Journal of Process Control 142 (2024) 103298

Contents lists available at ScienceDirect

Journal of Process Control

journal homepage: www.elsevier.com/locate/jprocont

Optimal switching of MPC cost function for changing active constraints

Lucas Ferreira Bernardino, Sigurd Skogestad *

Department of Chemical Engineering, Norwegian University of Science and Technology, Trondheim, 7491, Norway

ARTICLE INFO

Keywords:
Self-optimizing control
Model predictive control
Optimal operation

ABSTRACT

Model predictive control (MPC) allows for dealing with multivariable interactions, known future changes and dynamic satisfaction of constraints. Standard MPC has a cost function that aims at keeping selected controlled variables at constant setpoints. This work considers systems where the *steady-state optimal* active constraints change during operation. This situation is not handled optimally by standard MPC which uses fixed controlled variables for the unconstrained degrees of freedom. We propose a simple framework that detects the constraint changes and updates the controlled variables accordingly. The unconstrained controlled variables are chosen to be the reduced cost gradients, which when controlled to zero minimizes the steady-state economic cost. In this paper, the nullspace method for self-optimizing control is used to estimate the cost gradient using a static combination of the 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.

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, future known disturbances and setpoint changes, 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 the setpoints for the controlled variables (CVs) based on economics, as presented in Fig. 1. It is possible to eliminate the controlled variables by combining the RTO layer, the supervisory control (MPC) layer and the regulatory control (PID) layer into one layer. This alternative, known as Economic MPC (EMPC) [2], is popular in academia but almost never used in industry 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 J^{ec} is the scalar economic cost function, $u \in \mathbb{R}^{n_u}$ is the vector of inputs or manipulated variables (MVs), $d \in \mathbb{R}^{n_d}$ is the vector of disturbances, and $g(u,d) \in \mathbb{R}^{n_g}$ is the vector of inequality constraints. Note that the model equations and corresponding 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 Eq. (1) results in the optimal plant inputs u^*, but as shown in Fig. 1 the RTO layer usually implements the optimization results in the form of setpoint updates CV^{sp} to the MPC layer. We refer to this type of implementation as setpoint-tracking MPC, or standard MPC. As discussed in more detail later, see Eq. (5), standard MPC uses a control 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 part uses this 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

* Corresponding author.
E-mail address: skoge@ntnu.no (S. Skogestad).

https://doi.org/10.1016/j.jprocont.2024.103298
Received 28 April 2024; Received in revised form 19 July 2024; Accepted 19 August 2024
Available online 27 August 2024

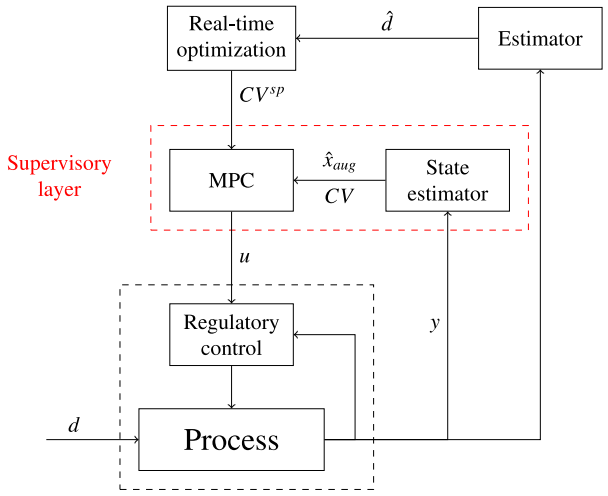

Fig. 1. Typical hierarchical control structure with *standard setpoint-tracking MPC* in the supervisory layer. The cost function for the RTO layer is J^{ec} and the cost function for the MPC layer is J^{MPC}. With no RTO layer (and thus constant setpoints CV^{sp}), this structure is not economically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

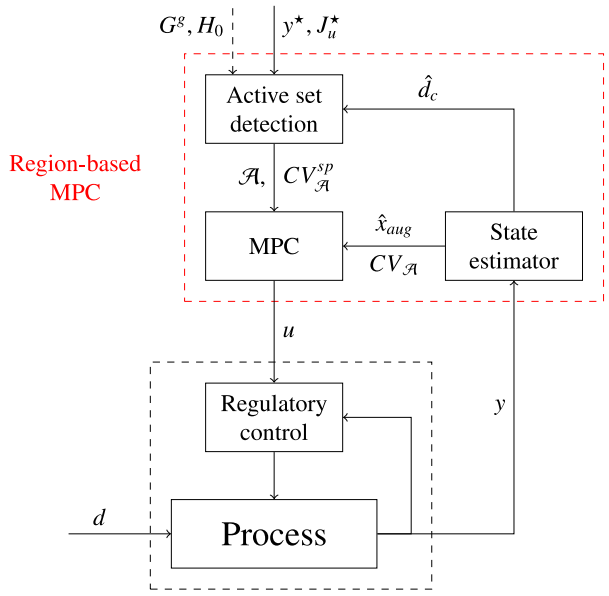

Fig. 2. Proposed region-based MPC structure with active set detection and change in controlled variables, see Eqs. (17) and (18). The possible updates from an upper RTO layer (y^*, J_u^* etc.) are not considered in the present work. Even with no RTO layer (and thus with constant setpoints $CV^{sp}_{\mathcal{A}}$ in each active constraint region), this structure is potentially economically optimal when there are changes in the active constraints.

problem or through the use of penalty terms. Additionally, most industrial MPC implementations include a target calculation block, which ensures that the setpoint that the MPC tracks is feasible at steady state [1].

In most 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 for changing disturbances as the main goal [3,4]. For quadratic problems, this gives for the unconstrained degrees of freedom, the controlled variables [5]:

$$CV = Hy$$

where y denotes the available process measurements (possibly including inputs u and measured disturbances d) 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]. For example, Graciano et al. [5] implemented MPC using the nominal self-optimizing CVs, that is, with the nominally active constraints. This can reject disturbances in the fast timescales and minimize the economic loss without the intervention of the RTO layer for the nominal set of active constraints. This is relatively simple to implement and also satisfies the constraints under changing operating conditions (disturbances). However, the approach of Graciano et al. [5] is not economically optimal for changing active constraints, because the optimal approach is to use different self-optimizing CVs for each set of active steady-state constraints [6], that is, to update the matrix H for each set of active constraints. Graciano et al. [5] write that: "When considering active set changes, there are two fundamental cases that must be handled by every method that attempts to give optimal operation: (1) Detecting when a new constraint becomes active: Here a degree of freedom is consumed for controlling the new constraint, and control of one of the unconstrained self-optimizing controlled variables must be given up. (2) Detecting when a previously active constraint becomes inactive: Here the constraint that was enforced previously must be released, and a new self-optimizing variable must be controlled instead". To handle case 2, the approach Graciano et al. [5] depends on RTO updates to be optimal. The authors write: "Our approach does not handle Case 2 in between RTO updates. That is, if a constraint becomes inactive in between RTO runs, it will be kept at its bound until the next RTO execution".

This work proposes a solution to this problem by providing a framework for self-optimizing control under changing active constraints (Fig. 2), labeled "region-based MPC". Which is optimal in both cases 1 and 2, without needing to rely on a RTO layer. Here, the self-optimizing CVs tracked by MPC are a function of the active constraint set. The detection of when a constraint becomes inactive (case 2) 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, whereas the proposed region-based MPC framework attains steady-state optimal operation, at least locally where nonlinearity in the model is negligible.

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, 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 Fig. 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:

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

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:

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

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 MPC optimization problem discretized using multiple shooting [10] can be of the form:

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

Here, CV are the selected controlled variables, x_k denotes the state at the kth time step, and $\phi(x_{k-1}, u_{k-1}, \hat{d}_c)$ is the result of the integration of the dynamic model in Eq. (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 the weight matrices Q and R are tuning parameters.

The output, state, and input constraints in Eq. (5) can be used to embed the RTO constraint $g(u, d) \le 0$ from Eq. (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 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 Eq. (5) has no stability guarantees, but it can converge if the prediction horizon N is large enough (the reader is referred to Mayne [11] 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. In the next section, we propose an optimal way of dealing with changing active constraints, without the need for RTO updates.

