Journal of Process Control 137 (2024) 103194

Contents lists available at ScienceDirect

Journal of Process Control

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

Decentralized control using selectors for optimal steady-state operation with changing active constraints

Lucas Ferreira Bernardino, Sigurd Skogestad *

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

ARTICLE INFO

Keywords:
Optimal operation
Decentralized control
Selectors

ABSTRACT

We study the optimal steady-state operation of processes where the active constraints change. The aim of this work is to eliminate or reduce the need for a real-time optimization layer, moving the optimization into the control layer by switching between appropriately selected controlled variables (CVs) in a simple way. The challenge is that the best CVs, or more precisely the reduced cost gradients associated with the unconstrained degrees of freedom, change with the active constraints. This work proposes a framework based on decentralized control that operates optimally in all active constraint regions, with region switching mediated by selectors. A key point is that the nullspace associated with the unconstrained cost gradient needs to be selected in accordance with the constraint directions so that selectors can be used. A main benefit is that the number of SISO controllers that need to be designed is only equal to the number of process inputs plus constraints. The main assumptions are that the unconstrained cost gradient is available online and that the number of constraints does not exceed the number of process inputs. The optimality and ease of implementation are illustrated in a simulated toy example with linear constraints and a quadratic cost function. In addition, the proposed framework is successfully applied to the nonlinear Williams–Otto reactor case study.

1. Introduction

The integration of optimization and control is very important when designing the control system for a process. The main objective of the control system is to keep the process stable and operating at the economically optimal operating point. Although these two objectives can be assessed simultaneously, for example, using economic model predictive control (EMPC) [1], a simpler, and in most cases equally optimal,[1] approach is to decompose the system hierarchically into an optimization and a control layer as shown in Fig. 1, where setpoints CV^{sp} are used to connect the two layers. The setpoints may need to be updated due to disturbances that affect the process economics. In the standard implementation in Fig. 1, the real-time optimization (RTO) and setpoints update is performed on a slow time scale based on a detailed nonlinear process model and the estimated states of the process. In most cases, the RTO layer is static.

Based on the concept of Morari et al. [2] of feedback optimizing control, the aim of the current paper is to move the real-time optimization, or at least parts of it, into the control layer. A recent review on this topic is given in Krishnamoorthy and Skogestad [3], where the authors state some of the challenges with RTO implementation, including the cost of developing the model, the uncertainty related to the model and its parameters (or disturbances), and human aspects related to the maintenance of an optimization layer in addition to the already existing digital control system (DCS). The importance of feedback optimizing control lies in being able to reject disturbances that affect economic performance in a simple manner, without relying on an upper optimization layer that may sometimes not even exist. To that end, an appropriate selection of the controlled variables (CVs) for the control layer is important. This is the main idea of self-optimizing control [4]. It is particularly important to include the active constraints as CVs, that is, the constraints that are optimally at their limiting value [2,5]. If information about the cost gradient is available, the optimal CVs are the active constraints plus the reduced cost gradients, and by controlling these at a constant setpoint of zero we may eliminate the optimization layer [6]. This choice of CVs is valid if the set of active constraints does not change in the considered operating region.

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

[1] In fact, in some cases a decomposed approach with separate optimization and control layers may be better performing economically than EMPC, because the control layer may be tuned to be fast, whereas this is likely difficult to achieve with a centralized solution like EMPC. This may give economic benefits, especially for fast-changing disturbances.

https://doi.org/10.1016/j.jprocont.2024.103194
Received 5 October 2023; Received in revised form 24 February 2024; Accepted 3 March 2024

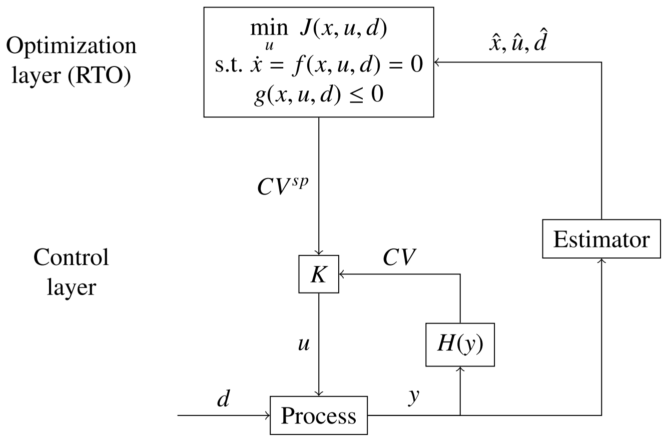

Fig. 1. Standard optimizing control implementation with separate layers for real-time optimization (RTO) and control (K, which can be *e.g.* MPC or PID). J denotes the (economic) cost function to be minimized, f the process model, g the process constraints, x the model states, d the disturbances, and u the process inputs (MVs).

Dealing with changes in active constraints has been a concern in previous works. For example, Cao [7] implemented a cascade control structure with selectors to avoid constraint violation by the lower self-optimizing layer, and Graciano et al. [8] applied MPC with zone control to the same end. A global self-optimizing control method for changing active constraints has been proposed by Ye et al. [9], where the goal is to minimize the average loss obtained with a single set of CVs. However, in a new active constraint region, not only do the active constraints change, but the directions related to the reduced cost gradient change accordingly. This means that to eliminate the RTO layer one needs to change the control layer in Fig. 1 during operation, both in terms of the selected CVs and the corresponding feedback controller K. With this perspective, Manum and Skogestad [10] has considered a centralized, steady-state analysis on switching control structures, with different CVs for each region.

However, the implementation of such a region-based control strategy quickly becomes impractical. This is because the number of active constraint regions grows exponentially with the number of constraints. Let n_u denote the number of process inputs or manipulated variables (MVs) and n_g the number of independent constraints. The upper bound on the number of active constraint regions is 2^{n_g}, which is reached when all constraint combinations are feasible [11]. In each region, we ideally need a new controller K, and if we want to use decentralized control then we need to design n_u single-input single-output (SISO) controllers in each region. For example, with $n_g = 4$ and $n_u = 5$, there could be up to $2^4 = 16$ constraint regions, which may require the tuning of $2^{n_g} \cdot n_u = 16 \cdot 5 = 80$ SISO controllers. Even though some CVs are reused between regions, the number of necessary SISO loops will be high.

The key contribution of this paper is to propose a simple and generic region-based control structure with only $n_g + n_u$ SISO controllers, as represented in Fig. 2, with the same set of unconstrained variables (CV^0 and CV^{0g}) in *all* operating regions.[2] Considering the previous example, this structure would have only $4 + 5 = 9$ SISO controllers. In the paper, we show that the unconstrained variables are obtained from n_u projections of the full cost gradient with respect to the inputs, $\nabla_u J$. This leads to n_u gradient controllers (K^0 and K^{0g}) and n_g constraint controllers (K^g). However, at any given time, only a subset with n_u of the $n_u + n_g$ controller outputs is implemented as process inputs, with the switching logic choosing between the controller outputs u^{0g} and u^g.

[2] The superscript 0 is used to indicate the unconstrained case and g the constrained case. The superscript $0g$ indicates unconstrained cases that are associated with switching constraints.

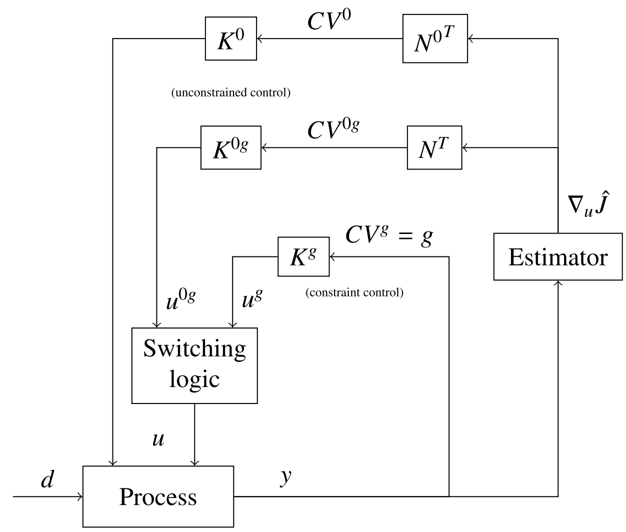

Fig. 2. Proposed optimizing control implementation, assuming $n_u \geq n_g$. The controllers K^0, K^{0g} and K^g are usually single-variable PID controllers. The projection (nullspace) matrices N^0 and N are defined in Eq. (7) and Eq. (8), respectively. There is no CV^0, N^0, and K^0 if $n_u = n_g$. Note that the optimization layer in Fig. 1 is eliminated, and an estimate $\nabla_u \hat{J}$ of the cost gradient is needed. The switching logic takes care of the change between active constraint regions. In this paper, this logic is decentralized to n_g individual blocks, see Fig. 3, which can be implemented as min or max selectors according to Theorem 3.

The second key contribution of this paper is to show that the switching logic in Fig. 2 can be effectively implemented using n_g min or max selectors, which are well-known advanced control elements and commonly used in practical control applications. An important decision is to pair each constraint to an MV, but this pairing problem is not addressed in this paper (the interested reader is referred to Skogestad and Postlethwaite [12]). The main assumptions in this work are that we have at least as many MVs as constraints ($n_u \geq n_g$), and that an estimator for the unconstrained cost gradient $\nabla_u J$ is available. In terms of cost gradient estimation, there are several methods available (see Krishnamoorthy and Skogestad [3]), and in this work, we use the simple model-based approach of dynamic state estimation and model linearization proposed by Krishnamoorthy et al. [13].

Selectors have been used in industry to switch between CVs since the 1940s [14]. Selectors are also used in academic case studies on optimal operation [11,15]. In these case studies, a control structure is proposed for the nominal operating region, with added logic elements and control loops to deal with the neighboring regions. However, the treatment of the unconstrained degrees of freedom is not clear. Krishnamoorthy and Skogestad [16] proposes a framework for constraint handling using min and max selectors, focusing on systems with a single MV, and therefore not considering the changes of reduced gradients for the unconstrained variables. To the best of the authors' knowledge, even though a general scheme for the paradigm of region-based control is proposed in the review paper by Krishnamoorthy and Skogestad [3], a systematic procedure for designing a decentralized control structure for optimal operation of generic multivariable systems has not yet been explored, as well as whether there are any fundamental limitations for the design of such systems. In this work, we explore these topics, and we describe a class of multivariable systems for which a decentralized control structure is always possible.

