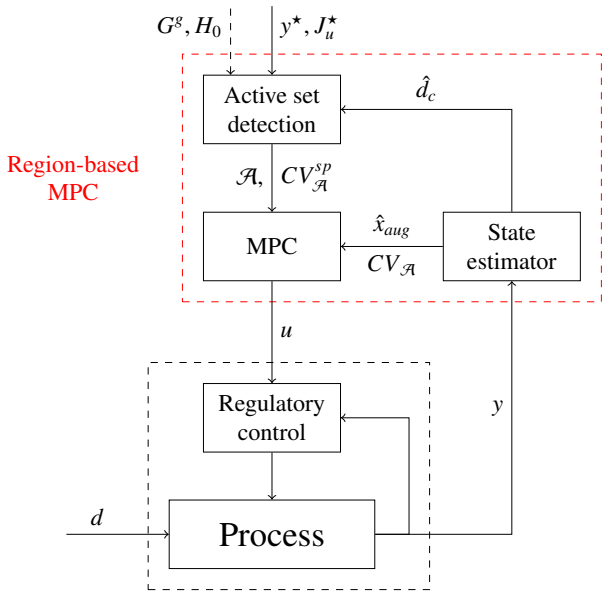

Figure 2: Proposed region-based MPC structure with active set detection and change in controlled variables. The possible updates from an upper RTO layer (y^*, J_u^* etc.) are not considered in the present work. Even with no RTO layer (and thus with constant setpoints $CV_{\mathcal{A}}^{sp}$, see (14) and (15), in each active constraint region), this structure is potentially economically optimal when there are changes in the active constraints.

and the results of its application in some case studies are presented in Section 4. Based on these results and the theoretical aspects of the control structure, we discuss the proposed framework in Section 5, and the paper is then concluded in Section 6.

2. Standard MPC implementation

We first briefly discuss the standard MPC implementation represented in Figure 1, which includes a state estimator and an open-loop optimizer (MPC block). For the estimator, consider the following dynamic model used as an internal model for MPC:

$$\begin{cases} \dfrac{dx}{dt} = f(x, u, d_c) \\ y = h(x, d_c) \end{cases} \tag{2}$$

Here, $x \in \mathbb{R}^{n_x}$ represents the vector of dynamic states, $u \in \mathbb{R}^{n_u}$ the vector of inputs (MVs), $y \in \mathbb{R}^{n_y}$ the vector of measurements and $d_c \in \mathbb{R}^{n_{d_c}}$ a vector of model disturbances.

There is usually one d_c for each controlled variable, which is used to account for uncertainty, for example, related to the "true" disturbances d, measurement bias, or model parameter changes. Note that d_c does not need to have a physical interpretation and is used mainly to include integral action in MPC. In other words, to attain offset-free control, the internal model is augmented with the integrating states d_c:

$$x_{aug} = \begin{bmatrix} x \\ d_c \end{bmatrix} \tag{3}$$

For a linear internal model, the number of additional integrating states must be at least the number of controlled variables, and it

need not be greater than the number of measurements [8]. The dynamic model considered by the state estimator is therefore of the form:

$$\begin{cases} \dfrac{d\hat{x}_{aug}}{dt} = \dfrac{d}{dt}\begin{bmatrix}\hat{x}\\ \hat{d}_c\end{bmatrix} = \begin{bmatrix} f(\hat{x}, u, \hat{d}_c)\\ 0\end{bmatrix} + \omega \\ \hat{y} = h(\hat{x}, \hat{d}_c) + \nu \end{cases} \tag{4}$$

where $\omega \sim \mathcal{N}(0, Q^e)$ and $\nu \sim \mathcal{N}(0, R^e)$ are the random variables present in most state estimation frameworks, and Q^e and R^e are the corresponding tuning parameters [9]. The estimated states $\hat{x}$ and $\hat{d}_c$ are then used to solve a moving-horizon optimization problem, which results in the next control action to be implemented. A simple discretized MPC optimization problem can be of the form:

$$\begin{aligned} \min_{u_k, x_k} \quad & J^{MPC} = \sum_{k=1}^{N} \|CV_k - CV^{sp}\|_Q^2 + \|\Delta u_k\|_R^2 \\ \text{s.t.} \quad & x_k = \phi(x_{k-1}, u_{k-1}, \hat{d}_c) \\ & y_k = h(x_k, \hat{d}_c) \\ & CV_k = Hy_k \\ & \Delta u_k = u_k - u_{k-1} \\ & x_0 = \hat{x} \\ & y_{min} \le y_k \le y_{max} \\ & x_{min} \le x_k \le x_{max} \\ & u_{min} \le u_k \le u_{max} \\ & -\Delta u_{max} \le \Delta u_k \le \Delta u_{max} \end{aligned} \tag{5}$$

Here, x_k denotes the state at the k-th time step, and $\phi(x_{k-1}, u_{k-1}, \hat{d}_c)$ is the result of the integration of the dynamic model (2) from t_{k-1} to $t_k = t_{k-1} + \Delta t$ with $u = u_{k-1}$, $d_c = \hat{d}_c$, and the initial condition as the previous state $x(t_{k-1}) = x_{k-1}$. The objective function J^{MPC} aims to minimize the tracking error $CV - CV^{sp}$ while penalizing large input changes Δu_k. N is the number of prediction steps, and Q and R are tuning matrices.

The output, state, and input constraints in (5) can be used to embed the RTO constraint $g(u, d) \le 0$ (1) in the MPC time scale. State constraints are not needed if we assume that the constraints $g(u, d)$ are measured (or estimated) and included as elements in the measurements vector y. Without loss of generality, we will assume that the economic constraints can be estimated from the dynamic model as:

$$g = h_g(x, u, d_c) \tag{6}$$

We remark that an MPC in the form of Equation (5) has no stability guarantees, but it can converge if the prediction horizon N is large enough (the reader is referred to Mayne [10] for an in-depth review of MPC formulations).

The focus of the present work is the case where the original setpoints CV^{sp} must be given up due to constraints becoming active at steady-state. In theory, the RTO layer may update the setpoints, but in most cases there is no RTO layer, so the setpoints are constant. Standard MPC satisfies the constraints, but it is suboptimal in terms of steady-state economic performance, and we propose a better way of dealing with economic constraints on standard setpoint-tracking MPC, without the need for RTO updates.

3. Region-based MPC framework

The structure of the proposed region-based MPC scheme is summarized in Figure 2. The state estimator, also present in standard MPC, serves as the feedback element for MPC as well as for the active set detection block, which is the new element of the framework when compared to standard MPC. The detected active set $\mathcal{A}$ along with the setpoints, is sent to the MPC block, which uses a different set of CVs for each active set $\mathcal{A}$.

We next describe how the CVs for each active set are determined, and how the set of active constraints can be estimated online.

3.1. Controlled variables for MPC

The controlled variables for each region are defined such that the steady-state economic problem (9) is solved by feedback control, meaning that we adjust u to keep $CV = CV^{sp}$. These CVs are defined as:

$$CV_{\mathcal{A}} = \begin{bmatrix} g_{\mathcal{A}} \\ c_{\mathcal{A}} \end{bmatrix} \tag{7}$$

Here, $g_{\mathcal{A}}$ denotes the active constraints, and $c_{\mathcal{A}}$ denotes the unconstrained CVs for optimal operation. The control action calculation for the proposed region-based MPC is very similar to that of Equation (5), but the objective function changes for each $\mathcal{A}$ according to:

$$J_{\mathcal{A}}^{MPC} = \sum_{k=1}^{N} \|CV_{\mathcal{A}} - CV_{\mathcal{A}}^{sp}\|_{Q_{\mathcal{A}}}^2 + \|\Delta u_k\|_{R_{\mathcal{A}}}^2 \tag{8}$$

where $Q_{\mathcal{A}}$ and $R_{\mathcal{A}}$ are tuning parameters that can be chosen independently for each active set $\mathcal{A}$.

