Systematic design of active constraint switching using selectors

Dinesh Krishnamoorthy, Sigurd Skogestad

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

Abstract

Selector logic is a simple and effective tool to switch between different controlled variables associated with change in active constraints. Selector blocks have been extensively used in the process control industry for decades, but their design has been based on engineering intuition and experience. Currently, there is a lack of systematic procedure to design selectors for active constraint switching. In this paper, we address this gap and provide a systematic procedure, which can be applied without the need for detailed process models. Illustrative examples are used to demonstrate the proposed framework.

1. Introduction

Selector logic blocks, also commonly known as *overrides* have been in use in the process industries for several decades to switch between a plurality of controllers, and are available as a part of any standard digital control system (DCS) software package. Despite their widespread use, not just in the process industries, but also in other application domains, selectors are currently designed in an ad-hoc fashion based on engineering know-how and experience [1, Section 1.17].

There is a lack of systematic design procedure that tells when to use a max-selector, when to use a min-selector, and when is it not feasible to use a selector. Industry-oriented and process control books such as [2, Ch. 6], [3, Ch. 12], [4, Ch. 18], [5, Ch. 10] and [6, Ch. 22] demonstrate the use of selectors using various illustrative examples, but there is no systematic procedure. [7] studies several control structures, however the analysis is limited to only one CV constraint, and does not address the case with several CV constraints. Although there have been a few works studying the stability of min-max selectors, see for example [8, 9, 10] and the references therein, the choices that affect design of the selectors itself still remains an open problem.

In this paper, we address this gap by providing a systematic design procedure. In particular, we study the use of selectors for active constraint switching in the context of economic optimal operation.

Often optimal operation occurs when some of the constraints are at their limiting values. For example, when the market price for the product is high, then the process should be operated at its maximum production capacity, in order to maximize the profit. Constraints that are optimally at their limiting values are known as *active constraints*. Because of changes in the operating conditions (disturbances) and market prices, the set of active constraints changes. An *active constraint region* is a disturbance space that is defined by the set of constraints that are active within it [11].

Typically, real-time optimization (RTO) involves solving a numerical optimization problem using rigorous nonlinear process models to compute the optimal setpoints, which are given to the control layer below. Recently, there is an increasing interest in achieving optimal operation, without the need to solve numerical optimization problems [12, 13, 14, 15, 16, 17, 18]. In other words, the economic objectives are translated into control objectives, thereby achieving optimal operation using simple feedback control structures [19]. The idea of "*feedback optimizing control*" dates back to [20]. The main idea of feedback optimizing control (also sometimes referred to as self-optimizing control[19], or direct input adaptation [13]) is to find the right set of controlled variables (CVs), which when held constant, leads to economically optimal operation [20, 19].

When some of the constraints are optimally active, then the simplest approach is to control the constraints at their limits, possibly with some safety margin, known as *back-off*. This is known as *active constraint control*. If there are any remaining unconstrained degrees of freedom (manipulated variables), one should then identify *self-optimizing* controlled variables, for example the steady-state cost gradient. Note that, in [19], the active constraints were included as the "obvious" self-optimizing variables, and the theory of self-optimizing control was developed for the less obvious unconstrained CVs [21].

Mathematically, feedback optimizing control aims to find a simple feedback solution to a steady-state real-time

☆The authors gratefully acknowledge the financial support from SUBPRO, which is financed by the Research Council of Norway, major industry partners and NTNU.

*Corresponding author

Email address: `dinesh.krishnamoorthy@ntnu.no`, `skoge@ntnu.no` (Dinesh Krishnamoorthy, Sigurd Skogestad)

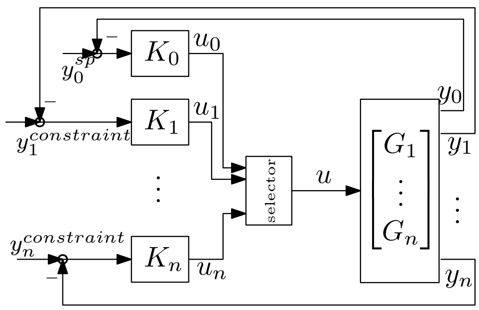


Figure 1: Schematic representation of active constraint switching using selectors.