2. Decentralized control framework for optimal operation

We consider a generic, steady-state optimization problem given by:

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

Here, $J : \mathbb{R}^{n_u} \times \mathbb{R}^{n_d} \to \mathbb{R}$ is a scalar cost function to be minimized, $g : \mathbb{R}^{n_u} \times \mathbb{R}^{n_d} \to \mathbb{R}^{n_g}$ is the function that returns the vector of inequality constraints, $u \in \mathbb{R}^{n_u}$ is the vector of decision variables (MVs), and $d \in \mathbb{R}^{n_d}$ is the vector of disturbances. Note that the states x (see Fig. 1) have been formally eliminated from the equations, such that J and g are functions only of the independent variables u and d. Introduce the Lagrange function $\mathcal{L}(u, \lambda, d) = J(u, d) + \lambda^T g(u, d)$. Then, for a given value of d, define u^* as the solution of Eq. (1), which satisfies the Karush–Kuhn–Tucker (KKT) conditions [17]:

$$\nabla_u \mathcal{L}(u^*, \lambda^*, d) = \nabla_u J(u^*, d) + \left(\nabla_u g(u^*, d)\right)^T \lambda^* = 0 \tag{2a}$$

$$g(u^*, d) \le 0 \tag{2b}$$

$$\lambda^* \ge 0 \tag{2c}$$

$$\lambda_i^* g_i(u^*, d) = 0, \qquad i = 1, \dots, n_g \tag{2d}$$

Here, λ is the vector of Lagrange multipliers associated with the inequality constraints, and λ^* is its optimal value. We remark that the KKT conditions only imply that the solution is a stationary point, and they are also satisfied by local minima or maximum and saddle points. We do not address these issues in this work, and we consider that the optimization problem in Eq. (1) is convex. While these optimization problems can be efficiently solved using numerical methods, we here focus on how to solve these problems with feedback control. For this, we rewrite the KKT conditions as control objectives, which allows us to embed the optimization into the control layer design.

The set of active constraints $\mathcal{A}$ is defined as the set that satisfies $g_i(u^*, d) = 0$ for $i \in \mathcal{A}$. For convenience, define $g_{\mathcal{A}} : \mathbb{R}^{n_u} \times \mathbb{R}^{n_d} \to \mathbb{R}^{n_a}$ as the function that returns the active constraints. Define the matrix:

$$G^g = \nabla_u g(u, d) \tag{3}$$

as the gradient of the constraints with respect to the MVs, and the matrix $G^g_{\mathcal{A}} = \nabla_u g_{\mathcal{A}}(u, d)$ as the gradient of the active constraints with respect to the MVs. If the set of active constraints $\mathcal{A}$ is known, Jäschke and Skogestad [6] prove that optimality can be attained by controlling to zero the active constraints and the associated reduced cost gradient. Their result is given by the following theorem:

Theorem 1 (***Optimal Controlled Variables***). *Consider the optimization problem in Eq. (1), where we assume that linear independence constraint qualification (LICQ) holds. We assume that the set of optimally active constraints $\mathcal{A}$ is known. Let $N_{\mathcal{A}} \in \mathbb{R}^{n_u \times (n_u - n_a)}$ be a basis for the nullspace of $G^g_{\mathcal{A}}$ such that:*

$$G^g_{\mathcal{A}} N_{\mathcal{A}} = 0 \tag{4}$$

Further, define the reduced cost gradient as:

$$\nabla_{u,\mathcal{A}} J(u, d) = N_{\mathcal{A}}^T \nabla_u J(u, d) \tag{5}$$

Then controlling $g_{\mathcal{A}}(u, d) = 0$ and $\nabla_{u,\mathcal{A}} J(u, d) = 0$ results in optimal steady-state operation.

Proof (*[6]*). If the active constraints $\mathcal{A}$ are known, the necessary optimality conditions (2) are equivalent to:

$$\begin{cases} \nabla_u \mathcal{L}(u^*, d) = \nabla_u J(u^*, d) + \left(G^g_{\mathcal{A}}\right)^T \lambda^*_{\mathcal{A}} = 0 \\ g_{\mathcal{A}}(u^*, d) = 0 \end{cases} \tag{6}$$

where $\lambda^*_{\mathcal{A}} > 0$ is the optimal vector of Lagrange multipliers for the active constraints. Premultiplying $\nabla_u \mathcal{L}(u^*, d)$ by $N_{\mathcal{A}}^T$ leads to:

$$N_{\mathcal{A}}^T \nabla_u \mathcal{L}(u^*, d) = N_{\mathcal{A}}^T \nabla_u J(u^*, d) + \left(G^g_{\mathcal{A}} N_{\mathcal{A}}\right)^T \lambda^*_{\mathcal{A}} = 0$$

Since by definition $G^g_{\mathcal{A}} N_{\mathcal{A}} = 0$, the optimality conditions are equivalent to $g_{\mathcal{A}}(u^*, d) = 0$ and $N_{\mathcal{A}}^T \nabla_u J(u^*, d) = 0$, which are n_u equations that fully determine u^* because $N_{\mathcal{A}}$ is full rank, and the associated optimal Lagrange multiplier can always be found as $\lambda^*_{\mathcal{A}} = -\left(G^g_{\mathcal{A}} (G^g_{\mathcal{A}})^T\right)^{-1} G^g_{\mathcal{A}} \nabla_u J(u^*, d)$. Therefore, enforcing $g_{\mathcal{A}}(u^*, d) = 0$ and $N_{\mathcal{A}}^T \nabla_u J(u^*, d) = 0$ leads to satisfying (6), which is equivalent to satisfying (2). □

In terms of feedback control, Theorem 1 says that $g_{\mathcal{A}}$ and $\nabla_{u,\mathcal{A}} J$ (both with setpoints 0) are the steady-state optimal CVs for a given operating region where the active constraints do not change. Here, the reduced cost gradient $\nabla_{u,\mathcal{A}} J = N_{\mathcal{A}}^T \nabla_u J$ is defined as the gradient in the unconstrained directions as given by the nullspace $N_{\mathcal{A}}$ of the active constraints [6]. If the system is to operate at another active constraint region, however, the CVs need to change, and if shifts in operating regions happen in real-time, the control system needs to automatically detect these region switches. The main idea of this work is to design a decentralized control structure, see Fig. 2, for all possible active constraint regions of the optimization problem in Eq. (1). The main assumption for guaranteeing the existence of this decentralized control structure is as follows:

Assumption 1. The matrix G^g is always full row rank, and the number of constraints is not greater than the number of MVs, that is, $\text{rank}(G^g) = n_g$, and $n_u \ge n_g$.

This not only guarantees LICQ for any set of constraints that may be optimally active, but it also guarantees the existence of decoupled CVs for optimal operation, as shown in the next theorem. For use in the next theorem, define N^0 as an orthonormal basis of the nullspace of G^g, that is:

$$G^g N^0 = 0 \tag{7}$$

The matrix N^0 represents the unconstrained directions that are never in conflict with constraint control. Note here that N^0 is an empty matrix (nonexistent) if we have as many constraints as inputs ($n_u = n_g$). Further define G^g_{-i} as the matrix containing all but the ith row of G^g, and define:

$$N = \begin{bmatrix} N_1 & \cdots & N_{n_g} \end{bmatrix} \tag{8}$$

as a matrix of n_g columns, where each column N_i is a unitary vector such that:

$$\begin{bmatrix} G^g_{-i} \\ N^{0T} \end{bmatrix} N_i = 0 \tag{9}$$

Each vector N_i represents the direction that may conflict with the corresponding constraint g_i, as shown next.

Theorem 2 (***Optimal Switching Between CVs***). *Given that Assumption 1 holds and that the active constraint index set is $\mathcal{A}$, the following control strategy allows for optimal operation:*

- *If $n_u > n_g$, which means N^0 is non-empty, control $CV^0 = N^{0T} \nabla_u J(u, d) = 0$;*
- *For $i = 1, 2, \dots, n_g$, if $i \in \mathcal{A}$, control $CV_i^g = g_i(u, d) = 0$; otherwise, control $CV_i^{0g} = N_i^T \nabla_u J(u, d) = 0$.*

Proof. To prove Theorem 2, it is sufficient to prove that the controlled variables are equivalent to the necessary first-order optimality conditions. Firstly, it is useful to note that, due to its construction, $G^g N_i = (G^g_i N_i)\hat{e}_i$, with G^g_i being the ith row of G^g, and $\hat{e}_i$ being the ith unit vector from the standard basis. Additionally, if the active constraint set is $\mathcal{A}$, and the inactive constraint set is $\mathcal{I} = \{1, \dots, n_g\} - \mathcal{A}$, the optimality conditions can be written as:

$$\begin{cases} \nabla_u \mathcal{L}(u^*, d) = \nabla_u J(u^*, d) + G^{gT} \lambda^* = 0 \\ g_i(u^*, d) = 0, \quad i \in \mathcal{A} \\ \lambda_i^* = 0, \quad i \in \mathcal{I} \end{cases}$$

Let $N_{\mathcal{I}}$ be the matrix with columns equal to N_i for $i \in \mathcal{I}$. Then, premultiplying $\nabla_u \mathcal{L}$ by $\begin{bmatrix} N_{\mathcal{I}} & N^0 \end{bmatrix}^T$ leads to:

$$\begin{bmatrix} N_{\mathcal{I}} & N^0 \end{bmatrix}^T \nabla_u \mathcal{L} = \begin{bmatrix} N_{\mathcal{I}} & N^0 \end{bmatrix}^T \nabla_u J + \left(G^g \begin{bmatrix} N_{\mathcal{I}} & N^0 \end{bmatrix}\right)^T \lambda^*$$
$$= \begin{bmatrix} N_{\mathcal{I}} & N^0 \end{bmatrix}^T \nabla_u J + \left(\begin{bmatrix} G^g N_{\mathcal{I}} & 0 \end{bmatrix}\right)^T \lambda^*$$

Here, $\left(G^g N_{\mathcal{I}}\right)^T \lambda^* = 0$, because $G^g N_i = (G_i^g N_i)\hat{e}_i$, and, from the optimality conditions, $\hat{e}_i^T \lambda^* = \lambda_i^* = 0$ for $i \in \mathcal{I}$. Therefore, the optimality conditions become

$$\begin{bmatrix} N_{\mathcal{I}} & N^0 \end{bmatrix}^T \nabla_u \mathcal{L} = \begin{bmatrix} N_{\mathcal{I}} & N^0 \end{bmatrix}^T \nabla_u J = 0,$$

which are the CVs proposed in addition to $g_i(u,d) = 0$ for $i \in \mathcal{A}$. Similarly to Theorem 1, this fully defines the operational degrees of freedom, and a suitable vector of Lagrange multipliers can be found. □

Note that the matrix

$$N_{(\mathcal{A})} = \begin{bmatrix} N_{\mathcal{I}} & N^0 \end{bmatrix} \tag{10}$$

used in the proof of Theorem 2 is a particular parametrization of the nullspace matrix $N_{\mathcal{A}}$ from Theorem 1, and therefore both results are equivalent for a given active constraint region. In Theorem 2 however, we specify an ideal association between CVs such that the handling of region switching may be done in a decentralized fashion, avoiding changes in the rest of the control structure. For instance, if the ith constraint changes from inactive to active, only the corresponding unconstrained degree of freedom $CV_i^{0g} = N_i^T \nabla_u J$ will become uncontrolled, and the remaining CVs are kept unaltered. In addition, the matrix $N_{(\mathcal{A})} = \begin{bmatrix} N_{\mathcal{I}} & N^0 \end{bmatrix}^T$ is designed to be full row rank, and therefore all operational degrees of freedom are filled for any active set $\mathcal{A}$. The choice of building the vectors N_i unitary and orthogonal to N^0 is purely for the uniqueness of the solution, as one could propose another projection $N_i' = \alpha N_i + N^0 w$ for any nonzero scaling factor α and any vector w, and optimal operation would still be attained, as ${N^0}^T \nabla_u J$ is always optimally zero.