3. Region-based MPC framework

The structure of the proposed region-based MPC scheme is summarized in Fig. 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 $\mathcal{A}$, is sent to the MPC block, which uses a different set of predetermined 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 region-based MPC

The controlled variables (CVs) in region (or active constraint set) $\mathcal{A}$ are:

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

Here, $g_{\mathcal{A}}$ denotes the active constraints, and $c_{\mathcal{A}}$ denotes the unconstrained CVs (which we will see are the reduced gradients) for optimal steady-state operation. The dynamic control problem for the proposed region-based MPC is very similar to that of standard MPC in Eq. (5), except that the objective function changes for each region $\mathcal{A}$:

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

where $Q_{\mathcal{A}}$ and $R_{\mathcal{A}}$ are tuning parameters that can be chosen independently for each region $\mathcal{A}$, and $CV_{\mathcal{A}}^{sp}$ are setpoints that will be calculated next.

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 this purpose, we follow Halvorsen et al. [12] and consider a local QP approximation of the steady-state economic optimization problem:

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

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

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, all evaluated at the steady state reference point $\star$. The linearized expression in Eq. (9b) for the constraints is not used by Halvorsen et al. [12], but it is needed here because we consider changes in active constraints. Additionally, the measurements y are locally approximated by a linear steady-state model of the form:

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

The optimal solution to the problem in Eq. (9) can be implemented in a feedback fashion by selecting as unconstrained controlled variables c a static linear combination of the measurements [12]:

$$c = H y \tag{11}$$

Assuming that there are enough ($n_y = n_u + n_d$)[1] independent noise-free measurements, we can find an analytical expression for H based on the nullspace method [14,15]:

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

[1] For cases with fewer measurements (i.e., $n_y < n_u + n_d$) and/or measurement noise we may apply the more general "exact local method" [13] for obtaining the matrix H.

where † denotes the pseudo inverse. The associated controlled variables are:

$$c_0 = H_0 y \tag{13}$$

Actually, the expression for H_0 is non-unique [14], and the particular formula given in Eq. (12) is chosen because it can be used to obtain an optimal first-order estimate of the unconstrained cost gradient J_u [13,15]:

$$\hat{J}_u = c_0 - c_0^{sp} = H_0(y - y^\star) + J_u^\star \tag{14}$$

As we shall see, the unconstrained gradient estimate $\hat{J}_u$ is very useful when considering the constrained case. The setpoint $c_0^{sp} = H_0 y^\star - J_u^\star$, is calculated based on the reference steady state *. If the reference steady state is an optimal operating point in the fully unconstrained region, then $J_u^\star = 0$. However, if the reference steady state is not optimal, or (more commonly) if the reference steady state is in a constrained region, then $J_u^\star \neq 0$.

For the constrained case, we introduce the reduced gradient, defined by:

$$\text{Reduced Gradient} = N_{\mathcal{A}}^T J_u \tag{15}$$

where the projection matrix $N_{\mathcal{A}}$ is a basis for the nullspace of the active constraints gradient, i.e.:

$$G_{\mathcal{A}}^g N_{\mathcal{A}} = 0 \tag{16}$$

Optimal steady-state operation is achieved when the reduced gradient is zero [6,16], that is, when $N_{\mathcal{A}}^T J_u = 0$.

Using the gradient estimate in Eq. (14), we then have that the optimal CVs related to the unconstrained degrees of freedom are $c_{\mathcal{A}} = N_{\mathcal{A}}^T H_0 y$ with setpoints $c_{\mathcal{A}}^{sp} = N_{\mathcal{A}}^T c_0^{sp}$. In summary, the full set of controlled variables for each active constraint region $\mathcal{A}$ becomes:

$$CV_{\mathcal{A}} = \begin{bmatrix} g_{\mathcal{A}} \\ c_{\mathcal{A}} \end{bmatrix} = \begin{bmatrix} g_{\mathcal{A}} \\ N_{\mathcal{A}}^T H_0 y \end{bmatrix} \tag{17}$$

with corresponding setpoints:

$$CV_{\mathcal{A}}^{sp} = \begin{bmatrix} 0 \\ N_{\mathcal{A}}^T (H_0 y^\star - J_u^\star) \end{bmatrix} \tag{18}$$

This choice of CVs minimizes the steady-state economic loss around the reference point $(u^\star, d^\star)$. Furthermore, even if the current operating point has a different active set $\mathcal{A}$ than that of the reference point, the use of $CV_{\mathcal{A}}$ minimizes the steady-state loss, as long as the approximations in Eqs. (9) and (10) hold. However, for non-quadratic/nonlinear problems the reference point matters and there will be a steady-state loss, for example, compare Figs. 11 and 13 for case study 2 which are linearized in two different reference points.

We now discuss how to detect the active set using the available measurements, so as to select the correct controlled variables $CV_{\mathcal{A}}$.

3.2. Active constraint set detection

Woodward et al. [7] describes an active set detection algorithm for a feedback optimizing strategy which only depends on the current value of the cost gradient J_u, the constraints g, and the constraints gradient G^g. In our case, we assume that g is directly measured, J_u is estimated using Eq. (14), and G^g is a constant, see Eq. (9b). Thus, in our case the algorithm depends directly on the available measurements.

The steady-state cost gradient estimate $\hat{J}_u$ is obtained at the (expected) steady state (denoted y^{ss}) that would result when the CVs are driven to their setpoints. This expected steady state can be determined using Eq. (2), leading to:

$$\begin{cases} 0 = f(x^{ss}, u^{ss}, \hat{d}_c) \\ CV_{\mathcal{A}}(x^{ss}, \hat{d}_c) = CV_{\mathcal{A}}^{sp} \end{cases} \tag{19}$$

Algorithm 1 Active set estimation, adapted from Woodward et al. [7].

1: $\hat{J}_u \leftarrow H_0(y^{ss} - y^\star) + J_u^\star$ ▷ from Eq. (14)
2: $\mathcal{A}^k \leftarrow \mathcal{A}^{k-1} \cup \{\ i \mid g_i^{ss} \geq 0\ \}$
3: $\delta u^* \leftarrow$ solution of Eq. (20)
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 Eq. (19)
7: $\mathcal{A}^k \leftarrow \mathcal{A}'$
8: **end if**

from which we obtain $y^{ss} = h(x^{ss}, u^{ss}, \hat{d}_c)$ and $g^{ss} = h_g(x^{ss}, u^{ss}, \hat{d}_c)$. With these values, we apply the algorithm from Woodward et al. [7] to find the current set of active constraints.

The algorithm is summarized in Algorithm 1. In step 1 we first find the expected steady-state measurements y^{ss} by solving Eq. (19), and then compute $\hat{J}_u$ using Eq. (14). In step 2 the constraints predicted to be violated, i.e. $g_i^{ss} \geq 0$, are added to the estimated active set. With this augmented active set $\mathcal{A}$, we solve in step 3 the following optimization problem:

$$\delta u^* = \arg\min_{\delta u} \ -\delta u^T \delta u \quad \text{s.t.} \quad \begin{cases} G_{\mathcal{A}}^g \delta u \leq 0 \\ \delta u^T \delta u = -\delta u^T \hat{J}_u \end{cases} \tag{20}$$

The idea is to find the largest projection of the negative of the estimated cost gradient, $-\hat{J}_u$, onto the feasible directions, i.e. find 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 removed from the active set $\mathcal{A}$, as controlling these constraints would hinder economic improvement. If the resulting active set has more than n_u elements, it is infeasible, because we can at most control n_u constraints with n_u inputs. In step 6, one picks a subset which is predicted to be feasible ($g_i^{ss} \leq 0\ \forall i$) by evaluating the corresponding expected steady states with Eq. (19).

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 Eq. (20) is an NLP due to the quadratic equality constraint. The work of Woodward et al. [7] solves it using an efficient algorithm tailored to this problem, but here we solve the optimization problem with a general-purpose NLP solver (IPOPT). To prevent premature switching, for example due to noise or estimator dynamics, the estimated constraint set $\mathcal{A}$ is only selected as the new set of active constraints (resulting in switching of the CVs) if the estimated new set has remained the same for N_{sw} time steps. N_{sw} is a tuning parameter.

4. Case studies

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

$$J_{\mathcal{A}}^{MPC} = \int_0^{N\Delta t} \|CV_{\mathcal{A}} - CV_{\mathcal{A}}^{sp}\|_{Q_{\mathcal{A}}}^2 \, dt + \sum_{k=1}^{N} \|\Delta u_k\|_{R_{\mathcal{A}}}^2 \tag{21}$$

This mixed 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 [17].

4.1. Case study 1 - toy example

We first consider a toy example with a quadratic cost function and linear dynamic and constraint models, for which the proposed region-

based MPC is steady-state optimal. On the other hand, standard MPC is expected to result in steady-state losses is some constrained regions.

The system has two dynamic states x, three inputs (MVs) u, two disturbances d, two constraints g and five independent measurements y. The first two measurements are the two constraints. Note that $n_u = 3$, $n_d = 2$ and $n_y = 5$ which means that we satisfy the requirement $n_y = n_u + n_d$ needed to apply the nullspace method for estimating the gradient J_u (for cases with fewer measurements and/or measurement noise we should instead apply the more general exact local method [13]).