The unconstrained controlled variables $c_{\mathcal{A}}$ should be selected to minimize the steady-state cost, given that the active constraints $g_{\mathcal{A}}$ are being controlled. For that, we follow Halvorsen et al. [11] and we consider a local QP approximation of the economic optimization problem of the form:

$$\begin{aligned} \min_{\Delta u} \quad & J^{ec} = J^{ec\star} + \begin{bmatrix}\Delta u^T & \Delta d^T\end{bmatrix}\begin{bmatrix} J_u^\star \\ J_d^\star \end{bmatrix} \\ & \quad + \frac{1}{2}\begin{bmatrix}\Delta u^T & \Delta d^T\end{bmatrix}\begin{bmatrix} J_{uu} & J_{ud} \\ J_{ud}^T & J_{dd}\end{bmatrix}\begin{bmatrix}\Delta u \\ \Delta d\end{bmatrix} \end{aligned} \tag{9a}$$

$$\text{s.t.} \quad g = g^\star + G^g \Delta u + G_d^g \Delta d \le 0 \tag{9b}$$

Here, $\Delta d = d - d^\star$ and $\Delta u = u - u^\star$ represent the disturbances and inputs as their deviation from their reference values $d^\star$ and $u^\star$ respectively, and $J^{ec\star}$, $J_u^\star$, and J_{uu} represent respectively the cost function, its gradient, and its Hessian with respect to the inputs, evaluated at steady state at the reference point. Additionally, the measurements y can be locally represented by a linear steady-state model of the form:

$$\Delta y = G^y \Delta u + G_d^y \Delta d \tag{10}$$

The linearized expression (9b) for the constraints is not used by Halvorsen et al. [11], but it is needed here because we consider changes in active constraints. The unconstrained CVs $c_{\mathcal{A}}$ are defined in terms of the CVs for the fully unconstrained region, c_0, which is itself a static linear combination of the measurements:

$$c_0 = H_0 y \tag{11}$$

Assuming that there are enough measurements ($n_y = n_u + n_d$), we can find an analytical expression for H_0 based on the nullspace method that is given by [12, 13]:

$$H_0 = \begin{bmatrix} J_{uu} & J_{ud} \end{bmatrix} \begin{bmatrix} G^y & G_d^y \end{bmatrix}^{\dagger} \tag{12}$$

which is used to obtain an optimal first-order estimate of the cost gradient J_u [13]:

$$\hat{J}_u = c_0 - c_0^{sp} = H_0(y - y^\star) + J_u^\star \tag{13}$$

The setpoint for the unconstrained CV, $c_0^{sp} = H_0 y^\star - J_u^\star$, is calculated based on the reference steady state and the corresponding value of the cost gradient. If the reference steady state is an optimal operating point at a fully unconstrained region, $J_u^\star$ is zero. However, if the reference steady state is not optimal, or if the system is operating at a constrained region, it is nonzero.

For the constrained case, the ideal self-optimizing variable is the reduced gradient $N_{\mathcal{A}}^T J_u$ [6], and the CVs related to the unconstrained degrees of freedom are given by:

$$c_{\mathcal{A}} = N_{\mathcal{A}}^T H_0 y$$

where $N_{\mathcal{A}}$ is a projection matrix, defined as a basis for the nullspace of the active constraints gradient, i.e. $G_{\mathcal{A}}^g N_{\mathcal{A}} = 0$. Because $c_{\mathcal{A}}$ is a linear combination of the unconstrained CVs c_0, the corresponding setpoint will be given by $c_{\mathcal{A}}^{sp} = N_{\mathcal{A}}^T c_0^{sp}$. The full set of CVs for each active constraint region $\mathcal{A}$ then becomes:

$$CV_{\mathcal{A}} = \begin{bmatrix} g_{\mathcal{A}} \\ c_{\mathcal{A}} \end{bmatrix} = \begin{bmatrix} g_{\mathcal{A}} \\ N_{\mathcal{A}}^T H_0 y \end{bmatrix} \tag{14}$$

with the corresponding setpoints being:

$$CV_{\mathcal{A}}^{sp} = \begin{bmatrix} 0 \\ N_{\mathcal{A}}^T (H_0 y^\star - J_u^\star) \end{bmatrix} \tag{15}$$

This choice of CVs minimizes the steady-state economic loss around the reference point $(u^\star, d^\star)$. Furthermore, even if the new operating point is such that the active set $\mathcal{A}$ is different than that of the reference point, the use of $CV_{\mathcal{A}}$ minimizes the steady-state loss, as long as the approximations in Equations (9) and (10) hold. We now discuss how to detect the active set using the available measurements, so as to select the correct controlled variables.

3.2. Active constraint set detection

In the previous section, we estimated the cost gradient J_u as a function of the available measurements y. The same idea is now applied to active set detection. Woodward et al. [7] describes an active set detection algorithm for a feedback optimizing strategy that only depends on the current value of the cost gradient J_u, the constraints g, and the constraints gradient G^g. Here, we adapt this strategy so that the method depends directly on the available measurements. In summary, we assume g to be directly measured, J_u is estimated using Equation (13), and G^g is assumed constant at its nominal value from Equation (9b).

In order to estimate J_u, which is the steady-state cost gradient, we use the value of the measurements at the expected (predicted optimal from the region-based MPC perspective using Equation (16)) steady state where the CVs are driven to their setpoints, which we call y^{ss}. This expected steady state can be determined using Equation (2), leading to:

$$\begin{cases} 0 = f(x^{ss}, u^{ss}, \hat{d}_c) \\ CV_{\mathcal{A}}(x^{ss}, \hat{d}_c) = CV_{\mathcal{A}}^{sp} \end{cases} \tag{16}$$

The steady-state measurements are then obtained as $y^{ss} = h(x^{ss}, \hat{d}_c)$, along with the corresponding predicted constraint values from Equation (6) as $g^{ss} = h_g(x^{ss}, u^{ss}, \hat{d}_c)$. With these values, we are ready to apply the method from Woodward et al. [7]. The algorithm is summarized in Algorithm 1, and we shall explain its main steps.

Algorithm 1 Active set estimation based on the MPC internal model, adapted from Woodward et al. [7]

1: $\hat{J}_u \leftarrow H_0(y^{ss} - y^\star) + J_u^\star$ ▷ from (13)
2: $\mathcal{A}^k \leftarrow \mathcal{A}^{k-1} \cup \{\, i \mid g_i^{ss} \geq 0 \,\}$
3: $\delta u^* \leftarrow$ solution of (17)
4: $\mathcal{A}^k \leftarrow \{\, i \in \mathcal{A}^k \mid G_i^g \delta u^* = 0 \,\}$
5: **if** $n\left(\mathcal{A}^k\right) > n_u$ **then** ▷ too many active constraints
6: Find $\mathcal{A}' \subset \mathcal{A}^k \mid g^{ss}(CV_{\mathcal{A}'} = CV_{\mathcal{A}'}^{sp}) \leq 0$ ▷ re-solve (16)
7: $\mathcal{A}^k \leftarrow \mathcal{A}'$
8: **end if**

The algorithm begins in step 1 by finding the expected steady-state measurements y^{ss} by solving Equation (16), and with it we find the predicted cost gradient $\hat{J}_u$ through Equation (13). Then, we include in step 2 the constraints predicted to be violated, i.e. $g_i^{ss} \geq 0$, into the estimated active set. With this augmented active set $\mathcal{A}$, we solve in step 3 the following optimization problem:

$$\delta u^* = \arg\min_{\delta u} \; -\delta u^T \delta u \quad \text{s.t.} \quad \begin{cases} G_{\mathcal{A}}^g \delta u \leq 0 \\ \delta u^T \delta u = -\delta u^T \hat{J}_u \end{cases} \tag{17}$$

With this problem, we wish to find the largest projection of the negative of the estimated cost gradient, $-\hat{J}_u$, onto the feasible directions, i.e. directions that do not violate $G_{\mathcal{A}}^g \delta u \leq 0$. The solution δu^* therefore dictates the best feasible descent direction for improving the economic cost function. In step 4,

the inactive constraints at the solution ($G_i^g \delta u^* < 0$) are then removed from the active set $\mathcal{A}$, as controlling these constraints would hinder economic improvement.

Because this method does not account for infeasibility, an additional step to obtain the active set $\mathcal{A}$ sent to the controller is necessary. If the active set resulting from the previous operations has more than n_u elements, it is deemed infeasible, because the controller cannot track more than n_u variables with the available inputs. One must then pick a subset that is predicted to be feasible ($g_i^{ss} \leq 0 \; \forall i$) by evaluating the corresponding closed-loop steady states with Equation (16), as represented in step 6. With this, the controller always tries to control a feasible set of active constraints. One may also add a constraint priority list, such that, when operation is infeasible for all candidate active sets, the less important constraint is given up.

The problem given in Equation (17) is presented as an NLP due to the quadratic equality constraint, but the work of Woodward et al. [7] solves this problem using a specific algorithm. Here, we simply solve the optimization problem directly. To prevent premature switching due to the estimator dynamics, the estimated $\mathcal{A}$ is only used for switching CVs after N_{sw} time steps where the estimated $\mathcal{A}$ is different than the one being implemented in the controller, and N_{sw} becomes a tuning parameter that improves switching performance.

4. Case studies

For the following case studies, consider the continuous-discrete time objective function for MPC:

$$J_{\mathcal{A}}^{MPC} = \int_0^{N\Delta t} \|CV_{\mathcal{A}} - CV_{\mathcal{A}}^{sp}\|_{Q_{\mathcal{A}}}^2 \, dt + \sum_{k=1}^{N} \|\Delta u_k\|_{R_{\mathcal{A}}}^2 \tag{18}$$

This formulation is used for convenience, as the integration of the MPC objective function and the internal model are done together using orthogonal collocation. The resulting problem is solved using CasADi/IPOPT [14].

4.1. Case study 1 - toy example

In this example, we illustrate the optimality of the proposed methodology for systems with quadratic cost function and linear dynamic and constraint models, which is sufficient for the exactness of the methodology described in Section 3, and the economic improvement when compared to a standard implementation of self-optimizing MPC.