Theorem 2 states a general set of feedback control objectives to attain optimal operation. It does not specify the type of controller to be used, and one may apply these results to obtain optimal operation with conventional tracking MPC with switching objectives to eliminate the RTO layer. This would be useful for cases where decentralized control performs poorly, but one still wishes to propose a simple control layer. In this work, however, we choose to explore the implications of this result for decentralized control, which is often more easily implemented in practice.

Pairing of MVs and CVs. It should be noted that Theorem 2 makes no distinction about the pairing between MVs and CVs, and it is left for the practitioner to make this pairing taking into account controllability and performance aspects. However, the theorem states the optimal association between CVs for region switching, which means that the control of $CV_i^g = g_i$ and $CV_i^{0g} = N_i^T \nabla_u J$ must be performed by the same MV in the case of a decentralized framework. From now on, it is considered that the MVs are ordered such that u_i is used to control the pair $CV_i^g = g_i$ and CV_i^{0g} for $i \leq n_g$. These considerations lead to the control structure presented in Fig. 3. Here, $\nabla_u \hat{J}$ represents the estimate of the cost gradient ($\nabla_u J$), K_i^g represent the individual constraint controllers, K_i^{0g} represent the individual gradient controllers that are conditionally active (*i.e.* only one of K_i^g and K_i^{0g} is active at any given time), and K_i^0 represent the individual gradient controllers that are always active. It is important that the controllers K_i^g and K_i^{0g} include anti-windup action so that the integral modes in the inactive controllers do not grow indefinitely.

We finally focus on the applicability of min/max selectors as the logic elements to switch between active constraint regions, which were left undetermined in Fig. 3 as "select" blocks. These selectors are applied on the controller outputs u_i^g and u_i^{0g} associated with the controlled variables g_i and CV_i^{0g}, respectively, resulting in the process input (MV) u_i to be applied to the system. This methodology was

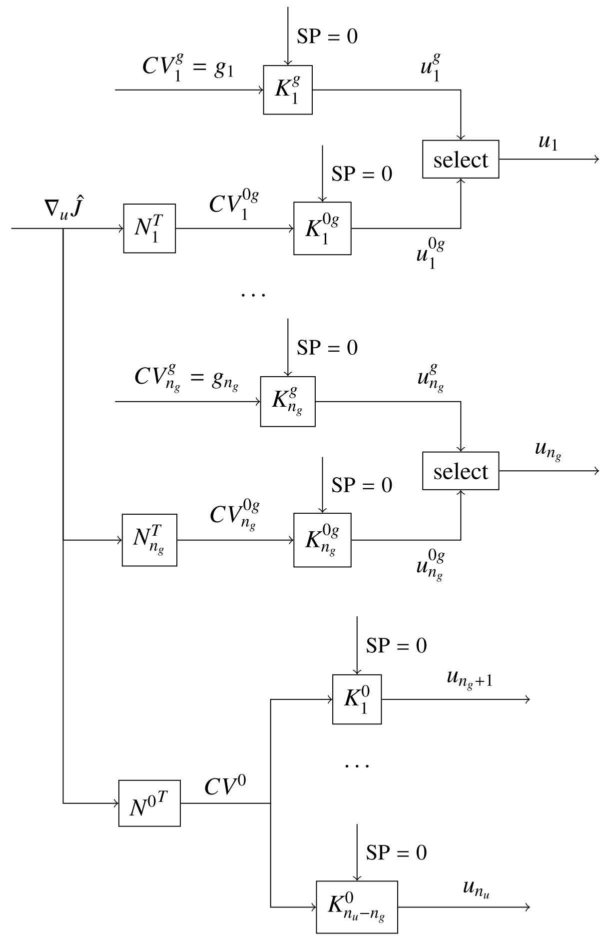

Fig. 3. Decentralized control structure for optimal operation according to Theorem 2. The "select" blocks are usually max or min selectors (see Theorem 3).

adopted in Krishnamoorthy and Skogestad [16] for optimal operation in the scalar case, *i.e.* with a single MV, where it was concluded that a constraint with a positive gain ($G^g > 0$) requires a min selector, whereas a negative gain ($G^g < 0$) requires a max selector. In the next theorem, we present similar results for the multivariable case.

Theorem 3 (***Decentralized Control. Applicability of Min/Max Selectors***). *In addition to Assumption 1, assume that the Hessian of the cost function with respect to the inputs is constant and positive definite, that is, $\nabla_u J(u,d) = J_{uu} u + J_{u,m}(d)$, with $J_{uu} > 0$ and arbitrary $J_{u,m}(d)$, and that G^g is constant. Consider the control structure in Fig. 3 and Theorem 2, and assume that every possible subsystem is stable.*

Let u_i^{0g} denote the value of u_i that controls $CV_i^{0g} = N_i^T \nabla_u J(u,d) = 0$, and let u_i^g denote the value of u_i that controls $g_i(u,d) = 0$. For a given active set $\mathcal{A}$, the associated nullspace of the active constraint gain matrix $G_{\mathcal{A}}^g$ is $N_{(\mathcal{A})} = \begin{bmatrix} N_{\mathcal{I}} & N^0 \end{bmatrix}$. Define the scaled projection matrix $P_{\mathcal{A}}$ and the transformed constraint gain matrix $G_{P_{\mathcal{A}}}^g$ as:

$$P_{\mathcal{A}} = N_{(\mathcal{A})} \left(N_{(\mathcal{A})}^T J_{uu} N_{(\mathcal{A})}\right)^{-1} N_{(\mathcal{A})}^T \tag{11}$$

$$G_{P_{\mathcal{A}}}^g = G^g P_{\mathcal{A}} \tag{12}$$

The optimal input is given by $u_i^* = \min(u_i^{0g}, u_i^g)$ if the ith diagonal element of the transformed gain matrix is positive ($(G_{P_{\mathcal{A}}}^g)_{ii} > 0$) for any active set $\mathcal{A}$ that does not include i. Conversely, the optimal input is given by $u_i^* = \max(u_i^{0g}, u_i^g)$ if $(G_{P_{\mathcal{A}}}^g)_{ii} < 0 \; \forall \mathcal{A} \not\ni i$.

Proof. See Appendix A □

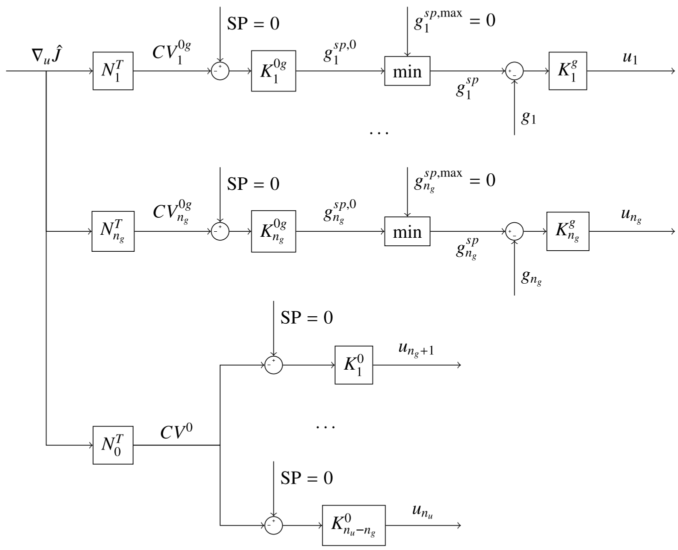

Fig. 4. Decentralized control structure for optimal operation, using an alternative cascade implementation.

It is worth noting that the ith row of $G^g_{P_{\mathcal{A}}}$ is identically zero for $i \in \mathcal{A}$, since $P_{\mathcal{A}}$ involves a projection to the nullspace of the active constraints. If $(G^g_{P_{\mathcal{A}}})_{ii}$ changes sign for different active sets, a single type of selector would not account for all theoretical regions. The single-input case of Theorem 3 can be easily verified by writing $u^g - u^{0g} = -\frac{1}{J_{uu}} G^g \lambda^i$. As $J_{uu} > 0$ for a convex optimization problem, $G^g > 0$ leads to $u^* = \min(u^{0g}, u^g)$, and $G^g < 0$ leads to $u^* = \max(u^{0g}, u^g)$, which is equivalent to the result in Krishnamoorthy and Skogestad [16].

Can some of the assumptions in Theorem 3 be removed? According to Theorem 3, the use of max- (or min-) selectors in Fig. 3 assumes that $(G^g_{P_{\mathcal{A}}})_{ii}$ remains positive (or negative) for any active set $\mathcal{A}$ that does not include i. This is to rule out cases where the steady-state gain for control of the constraint g_i changes sign, as this would lead to instability with integral action in the controller. In other words, this is to rule out interacting processes where $u_i^* = \min(u_i^{0g}, u_i^g)$ for a given active set $\mathcal{A}$, and $u_i^* = \max(u_i^{0g}, u_i^g)$ for another. However, it is not clear whether this is a restriction in practice. Thus, it is possible that the assumption about no sign change for the diagonal elements $(G^g_{P_{\mathcal{A}}})_{ii}$ is not needed. This is left as an open research issue.

Cascade implementation. It is anyway possible to avoid this restriction by using the cascade switching implementation in Fig. 4. That is, for this implementation the simple selector logic is always optimal without the assumption about the sign of $(G^g_{P_{\mathcal{A}}})_{ii}$ in Theorem 3. In the cascade implementation in Fig. 4, the constraints are always controlled in the lower layer, and the optimal constraint setpoint g_i^{sp} will either be the value $g_i^{sp,0}$ that controls $CV_i^{0g} = 0$ or the constraint's limit value itself, such that $g_i^{sp} = \min(g_i^{sp,0}, 0)$ leads to optimal operation. This result can also be obtained by rewriting Theorem 3 in terms of the transformed inputs $v = \begin{bmatrix} g_1 & \cdots & g_{n_g} & u_{n_g+1} & \cdots & u_{n_u} \end{bmatrix}^T$, where it can be verified that the condition $(G^{g,v}_{P_{\mathcal{A}}})_{ii} > 0$ is always satisfied. This result is presented in Appendix B.

The idea of using cascade control for self-optimizing control and constraint satisfaction has been previously proposed in Cao [7]. There, the cost gradient is controlled in the unconstrained case, while the lower layer keeps the system feasible by saturating the setpoint from the upper layer. This approach ensures feasibility and self-optimizing behavior at the unconstrained region, but optimality at all active constraint regions is only ensured by carefully selecting the CVs at the upper layer, which is the main idea of the present work. In addition, even though the cascade structure will always operate optimally at steady state, it requires that the outer controllers K_i^{0g} are sufficiently slower than the inner controllers K_i^g, and therefore the generic structure in Fig. 3 offers more flexibility in terms of loop tuning and implementation of further control overrides. The simulations presented in this paper are for the implementation in Fig. 3.