The economic objective and constraints are represented by the following steady-state optimization problem:

$$\min_u J^{ec} = \frac{1}{2}x^T \begin{bmatrix} 1 & 0 \\ 0 & 10 \end{bmatrix} x + \frac{1}{2}u^T \begin{bmatrix} 1 & -0.1 & -0.2 \\ -0.1 & 0.8 & -0.1 \\ -0.2 & -0.1 & 0.3 \end{bmatrix} u \quad \text{s.t.} \begin{cases} g_1 = x_1 - 0.8x_2 \le 0 \\ g_2 = u_1 + u_2 + u_3 \le 0 \end{cases} \tag{22}$$

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

$$\begin{cases} \dot{x} = \begin{bmatrix} -\frac{1}{\tau_1} & 0 \\ 0 & -\frac{1}{\tau_2} \end{bmatrix} x + \begin{bmatrix} \frac{0.2}{\tau_1} & 0 & 0 \\ 0 & \frac{0.2}{\tau_2} & 0 \end{bmatrix} u + \begin{bmatrix} \frac{1}{\tau_1} & 0 \\ 0 & \frac{1}{\tau_2} \end{bmatrix} d \\ y = \begin{bmatrix} g_1 \\ g_2 \\ x_2 \\ u_2 \\ u_3 \end{bmatrix} = \begin{bmatrix} 1 & -0.8 \\ 0 & 0 \\ 0 & 1 \\ 0 & 0 \\ 0 & 0 \end{bmatrix} x + \begin{bmatrix} 0 & 0 & 0 \\ 1 & 1 & 1 \\ 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} u \end{cases} \tag{23}$$

with $\tau_1 = 1$ and $\tau_2 = 2$ [h]. For the vector of measurements y, we consider 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 ($n_y = n_u + n_d$). We may eliminate the state variables x from the steady-state optimization problem in Eq. (22), since from Eq. (23) we have at steady state:

$$x = \begin{bmatrix} 0.2 & 0 & 0 \\ 0 & 0.2 & 0 \end{bmatrix} u + \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} d \tag{24}$$

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

$$\min_u J^{ec} = \frac{1}{2} u^T \underbrace{\begin{bmatrix} 1.04 & -0.1 & -0.2 \\ -0.1 & 1.2 & -0.1 \\ -0.2 & -0.1 & 0.3 \end{bmatrix}}_{J_{uu}} u + u^T \underbrace{\begin{bmatrix} 0.2 & 0 \\ 0 & 2 \\ 0 & 0 \end{bmatrix}}_{J_{ud}} d \quad \text{s.t. } g = \underbrace{\begin{bmatrix} 0.2 & -0.16 & 0 \\ 1 & 1 & 1 \end{bmatrix}}_{G^g} u + \underbrace{\begin{bmatrix} 1 & -0.8 \\ 0 & 0 \end{bmatrix}}_{G^g_d} d \le 0 \tag{25}$$

The following static measurements model is assumed:

$$y = \underbrace{\begin{bmatrix} 0.2 & -0.16 & 0 \\ 1 & 1 & 1 \\ 0 & 0.2 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}}_{G^y} u + \underbrace{\begin{bmatrix} 1 & -0.8 \\ 0 & 0 \\ 0 & 1 \\ 0 & 0 \\ 0 & 0 \end{bmatrix}}_{G^y_d} d \tag{26}$$

This problem has two inequality constraints and three MVs, so optimal operation will always have between one and three unconstrained degrees of freedom. As the system has only two disturbances, we can graphically illustrate the active constraint regions as in Fig. 3, where we can see all four possible combinations of active constraints. Note that this map is not needed to apply the proposed region-based method; it is only made to visualize the regions.

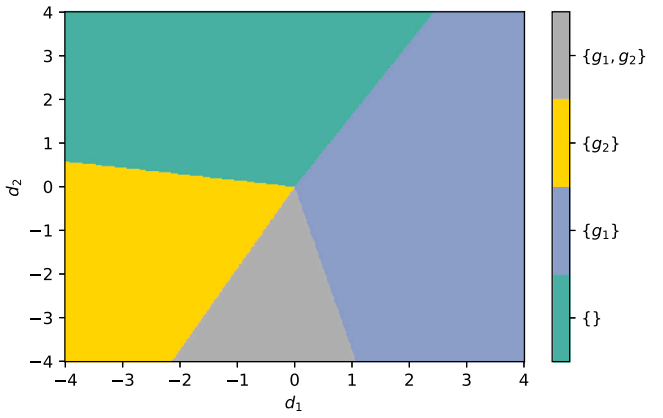

Fig. 3. The four active constraint regions for case study 1 as a function of the two disturbances. The upper green region is unconstrained.

Table 1
Gradient projection matrices for the four regions for case study 1.

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

Table 2
Tuning parameters for MPC controllers and estimator for case study 1. The standard MPC uses only the tuning parameters for the fully unconstrained region ($\mathcal{A} = \{\}$), that is $Q = \text{diag}([1, 1, 100])$.

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])
N_{sw}		2

For implementing a standard MPC controller, we follow the strategy described by Graciano et al. [5]. We design the self-optimizing controlled variables $c = H_0 y$ in the unconstrained region and from the nullspace method in Eq. (12) we obtain:

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

As the reference steady state, we choose the optimal operating point at $d^\star = [-4;\ \ +4]$. The measurements at the optimal operating point are $y^\star = [-6.17 \quad -9.86 \quad 2.62 \quad -6.91 \quad -2.56]^T$, and the resulting setpoints become $c^{sp} = H_0 y^\star = [0 \quad 0 \quad 0]^T$.

For the region-based MPC controller, we use the same unconstrained nominal point, resulting in the same $c_0 = H_0 y$. The projection matrices $N_{\mathcal{A}}$ for the constrained regions are presented in Table 1.

The tuning parameters for the MPC controllers are given in Table 2, where the standard MPC uses only the tuning parameters for the fully unconstrained region ($\mathcal{A} = \{\}$).

The closed-loop dynamic responses for both standard and region-based MPC are shown in Fig. 4. The responses go through eight sets of

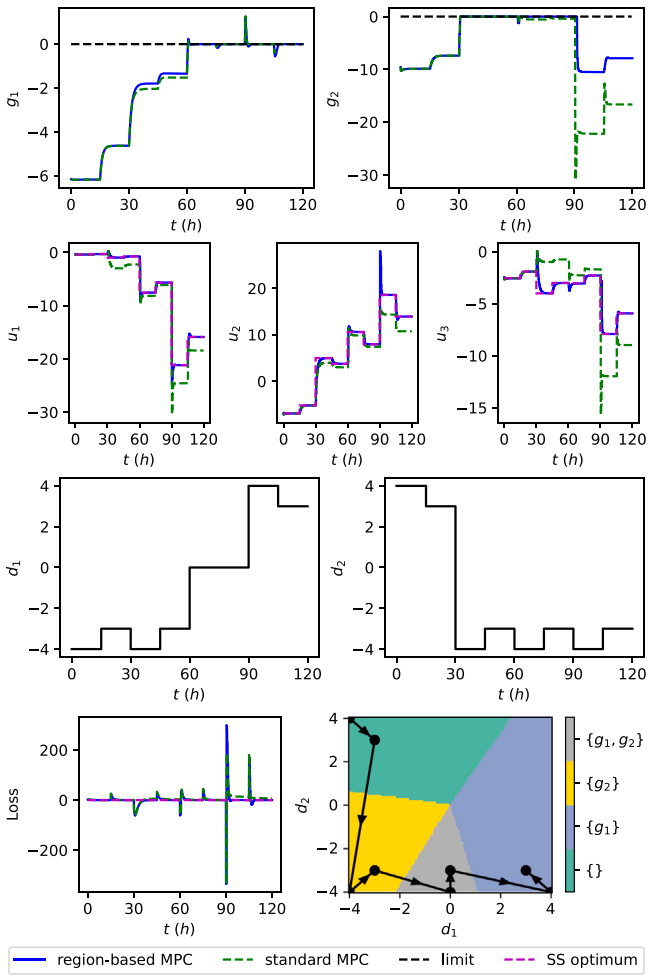

Fig. 4. Dynamic simulation results for case study 1 - comparison between standard MPC (green) and the proposed region-based MPC (blue).