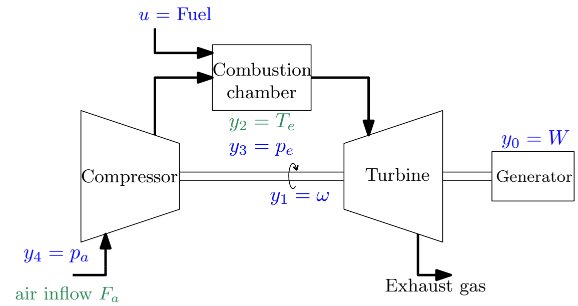


Figure 2: Schematic representation of a gas turbine engine

optimization problem of the form

$$\min_{\mathbf{u}} J(\mathbf{x}, \mathbf{u}, \mathbf{d}) \tag{1a}$$

s.t.

$$\mathbf{g}(\mathbf{x}, \mathbf{u}, \mathbf{d}) \leqslant 0 \tag{1b}$$

where $\mathbf{u} \in \mathbb{R}^{n_u}$ are the set of manipulated variables (MV), $\mathbf{x} \in \mathbb{R}^{n_x}$ denotes the internal variables, $\mathbf{d} \in \mathbb{R}^{n_d}$ denotes the set of disturbances, $J : \mathbb{R}^{n_u} \times \mathbb{R}^{n_x} \times \mathbb{R}^{n_d} \to \mathbb{R}$ is the cost function, and $\mathbf{g} : \mathbb{R}^{n_u} \times \mathbb{R}^{n_x} \times \mathbb{R}^{n_d} \to \mathbb{R}^{n_g}$ are the set of constraints. A constraint $g_i(\mathbf{x}, \mathbf{u}, \mathbf{d}) \leqslant 0$ is said to be active if $g_i(\mathbf{x}, \mathbf{u}, \mathbf{d}) = 0$. Let the set of $n_a \leqslant n_g$ active constraints be denoted by $\mathbf{g}_{\mathbb{A}} \subseteq \mathbf{g}$ (i.e. $\mathbf{g}_{\mathbb{A}}(\mathbf{x}, \mathbf{u}, \mathbf{d}) = 0$).

Note that here, the manipulated variables $\mathbf{u}$ denote the available degrees of freedom for the given problem. It may either be the actual physical manipulated variable, or the setpoint to a lower level regulatory controller.

To transform the optimization problem into a feedback control problem, we need to identify controlled variables (CVs y_i) associated with the constraints and the with the goal of minimizing the cost J. The latter is a bit complicated, as the best "self-optimizing" variables to control for minimizing the cost J depends on what the active constraints are. Thus, for a steady-state optimization problem of the form (1), the first step towards designing a feedback optimizing control structure involves identifying the relevant active constraint combinations. Once the relevant active constraint combinations are identified, one must make a decision on which unconstrained variables to control. That is, in each active constraint region, we must control [15],

1. The n_a active constraints at their limits (i.e. CV:= $\mathbf{g}_{\mathbb{A}} \to 0$),
2. For the remaining $n_u - n_a$ unconstrained degrees of freedom, control a self-optimizing controlled variable (unconstrained CV), e.g. the steady-state cost gradient.

It is then evident, that the active constraint regions play an important role in feedback optimizing control, since this determines "what to control". However, most of the works considering feedback optimizing control (studied in the context of extremum seeking control [22], NCO-tracking [23], self-optimizing control [19], hill-climbing control [24], neighboring extremal control [25] etc. to name a few) focus on the unconstrained optimum, and switching between active constraint sets has received relatively little attention. One of the main reasons for this is probably that active constraint control is often seen as simple/obvious.