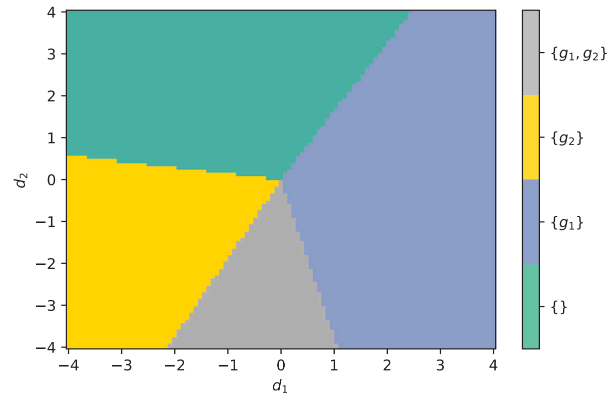

Fig. 5. Active constraint regions for case study 1 as a function of disturbances.

3. Case study 1 - Toy example

In this section, to illustrate the implementation of the proposed control structure, we consider a linear process with a quadratic cost function and 2 linear constraints. The process has 2 dynamic states x, 3 inputs (MVs) u, and 2 disturbances d. The linear state-space model is:

$$\dot{x} = \underbrace{\begin{bmatrix} -\frac{1}{\tau_1} & 0 \\ 0 & -\frac{1}{\tau_2} \end{bmatrix}}_{A} x + \underbrace{\begin{bmatrix} \frac{0.2}{\tau_1} & 0 & 0 \\ 0 & \frac{0.2}{\tau_2} & 0 \end{bmatrix}}_{B} u + \underbrace{\begin{bmatrix} \frac{1}{\tau_1} & 0 \\ 0 & \frac{1}{\tau_2} \end{bmatrix}}_{B_d} d \tag{13}$$

with $\tau_1 = 1$ and $\tau_2 = 2$. It is assumed that both states are measured, that is, $y = Cx + Du$ with $C = I$ and $D = 0$.

The steady-state optimization problem in terms of the states is:

$$\begin{aligned} \min_u \quad & \frac{1}{2} x^T \begin{bmatrix} 1 & 0 \\ 0 & 10 \end{bmatrix} x + \frac{1}{2} u^T \begin{bmatrix} 1 & -0.1 & -0.2 \\ -0.1 & 0.8 & -0.1 \\ -0.2 & -0.1 & 0.3 \end{bmatrix} u \\ \text{s.t.} \quad & \begin{cases} g_1 = x_1 - 0.8 x_2 \leq 0 \\ g_2 = u_1 + u_2 + u_3 \leq 0 \end{cases} \end{aligned} \tag{14}$$

At steady state, the states can be eliminated to give the following static optimization problem:

$$\begin{aligned} \min_u \quad & J(u,d) = \frac{1}{2} u^T \begin{bmatrix} 1.04 & -0.1 & -0.2 \\ -0.1 & 1.2 & -0.1 \\ -0.2 & -0.1 & 0.3 \end{bmatrix} u + u^T \begin{bmatrix} 0.2 & 0 \\ 0 & 2 \\ 0 & 0 \end{bmatrix} d \\ \text{s.t.} \quad & g(u,d) = \underbrace{\begin{bmatrix} 0.2 & -0.16 & 0 \\ 1 & 1 & 1 \end{bmatrix}}_{G^g} u + \begin{bmatrix} 1 & -0.8 \\ 0 & 0 \end{bmatrix} d \leq 0 \end{aligned} \tag{15}$$

For given disturbances d, we can solve the problem in Eq. (15) to find the optimal steady-state inputs u^* and the active set $\mathcal{A}$. From this, we can graphically represent the active constraint regions as a function of the two disturbances as shown in Fig. 5. Note that this is done for visualization purposes only and is not a part of the proposed method. In fact, for the proposed method we do not need to know what the disturbances are; what is needed is measured or estimated values for the constraints g and the unconstrained cost gradient $\nabla_u \hat{J}$. We see in Fig. 5 that all $2^{n_g} = 2^2 = 4$ combinations of constraints are possible. Each region has a specific set of CVs for optimal operation, namely the active constraints and the corresponding reduced gradients, as given in Theorem 2.

Table 1
Diagonal elements of $G^g_{P_{\mathcal{A}}}$ for all relevant sets $\mathcal{A}$ for case study 1.

$\mathcal{A}$	$(G^g_{P_{\mathcal{A}}})_{11}$	$(G^g_{P_{\mathcal{A}}})_{22}$
{ }	0.201	1.443
{1}	–	1.801
{2}	0.155	–

Table 2
PI controller tunings for case study 1. Note that $K_I = K_c/\tau_I$ is the integral gain. The first four controllers have anti-windup with tracking time $\tau_T = 0.01$ s.

Controller	Parameter	Value
K^g_1	K_c	50
	τ_I	1.0
K^g_2	K_I	100
K^{0g}_1	K_I	−2.382
K^{0g}_2	K_I	3.055
K^0	K_I	5.523

We have $n_u = 3$ and $n_g = 2$, so with the proposed method, we need to design $n_u + n_g = 5$ SISO controllers with $n_g = 2$ selectors to obtain optimal steady-state operation. Since $n_u > n_g$, we always have $n_u - n_g = 1$ unconstrained degree of freedom corresponding to the controlled variable $CV^0 = N^{0^T} \nabla_u \hat{J}$. From the nullspace of the full G^g matrix, we find that this direction is given by

$$N^0 = \begin{bmatrix} -0.36214 & -0.45268 & 0.81482 \end{bmatrix}^T.$$

In addition, there are two unconstrained directions related to the two constraints. We have that $CV^{0g}_1 = N^T_1 \nabla_u \hat{J}$ should be controlled when g_1 is not active, and $CV^{0g}_2 = N^T_2 \nabla_u \hat{J}$ should be controlled when g_2 is not active. These directions are:

$$\begin{bmatrix} 1 & 1 & 1 \\ -0.36214 & -0.45268 & 0.81482 \end{bmatrix} N_1 = 0 \Longrightarrow N_1 = \begin{bmatrix} -0.73179 \\ 0.67952 \\ 0.052271 \end{bmatrix}$$

$$\begin{bmatrix} 0.2 & -0.16 & 0 \\ -0.36214 & -0.45268 & 0.81482 \end{bmatrix} N_2 = 0 \Longrightarrow N_2 = \begin{bmatrix} 0.50902 \\ 0.63627 \\ 0.57971 \end{bmatrix}$$

For designing a decentralized control structure, a pairing between the constraints and the MVs must be performed. From the steady-state gain matrix G^g we see that u_3 should not be used to control g_1 (because of zero gain). Otherwise, there are no clear restrictions, and g_1 is arbitrarily paired to u_1, and g_2 is paired to u_2. We must require that the corresponding unconstrained optimal CVs are paired accordingly, meaning that CV^{0g}_1 is paired to u_1, CV^{0g}_2 is paired to u_2, and CV^0 is paired to u_3.

For selector design, Table 1 shows the transformed constraint gains calculated using Eq. (12) for all active constraint sets, and we verify that the gains are always positive for both constraints. This means that selectors are possible for both control loops and that both selectors should be "min"-selectors. The resulting control structure is shown in Fig. 6.

The cost gradient is estimated through a relinearization of the dynamic model at the current estimated state to obtain the following linear model:

$$\begin{cases} \dot{x} = Ax + Bu \\ J = C_J x + D_J u \end{cases} \tag{16}$$

where, by setting $\dot{x} = 0$, the estimated steady-state cost gradient becomes [13]

$$\nabla_u \hat{J} = -C_J A^{-1} B + D_J \tag{17}$$

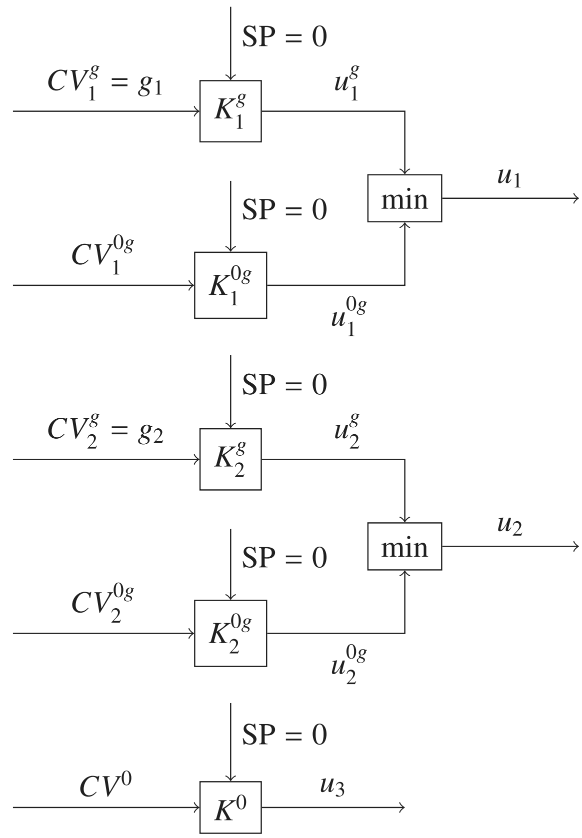

Fig. 6. Decentralized control structure for case study 1.

For state estimation, the model is augmented to include the disturbances as integrating states, according to:

$$\begin{cases} \begin{bmatrix} \dot{x} \\ \dot{d} \end{bmatrix} = \begin{bmatrix} A & B_d \\ 0 & 0 \end{bmatrix} \begin{bmatrix} x \\ d \end{bmatrix} + \begin{bmatrix} B \\ 0 \end{bmatrix} u \\ y = \begin{bmatrix} C & 0 \end{bmatrix} \begin{bmatrix} x \\ d \end{bmatrix} \end{cases} \tag{18}$$

To estimate the states, a continuous-time Kalman filter is implemented with this augmented model, and the estimated state $\hat{x}$ and current input u are used to evaluate the matrices in Eq. (16) at all times, leading to the estimated cost gradient $\nabla_u \hat{J}$ in (17) . The matrices A, B, B_d, C, and D are as defined in Eq. (13), and C_J and D_J are calculated from Eq. (14) to give:

$$C_J = \hat{x}^T \begin{bmatrix} 1 & 0 \\ 0 & 10 \end{bmatrix}$$

$$D_J = u^T \begin{bmatrix} 1 & -0.1 & -0.2 \\ -0.1 & 0.8 & -0.1 \\ -0.2 & -0.1 & 0.3 \end{bmatrix}$$

We emphasize that analytical expressions for these derivatives are available due to the simplicity of this case study, and we encourage the use of automatic differentiation tools to obtain these matrices in more realistic case studies.