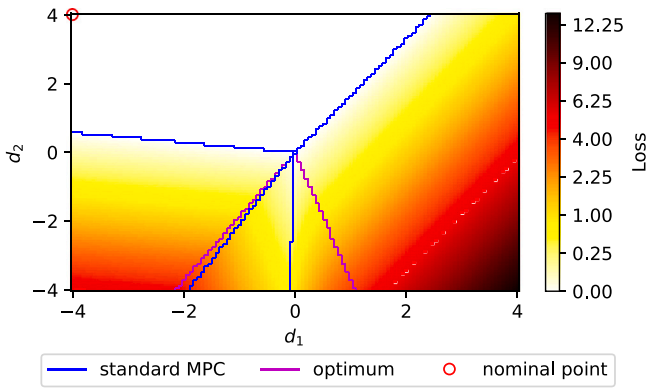

Fig. 5. Economic loss of standard MPC for case study 1 as a function of disturbances. Magenta lines delimit optimal active constraint regions, blue lines delimit operating regions of standard MPC. Region-based MPC attains zero loss for all disturbance values.

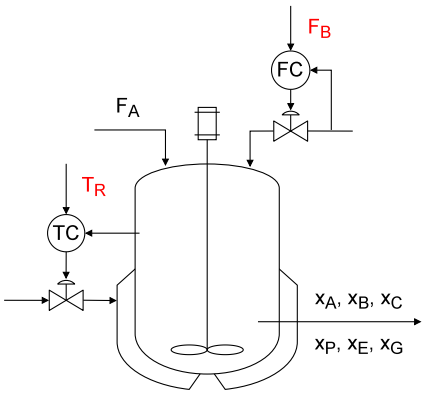

Fig. 6. Schematic representation of Williams-Otto reactor, with MVs in red.

disturbances; two inside each region as illustrated by the lower right plot.

With standard MPC, the controlled variables $c_0 = H_0 y$ obtained in the unconstrained region ($\mathcal{A} = \{\}$) are not steady-state optimal when applied to the constrained regions. This is most easily seen by comparing the inputs (u_1, u_2, u_3) from standard MPC (green) with the steady-state optimal inputs (magenta). On the other hand, the inputs obtained with the proposed region-based MPC (blue) are optimal at steady state in all four regions. In addition, the switching of CVs between regions is seen to be smooth. Both strategies satisfy the constraints at steady state but with some dynamic violations. The "loss" in the lower left plot in Fig. 4 is the difference between the current (dynamic) economic cost and the optimal value of the cost at steady state for the corresponding disturbances, and therefore both negative and positive dynamic spikes are possible.

The steady-state economic loss is of more interest. This loss is zero for the proposed region-based MPC. For standard MPC, the steady-state loss is shown in more detail in Fig. 5. As expected, the steady-state loss is strictly positive. It can be seen that the loss is nonzero whenever 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 controlled variables is not optimal. Therefore, standard MPC may result in control of constraints which should not be active.

It is worth mentioning that the proposed region-based MPC method does not rely on RTO updates of the setpoints for dealing with changes in active constraints. Instead, it relies on a switching logic for the CVs which is solely based on measurements.

4.2. Case study 2 - Williams-Otto reactor

This case study is based on the process described by Williams and Otto [18], see Fig. 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:

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

The component balances result in the following system of ODEs:

$$\frac{dx_A}{dt} = \frac{F_A}{W} - \frac{(F_A + F_B)x_A}{W} - k_1 x_A x_B \tag{27a}$$
$$\frac{dx_B}{dt} = \frac{F_B}{W} - \frac{(F_A + F_B)x_B}{W} - k_1 x_A x_B - k_2 x_C x_B \tag{27b}$$
$$\frac{dx_C}{dt} = -\frac{(F_A + F_B)x_C}{W} + 2k_1 x_A x_B - 2k_2 x_C x_B - k_3 x_P x_C \tag{27c}$$
$$\frac{dx_P}{dt} = -\frac{(F_A + F_B)x_P}{W} + k_2 x_C x_B - 0.5 k_3 x_P x_C \tag{27d}$$
$$\frac{dx_E}{dt} = -\frac{(F_A + F_B)x_E}{W} + 2k_2 x_C x_B \tag{27e}$$
$$\frac{dx_G}{dt} = -\frac{(F_A + F_B)x_G}{W} + 1.5 k_3 x_P x_C \tag{27f}$$

Table 3
Model parameters for case study 2.

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	11 111 K
p_A	79.23 $/kg
p_B	118.34 $/kg
p_P	1043.38 $/kg
p_E	20.92 $/kg

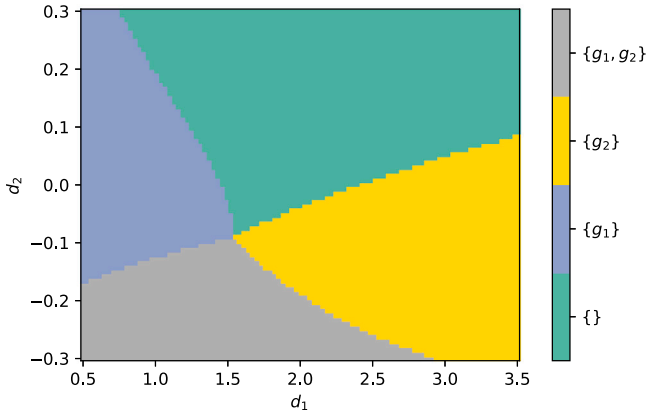

Fig. 7. The four active constraint regions for case study 2 as a function of the two disturbances.

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:

$$\begin{aligned} \min_u \; & J^{ec} = p_A F_A + p_B F_B - (F_A + F_B)\left[p_P(1+\Delta p_P)x_P + p_E x_E\right] \\ \text{s.t.} \quad & x_E \le 0.30 \\ & x_A \le 0.12 \end{aligned} \tag{28}$$

The available degrees of freedom for the MPC layer are $u = \begin{bmatrix} F_B & T_R \end{bmatrix}^T$, which are the mass inflow of pure B and the reactor temperature. Note that the physical degrees of freedom are valve positions, so in practice there will be a lower-layer regulatory (PID) layer, with a flow controller and a temperature controller. However, the regulatory layer is not considered in this paper, that is, we assume perfect regulatory control where the inflow rate and reactor temperature are set directly. As long as we have an acceptable time scale separation (typically, a factor 4 or higher [19]) between the MPC and regulatory layers, this assumption has no effect on the results.

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 optimize the system for various disturbances, and visualize in Fig. 7 the resulting active constraint regions as a function of the two disturbances.

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

$$\begin{aligned} g_1 &= \frac{x_E - 0.30}{0.0287329} \le 0 \\ g_2 &= \frac{x_A - 0.12}{0.0714527} \le 0 \end{aligned} \tag{29}$$

The measurements are the two constraints, the fraction of component P and the price of P, that is, $y = \begin{bmatrix} g_1 & g_2 & x_p & \Delta p_P \end{bmatrix}^T$.

Table 4
Gradient projections for case study 2 with linearization at vertex ($d_I^\star$ = [1.543; −0.089]).

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

Table 5
Tuning parameters for MPC controllers and estimator for case study 2 - linearization at vertex ($d_I^\star$ = [1.543; −0.089]).

Parameter	$\mathcal{A}$	Value
$Q_{\mathcal{A}}$	{}	diag([0.01, 1])
	{1}	diag([30, 1])
	{2}	diag([30, 1])
	{1, 2}	diag([3, 30])
$R_{\mathcal{A}}$		diag([0.5, 0.02])
N		60
Δt		0.0333 h
Q^e		diag([10^{-3}, 10^{-3}, 10^{-3}, 10^{-3}, 10^{-3}, 10^{-3}, 8, 8, 0.8, 0.8])
R^e		diag([10^{-12}, 10^{-12}, 10^{-12}, 10^{-12}])
N_{sw}		2

For the two MPC controllers, we use a linear approximation of the dynamic model, and estimate disturbances and additional integrating states using a linear Kalman filter that ensures zero offset [20]. Similarly, to find the matrix H_0 needed to design the controlled variables (CVs) for MPC, we must first obtain, by linearization at the nominal design point (*), the matrices J_{uu}, J_{ud}, G^y, G_d^y, and G_u. The choice of nominal design point is important, especially for the CV selection, as it affects the steady-state losses that we may result when operating away from the nominal point. To understand this better we shall consider two different nominal design points (I and II).

The first set of results are based on linearization in design point I, at the vertex between the regions ($d_I^\star$ = [1.543; −0.089]), 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.