The hypothetical system considered here has 2 dynamic states x and 3 MVs u, with economic objectives and constraints being represented by the following optimization problem:

$$\begin{aligned} \min_u \quad & \frac{1}{2} x^T \begin{bmatrix} 1 & 0 \\ 0 & 10 \end{bmatrix} x + \frac{1}{2} u^T \begin{bmatrix} 1 & -0.1 & -0.2 \\ -0.1 & 0.8 & -0.1 \\ -0.2 & -0.1 & 0.3 \end{bmatrix} u \\ \text{s.t.} \quad & \begin{cases} g_1 = x_1 - 0.8 x_2 \leq 0 \\ g_2 = u_1 + u_2 + u_3 \leq 0 \end{cases} \end{aligned} \tag{19}$$

The dynamic states x are affected by the MVs u and the disturbances d according to the following linear state-space dynamic model:

$$\begin{cases} \dot{x} = \begin{bmatrix} -\frac{1}{\tau_1} & 0 \\ 0 & -\frac{1}{\tau_2} \end{bmatrix} x + \begin{bmatrix} \frac{0.2}{\tau_1} & 0 & 0 \\ 0 & \frac{0.2}{\tau_2} & 0 \end{bmatrix} u + \begin{bmatrix} \frac{1}{\tau_1} & 0 \\ 0 & \frac{1}{\tau_2} \end{bmatrix} d \\ y = \begin{bmatrix} g_1 \\ g_2 \\ x_2 \\ u_2 \\ u_3 \end{bmatrix} = \begin{bmatrix} 1 & -0.8 \\ 0 & 0 \\ 0 & 1 \\ 0 & 0 \\ 0 & 0 \end{bmatrix} x + \begin{bmatrix} 0 & 0 & 0 \\ 1 & 1 & 1 \\ 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} u \end{cases} \tag{20}$$

with $\tau_1 = 1$ and $\tau_2 = 2$. For the vector of measurements y, as discussed in Section 3, we follow the convention of considering the constraints as direct measurements, figuring in the first two rows of the measurement vector. The remaining measurements are chosen with the goal of satisfying a sufficient number of independent measurements ($n_y = n_u + n_d$).

Using the information above, we may eliminate the state variables x, since at steady state, we have that:

$$x = \begin{bmatrix} 0.2 & 0 & 0 \\ 0 & 0.2 & 0 \end{bmatrix} u + \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} d \tag{21}$$

We then write the steady-state optimization problem in the standard form:

$$\begin{aligned} \min_u \quad & J = \frac{1}{2} u^T \begin{bmatrix} 1.04 & -0.1 & -0.2 \\ -0.1 & 1.2 & -0.1 \\ -0.2 & -0.1 & 0.3 \end{bmatrix} u + u^T \begin{bmatrix} 0.2 & 0 \\ 0 & 2 \\ 0 & 0 \end{bmatrix} d \\ \text{s.t.} \quad & g = \begin{bmatrix} 0.2 & -0.16 & 0 \\ 1 & 1 & 1 \end{bmatrix} u + \begin{bmatrix} 1 & -0.8 \\ 0 & 0 \end{bmatrix} d \leq 0 \end{aligned} \tag{22}$$

along with the steady-state expression for the measurements:

$$y = \begin{bmatrix} 0.2 & -0.16 & 0 \\ 1 & 1 & 1 \\ 0 & 0.2 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} u + \begin{bmatrix} 1 & -0.8 \\ 0 & 0 \\ 0 & 1 \\ 0 & 0 \\ 0 & 0 \end{bmatrix} d \tag{23}$$

for which the matrices presented in Equations (9) and (10) are recognizable. This problem has two inequality constraints and three MVs, so optimal operation has always between one and three unconstrained DOFs. As the system has only two disturbances, we can graphically illustrate the active constraint regions as in Figure 3, where we can see all possible combinations of active constraints. This map of disturbances is not used in the method, and it is only made with the goal of visualizing the optimal operation mode for each disturbance.

For implementing a standard self-optimizing MPC controller with dynamic constraint handling, we follow the strategy described by Graciano et al. [5]. We design the self-optimizing CVs $c = Hy$ at the unconstrained region, choosing as the reference steady state the optimal value for $d^\star = \begin{bmatrix} -4; & +4 \end{bmatrix}$. We can therefore use the matrix $H = H_0$ as defined in Equation (12), leading to:

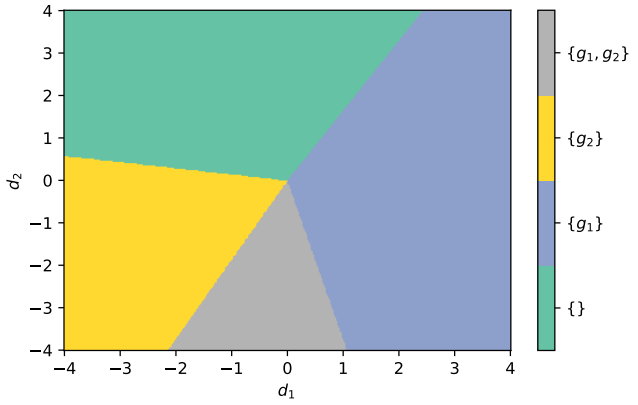

Figure 3: Active constraint regions for case study 1 as a function of disturbances

$$H_0 = \begin{bmatrix} 0.2 & 1 & 0.16 & -1.1 & -1.2 \\ 0 & -0.1 & 2 & 0.9 & 0 \\ 0 & -0.2 & 0 & 0.1 & 0.5 \end{bmatrix}$$

The measurements at the optimal operating point for $d^\star$ are $y^\star = [-6.17 \quad -9.86 \quad 2.62 \quad -6.91 \quad -2.56]^T$, and the corresponding setpoint is $c^{sp} = H_0 y^\star = \begin{bmatrix} 0 & 0 & 0 \end{bmatrix}^T$. The implementation of the region-based MPC controller leads to c_0 being this same set of CVs, and it depends additionally on the calculation of projection matrices $N_{\mathcal{A}}$ for the constrained regions. These are presented in Table 1. In terms of the tuning of the controllers, these are presented in Table 2, with the standard MPC using only the tuning corresponding to the fully unconstrained region ($\mathcal{A} = \{\}$).

$\mathcal{A}$	$N_{\mathcal{A}}^T$
$\{\}$	$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$
$\{1\}$	$\begin{bmatrix} 0.625 & 0.781 & 0 \\ 0 & 0 & 1 \end{bmatrix}$
$\{2\}$	$\begin{bmatrix} -0.577 & 0.789 & -0.211 \\ -0.577 & -0.211 & 0.789 \end{bmatrix}$
$\{1,\ 2\}$	$\begin{bmatrix} -0.362 & -0.453 & 0.815 \end{bmatrix}$

Table 1: Optimal gradient projections for case study 1

The process was simulated at two different points inside each region. Results of the closed-loop simulation of both control strategies are given in Figure 4. The three CVs $c = H_0 y$ obtained in the unconstrained region ($\mathcal{A} = \{\}$) are not suitable for the constrained regions in terms of optimal operation. This is most easily seen by comparing the inputs (u_1, u_2, u_3) from standard MPC (green) with the optimal inputs (magenta). The inputs obtained with the proposed region-based MPC (blue) are optimal at steady state in all four regions, and the switching of CVs is seen to be smooth.

The economic loss of standard MPC is shown in more detail in Figure 5. It can be seen that the loss is nonzero whenever

Parameter	$\mathcal{A}$	Value
$Q_{\mathcal{A}}$	$\{\}$	diag([1, 1, 100])
	$\{1\}$	diag([1, 1, 1])
	$\{2\}$	diag([1, 1, 1])
	$\{1,\ 2\}$	diag([1, 1, 1])
$R_{\mathcal{A}}$		diag([0.01, 0.01, 0.01])
N		30
Δt		0.333
Q^e		diag([0.05, 0.05, 1, 1])
R^e		diag([0.01, 0.01, 0.01, 0.01, 0.01])

Table 2: Tuning of controllers and estimator for case study 1

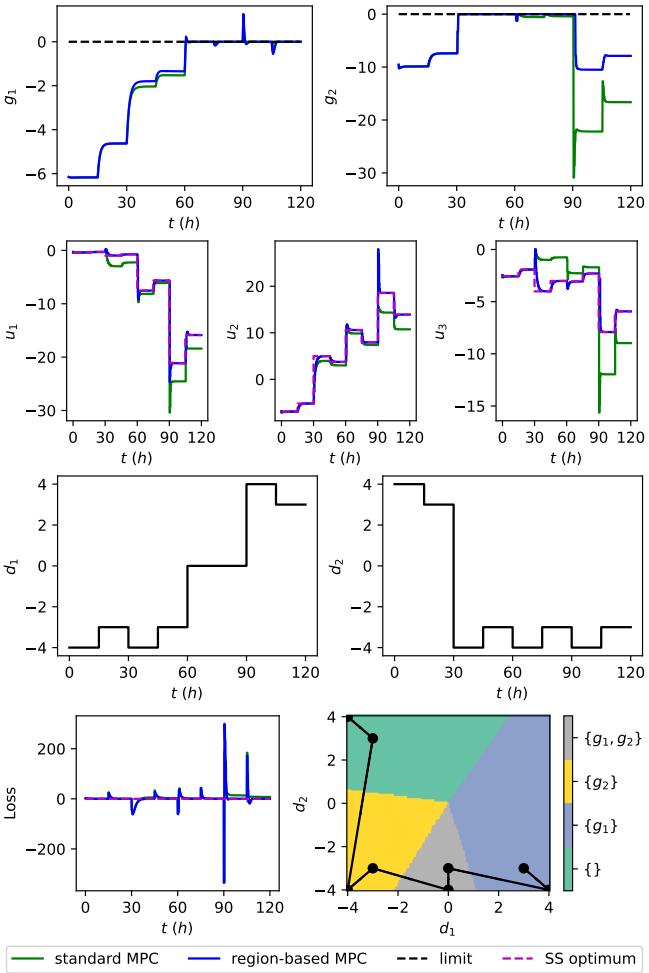

Figure 4: Dynamic simulation results for case study 1 - comparison between standard MPC (green) and the proposed region-based MPC (blue)

the system leaves the unconstrained region. Note that the lines delimiting the operating regions (blue) do not coincide with the optimal boundaries (magenta) at the partly constrained regions, because the use of fixed CVs is not optimal. Therefore, standard MPC does not guarantee control of the correct active constraints.