The constraint controllers K^g_1 and K^g_2 were designed according to the SIMC rules [18] with the choice $\tau_{C,1} = 0.1$ s and $\tau_{C,2} = 0.01$ s. In terms of gradient control, we assume that the effect of the inputs on the estimated $\nabla_u \hat{J}$ is that of a pure gain process, neglecting any dynamics associated with the gradient estimation, and therefore the gradient controllers K^{0g}_1, K^{0g}_2, and K^0 become pure integral controllers. These were tuned according to the SIMC rules with $\tau_C = 0.5$ s. All controllers linked to selectors are implemented with anti-windup action

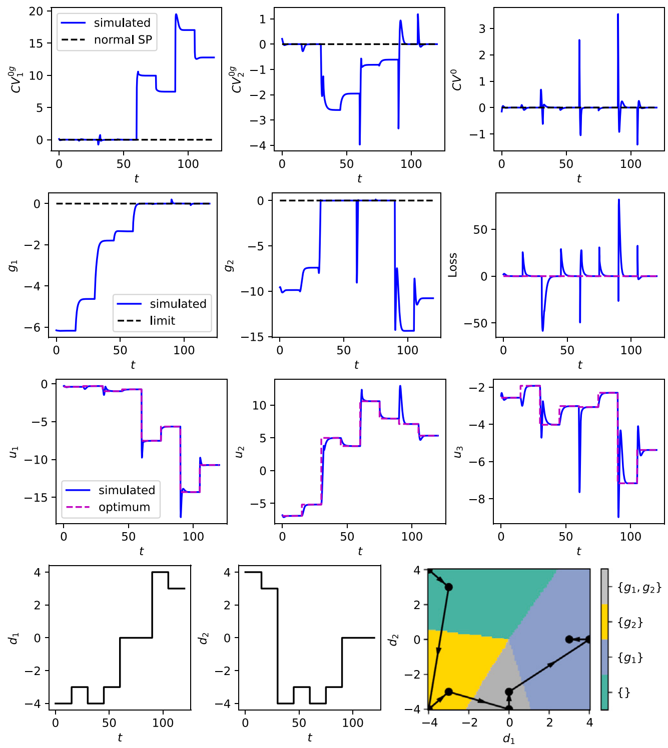

Fig. 7. Closed-loop simulation results for case study 1.

based on the back-calculation strategy [14,19], with a tracking time of $\tau_T = 0.01$ s. The resulting controller tunings are summarized in Table 2.

The closed-loop simulations are shown in Fig. 7. To validate the optimality of the control structure, the disturbances were changed stepwise every 15 s (see lower left plots) to make the system operate in all four active constraint regions (see lower right plot). It can be seen that constraint changes are effectively handled, giving up the corresponding gradient projection when a constraint becomes active, and that operation is driven to the optimal steady state for all disturbances.

4. Case study 2 - Williams–Otto reactor

The control structure proposed in Section 2 depends on using projection matrices. These are constant only when the constraints are linear in the MVs. We now consider a nonlinear case study where this assumption is not satisfied and one may expect economic losses in some regions. The case study is based on the process described by Williams and Otto [20] and studied in [16], see Fig. 8. It consists of a continuously stirred reactor tank with perfect level control, in which A and B are mixed, generating the main product P, the less interesting product E and the undesired byproduct G. The reactions and reaction rates are given by:

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

The component mass balances for the six components give the following set of ODEs:

$$\frac{dx_A}{dt} = \frac{F_A}{W} - \frac{(F_A+F_B)x_A}{W} - k_1 x_A x_B \tag{19a}$$
$$\frac{dx_B}{dt} = \frac{F_B}{W} - \frac{(F_A+F_B)x_B}{W} - k_1 x_A x_B - k_2 x_C x_B \tag{19b}$$

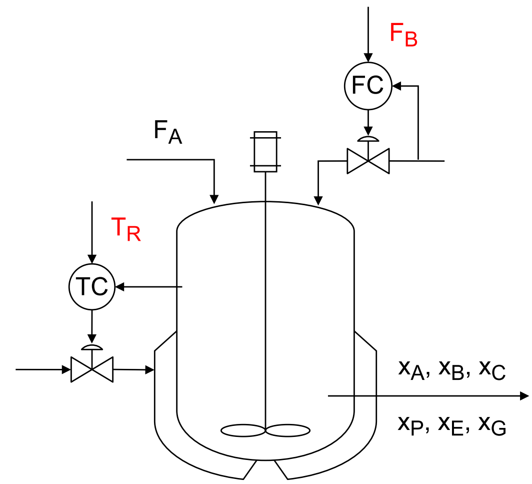

Fig. 8. Schematic representation of Williams–Otto reactor, with MVs in red.

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	11111 K
p_A	79.23 $/kg
p_B	118.34 $/kg
p_P	1043.38 $/kg
p_E	20.92 $/kg

$$\frac{dx_C}{dt} = -\frac{(F_A+F_B)x_C}{W} + 2k_1 x_A x_B - 2k_2 x_C x_B - k_3 x_P x_C \tag{19c}$$
$$\frac{dx_P}{dt} = -\frac{(F_A+F_B)x_P}{W} + k_2 x_C x_B - 0.5 k_3 x_P x_C \tag{19d}$$
$$\frac{dx_E}{dt} = -\frac{(F_A+F_B)x_E}{W} + 2k_2 x_C x_B \tag{19e}$$
$$\frac{dx_G}{dt} = -\frac{(F_A+F_B)x_G}{W} + 1.5 k_3 x_P x_C \tag{19f}$$

The model parameters for this case study are summarized in Table 3. The economic cost J includes the cost of reactants p_A and p_B and the selling price of products p_P and p_E, and the operational constraints are related to maximum allowed values for x_A and x_E. The steady-state optimization problem becomes

$$\begin{aligned} \min_u J &= p_A F_A + p_B F_B - (F_A+F_B)\left[p_P(1+\Delta p_P)x_P + p_E x_E\right] \\ \text{s.t.} \quad & g_1 = x_E - 0.30 \le 0 \\ & g_2 = x_A - 0.12 \le 0 \end{aligned} \tag{20}$$

The degrees of freedom (MVs) are $u = \begin{bmatrix} F_B & T_r \end{bmatrix}^T$, and the disturbances are $d = \begin{bmatrix} F_A & \Delta p_P \end{bmatrix}^T$, where Δp_P is the relative change in the price of the main product, as defined in Eq. (20).

The active constraint regions as a function of the two disturbances are shown in Fig. 9. In contrast to the previous case study, the lines delimiting each region are not straight. This alone should not affect the optimality of the proposed framework, as the optimality only requires that the constraints are linear in the MVs. However, since the latter does not hold for the case study, the use of constant projection matrices will lead to some economic loss.

We have $n_u = n_g = 2$ so Assumption 1 is satisfied. With the proposed method we need to design $n_u + n_g = 4$ SISO controllers with $n_g = 2$

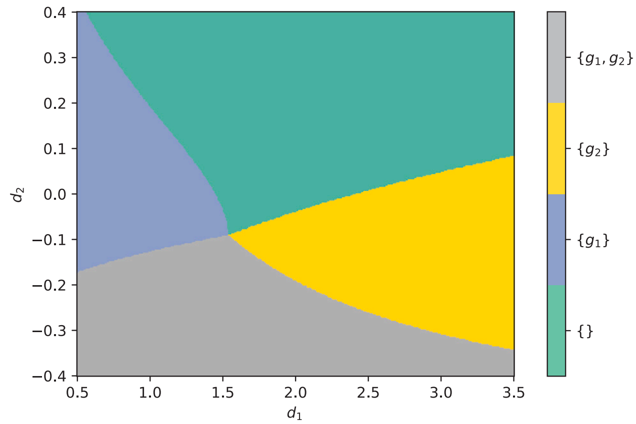

Fig. 9. Active constraint regions for case study 2 as a function of disturbances.

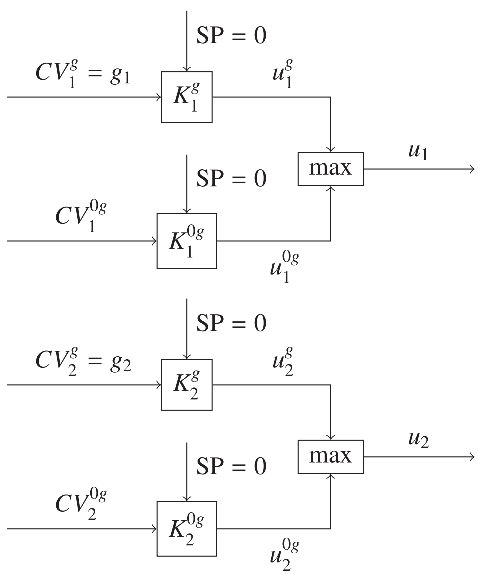

Fig. 10. Complete control structure for case study 2.

Table 4
Nominal operating point for case study 2.

Variable	Value
F_A	0.5 kg/s
Δp_P	0
F_B	1.4587 kg/s
T_r	342.537 K
x_A	0.0712 kg/kg
x_B	0.4107 kg/kg
x_C	0.0173 kg/kg
x_P	0.1246 kg/kg
x_E	0.3 kg/kg
x_G	0.0762 kg/kg

selectors to obtain optimal steady-state operation. To obtain the gain matrix G^g from the MVs to the constraints, we need to linearize the steady-state model of the constraints. In the following simulations, we use the linearization performed at the nominal operating point presented in Table 4, leading to fixed CVs for operation in all regions. This linearization gives:

$$G^g = \begin{bmatrix} -0.1045 & 0.003268 \\ -0.04379 & -0.00241 \end{bmatrix}$$

Table 5
Diagonal of $G^g_{P_\mathcal{A}}$ for all relevant sets $\mathcal{A}$ for case study 2.

$\mathcal{A}$	$(G^g_{P_\mathcal{A}})_{11}$	$(G^g_{P_\mathcal{A}})_{22}$
$\{\}$	-6.01×10^{-4}	−0.0279
$\{1\}$	–	−0.0287
$\{2\}$	-5.05×10^{-4}	–

Table 6
PI controller tunings for case study 2.

Controller	Parameter	Value
K_1^g	K_c	−430.6
	τ_I	0.225 h
K_2^g	K_c	−2988
	τ_I	0.072 h
K_1^{0g}	K_I	−1.833
K_2^{0g}	K_I	202.5

Since $n_u = n_g$, the system has no completely unconstrained degrees of freedom, so there are no variables CV^0 that are always controlled. The gradient projections N_1 and N_2 for the two potentially unconstrained degrees of freedom become:

$$\begin{bmatrix} -0.04379 & -0.00241 \end{bmatrix} N_1 = 0 \Longrightarrow N_1 = \begin{bmatrix} -0.05499 \\ 0.9985 \end{bmatrix}$$

$$\begin{bmatrix} -0.1045 & 0.003268 \end{bmatrix} N_2 = 0 \Longrightarrow N_2 = \begin{bmatrix} 0.03126 \\ 0.9995 \end{bmatrix}$$

For MV–CV pairing, we choose $u_1 = F_B$ for controlling g_1 and $CV_1^{0g} = N_1^T \nabla_u \hat{J}$, and $u_2 = T_r$ controlling g_2 and $CV_2^{0g} = N_2^T \nabla_u \hat{J}$. This pairing choice was made based on the steady-state RGA for constraint control, which gives $\lambda = 0.638$ for the chosen pairing. For designing the selectors according to Theorem 3, a local analysis of the transformed constraint gains given in Eq. (12) was made at the nominal point and is summarized in Table 5. The projected gains are negative for both constraints regardless of the active set, which means that both selectors should be "max" selectors. The resulting control structure is presented in Fig. 10.