From the closed-loop dynamic simulations in Fig. 8, we see that in the unconstrained region (until t = 4 h), the behavior of standard MPC and region-based MPC is identical (as expected, since H_0, and thus the two CVs, are the same). However, note that the presence of nonlinearity leads to non-optimal steady-state optimal operation, even for the region-based MPC. This is most easily seen by comparing the process inputs u_1 and u_2 with the steady-state optimal dashed magenta lines. The operation in the subsequent region with constraint g_1 active (from t = 4 to t = 8 h) highlights the difference between the two MPC approaches. The region-based MPC framework detects quite accurately the region change and switches to two new CVs (one constraint and one reduced gradient), whereas standard MPC, while using one degree of freedom to control the constraint, attempts to track (one average, with the remaining single degree of freedom) both the two original CVs from the unconstrained region. This is not optimal, as seen most easily from the plot of u_2. 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.

In Figs. 9 and 10 we can see in more detail the steady-state loss for the two MPC schemes as a function of the two disturbances. The

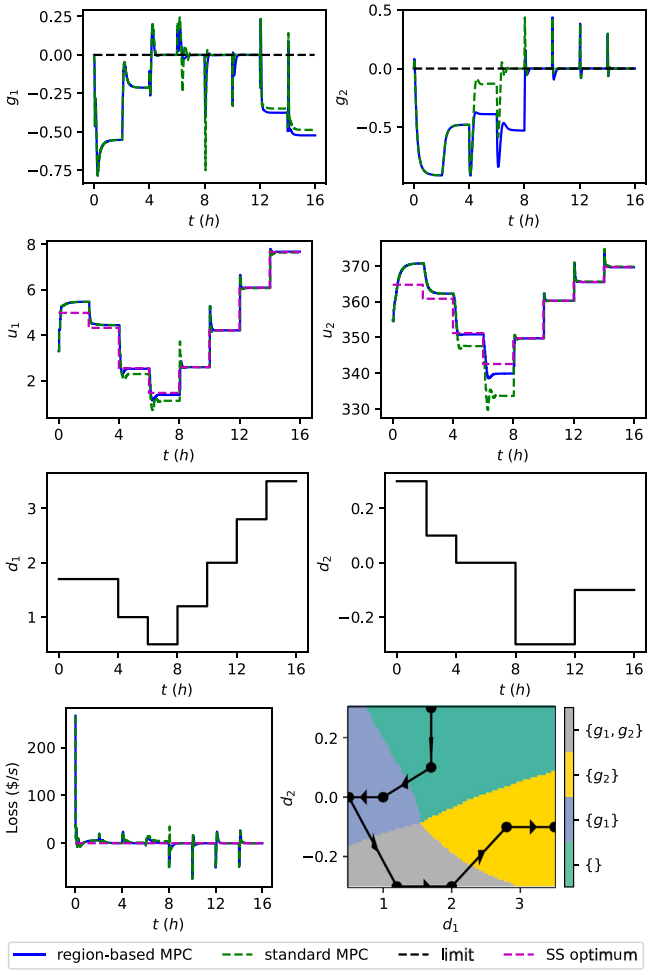

Fig. 8. Dynamic closed-loop simulation results for case study 2 - comparison between standard MPC (green) and the proposed region-based MPC (blue) - linearized at vertex ($d_I^\star = [1.543;\ -0.089]$).

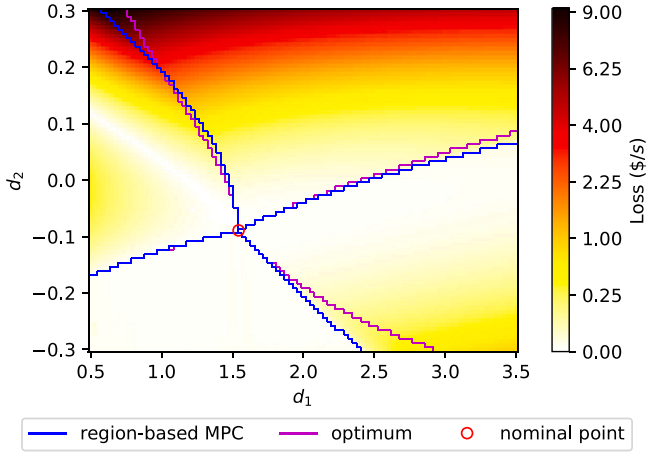

Fig. 9. Steady-state economic loss for region-based MPC for case study 2 - linearized at vertex ($d_I^\star = [1.543;\ -0.089]$).

difference between the two MPC controllers is not so large because there are losses also with region-based MPC due to nonlinearity in the model. However, in standard MPC, the prioritization of CVs in the constrained regions depends on the cost function weights, and therefore the steady state economic loss depends on the MPC tuning parameters.

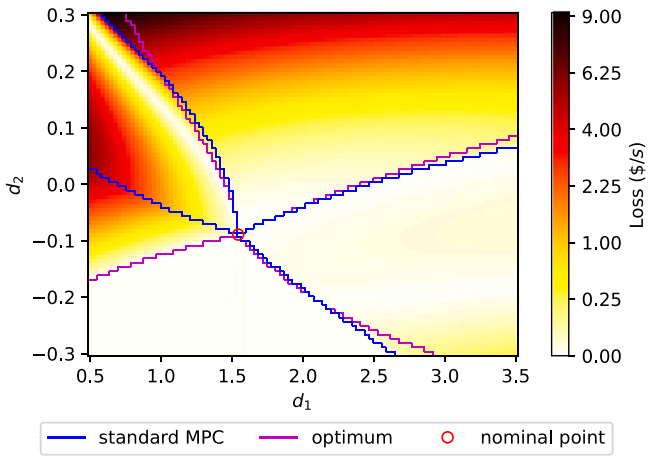

Fig. 10. Steady-state economic loss for standard MPC for case study 2 - linearized at vertex ($d_I^\star = [1.543;\ -0.089]$). $Q = \mathrm{diag}([0.01,\ 1])$.

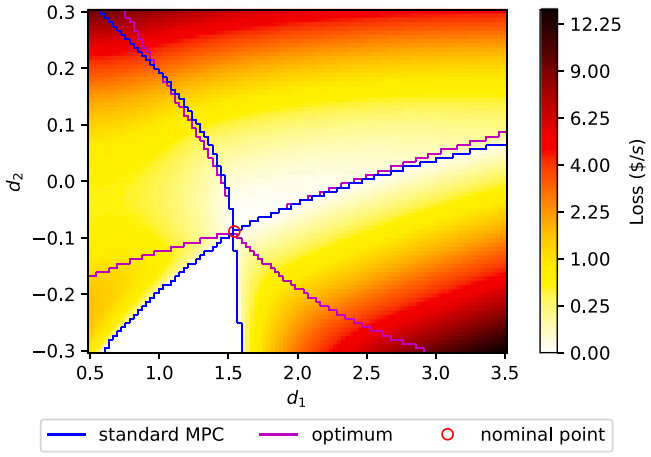

Fig. 11. Steady-state economic loss for standard MPC depends on tuning (choice of weight Q). Results are for case study 2 - linearized at vertex ($d_I^\star = [1.543;\ -0.089]$) with $Q = \mathrm{diag}([5 \times 10^{-4},\ 5])$.

This fact is illustrated in Fig. 11, where a different standard MPC tuning (different choice of the weight Q) led to much larger losses 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, making the steady-state performance independent of the MPC tuning.

We now design the region-based MPC and the standard MPC by linearizing at a different optimal nominal point, namely at $d_{II}^\star = [2.0;\ +0.2]$, 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 tunings are given in Tables 6 and 7, respectively.

For this linearization point, the results are shown in Figs. 12 to 14. In this case, the region-based MPC overall gives much smaller economic losses than standard MPC. Also, the regions obtained in Fig. 13 are shaped similarly to the optimal regions, which does not happen with standard MPC, see Fig. 14. Because the linearization of the system is in the interior of the unconstrained region, the economic loss in that region is smaller when compared to that of Fig. 9, while (somewhat surprisingly) not resulting in a larger loss for the remaining regions for the region-based MPC.

Table 6
Gradient projections for case study 2 with linearization at $d_{II}^{\star} = [2.0;\ +0.2]$.

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

Table 7
Tuning of controllers and estimator for case study 2 - linearization at $d_{II}^{\star} = [2.0;\ +0.2]$ (omitted parameters are the same as in Table 5).