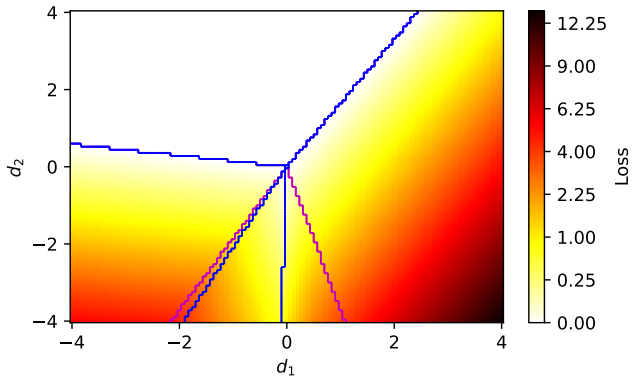

Figure 5: Economic loss of standard MPC for case study 1 as a function of disturbances. Magenta lines delimit optimal active constraint regions, blue lines delimit operating regions of standard MPC. Region-based MPC attains zero loss for all disturbance values.

It is worth mentioning that the proposed method does not rely on RTO updates for dealing with changes in active constraints. Instead, it relies on a switching logic for the CVs solely based on measurements and the nominal plant behavior, and on the self-optimizing property of the chosen CVs, which means that no setpoint updates are required for the system to operate optimally.

4.2. Case study 2 - Williams-Otto reactor

This case study is based on the process described by Williams and Otto [15], see Figure 6. It consists of a continuously stirred reactor tank with perfect level control, in which A and B are mixed, generating the main product of interest P, along with the less interesting product E and the undesired byproduct G. The three reactions are:

$$\begin{aligned} A + B &\xrightarrow{k_1} C & k_1 &= k_{0,1} e^{-E_1/T_r} \\ C + B &\xrightarrow{k_2} P + E & k_2 &= k_{0,2} e^{-E_2/T_r} \\ P + C &\xrightarrow{k_3} G & k_3 &= k_{0,3} e^{-E_3/T_r} \end{aligned}$$

The component mass balances result in the following system of ODEs:

$$\frac{dx_A}{dt} = \frac{F_A}{W} - \frac{(F_A + F_B)x_A}{W} - k_1 x_A x_B \tag{24a}$$

$$\frac{dx_B}{dt} = \frac{F_B}{W} - \frac{(F_A + F_B)x_B}{W} - k_1 x_A x_B - k_2 x_C x_B \tag{24b}$$

$$\frac{dx_C}{dt} = -\frac{(F_A + F_B)x_C}{W} + 2k_1 x_A x_B - 2k_2 x_C x_B - k_3 x_P x_C \tag{24c}$$

$$\frac{dx_P}{dt} = -\frac{(F_A + F_B)x_P}{W} + k_2 x_C x_B - 0.5 k_3 x_P x_C \tag{24d}$$

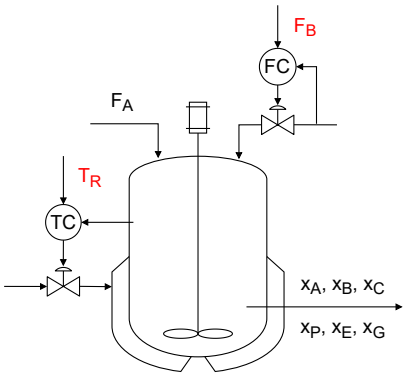

Figure 6: Schematic representation of Williams-Otto reactor, with MVs in red

$$\frac{dx_E}{dt} = -\frac{(F_A + F_B)x_E}{W} + 2k_2 x_C x_B \tag{24e}$$

$$\frac{dx_G}{dt} = -\frac{(F_A + F_B)x_G}{W} + 1.5 k_3 x_P x_C \tag{24f}$$

Here, x_i represents the mass fraction of component i. The model parameters for this case study are summarized in Table 3. The economic optimization problem to be considered is:

$$\begin{aligned} \min_u \; & J = p_A F_A + p_B F_B - (F_A + F_B)\left[p_P(1 + \Delta p_P)x_P + p_E x_E\right] \\ \text{s.t.} \quad & x_E \le 0.30 \\ & x_A \le 0.12 \end{aligned} \tag{25}$$

Parameter	Value
W	2105 kg
$k_{0,1}$	1.6599×10^{-6} kg/s
$k_{0,2}$	7.2117×10^{-8} kg/s
$k_{0,3}$	2.6745×10^{-12} kg/s
E_1	6666.7 K
E_2	8333.3 K
E_3	11111 K
p_A	79.23 \$/kg
p_B	118.34 \$/kg
p_P	1043.38 \$/kg
p_E	20.92 \$/kg

Table 3: Model parameters for case study 2

The available DOFs for operation are $u = \begin{bmatrix} F_B & T_R \end{bmatrix}^T$, namely the mass inflow of pure B and the reactor temperature, and the considered disturbances are $d = \begin{bmatrix} F_A & \Delta p_P \end{bmatrix}^T$, namely the mass inflow of pure A and the relative variation of the price p_P. Similar to case study 1, we can visualize the active constraint regions as a function of the two disturbances, as shown in Figure 7.

We choose to scale the constraints relative to the maximum optimal constraint value in the disturbance window shown in Figure 7. This gives the following scaled problem:

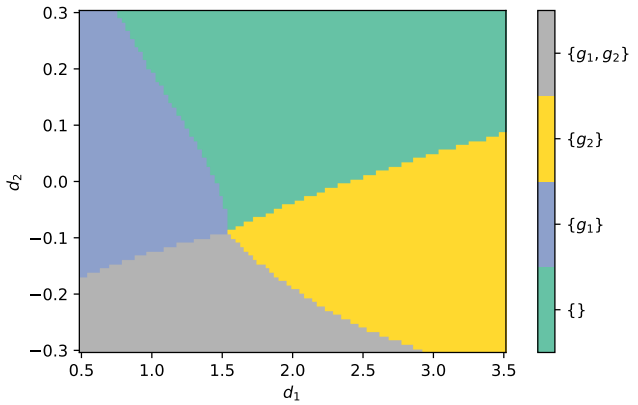

Figure 7: Active constraint regions for case study 2 as a function of disturbances

$$\begin{aligned} \min_u J &= p_A F_A + p_B F_B - (F_A + F_B)\left[p_P(1+\Delta p_P)x_P + p_E x_E\right] \\ \text{s.t.} \quad g_1 &= \frac{x_E - 0.30}{0.0287329} \leq 0 \\ g_2 &= \frac{x_A - 0.12}{0.0714527} \leq 0 \end{aligned} \tag{26}$$

The measurements are the two constraints, the fraction of component P and the price of P, that is, $y = \begin{bmatrix} g_1 & g_2 & x_p & \Delta p_P \end{bmatrix}^T$. To design the region-based MPC, we must first obtain the matrices J_{uu}, J_{ud}, G^y, G^y_d, and G_u, which depend on the operating point. One simple approach is to calculate those matrices at the design point and keep them constant during operation. We shall consider this strategy using two design points, in order to evaluate the effect of nonlinearity on the proposed framework.

For the MPC controllers, we use a linear approximation of the dynamic model at the design point, and estimate disturbances and additional integrating states using a linear Kalman filter that ensures zero offset [16]. The use of a mismatch between the internal linear MPC model and the true nonlinear plant model is to show that the steady-state economic performance is a result of a correct choice of CVs, and not necessarily of a correct dynamic process model.

The first set of simulations refers to a linearization at the vertex between the regions ($d^\star = [1.54265 - 0.0891]$), where the resulting H_0 is given by:

$$H_0 = \begin{bmatrix} -42.0785 & -36.0878 & 1153.68 & -126.066 \\ 3.42257 & -0.370313 & -232.921 & 0.369661 \end{bmatrix}$$

and the corresponding gradient projections and controller tunings are given in Tables 4 and 5, respectively.