To tune the controllers, we obtained the following transfer functions from the MVs to the constraints (with time in hours):

$$G_{11}(s) = \frac{-0.1045}{0.225\ s + 1}, \quad G_{22}(s) = \frac{-0.00241}{0.072\ s + 1}$$

Based on this, the PI controllers for the constraints were tuned using the SIMC rules [18] with $\tau_{C,1} = 0.005$ h and $\tau_{C,2} = 0.01$ h. Similar to the previous case study, the method for gradient estimation is again considered to be instantaneous with respect to the inputs, meaning that the gradient controllers become integral controllers, tuned using the SIMC rules with $\tau_C = 0.05$ h. All four controllers were implemented with anti-windup action based on back-calculation with a tracking time of $\tau_T = 0.01$ h. The controller tunings are summarized in Table 6.

Closed-loop dynamic simulations are presented in Fig. 11. The disturbances were changed so that all four active constraint regions were explored. Since we consider that the cost gradient $\nabla_u \hat{J}$ is an available measurement (from an estimator with a perfect model), operation in the fully unconstrained region (from $t = 21$ h to $t = 27$ h) is optimal at steady state, which can be seen by the input values converging to the exact steady-state optimal value. Since we assume that the constraints are directly measured (which is a mild assumption), the same logic applies to the fully constrained region from $t = 9$ h to $t = 15$ h. In addition, operation is optimal at the nominal point by design. In the two remaining partly constrained regions, the system does not converge exactly to the steady-state optimum, but the constraints are always satisfied (except for short dynamic transients, which may be avoided by introducing a back-off for the constraints).

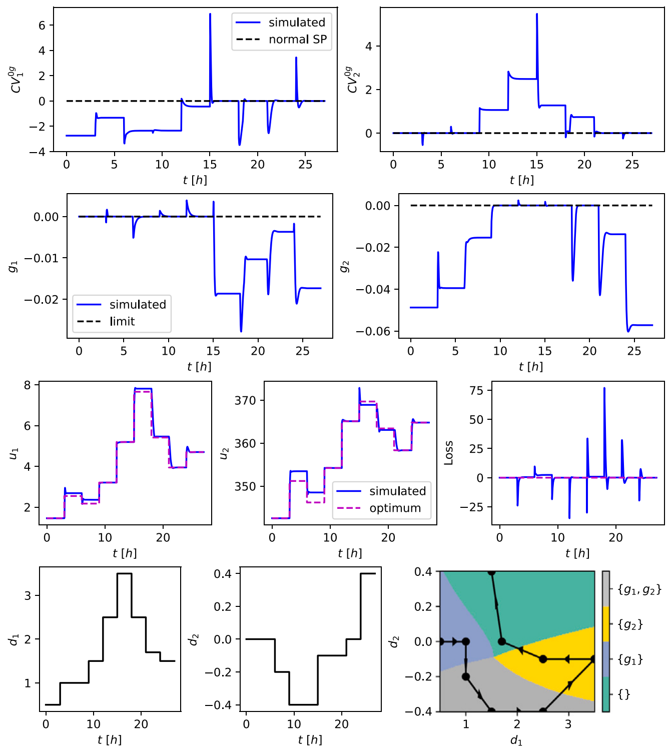

Fig. 11. Closed-loop simulation results for case study 2.

It is interesting to note that for the third set of disturbances ($d = [1.0,\ -0.2]^T$ from $t = 6$ h to 9 h), the second constraint ($g_2 = 0$) is not controlled, even though it should be optimally controlled together with $g_1 = 0$. Instead, the selector logic results in the control of $CV_2^{0g} = 0$, which can be done without violation of g_2, that is, constraint g_2 is "over-satisfied". The reason for this non-optimal operation is that the selected value for projection matrix N_2 is not optimal in this operating region.

The steady-state economic loss is better visualized as a function of the disturbances in Fig. 12. The highest losses are observed around where we ideally should switch between the partly constrained and the fully constrained regions. The optimal switch between these regions (black lines) does not coincide with the actual switch obtained with the selectors (blue lines). Economic loss is observed before the optimal switch due to the inaccuracy of the projection matrices. For the same reason, and because this further leads to suboptimal performance of the selectors, economic loss is also seen between the optimal and actual switch of CVs. However, the optimal switch between the fully unconstrained and the partly constrained regions coincides with the actual switch between the corresponding CVs. This happens because, before the switch, the full cost gradient $\nabla_u \hat{J}$ is controlled to zero, leading to zero economic loss, and the constraint becomes active immediately at the switch. Therefore, at this switch, the economic loss is zero, and it continuously grows as the system moves further into the partly constrained region.

5. Discussion

5.1. Steady-state cost gradient estimation

The results in this paper assume the availability of the steady-state cost gradient $\nabla_u J$ during operation. This can be fulfilled through model-based estimation, model-free estimation, or a combination of both methods [3]. Model-free methods usually depend on the perturbation of the inputs, and when the constraints are being controlled the perturbation can be done in their setpoints instead. In the presented case studies, we used a model-based approach, where a Kalman Filter was used to estimate the current dynamic state x and disturbance d with an augmented model (18), and then setting $\dot{x} = 0$ in the linearized model (16) leads to the gradient estimate (17).

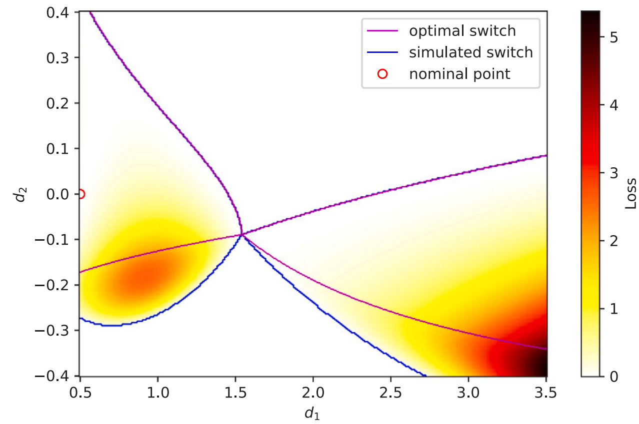

Fig. 12. Steady-state closed-loop economic loss for case study 2.

Because the cost gradient $\nabla_u J$ is, by definition, a steady-state variable, it is not well-defined during a dynamic transition, and any gradient estimator must make some steady-state assumption or prediction. The necessity of estimating the cost gradient is related to ensuring exact optimality. In practice, one would wish to use an approximation of the cost gradient that is more easily implementable, even if that means accepting some economic loss. In that sense, data-driven approaches for this estimation would be appealing, as well as self-optimizing control methods that provide an approximation for the cost gradient through a static combination of measurements [6].

However, a simpler approach is to use a static estimation of $\nabla_u J$ directly based on the measurements y. In another paper [21], we prove the optimality of a simple linear steady-state gradient estimate of the form

$$\nabla_u \hat{J} = H y - c_s$$

where y are the measurements, and the constant vector c_s and the constant matrix H are obtained using the "exact local method" of self-optimizing control. In addition, a correction of c_s from a more accurate gradient estimator may be applied on a slower time scale, for example, using a model-based approach like RTO or a data-based perturbation method like extremum-seeking control.

5.2. Handling of constraints

Constraints in process systems are usually measured or estimated, and our approach is optimal for such cases, as control loops are implemented to handle these constraints. In MPC applications, where the problem is formulated as a dynamic trajectory optimization, it is common that process constraints are posed as constraints on the dynamic states, but this is not how our approach handles this issue. Rather, our method is focused on process constraints that may become active at steady state and influence process economics.

5.3. Updating of projection matrices

In the simulations presented in this paper, we assumed that a linearization of the constraints at a nominal operating point would be sufficiently accurate for capturing the transitions between active constraint regions. This simplification was primarily made to ensure a control structure that can be easily implemented, and it led to acceptable results even for a nonlinear case study (see Fig. 12). However, it is possible to enhance economic performance by updating the projection matrices N and N^0 during operation. To accomplish this, an accurate estimator for the complete constraint gradient matrix G^g is required, and typically such an estimator is only available at a time scale similar to that of RTO. However, our primary objective is to achieve acceptable economic losses in fast time scales, and this could be accomplished by using constant projection matrices.

5.4. Controller tuning

Even though the proposed control structure only has $n_u + n_g$ SISO controllers to be designed, the tuning of these controllers may prove to be challenging. This is because these controllers must work in many different regions (up to 2^{n_g} theoretical regions), and the interaction between loops will change depending on which controllers are active. The pairing between MVs and CVs should consider this, and the tuning for the loops should be robust in the sense that acceptable performance is attained for every operation mode. This issue was not noticed in the case studies in this work, but it is easy to see that it may arise in practice.

5.5. Limitations for systems with many constraints

In this paper, we consider a class of problems with $n_u \geq n_g$, so it is possible to devise a simple, decentralized control structure. There is a particular case of systems with more constraints than inputs that can fit into the framework proposed in this work. That would be the case where the constraints can be arranged into n_g groups, where each group is comprised of constraints that have parallel gain vectors with respect to the inputs, *i.e.* the constraints g_i and g_j would belong to the same group if $\nabla_u g_i = \alpha \nabla_u g_j$ for some nonzero α. In practice, this would represent a process variable with lower and upper bounds, or constraints of similar nature caused by different factors, *e.g.* a maximum processing rate due to upstream or downstream conditions. Each of these groups has a unique characteristic direction in terms of the rows of G^g, which can be used to calculate the gradient projections with the methodology described in Theorem 2. Each of these groups should then be organized internally following the single-variable methodology described by Krishnamoorthy and Skogestad [16]. As the methodology devised in this paper mitigates the correlation between each group of constraints, the gradient projections that serve as unconstrained CVs remain constant with respect to changes in the remaining loops, and therefore no additional logic is required in the implementation of max/min selectors for constraint handling.

The main case not covered by the present methodology is when there are $n_g > n_u$ *independent* constraints that may become active, expressly violating Assumption 1. In this case, considering that some pairing between MVs and constraints is done, the first problem that arises is the possibility of constraints paired with the same MV becoming active at the same time, requiring that one or several constraints become controlled by other MVs. A heat exchanger case with $n_g = 3 > n_u = 2$ was studied in Bernardino et al. [22], where it was shown that a region-based approach similar to the one studied in the present paper fails to achieve optimality for some disturbance scenarios, whereas the primal–dual approach always reaches the optimal steady state. To achieve optimality with a region-based approach, an adaptive pairing strategy may be used, as described for this case study in Bernardino et al. [23]. This gives optimal operation for all disturbances, but the adaptive pairing becomes quite complicated (see Figure 2 in [23]).