Parameter	$\mathcal{A}$	Value
$Q_{\mathcal{A}}$	{}	diag([5×10^{-4}, 5])
	{1}	diag([30, 1])
	{2}	diag([30, 1])
	{1, 2}	diag([3, 30])

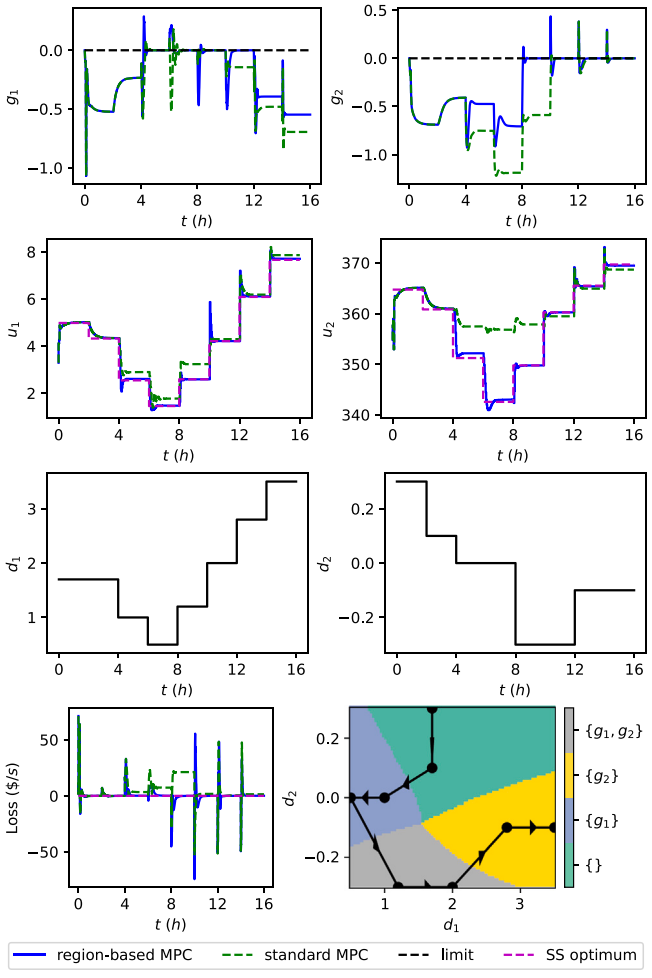

Fig. 12. Dynamic simulation results for case study 2 - comparison between standard MPC (green) and the proposed region-based MPC (blue) - linearized at $d_{II}^{\star} = [2.0;\ +0.2]$.

4.3. Case study 3 - Williams-Otto reactor revisited

We now revisit case study 2, and consider a case with three rather than two constraints. The main reason for adding a constraint is because then the standard decentralized selector-based region-based control does not work because $n_g > n_u$ [21]. The cost function is the same

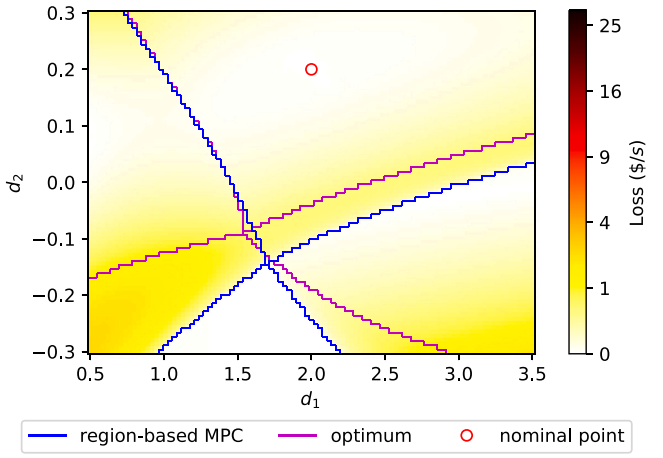

Fig. 13. Steady-state economic loss for region-based MPC for case study 2 - linearized at $d_{II}^{\star} = [2.0;\ +0.2]$.

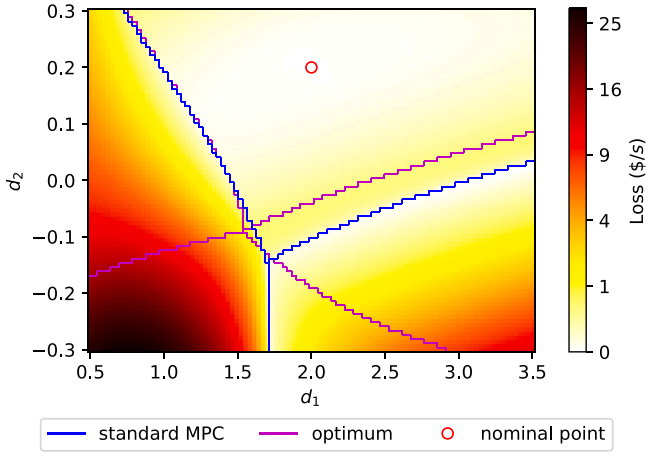

Fig. 14. Steady-state economic loss for standard MPC for case study 2 - linearized at $d_{II}^{\star} = [2.0;\ +0.2]$.

as in case study 2, except that we do not consider price changes. We consider the following constraints (all of which are new):

$$\begin{aligned} F_B &\le 4.0 \\ T_r &\le 355.0 \\ x_G &\le 0.105 \end{aligned} \tag{30}$$

The MVs are the same, $u = \begin{bmatrix} F_B & T_r \end{bmatrix}^T$, but to make it easier to plot the results, we only consider one disturbance, $d = F_A$, which is in the range $0.5 \le d \le 3.5$. The nominal value is $d^{\star} = 1$, which corresponds to an optimal operating point where constraint g_3 is active. We again normalize the constraints, and the normalized steady-state optimization problem is given by:

$$\begin{aligned} \min_{u} \quad & J = p_A F_A + p_B F_B - (F_A + F_B)\left(p_P x_P + p_E x_E\right) \\ \text{s.t.} \quad & g_1 = \frac{F_B - 4.0}{2.68018} \le 0 \\ & g_2 = \frac{T_r - 355.0}{9.55095} \le 0 \\ & g_3 = \frac{x_G - 0.105}{0.00411912} \le 0 \end{aligned} \tag{31}$$

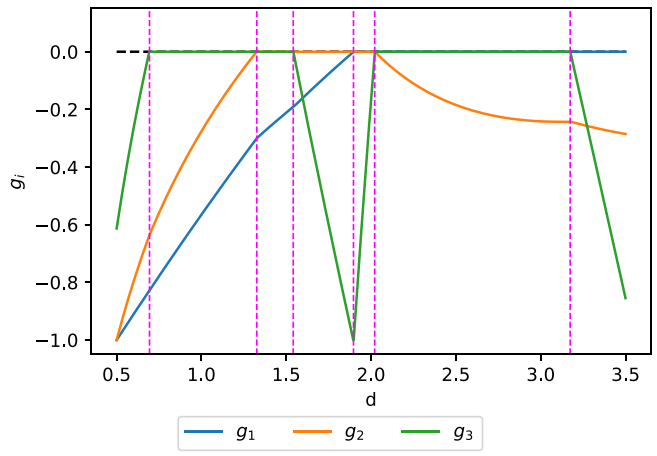

Fig. 15. Optimal values of the three constraints g_i for case study 3 as a function of the disturbance d. The vertical lines represent region switches. Fully unconstrained operation occurs when d is small.

Table 8
Gradient projections for case study 3.

$\mathcal{A}$	$N_{\mathcal{A}}^T$
{}	$\begin{bmatrix}1 & 0\\ 0 & 1\end{bmatrix}$
{1}	$\begin{bmatrix}0 & 1\end{bmatrix}$
{2}	$\begin{bmatrix}1 & 0\end{bmatrix}$
{3}	$\begin{bmatrix}0.07129 & 0.99746\end{bmatrix}$
{1, 2}	–
{1, 3}	–
{2, 3}	–

Fig. 15 shows the active constraint regions as a function of the disturbance F_A. It can be seen that all seven feasible combinations of active constraints appear in the considered disturbance range.[2]

The available measurements are the normalized constraints, $y = g$. We design the unconstrained controlled variables based on the nominal point, for which the resulting H_0 is given by:

$$H_0 = \begin{bmatrix}-36.6848 & 36.887 & -6.57349\\ -2.0782 & 1.18001 & 0.170281\end{bmatrix}$$

and the corresponding gradient projections and controller tunings are given in Tables 8 and 9, respectively.

Dynamic closed-loop simulations with standard and region-based MPC appear in Fig. 16. The tuning used for standard MPC was done such that it operates acceptably when constraints become active, which reduces the performance in some regions. Because the region-based MPC can be tuned independently for every active constraint region, its dynamic performance is better (although it is not so easy to see from the graphs).