The results are shown in Figures 8 to 10, From the dynamic simulation in Figure 8, we see that the behavior of standard MPC and region-based MPC is identical in the unconstrained region (until $t = 4$ h), but the operation at the subsequent region with g_1 active (from $t = 4$ to $t = 8$ h) highlights the difference between the approaches. The region-based MPC framework detects quite accurately the region change and switches

$\mathcal{A}$	$N^T_{\mathcal{A}}$
{}	$\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$
{1}	$\begin{bmatrix} 0.1208 & 0.9927 \end{bmatrix}$
{2}	$\begin{bmatrix} -0.0849 & 0.9964 \end{bmatrix}$
{1, 2}	-

Table 4: Optimal gradient projections for example 2 - linearization at vertex

the CVs accordingly, whereas standard MPC attempts to track the CVs from the unconstrained region, which is not always optimal. There is some loss with region-based MPC associated with the nonlinearity in the model. For the fully constrained region ($\{g_1, g_2\}$ from $t = 8$ to $t = 12$ h), the two MPC schemes behave similarly, attaining zero steady-state loss by taking all the constraints to their limit values. The region-based MPC attains this through direct constraint control, whereas standard MPC relies on its dynamic constraint handling, which has its own issues regarding stability and performance.

The steady-state behavior for both the region-based MPC and the standard MPC was simulated for the whole domain displayed in Figure 7, and the results are presented in Figures 9 and 10, respectively. Due to the linearization being performed at the vertex between the four regions, the description of the boundaries between the regions is fairly accurate for the region-based MPC in Figure 9, which illustrates the local exactness of the method. However, we can see a large economic loss at the unconstrained region as the system moves further from the reference point, and this can be attributed to the errors associated with Equation (12) for nonlinear systems. The standard MPC in Figure 10 does not reproduce the optimal behavior locally in terms of region boundaries, and it creates an economic loss peak at the region with g_1 active, which is not seen in the same magnitude for the region-based MPC. It can also be seen that there are some disturbance combinations with a smaller economic loss for standard MPC than for region-based MPC, although this is not the general trend. This curious behavior is a combination of the inaccuracy of the CVs calculated locally with the giving up of those CVs by the MPC algorithm, which in itself depends on the tuning parameters of the MPC. This fact is illustrated in Figure 11, where a different standard MPC tuning than that of Table 5 led to much worse overall performance in the constrained regions. For the proposed region-based MPC this is not an issue because the choice of CVs is consistent with the active constraints, and therefore the control offset will be zero at steady state, making the steady-state performance independent of the dynamic tuning.

We now design the region-based MPC and the standard MPC to operate around $d^\star = [2.0, +0.2]$, which lies in the interior of the unconstrained region. Here, the resulting H_0 is:

$$H_0 = \begin{bmatrix} -46.0296 & -32.6404 & 1577.81 & -96.6946 \\ 5.50426 & -2.61125 & -393.808 & 0.342395 \end{bmatrix}$$

and the corresponding gradient projections and controller tun-

Parameter	$\mathcal{A}$	Value
$Q_{\mathcal{A}}$	$\{\}$	$\text{diag}([0.01, 1.0])$
	$\{1\}$	$\text{diag}([30.0, 1.0])$
	$\{2\}$	$\text{diag}([30.0, 1.0])$
	$\{1,\ 2\}$	$\text{diag}([3.0, 30.0])$
$R_{\mathcal{A}}$		$\text{diag}([0.5, 0.02])$
N		60
Δt		0.0333 h
Q^e		$\text{diag}([10^{-3}, 10^{-3}, 10^{-3}, 10^{-3}, 10^{-3}, 10^{-3}, 8, 8, 0.8, 0.8])$
R^e		$\text{diag}([10^{-12}, 10^{-12}, 10^{-12}, 10^{-12}])$

Table 5: Tuning of controllers and estimator for example 2 - linearization at vertex

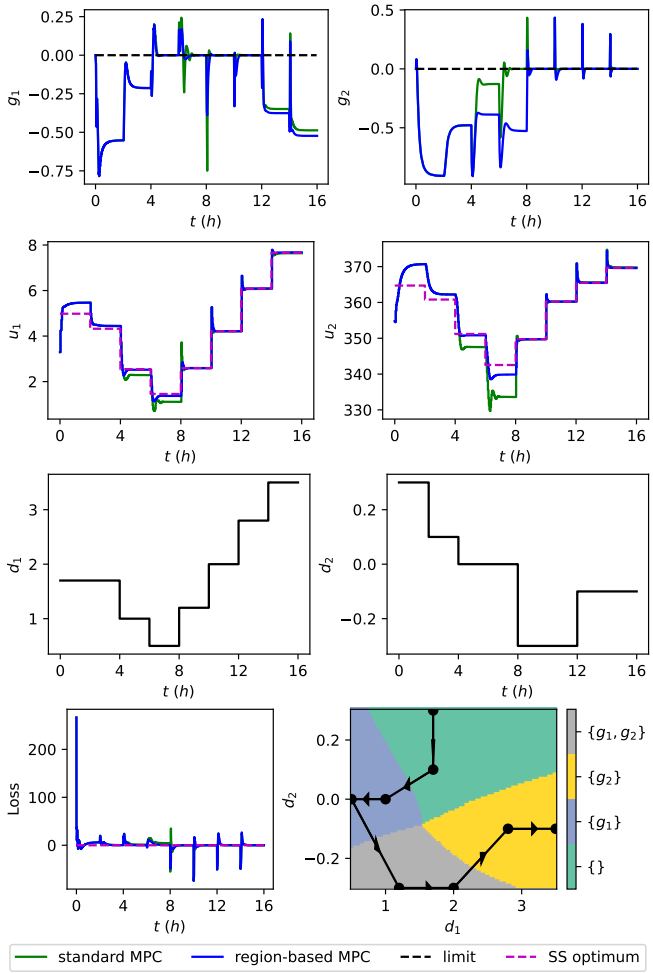

Figure 8: Dynamic simulation results for case study 2 - comparison between standard MPC (green) and the proposed region-based MPC (blue) - linearized at vertex

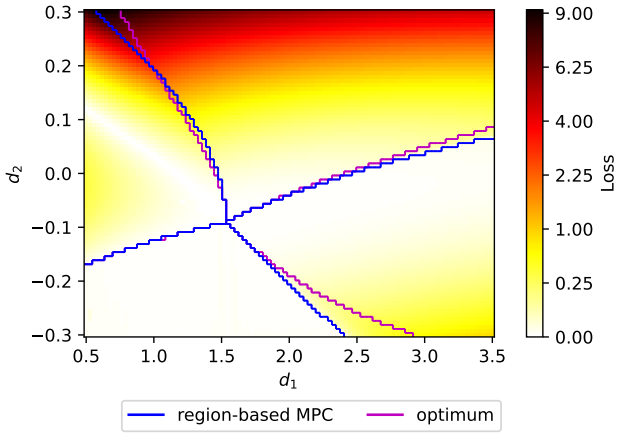

Figure 9: Steady-state economic loss for region-based MPC on case study 2 - linearized at vertex

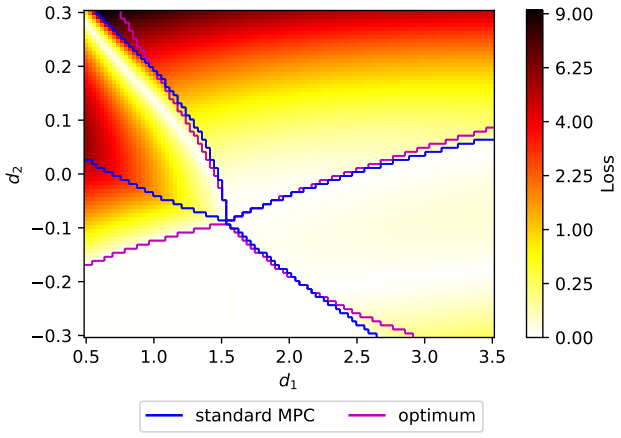

Figure 10: Steady-state economic loss for standard MPC on case study 2 - linearized at vertex

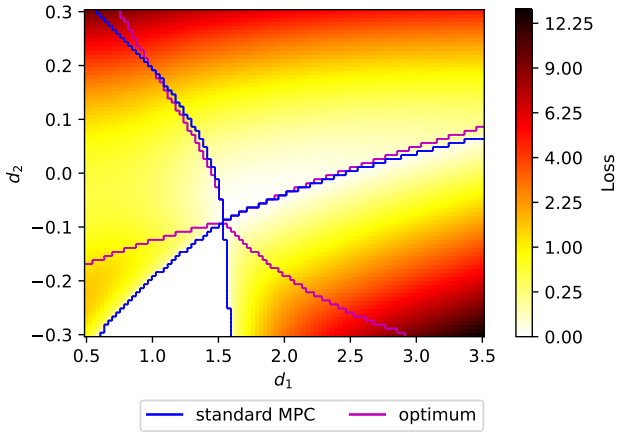

Figure 11: Steady-state economic loss for standard MPC on case study 2 - linearized at vertex, $Q = \text{diag}([5 \times 10^{-4},\ 5])$

ings are given in Tables 6 and 7, respectively.