When the set of active constraints changes, then this requires a change in the controlled variables and reconfiguration of the control loops. The active constraints may be on the controlled variables (CV), or on the manipulated variables (MV). Therefore, optimal operation may involve either CV-CV switching, MV-MV switching, or CV-MV switching.

CV-CV switching occurs when there are several CV constraints for one MV, but only one CV constraint is active and can be controlled at any given time. To achieve this using selectors, independent controllers may be designed for each controlled variable, and a max/min-selector is used to select either the maximum or the minimum value of all the inputs computed by the different controllers as shown in Fig. 1 for the case where all the constraints are associated with a single input u. This is the case studied in this paper.

The main contribution of this paper is the systematic design procedure for active constraint switching using selectors for the case when all the constraints are associated with a single input. Several illustrative examples are provided to demonstrate the proposed systematic design procedure.

2. Motivating example 1: Gas Turbine Control

Consider a gas turbine engine in Fig. 2 (e.g. for power generation or for aircraft propulsion) with two MVs (degrees of freedom), namely the air inflow rate F_a and the fuel injection rate u Suppose the objective is to control the engine power $y_0 = W$ to a desired setpoint, subject to a max-constraint on the rotational speed ω, a max-constraint on the engine temperature T_e, a min-constraint on the engine pressure p_e, a min-constraint on the air inlet pressure p_a, and a max-constraint on the fuel injection rate u (MV),

$$\begin{aligned} \text{Desired: } & y_0 = W = W_{sp} \\ \text{CV constraints: } & y_1 = \omega \leqslant \omega_{max} \\ & y_2 = T_e \leqslant T_{e,max} \\ & y_3 = p_e \geqslant p_{e,min} \\ & y_4 = p_a \geqslant p_{a,min} \\ \text{MV constraint: } & u \leqslant u_{max} \end{aligned}$$

Assuming that the air inflow rate F_a is used to control the engine temperature T_e to its maximum value $T_{e,max}$, the fuel injection rate must be used to achieve the other objectives. In the reduced system, we have one CV ($y_0 = W$) with a desired setpoint which may be given up if necessary, three CVs y_i, $i = \{1, 3, 4\}$ with inequality constraints, and one MV u with an upper limit.

Process insight tells us that increasing the fuel u will increase the speed $y_1 = \omega$, increase the engine pressure $y_3 = p_e$, and decrease the air inlet pressure $y_4 = p_a$. Based on this, if one uses a min-selector to switch between the CVs, the constraints on u, y_1, and y_4 would remain feasible, but the minimum constraint on y_3 may be violated. Similarly, with a max-selector, y_3 would remain feasible, but constraints on u, y_1, y_2 and y_4 may be violated. It is also not evident if the performance would be the same if one were to use a max-min selector or a min-max selector in series, if so under what conditions.

This simple example clearly shows that although one can often design selector blocks using process insight in such an ad-hoc fashion, it is clearly desirable to have a more systematic procedure in order to design verifiable control structures. Specifically, it may not be evident to an engineer when a set of constraints are conflicting, whether to use a min or a max-selector, or a combination of both, or if such a switching scheme is feasible at all. Also, the lack of a systematic design procedure means that all the feasible alternatives may not be explored.

3. Systematic design of active constraint switching using selectors

Consider a process with a single input (u) and many potential controlled variables (CVs), such that we have

- At most one CV (y_0) with a desired setpoint, that may be given up if necessary.
- Any number of CV y_i with inequality constraints $y_{i,lim}$, that may be optimally active.
- MV inequality constraints $u_{min} \leqslant u \leqslant u_{max}$

Note that the objective is to achieve feasible and optimal steady-state operation using simple feedback controllers and selectors to switch between the active constraints. Therefore, the following results are based on steady-state.

Notation:

- Let G_i denote the steady state gain from the input u to the output y_i.
- Let $\mathbb{Y}$ be the set of all inequality constraints $y_{i,lim}$ (where $y_{i,lim}$ could be both $y_{i,max}$ or $y_{i,min}$) that needs to be satisfied at steady-state for any given operating condition.