Figs. 17 and 18 shows the steady-state values of the constraints as a function of the disturbance for region-based MPC and standard MPC, respectively. The corresponding steady-state losses in Fig. 19 show large benefits of the proposed region-based MPC. The region-based MPC is optimal at and around $d = d^\star = 1.0$. But this not the case for standard MPC. The reason is that the nominal operating point is partly constrained (with $g_3 = 0$ being active). While the region-based MPC is able to use the correction $J_u^\star$ in Eq. (14), 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 large steady-state losses for $d > 2$, approximately.

[2] In general, with n_c constraints, there are $2^{n_c} = 2^3 = 8$ possible constraint combinations. However, since there are only $n_u = 2$ degrees of freedom, it is not possible to control three constraints at the same time, thus only seven of the eight regions are feasible.

Table 9
Tuning parameters for MPC controllers and estimator for case study 3.

Parameter	$\mathcal{A}$	Value
$Q_{\mathcal{A}}$	{}	diag([1, 8])
	{1}	diag([10, 2])
	{2}	diag([50, 1])
	{3}	diag([5, 20])
	{1, 2}	diag([100, 100])
	{1, 3}	diag([50, 10])
	{2, 3}	diag([50, 5])
$R_{\mathcal{A}}$		diag([0.5, 2])
N		40
Δt		0.025 h
Q^e		diag([10^{-3}, 10^{-3}, 10^{-3}, 10^{-3}, 10^{-3}, 10^{-3}, 5000, 500])
R^e		diag([10^{-6}, 10^{-6}, 10^{-12}])
N_{sw}		5

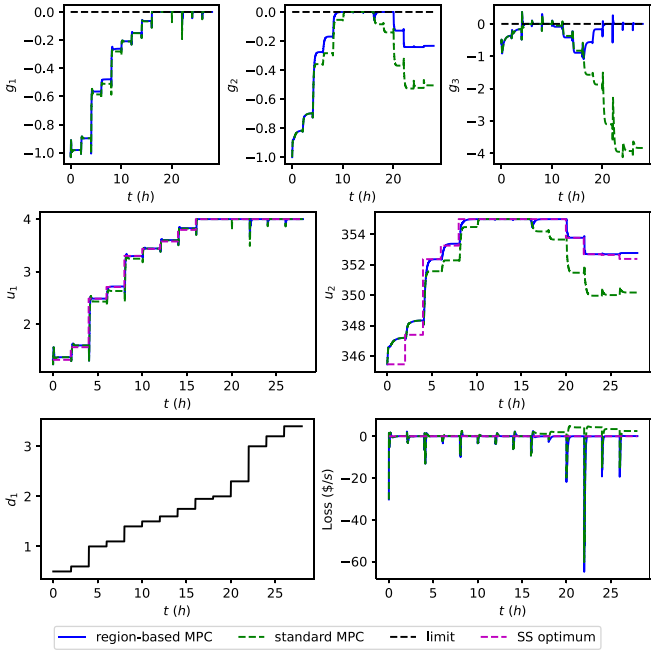

Fig. 16. Dynamic simulation results for case study 3 - comparison between standard MPC (green) and the proposed region-based MPC (blue).

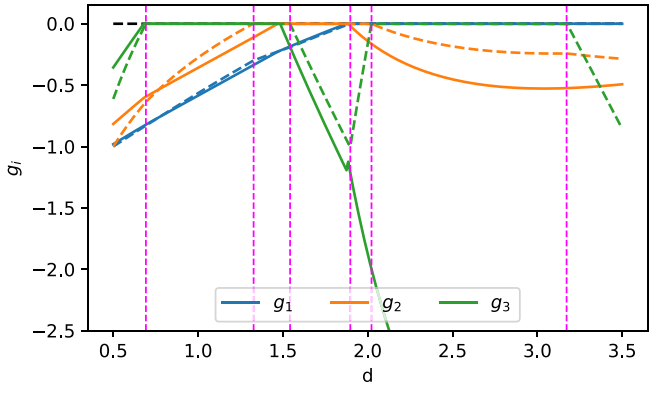

Fig. 17. Steady-state constraint values for standard MPC for case study 3 (optimal values as dashed lines).

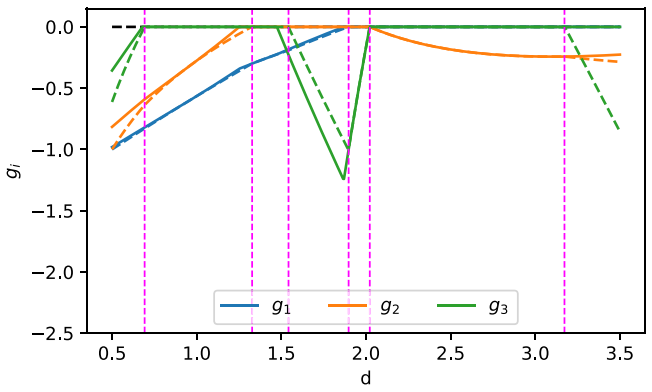

Fig. 18. Steady-state constraint values for region-based MPC for case study 3 (optimal values as dashed lines).

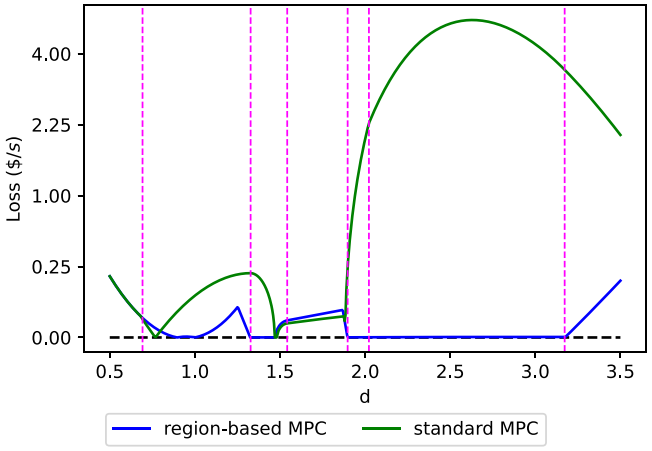

Fig. 19. Steady-state loss as a function of the disturbance for case study 3 (the y-axis is scaled quadratically for better visualization of low loss values).

As mentioned, the standard decentralized selector-based region-based control cannot cover all regions. For example, assume u_1 is paired to g_1, and u_2 is paired to g_2, then we cannot allow g_3 to be active at the same time as any of the other constraints. Thus, we would need to use some sort of adaptive pairing, if decentralized control is to be achieved (see Bernardino et al. [22] for an example).

5. Discussion

5.1. Exact local method for gradient estimation

The unconstrained gradient J_u is a key element in the proposed region-based MPC approach, as it used both to determine the unconstrained CVs and to detect switches in active constraints. In this work, see Eq. (14), the gradient is estimated using the nullspace method of self-optimizing control, as first suggested by Jäschke and Skogestad [15]. The nullspace method is simple, but it assumes that we have a sufficient number of measurements and it does not take into account measurement error. In another paper [13], we propose a more general gradient estimation approach, based on the exact local method of self-optimizing control [14] which accounts for static measurement error and can be applied with any number of measurements. It is not much more difficult to use than the nullspace methods, but it requires the user to specify the expected magnitude of disturbances and measurement errors.

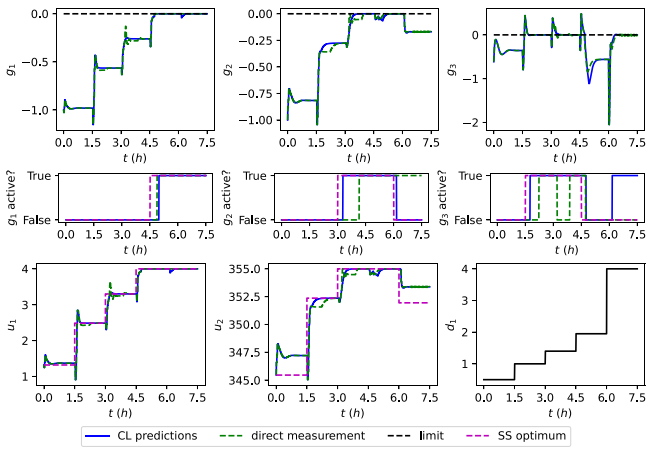

Fig. 20. Comparison of active set detection using the proposed prediction of the new steady state (blue) and using direct measurements (green dashed; not recommended). The simulations are for region-based MPC for case study 3.

5.2. Optimal operation under changing active constraints

The region-based MPC proposed in this work depends on a logic element (the active set detection block) that detects the current active constraint set. From this, we obtain the optimal controlled variables (which are the active constraints plus the corresponding reduced gradients, see Eq. (17)). The use of simple logic elements for switching control structures is very common in industrial practice, but we lack a good theory of such systems, including stability analysis [19]. In general, it is necessary to analyze and possibly simulate the proposed control system for the range of expected disturbances. The proposed system includes a switching strategy, along with a pairing between MVs and CVs. This approach, however, depends on the case study and the engineering insight, and, as mentioned, one may find cases where a decentralized strategy is impossible or at least too complex to be considered for practical use [22].