$\mathcal{A}$	$N_{\mathcal{A}}^T$
{}	$\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$
{1}	$\begin{bmatrix} 0.1110 & 0.9938 \end{bmatrix}$
{2}	$\begin{bmatrix} -0.1685 & 0.9857 \end{bmatrix}$
{1, 2}	-

Table 6: Optimal gradient projections for example 2 - linearization at $d^\star = [2.0, +0.2]$

Parameter	$\mathcal{A}$	Value
$Q_{\mathcal{A}}$	{}	$\text{diag}([5 \times 10^{-4},\ 5])$
	{1}	$\text{diag}([30.0, 1.0])$
	{2}	$\text{diag}([30.0, 1.0])$
	{1, 2}	$\text{diag}([3.0, 30.0])$

Table 7: Tuning of controllers and estimator for example 2 - linearization at $d^\star = [2.0,\ +0.2]$ (omitted parameters are the same as in Table 5)

For this linearization, the results are shown in Figures 12 to 14. In this case, the region-based MPC overall gives a much smaller economic loss on the partly constrained regions compared to standard MPC. Also, the regions obtained in Figure 13 are shaped similarly to the optimal regions, which does not happen with the standard MPC in Figure 14. Because the linearization of the system happened in the interior of the unconstrained region, the economic loss on that region is smaller when compared to that of Figure 9, while not resulting in a larger loss for the remaining regions in the case of the region-based MPC.

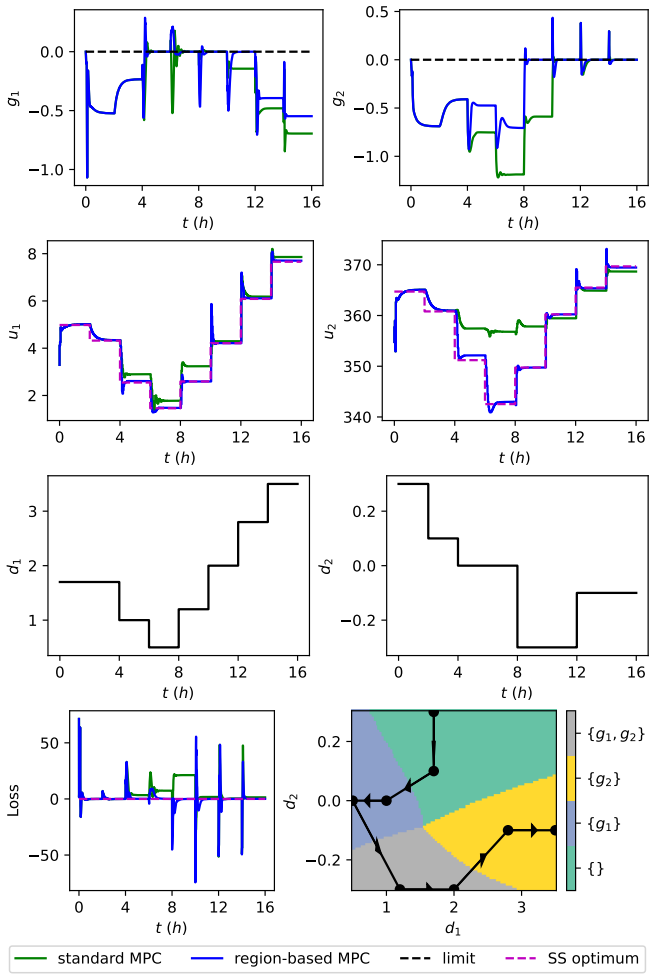

Figure 12: Dynamic simulation results for case study 2 - comparison between standard MPC (green) and the proposed region-based MPC (blue) - linearized at $d^\star = [2.0, +0.2]$

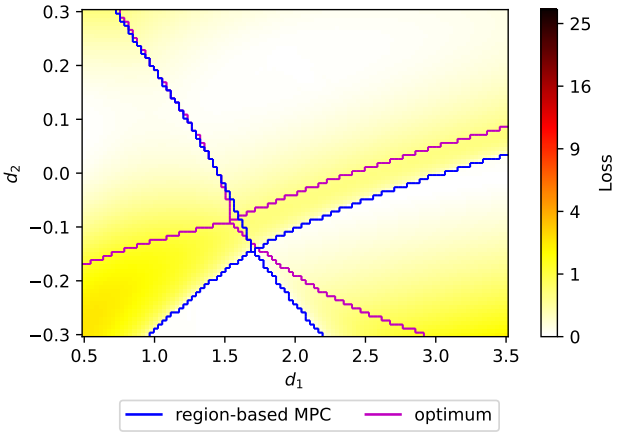

Figure 13: Steady-state economic loss for region-based MPC on case study 2 - linearized at $d^\star = [2.0, +0.2]$

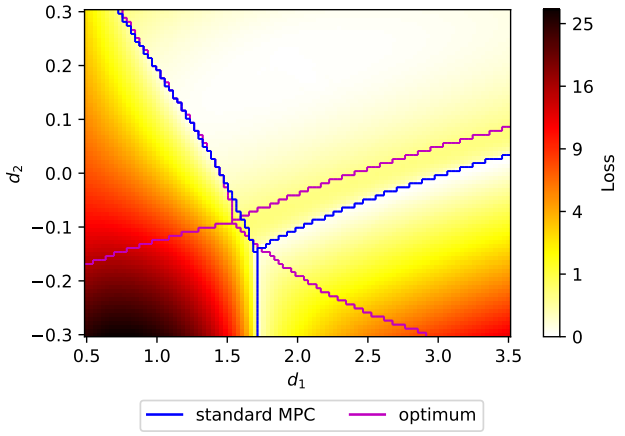

Figure 14: Steady-state economic loss for standard MPC on case study 2 - linearized at $d^\star = [2.0, +0.2]$

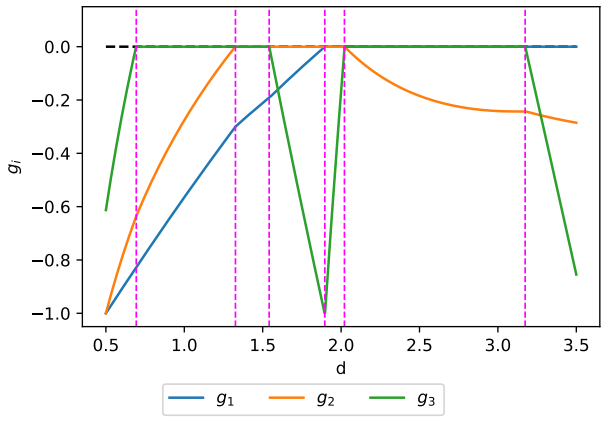

Figure 15: Active constraint regions for case study 3 in terms of optimal constraint values as a function of d (magenta dashed lines represent region switches)

4.3. Case study 3 - Williams-Otto reactor revisited

We now revisit the previous case study, to consider a case with three rather than two constraints and only one disturbances. This is a case where stanbdard decentralized selector-based region-based control would not work, because $n_g > n_u$ [17]. With only one disturabances it is easier to visualize some of results. We consider the following optimization problem:

$$\begin{aligned} \min_{u} \quad & J = p_A F_A + p_B F_B - (F_A + F_B)(p_P x_P + p_E x_E) \\ \text{s.t.} \quad & F_B \leq 4.0 \\ & T_r \leq 355.0 \\ & x_G \leq 0.105 \end{aligned} \tag{27}$$

Here, the operational constraints are related to maximum allowed values for F_B, T_r, and x_G. The MVs are the same, $u = \begin{bmatrix} F_B & T_r \end{bmatrix}^T$, but we only consider one disturbance, $d = F_A$, which is in the reange $0.5 \leq d \leq 3.5$. We again normalize the constraints. The normalized problem is given by:

$$\begin{aligned} \min_{u} \quad & J = p_A F_A + p_B F_B - (F_A + F_B)(p_P x_P + p_E x_E) \\ \text{s.t.} \quad & g_1 = \frac{F_B - 4.0}{2.68018} \leq 0 \\ & g_2 = \frac{T_r - 355.0}{9.55095} \leq 0 \\ & g_3 = \frac{x_G - 0.105}{0.00411912} \leq 0 \end{aligned} \tag{28}$$

Figure 15 present the active constraint regions as a function of the disturbance F_A. It can be seen that all possible feasible combinations of active constraints appear in the considered disturbance range.

This problem has three constraints and two MVs, and therefore a fixed pairing between constraints and MVs would not account for all possible active constraint regions. For instance, if u_1 is paired to g_1, u_2 is paired to g_2, and g_3 may be active at the same time as the other constraints, control of g_3 must have some sort of adaptive pairing, if decentralized control is to be achieved (see Bernardino et al. [18] for an example).

In Figure 16, we present results for the dynamic simulation of the system. It should be noted that the tuning used for the standard MPC is done such that it can operate acceptably even when constraints become active, which hinders the overall attainable performance. Because the region-based MPC can be tuned independently for every active constraint region, dynamic performance can be expected to be better.