In this sense, general strategies for switching pairings require more complex logic, and currently, there is no systematic arranging of classical control logic blocks that can account for that. On top of that, even if conflicting constraints are not an issue for the considered operating window, *i.e.* constraints paired to the same MV do not become active at the same time for the considered disturbances, the design of controllers for the unconstrained degrees of freedom becomes more complicated. As the constraints are assumed to be independent, the gradient projections optimally controlled to zero will be different when each of them is active. This entails that the remaining control loops have to change depending on which controller related to this MV is active. Therefore, proposing decentralized control structures for the optimal operation of systems with more constraints than MVs inevitably leads to complex and interacting control loops, and centralized strategies such as the primal–dual feedback optimizing control presented in Krishnamoorthy [24] or MPC become more appealing.

For the same reason, the proposed framework has limitations in optimal dealing with input saturation. In real systems, every MV has physical bounds $u^{lb} \leq u \leq u^{ub}$ in addition to the process constraints g. Therefore, every physical system in a way has more constraints than MVs, and one must identify the constraints that are more likely to become active if one follows Theorem 2 for designing a control structure. The choice of not pairing an MV that may saturate with an important CV, in this case, an economic constraint, agrees with the rule of thumb "pair an MV that may saturate with a CV that may be given up" [25], as the gradient projections paired to that MV should by design be given up in case of MV saturation.

The proposed framework attains optimal operation in a wide operating range, by enforcing optimality conditions at steady state for all possible active set combinations for a maximum of n_u independent constraints. It should also be emphasized that less frequent constraints can still be dealt with in the current framework by the implementation of more selectors, even if $n_g > n_u$, bearing in mind that steady-state optimal operation will not be guaranteed when those become active due to changes in the unconstrained CVs. However, violation of such infrequent constraints would be prevented, which is the main goal of such additional control loops.

5.6. Stability and optimal convergence of selectors

The control strategy proposed in this work relies on switching blocks to perform optimal operation in different operating regions. Analyzing the stability of switched systems is more complex, as the stability of a switched system may not necessarily match that of its corresponding continuous subsystems [26]. In Theorem 3, we assume that each subsystem within the switching system is stable, which is a condition already present and well described when using decentralized control in multivariable systems. By ensuring that every subsystem is stable, the overall stability of the switching system can be guaranteed. This can be achieved by enforcing a sufficiently large average dwell time [27], which is a practical and easily implementable solution.

The conditions for implementing min/max selectors to detect switches in active constraints optimally are outlined in Theorem 3. This theorem is based on a local analysis of the optimization problem and is rigorously applicable to problems with a constant positive definite Hessian J_{uu} and constant constraints gain G^g. A relevant case in practice is that of linear economic objectives, for which J_{uu} is positive semidefinite, but this case is always solved by active constraint control, as there are no unconstrained degrees of freedom to be determined. While the presented proof does not address generic nonlinear optimization problems, it provides a useful local test that can eliminate certain impossible configurations resulting from the chosen MV–CV pairings or the formulation of the optimal operation problem itself. If the conditions specified in Theorem 3 are not satisfied, we recommend utilizing the cascade framework presented in Fig. 4.

The condition derived in Theorem 3 for applicability of selectors would only be violated by highly interacting systems, where the sign of the transformed constraint gain ($G^g_{P_{\mathcal{A}}}$) would change depending on the active loops. This condition is conjectured to be associated with the decentralized integral controllability (DIC) of each potential subsystem [28]. In our study, we could not find an example of a linear system with a convex objective function and without DIC that does not satisfy the conditions stated in the selector theorem. This further suggests a connection between these concepts and that the DIC conditions possibly imply the applicability of selectors. The link between the conditions of Theorem 3 and controllability aspects remains an open challenge that requires further investigation.

5.7. Other switching approaches

In this work, we have proposed the use of selectors in the controller outputs for detecting switches in active constraints. However, other strategies for adaptively controlling constraints in the context of optimal operation have been proposed. Manum and Skogestad [10] studied the problem of active constraint switching in self-optimizing control by tracking the self-optimizing CVs in neighboring regions, where the switching happens when there is a change of sign in the monitored variable. In the notation herein presented, this would be equivalent to the following switching logic:

- If $CV_i^{0g} = N_i^T \nabla_u J$ is being controlled to zero, a change of sign in g_i means that the ith constraint became active, as this sign change corresponds to constraint violation;
- Conversely, if g_i is being controlled to zero, a change of sign in $CV_i^{0g} = N_i^T \nabla_u J$ means that the ith constraint became inactive, as this sign change corresponds to a change in the objective function slope.

The problem with implementing such logic lies in the resulting dynamics of the control system. As this logic implies that the reference variable is perfectly controlled for accurate detection, the logic should operate in a slower time scale than that of the closed-loop system, which would in turn result in undesired behavior, especially constraint violation. Operating the switching logic in fast time scales could in turn lead to the appearance of limit cycles, due to self-sustained switching between control loops.

We have also presented the cascade control structure in Fig. 4 as an alternative switching strategy. A similar idea has been proposed by Cao [7] to promote self-optimizing operation at the nominal region while sub-optimally coping with constraint satisfaction. There are however some disadvantages to this approach related to the limitations that the cascade structure imposes. If constraint control is slow, controlling the corresponding gradient projection becomes unnecessarily slow. Moreover, even though constraint control may help with decoupling the system, it may also cause the opposite problem, and the interaction between loops may impose limitations on the performance of the upper layer. Therefore, the use of a cascade framework for optimal operation may be beneficial, but the improvement that it may bring must be assessed for each particular case study.

Recently, the work of Ye et al. [29] has tackled the problem of changing active constraints by embedding the switching constraints into the CV design, generating a single nonlinear CV. Because the resulting CV design problem was deemed intractable in most cases, a neural network was used to approximate the behavior of this theoretical CV. It is interesting to note that the switching behavior still happens in the designed CV, with the exact ideal CV being in general non-smooth. This is expected because of the nature of the problem, and although neural networks can approximate these variables, the interpretability of the resulting CV is lost, and constraint control must be explicitly performed elsewhere. In the present work, we deal with the switching explicitly, controlling the constraints directly when it is optimal.

6. Conclusion

We propose a simple framework for decentralized optimizing control with changing active constraints. The starting point is that at steady state, optimal economic operation in a given active constraint region $\mathcal{A}$ is achieved by keeping the controlled variables $CV = \left[g_{\mathcal{A}}; N_{\mathcal{A}}^T \nabla_u J(u^*, d)\right]$ at constant setpoints $CV^{sp} = 0$ (Theorem 1 [6]). Here $g_{\mathcal{A}}$ denotes the set of active steady-state constraints, and $N_{\mathcal{A}}^T \nabla_u J(u^*, d)$ is the reduced steady-state cost gradient for the remaining unconstrained degrees of freedom.

There are some degrees of freedom in the choice of the directions in the unconstrained nullspace $N_{\mathcal{A}}$ and to implement constraint switching in a simple manner, these should be chosen in accordance with the constraint directions. The main contribution of this paper is to prove in Theorem 2 for the case with $n_u \geq n_g$, that we should control the unconstrained variables $CV^0 = N^{0^T} \nabla_u J$ (which are not affected by the constraints), and in addition, depending on whether the constraint g_i is active or not, either control the constraint $CV_i^g = g_i$ or the associated unconstrained variable $CV_i^{0g} = N_i^T \nabla_u J$, where N^0 and N_i are calculated according to (7)–(9). This can be implemented with the simple control structure in Fig. 2. Furthermore, Theorem 3 shows that the switching can be performed with min/max selectors, which leads to the simple control structure in Fig. 3. Here, no centralized supervisor is needed to determine the active constraints, as the switching logic uses local feedback controllers.

CRediT authorship contribution statement

Lucas Ferreira Bernardino: Writing – review & editing, Writing – original draft, Visualization, Validation, Software, Methodology, Conceptualization. **Sigurd Skogestad:** Writing – review & editing, Visualization, 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.

Acknowledgments

The authors thank Johannes Jäschke for fruitful discussions.

Funding

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

Appendix A. Proof of Theorem 3

Proof. Define a set $A \subset \{1, 2, \ldots, n_g\}$ and an index i such that $i \notin A$, and another set A^* such that $A^* = A \cup \{i\}$. We prove the theorem by comparing the solution of the optimization problems with active sets A^* and A, *i.e.* what is the effect of controlling $g_i = 0$ instead of the corresponding unconstrained degree of freedom $CV_i^{0g} = N_i^T \nabla_u J = 0$ for arbitrary A and i.

The optimality conditions for A^* as the active set are given by:

$$\begin{cases} \nabla_u \mathcal{L} = J_{uu} u^{A^*} + J_{u,m}(d) + (G_{A^*}^g)^T \lambda^{A^*} = 0 \\ g_{A^*}(u^{A^*}, d) = G_{A^*}^g u^{A^*} + g_{A^*}^m(d) = 0 \end{cases}$$

in which the constraint-related matrices are partitioned with relation to the active set A and the remaining index i as $G_{A^*}^g = \begin{bmatrix} (G_A^g)^T & (G_i^g)^T \end{bmatrix}^T$ and $g_{A^*}^m(d) = \begin{bmatrix} g_A^m(d)^T & g_i^m(d)^T \end{bmatrix}^T$.

Eliminating u^{A^*} from the first equation gives:

$$u^{A^*} = -J_{uu}^{-1} J_{u,m}(d) - J_{uu}^{-1} (G_{A^*}^g)^T \lambda^{A^*}$$

Substituting this equation into the second optimality condition gives the following expression for λ^{A^*}:

$$(G^g_{A^*}J^{-1}_{uu}(G^g_{A^*})^T)\lambda^{A^*} = g^m_{A^*}(d) - G^g_{A^*}J^{-1}_{uu}J_{u,m}(d)$$

The variables can be partitioned as follows:

$$\begin{bmatrix} G^g_A J^{-1}_{uu}(G^g_A)^T & G^g_A J^{-1}_{uu}(G^g_i)^T \\ G^g_i J^{-1}_{uu}(G^g_A)^T & G^g_i J^{-1}_{uu}(G^g_i)^T \end{bmatrix}\begin{bmatrix}\lambda^{(A)} \\ \lambda^i\end{bmatrix} = \begin{bmatrix} g^m_A(d) - G^g_A J^{-1}_{uu}J_{u,m}(d) \\ g^m_i(d) - G^g_i J^{-1}_{uu}J_{u,m}(d)\end{bmatrix}$$

The same procedure with A as the active set leads to:

$$\begin{cases} u^A = -J^{-1}_{uu}J_{u,m}(d) - J^{-1}_{uu}(G^g_A)^T\lambda^A \\ (G^g_A J^{-1}_{uu}(G^g_A)^T)\lambda^A = g^m_A(d) - G^g_A J^{-1}_{uu}J_{u,m}(d)\end{cases}$$