In addition, even if the disturbance range is such that constraints paired to the same MV are never active at the same time, the control structure should be changed according to which set of constraints is active, because the optimal CVs related to the unconstrained degrees of freedom will change, that is, the reduced gradient will change. In summary, in terms of optimal operation of such systems, there needs to be some centralized logic, as the region-based MPC proposed in this paper.

Alternatively, a general decentralized approach is the primal–dual feedback optimizing control scheme proposed by [23,24]. Here, there is a decentralized master constraint controller which controls the constraints on a slow time scale by manipulating the corresponding Lagrange multipliers. The active constraint set is detected by using max-selectors on the Lagrange multiplier. The approach has been further developed by [25] to include override control on the fast time scale for critical 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 track the values of the CVs in the neighboring regions [6,26]. 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 or lack of convergence where the control structures switch 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 for the constraint switching block

To simplify the active constraint detection block, one may consider using a gradient estimate based directly on the current measurements y, instead of solving for the predicted steady-state y^{ss} as described in Algorithm 1. A simulation with direct measurements for case study 2 is shown in Fig. 20 (dashed green lines), and is compared to the approach proposed in this work (blue lines). However, this is not a good solution, for example, consider the controller activation plots (shown as true/false) for constraints g_2 and g_3. Here, we see that the use of direct measurements in some cases delays the detection of an active set change, and in other cases switches unnecessarily.

5.5. Region-based MPC tuning

The proposed 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 well-known problem 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 [27]. In this work, this is attained by the tuning parameter N_{sw}.

Additionally, the cost functions in Eq. (8) for the region-based MPC must be independently tuned. This is necessary because different CVs usually have different dynamic behaviors. Careful evaluation of the MPC tunings for different regions is therefore advised, so that good dynamic performance is attained in all relevant operating conditions.

5.6. Constraint handling with standard MPC

The simulations verify that standard setpoint-tracking MPC is unable to deal optimally with changing steady-state constraints. To satisfy a steady-state constraint which is not in the nominal region, standard MPC gives up on tight control of its CVs. In our simulations of standard MPC, this is done in an average manner, as determined indirectly by the weight Q for setpoint tracking. A better way of implementing this, used in most industrial implementations before solving the dynamic MPC problem, is to use a steady-state target calculation block [1], which, based on the predicted disturbances, computes the new steady state and from this updates feasible setpoints for the unconstrained CVs. A further improvement, used in many industrial MPC implementations, is to solve a *sequence* of steady-state target calculations, giving up constraints in a predefined order, until one obtains a feasible steady-state solution (satisfying all constraints) [28].

The approach presented in this work could then be used in the target calculation block only, and the dynamic 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.

5.7. Addition of RTO layer

The present work has not focused on integrating the proposed tool with RTO, as other works have covered [29]. 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 Fig. 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.

5.8. EMPC

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 optimal switching of the MPC cost function under changing active constraints was presented, see Fig. 2. The two main elements are (1) the active set detection block (see Algorithm 1), and (2) the use of self-optimizing CVs (reduced gradients) in each active constraint region for the unconstrained degrees of freedom; see $c_{\mathcal{A}} = N_{\mathcal{A}}^T H_0 y$ in Eq. (17) with corresponding setpoint $c_{\mathcal{A}}^{sp}$ in Eq. (18). In this paper, we estimated the measurement combination matrix H_0 using the nullspace method from self-optimizing control, but more generally, with measurement error and any number of measurements y, it is recommended to obtain H_0 using the exact local method [13]. 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 of the setpoints, making it applicable to a wide class of problems.

Funding

This work was funded by the Research Council of Norway through the IKTPLUSS programme (project number 299585).

CRediT authorship contribution statement

Lucas Ferreira Bernardino: Writing – original draft, Visualization, Software, Methodology, Formal analysis, Conceptualization. **Sigurd Skogestad:** Writing – review & editing, Supervision, Project administration, Conceptualization.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

References

[1] J.B. Rawlings, Tutorial overview of model predictive control, IEEE Control Syst. Mag. 20 (3) (2000) 38–52.
[2] M. Ellis, H. Durand, P.D. Christofides, A tutorial review of economic model predictive control methods, J. Process Control 24 (8) (2014) 1156–1178.
[3] S. Skogestad, Plantwide control: The search for the self-optimizing control structure, J. Process Control 10 (5) (2000) 487–507.
[4] J. Jäschke, Y. Cao, V. Kariwala, Self-optimizing control–A survey, Annu. Rev. 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, J. Process Control 34 (2015) 35–48.
[6] J. Jäschke, S. Skogestad, Optimal controlled variables for polynomial systems, J. 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, Comput. Chem. Eng. 34 (11) (2010) 1863–1872.
[8] U. Maeder, F. Borrelli, M. Morari, Linear offset-free model predictive control, Automatica 45 (10) (2009) 2214–2222.
[9] D. Simon, Optimal State Estimation: Kalman, H Infinity, and Nonlinear Approaches, John Wiley & Sons, 2006.
[10] A. Cervantes, L.T. Biegler, Optimization strategies for dynamic systems, Encyclopedia Optim. 4 (2009) 216–227.
[11] D.Q. Mayne, Model predictive control: Recent developments and future promise, Automatica 50 (12) (2014) 2967–2986.
[12] I.J. Halvorsen, S. Skogestad, J.C. Morud, V. Alstad, Optimal selection of controlled variables, Ind. Eng. Chem. Res. 42 (14) (2003) 3273–3284.
[13] L.F. Bernardino, S. Skogestad, Optimal measurement-based cost gradient estimate for feedback real-time optimization, Comput. Chem. Eng. 189 (2024) 108815.
[14] V. Alstad, S. Skogestad, E.S. Hori, Optimal measurement combinations as controlled variables, J. Process Control 19 (1) (2009) 138–148.
[15] J. Jäschke, S. Skogestad, NCO tracking and self-optimizing control in the context of real-time optimization, J. Process Control 21 (10) (2011) 1407–1416.
[16] J. Nocedal, S.J. Wright, Numerical Optimization, second ed., Springer New York, NY, 2006.
[17] J.A.E. Andersson, J. Gillis, G. Horn, J.B. Rawlings, M. Diehl, CasADi – A software framework for nonlinear optimization and optimal control, Math. Program. Comput. 11 (1) (2019) 1–36, http://dx.doi.org/10.1007/s12532-018-0139-4.
[18] T.J. Williams, R.E. Otto, A generalized chemical processing model for the investigation of computer control, Trans. Am. Inst. Electr. Eng. I 79 (5) (1960) 458–473.
[19] S. Skogestad, Advanced control using decomposition and simple elements, Annu. Rev. Control 56 (2023) 100903.
[20] G. Pannocchia, M. Gabiccini, A. Artoni, Offset-free MPC explained: Novelties, subtleties, and applications, IFAC-PapersOnLine 48 (23) (2015) 342–351.
[21] L.F. Bernardino, S. Skogestad, Decentralized control using selectors for optimal steady-state operation with changing active constraints, J. Process Control 137 (2024) 103194.
[22] L.F. Bernardino, D. Krishnamoorthy, S. Skogestad, Optimal operation of heat exchanger networks with changing active constraint regions, in: Computer Aided Chemical Engineering, vol. 49, Elsevier, 2022, pp. 421–426.
[23] D. Krishnamoorthy, A distributed feedback-based online process optimization framework for optimal resource sharing, J. Process Control 97 (2021) 72–83.
[24] R. Dirza, S. Skogestad, D. Krishnamoorthy, Optimal resource allocation using distributed feedback-based real-time optimization, IFAC-PapersOnLine 54 (3) (2021) 706–711.
[25] R. Dirza, S. Skogestad, Primal-dual feedback-optimizing control with override for real-time optimization, J. Process Control 138 (2024) 103208.
[26] H. Manum, S. Skogestad, Self-optimizing control with active set changes, J. Process Control 22 (5) (2012) 873–883.
[27] H. Lin, P.J. Antsaklis, Stability and stabilizability of switched linear systems: A survey of recent results, IEEE Trans. Autom. Control 54 (2) (2009) 308–322.
[28] S. Strand, J.R. Sagli, MPC in Statoil–advantages with in-house technology, IFAC Proc. Vol. 37 (1) (2004) 97–103.
[29] 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, J. Process Control 106 (2021) 1–19.