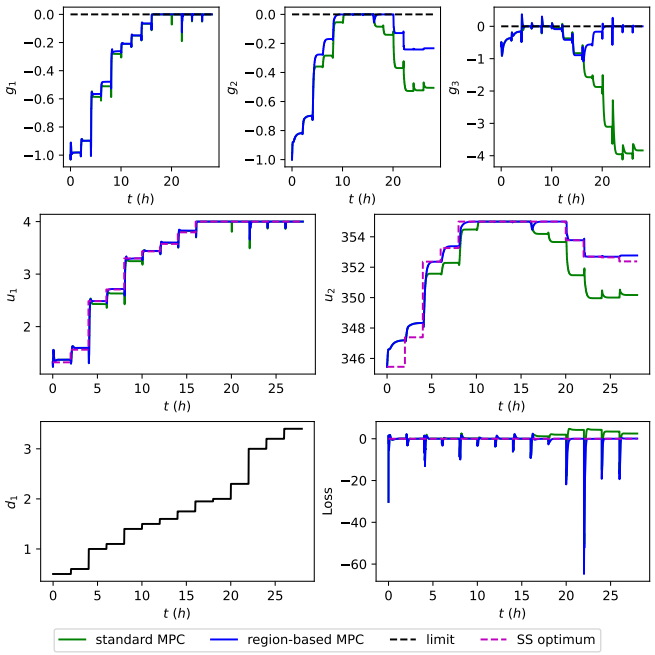

Figure 16: Dynamic simulation results for case study 3 - comparison between standard MPC (green) and the proposed region-based MPC (blue)

In Figures 17 to 19 we compare the steady-state behavior of the region-based MPC and the standard MPC, respectively, in terms of the constraints' values and economic losses. We can see that the linearization strategy is such that the operation is

exactly optimal at $d = d^\star = 1.0$ for the region-based MPC, but the same cannot be said for the standard MPC. This is because the system was linearized at a partly constrained region, and while the region-based MPC is able to use the correction $J_u^\star$ in Equation (13), the same correction applied to the standard MPC does not lead to optimal operation. In addition, standard MPC performs poorly at driving the system to the correct constraints to be controlled, which leads to huge discrepancies with relation to optimality for $d > 1.9$ and large economic losses.

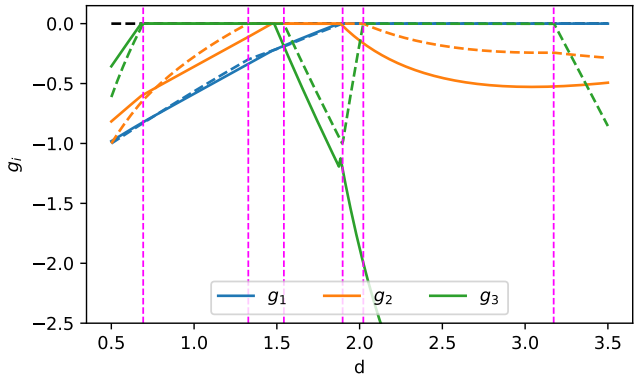

Figure 17: Steady-state constraint values for standard MPC on case study 3 (optimal values as dashed lines)

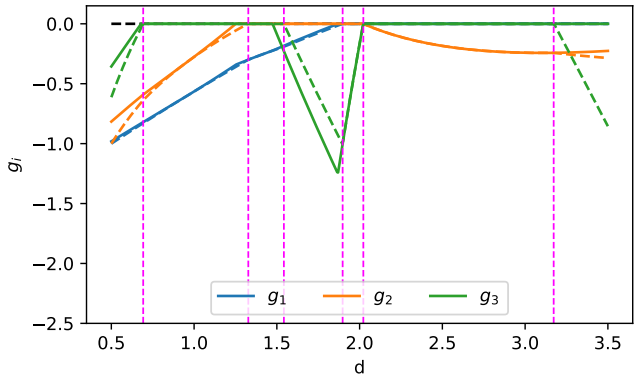

Figure 18: Steady-state constraint values for region-based MPC on case study 3 (optimal values as dashed lines)

5. Discussion

5.1. *Exact local method for H (for case with measurements error and any number of measurements)*

The gradient esitmationm in this work is paper on the nullspace method of self-optimizing control and therefore disregards measurement error. In another paper [19], we propose a gradient estimation method that accounts for static measurement error (n^y), based on the exact local method, which results in an optimal linear combination of any number (n_y) of measurements.

TODO: The rest of the discussion is very long. Try to reduce!

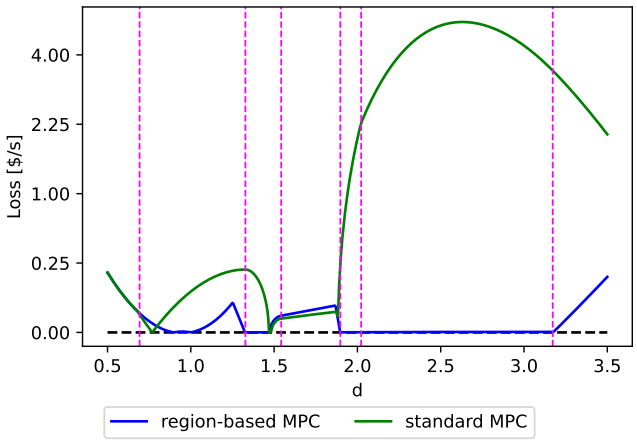

Figure 19: Closed-loop steady-state economic loss for controllers on case study 3 (y-axis is scaled quadratically for better visualization)

5.2. *Optimal operation under changing active constraints*

The region-based MPC proposed in this work depends on a logic element that detects the current active constraint set so that the corresponding self-optimizing CVs, defined by $CV_{\mathcal{A}}$, are selected and controlled. The use of simple logic elements for changing control structures is very common among practitioners, but it is not generally clear how to use these elements optimally [20]. This issue has received attention in recent developments, especially when a low number of switching variables is involved [21, 22]. In general, it is necessary that one has analyzed the range of disturbances to be handled by the control structure, in order to propose a switching strategy, along with a pairing between MVs and CVs, that accommodates all control objectives. That procedure is, however, dependent on the case study and the engineering insight, and one may find cases where a decentralized strategy would be impossible or too complex to be considered in practice [18].

In addition to this issue, even if the disturbance range is such that constraints paired to the same MV are never active at the same time, the whole control structure should in principle be changed according to which set of constraints is active because the optimal CVs related to the unconstrained degrees of freedom will change. Therefore, in terms of self-optimizing control of such systems, we can say that the general case of a switching logic between CVs must be in some sense centralized, as the complexity of the decision process becomes combinatorial. Because of these intrinsic limitations of decentralized SOC structures, centralized approaches for SOC become vital for guaranteeing optimal operation of systems with several changing constraints.

5.3. *Estimation of active constraints*

To determine the active set during operation, we use the method by Woodward et al. [7], which is proven optimal for measured gradients. In this work, the cost gradient is estimated through a linear combination of the measurements, which is consistent with the CVs being used.

Another approach for detecting changes in the active constraint region, is to tracks the values of the CVs in the neighboring regions. [6, 23]. The CVs determined for each region must be consistent to result in a unique solution to the switching problem. If this is not the case, one may encounter multiple steady-state solutions and lack of convergence where the control structures switches indefinitely. This was observed when applying this approach to the case studies.

On the other hand, the solution presented in this work relies on a single model realization, and all CVs obtained from it are consistent.

5.4. Use of direct measurements

To simplify the active constraint prediction, one mau consider basing the gardient esimate based .... In Algorithm 1, we describe the estimation of the active set considering the predicted closed-loop behavior (TODO: you keep using the "close-loop"; I think meaning that you are looking at the final steady state). of the internal model. This is done to accelerate convergence, and one could instead use the current value of the measurements y for determining both cost gradient estimation and constraint violation. The use of direct measurements is illustrated in Figure 20, (TODO: ???? not clear what you are trying to say) compared to the approach used in the course of this work, with all other parameters being the same across the simulations.

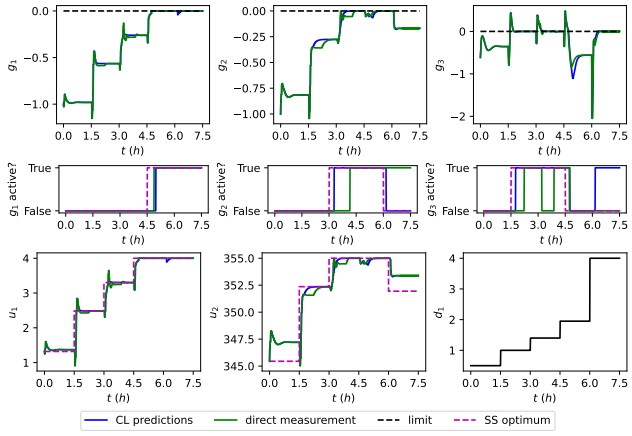

Figure 20: Comparison between region-based MPC with active set detection from closed-loop predictions (blue) and from direct measurements (green)

We can see that the use of direct measurements on the active set detection algorithm gives a worse overall closed-loop behavior in this case. For the first step change on the disturbance (t = 1.5 h), even though g_3 is dynamically violated, the cost gradient estimated directly from the measurements dictates that it will be given up at steady state, and the constraint is therefore not deemed active. When the system is close to steady state, the cost gradient estimate becomes more accurate, and the constraint is then considered active. A similar behavior happens for the second step change on the disturbance (t = 3 h). For the last disturbance value (from t = 6 h), the closed-loop system faces problems, as the e estimated active set does not match the operation of the system, and the state estimator is in conflict with the dynamic constraint. Because of this, we see oscillation in the result, and the system does not seem to converge.

TODO: The following paragraph is unclear and not needed.