The term $(G^g_{A^*})^T\lambda^{A^*} = (G^g_A)^T\lambda^{(A)} + (G^g_i)^T\lambda^i$ can be expressed in terms of the solution for A, as follows:

$$\begin{aligned}\lambda^{(A)} &= (G^g_A J^{-1}_{uu}(G^g_A)^T)^{-1}\left(g^m_A(d) - G^g_A J^{-1}_{uu}J_{u,m}(d) - G^g_A J^{-1}_{uu}(G^g_i)^T\lambda^i\right) \\ &= \lambda^A - (G^g_A J^{-1}_{uu}(G^g_A)^T)^{-1}G^g_A J^{-1}_{uu}(G^g_i)^T\lambda^i \\ \Longrightarrow (G^g_{A^*})^T\lambda^{A^*} &= (G^g_A)^T\lambda^A + \left(I - (G^g_A J^{-1}_{uu}(G^g_A)^T)^{-1}G^g_A J^{-1}_{uu}\right)(G^g_i)^T\lambda^i\end{aligned}$$

Therefore, u^{A^*} can be expressed in terms of u^A as follows:

$$\begin{aligned}u^{A^*} &= -J^{-1}_{uu}J_{u,m}(d) - J^{-1}_{uu}(G^g_{A^*})^T\lambda^{A^*} \\ &= u^A - J^{-1}_{uu}\left(I - (G^g_A J^{-1}_{uu}(G^g_A)^T)^{-1}G^g_A J^{-1}_{uu}\right)(G^g_i)^T\lambda^i\end{aligned}$$

The transformation $P_{\mathcal{A}} = J^{-1}_{uu}\left(I - (G^g_A J^{-1}_{uu}(G^g_A)^T)^{-1}G^g_A J^{-1}_{uu}\right)$ is equivalent to a scaled projection to the nullspace of G^g_A, $N_{(\mathcal{A})}$ [17], according to the identity $P_{\mathcal{A}} = N_{(\mathcal{A})}\left(N^T_{(\mathcal{A})}J_{uu}N_{(\mathcal{A})}\right)^{-1}N^T_{(\mathcal{A})} = J^{-1}_{uu}\left(I - (G^g_A J^{-1}_{uu}(G^g_A)^T)^{-1}G^g_A J^{-1}_{uu}\right)$, and is therefore positive semidefinite. The effect of inclusion of an arbitrary constraint g_i is therefore given by:

$$u^{A^*} - u^A = -P_{\mathcal{A}}(G^g_i)^T\lambda^i \tag{A.1}$$

We can see that the ith component of the vector $P_{\mathcal{A}}(G^g_i)^T$ dictates the steady-state behavior of the ith MV when the ith constraint becomes active and the system is operating at the active set A. Following the notation introduced in the statement of Theorem 3 and in Fig. 3, we have $u^g_i = (u^{A^*})_i$ and $u^{0g}_i = (u^A)_i$, since these are the MV values such that $g_i = 0$ and $CV^{0g}_i = N^T_i\nabla_u J = 0$, respectively. This means that, for $(P_{\mathcal{A}}(G^g_i)^T)_i > 0$, $u^g_i - u^{0g}_i < 0$ when $\lambda^i > 0$ and consequently u^g_i should be selected, and $u^g_i - u^{0g}_i > 0$ when $\lambda^i < 0$ and consequently u^{0g}_i should be selected, meaning that we must use $u^*_i = \min(u^{0g}_i, u^g_i)$ for guaranteeing optimality. Similar analysis can be performed for $(P_{\mathcal{A}}(G^g_i)^T)_i < 0$, leading to $u^*_i = \max(u^{0g}_i, u^g_i)$. Since this analysis is performed for arbitrary i and $\mathcal{A} \not\ni i$, guaranteeing that $(P_{\mathcal{A}}(G^g_i)^T)_i = (G^g_{P_{\mathcal{A}}})_{ii}$ has the same sign for any possible $\mathcal{A}$ is sufficient for guaranteeing the theorem statement, which completes the proof. □

Appendix B. Optimality of min selectors in cascade structure

In this section, we restate Theorem 3 for the cascade case illustrated in Fig. 4, proving its optimality. In this control structure, the manipulated variables as seen from the higher layer can be represented at steady-state as:

$$v = \begin{bmatrix} g_1 \\ \vdots \\ g_{n_g} \\ u_{n_g+1} \\ \vdots \\ u_{n_u}\end{bmatrix} \Longrightarrow \Delta v = \begin{bmatrix} G^g \\ \hline 0_{(n_u-n_g)\times n_g} \;\vert\; I_{n_u-n_g}\end{bmatrix}\Delta u = W\Delta u$$

With this change of variables, the transformed optimization problem becomes:

$$\begin{aligned}\min_v \quad & J(v,d) \\ \text{s.t.} \quad & v_i \le 0, \quad i = 1,\dots,n_g\end{aligned} \tag{B.1}$$

Note that we must require that W, the Jacobian for the change of variables, is full rank, such that the optimality conditions for Eq. (B.1) and Eq. (1) are equivalent. This is a mild assumption related to the steady-state controllability of the constraints in the lower layer with the chosen pairing, and it results in a transformed Hessian $J_{vv} = W^T J_{uu} W$ that is also positive definite. Also, for the transformed problem, we have the gain matrix $G^{g,v}$ from the transformed inputs to the constraints:

$$G^{g,v} = \begin{bmatrix} I_{n_g} & 0_{n_g\times(n_u-n_g)}\end{bmatrix}$$

This allows us to write the optimal CVs in terms of the projection matrices N_i and N^0 for the transformed problem as:

$$N^0 = \begin{bmatrix} 0_{n_g\times(n_u-n_g)} \\ I_{n_u-n_g}\end{bmatrix}$$

$$N_i = \hat{e}_i, \quad i = 1,\dots,n_g$$

The procedure for obtaining the difference in the optimal solution for neighboring regions presented in Appendix A is also valid for the transformed problem, and we must therefore analyze the sign of the diagonal of the matrix product $G^{g,v}P_{\mathcal{A}}$. Recall that $P_{\mathcal{A}} = N_{(\mathcal{A})}\left(N^T_{(\mathcal{A})}J_{vv}N_{(\mathcal{A})}\right)^{-1}N^T_{(\mathcal{A})}$, where the matrix $N^T_{(\mathcal{A})}J_{vv}N_{(\mathcal{A})}$ is positive definite, being here a principal submatrix of J_{vv} which selects the rows and columns with indexes not in $\mathcal{A}$. Therefore, $P_{\mathcal{A}}$ becomes a positive semidefinite $n_u\times n_u$ matrix, where the diagonal elements $P_{\mathcal{A}_{ii}}$ are zero for $i\in\mathcal{A}$ and positive for $i\notin\mathcal{A}$.

Finally, we can see that $(G^{g,v}P_{\mathcal{A}})_{ii} > 0$ for $i\notin\mathcal{A}$, since the first n_g elements of the diagonal of $G^{g,v}P_{\mathcal{A}}$ are the same as those of $P_{\mathcal{A}}$. It follows that, according to Eq. (A.1) and the results here obtained, the optimal solution is given by $v^*_i = \min(v^{0g}_i, 0)$ for $i = 1,\dots,n_g$, which completes the proof.

References

[1] M. Ellis, H. Durand, P.D. Christofides, A tutorial review of economic model predictive control methods, J. Process Control 24 (8) (2014) 1156–1178.

[2] M. Morari, Y. Arkun, G. Stephanopoulos, Studies in the synthesis of control structures for chemical processes: Part I: Formulation of the problem. Process decomposition and the classification of the control tasks. Analysis of the optimizing control structures, AIChE J. 26 (2) (1980) 220–232.

[3] D. Krishnamoorthy, S. Skogestad, Real-time optimization as a feedback control problem - a review, Comput. Chem. Eng. (2022) 107723.

[4] S. Skogestad, Plantwide control: The search for the self-optimizing control structure, J. Process Control 10 (5) (2000) 487–507.

[5] A. Maarleveld, J. Rijnsdorp, Constraint control on distillation columns, Automatica 6 (1) (1970) 51–58.

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

[7] Y. Cao, Constrained self-optimizing control via differentiation, IFAC Proc. Vol. 37 (1) (2004) 63–70.

[8] 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.

[9] L. Ye, Y. Cao, S. Skogestad, Global self-optimizing control for uncertain constrained process systems, IFAC-PapersOnLine 50 (1) (2017) 4672–4677.

[10] H. Manum, S. Skogestad, Self-optimizing control with active set changes, J. Process Control 22 (5) (2012) 873–883.

[11] A. Reyes-Lúa, S. Skogestad, Systematic design of active constraint switching using classical advanced control structures, Ind. Eng. Chem. Res. 59 (6) (2019) 2229–2241.

[12] S. Skogestad, I. Postlethwaite, Multivariable Feedback Control: Analysis and Design, john Wiley & sons, 2005.

[13] D. Krishnamoorthy, E. Jahanshahi, S. Skogestad, Feedback real-time optimization strategy using a novel steady-state gradient estimate and transient measurements, Ind. Eng. Chem. Res. 58 (1) (2018) 207–216.

[14] S. Skogestad, Advanced control using decomposition and simple elements, Annu. Rev. Control 56 (2023) 100903.

[15] D. Krishnamoorthy, S. Skogestad, Online process optimization with active constraint set changes using simple control structures, Ind. Eng. Chem. Res. 58 (30) (2019) 13555–13567.

[16] D. Krishnamoorthy, S. Skogestad, Systematic design of active constraint switching using selectors, Comput. Chem. Eng. 143 (2020) 107106.

[17] J. Nocedal, S.J. Wright, Numerical Optimization, second ed., Springer New York, NY, 2006.

[18] S. Skogestad, Simple analytic rules for model reduction and PID controller tuning, J. Process Control 13 (4) (2003) 291–309.
[19] K.J. Åström, L. Rundqwist, Integrator windup and how to avoid it, in: 1989 American Control Conference, IEEE, 1989, pp. 1693–1698.
[20] 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 (5) (1960) 458–473.
[21] L.F. Bernardino, S. Skogestad, Optimal measurement-based estimate of the cost gradient for real-time optimization. (in preparation).
[22] L.F. Bernardino, D. Krishnamoorthy, S. Skogestad, Comparison of simple feedback control structures for constrained optimal operation, IFAC-PapersOnLine 55 (7) (2022) 883–888.
[23] 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.
[24] D. Krishnamoorthy, A distributed feedback-based online process optimization framework for optimal resource sharing, J. Process Control 97 (2021) 72–83.
[25] S. Skogestad, Control structure design for complete chemical plants, Comput. Chem. Eng. 28 (1–2) (2004) 219–234.
[26] D. Liberzon, A.S. Morse, Basic problems in stability and design of switched systems, IEEE Control Syst. Mag. 19 (5) (1999) 59–70.
[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] J. Lee, T.F. Edgar, Conditions for decentralized integral controllability, J. Process Control 12 (7) (2002) 797–805.
[29] L. Ye, Y. Cao, Y. He, C. Zhou, H. Su, X. Tang, S. Yang, Generalized global self-optimizing control for chemical processes part I. The existence of perfect controlled variables and numerical design methods, Ind. Eng. Chem. Res. 62 (37) (2023) 15051–15069.