It is important to consider that the cost gradient is only correctly evaluated at steady state, because the derivation of H_0, see Equation (12), is based on the steady-state optimization problem. Indeed, the cost gradient J_u is only defined at steady state. This means that one may find inconsistent results for the cost gradient during transients, and the use of predicted steady-state values would be in general beneficial, which is why the estimator is also important for active set detection. However, we remark that the state estimator is not used to re-optimize the plant like in the hybrid RTO scheme in Krishnamoorthy et al. [24], and therefore estimating the disturbances d is not necessary. We simply require that the estimator matches the measurements at steady state, similar to the conditions for offset-free control [8].

5.5. Tuning of region-based MPC

TODO: Also this is unclear to me:

The region-based MPC can be seen as a set of multivariable feedback controllers coordinated by a logic element. This logic element introduces an additional information loop, besides the feedback controller itself, which may cause stability problems. Rapid changes in which of the controllers is active may occur from the interaction between the switching element and the closed-loop dynamics, generating high-frequency, self-sustained switching. This is a known issue in closed-loop systems with selectors or other logical elements, and it may be counteracted by restricting how fast the logical element may change, leading to overall system stability [25]. In this work, this is attained by the tuning parameter N_{sw}. Additionally, viewing the controllers of each region as independent is an opportunity for independent tuning of those controllers, as different CVs usually have different dynamic behaviors. Careful evaluation of MPC tunings for different regions is therefore advised, so that good dynamic performance is attained in all relevant operating conditions.

5.6. Comparison between region-based MPC and other MPC approaches

The simulations verify that standard setpoint-tracking MPC is unable to deal with changing steady-state constraints. To satisfy a steady-state constraint which is not in the nominal region, the standard MPC gives up on tight control of its CVs, usually all at the same time. In our simulations, the prioritization of CVs is automatically done by the tuning the MPC control weights, and the steady-state offset of the CVs will therefore be indirectly determined by such tuning. This aspect is not usually prioritized in the design of MPC controllers, and therefore we have no quantitative control over how much offset we tolerate for each CV. Another possibility, which is used in industrial implementations of MPC, is solving a sequence of steady-state calculations, assessing constraint satisfaction, before solving the MPC problem itself [26]. This allows for adapting the MPC problem, changing control specifications so that constraints are

considered when necessary, and avoiding the use of dynamic constraints that may cause stability issues. The problem here becomes determining the control specifications, which is often based on process experience.

The present work has not focused on integrating the proposed tool with RTO, as other works have covered [27]. Instead, the region-based MPC was formulated to be independent of the RTO layer, such that it operates near optimally without its updates. Naturally, the proposed tool can be integrated with RTO, by updating the gain matrices and reference values in Figure 2. Because these updates are associated with the steady-state conditions of optimality, the economic performance of the region-based MPC will be as good as the quality of these updates.

Some MPC frameworks (for example, that of Rawlings [1]) include a target calculation block, which will define to what steady state the MPC will converge. In these frameworks, it seems possible that the approach presented in this work can be used at the target calculation block only, and the MPC problem remains unchanged for every active constraint region. The main benefit of this would be that the stability properties of the MPC problem would remain the same regardless of the detected active set. This does not completely solve the stability issue, as the estimator and the target calculator blocks must converge, but it would still be an appealing approach.

We must also note that the proposal of this work is fundamentally different from that of centralized approaches such as economic model predictive control (EMPC). In these approaches, the dynamic and economic problems are solved together, which requires a high level of detail in the available dynamic model [2]. In the proposed region-based MPC, we only require a reasonable dynamic model to ensure closed-loop stability for the tracking of CVs and an accurate economic steady-state problem that will define these CVs.

6. Conclusion

A framework for self-optimizing control under changing active constraints was presented, see Figure 2. Its main elements are (1) the active set detection block (see Algorithm 1), and (2) the design of self-optimizing CVs in each active constraint region for the unconstrained degrees of freedom, see $c_{\mathcal{A}} = N_{\mathcal{A}}^T H_0 y$ in Equation (14) and the respective setpoint in Equation (15). In this paper, we estimated the cost gradient J_u using the nullspace method from self-optimizing control, by obtaining a measurement combination matrix H_0. More generally, with measurement bias and any number of measurements y, it is recommended to obtain H_0 for estimating the cost gradient J_u using the exact local method [19]. The setpoints $c_{\mathcal{A}}^{sp}$ are calculated based on the nominal operating point and were not updated during the simulations for the three case studies, to show the self-optimizing nature of the chosen CVs. We highlight that the switching of control objectives is done without the need for pairing MVs and CVs and without the need for RTO updates, making it applicable to a wide class of problems.

References

[1] J. B. Rawlings, Tutorial overview of model predictive control, IEEE control systems magazine 20 (2000) 38–52.

[2] M. Ellis, H. Durand, P. D. Christofides, A tutorial review of economic model predictive control methods, Journal of Process Control 24 (2014) 1156–1178.

[3] S. Skogestad, Plantwide control: The search for the self-optimizing control structure, Journal of process control 10 (2000) 487–507.

[4] J. Jäschke, Y. Cao, V. Kariwala, Self-optimizing control–A survey, Annual Reviews in Control 43 (2017) 199–223.

[5] J. E. A. Graciano, J. Jäschke, G. A. Le Roux, L. T. Biegler, Integrating self-optimizing control and real-time optimization using zone control MPC, Journal of Process Control 34 (2015) 35–48.

[6] J. Jäschke, S. Skogestad, Optimal controlled variables for polynomial systems, Journal of Process Control 22 (2012) 167–179.

[7] L. Woodward, M. Perrier, B. Srinivasan, Real-time optimization using a jamming-free switching logic for gradient projection on active constraints, Computers & Chemical Engineering 34 (2010) 1863–1872.

[8] U. Maeder, F. Borrelli, M. Morari, Linear offset-free model predictive control, Automatica 45 (2009) 2214–2222.

[9] D. Simon, Optimal state estimation: Kalman, H infinity, and nonlinear approaches, John Wiley & Sons, 2006.

[10] D. Q. Mayne, Model predictive control: Recent developments and future promise, Automatica 50 (2014) 2967–2986.

[11] I. J. Halvorsen, S. Skogestad, J. C. Morud, V. Alstad, Optimal selection of controlled variables, Industrial & Engineering Chemistry Research 42 (2003) 3273–3284.

[12] V. Alstad, S. Skogestad, E. S. Hori, Optimal measurement combinations as controlled variables, Journal of Process Control 19 (2009) 138–148.

[13] J. Jäschke, S. Skogestad, NCO tracking and self-optimizing control in the context of real-time optimization, Journal of Process Control 21 (2011) 1407–1416.

[14] J. A. E. Andersson, J. Gillis, G. Horn, J. B. Rawlings, M. Diehl, CasADi – A software framework for nonlinear optimization and optimal control, Mathematical Programming Computation 11 (2019) 1–36. doi:10.1007/s12532-018-0139-4.

[15] T. J. Williams, R. E. Otto, A generalized chemical processing model for the investigation of computer control, Transactions of the American Institute of Electrical Engineers, Part I: Communication and Electronics 79 (1960) 458–473.

[16] G. Pannocchia, M. Gabiccini, A. Artoni, Offset-free MPC explained: Novelties, subtleties, and applications, IFAC-PapersOnLine 48 (2015) 342–351.

[17] L. F. Bernardino, S. Skogestad, Decentralized control using selectors for optimal steady-state operation with changing active constraints, Journal of Process Control 137 (2024) 103194.

[18] L. F. Bernardino, D. Krishnamoorthy, S. Skogestad, Optimal operation of heat exchanger networks with changing active constraint regions, in: Computer Aided Chemical Engineering, volume 49, Elsevier, 2022, pp. 421–426.

[19] L. F. Bernardino, S. Skogestad, Optimal measurement-based cost gradient estimate for real-time optimization, 2024. *In preparation*.

[20] S. Skogestad, Advanced control using decomposition and simple elements, Annual Reviews in Control 56 (2023) 100903.

[21] A. Reyes-Lúa, S. Skogestad, Multi-input single-output control for extending the operating range: Generalized split range control using the baton strategy, Journal of Process Control 91 (2020) 1–11.

[22] D. Krishnamoorthy, S. Skogestad, Online process optimization with active constraint set changes using simple control structures, Industrial & Engineering Chemistry Research 58 (2019) 13555–13567.

[23] H. Manum, S. Skogestad, Self-optimizing control with active set changes, Journal of process control 22 (2012) 873–883.

[24] D. Krishnamoorthy, B. Foss, S. Skogestad, Steady-state real-time optimization using transient measurements, Computers & Chemical Engineering 115 (2018) 34–45.

[25] H. Lin, P. J. Antsaklis, Stability and stabilizability of switched linear systems: A survey of recent results, IEEE Transactions on Automatic control 54 (2009) 308–322.

[26] S. Strand, J. R. Sagli, MPC in Statoil–advantages with in-house technology, IFAC Proceedings Volumes 37 (2004) 97–103.

[27] P. d. A. Delou, R. Curvelo, M. B. de Souza Jr, A. R. Secchi, Steady-state real-time optimization using transient measurements in the absence of a dynamic mechanistic model: A framework of HRTO integrated with Adaptive Self-Optimizing IHMPC, Journal of Process Control 106 (2021) 1–19.