Assumption 1. *The gain G_i from u to y_i does not change sign (that is, $\frac{\mathrm{d}y_i}{\mathrm{d}u}$ has the same sign for any* du *and any operating point).*

The set of constraints can be divided into two subsets $\mathbb{Y} = \mathbb{Y}^+ \cup \mathbb{Y}^-$ according to the following criteria.

1. $\mathbb{Y}^+$ is the set of constraints where reducing the input u is better in terms of satisfying the constraints. This means that
   - For CVs where the gain G_i from u to y_i is positive ($G_i > 0$), the set $\mathbb{Y}^+$ includes the corresponding max-constraints ($y_{i,max}$).
   - For CVs where the gain G_i from u to y_i is negative ($G_i < 0$), the set $\mathbb{Y}^+$ includes the corresponding min-constraints ($y_{i,min}$).
   - The set $\mathbb{Y}^+$ also includes a possible max-constraint on u (u_{max}).
2. $\mathbb{Y}^-$ is the set of constraints where increasing the input u is better in terms of satisfying the constraints. This means that
   - For CVs where the gain G_i from u to y_i is positive ($G_i > 0$), the set $\mathbb{Y}^-$ includes the corresponding min-constraints ($y_{i,min}$).
   - For CVs where the gain G_i from u to y_i is negative ($G_i < 0$), the set $\mathbb{Y}^-$ includes the corresponding max-constraints ($y_{i,max}$).
   - The set $\mathbb{Y}^-$ also includes a possible min-constraint on u (u_{min}).

Define

$$u_i = u(y_i = y_{i,lim}) \tag{2}$$

as the value of the input u which at steady-state satisfies the constraint $y_{i,lim}$, i.e., u_i gives $y_i = y_{i,lim}$ at steady state.

Theorem 1 (Feasibility)**.** *Let*

$$\overline{u} = \min_{i \in \mathbb{Y}^+} u_i \tag{3}$$

denote the largest allowed input u that satisfies all the constraint in the set $\mathbb{Y}^+$, and similarly, let

$$\underline{u} = \max_{i \in \mathbb{Y}^-} u_i \tag{4}$$

denote the smallest allowed input u that satisfies all the constraint in the set $\mathbb{Y}^-$. Then to have feasible operation using u as the manipulated variable, we must at any given steady-state operating point have $\underline{u} \leqslant \overline{u}$.

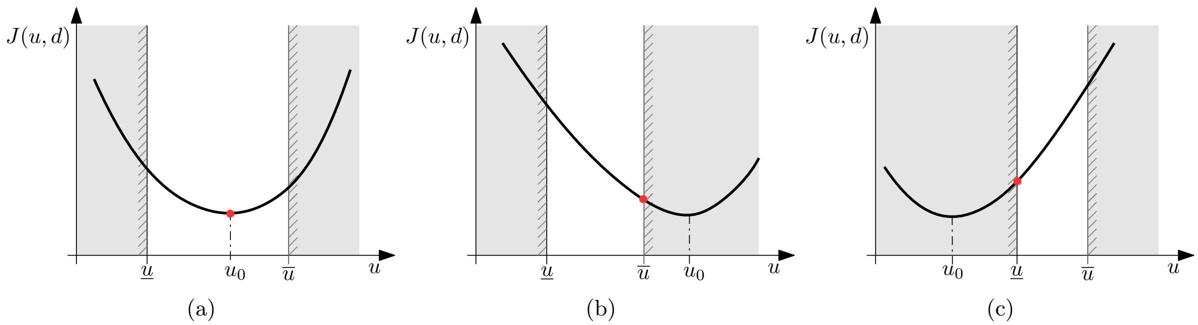


Figure 3: Schematic representation showing that the optimal input (marked with a red dot) can be achieved using a mid-selector. (a) Unconstrainted Optimum $\underline{u} < u_0 < \overline{u}$. (b) Constrained optimum $\underline{u} < \overline{u} < u_0$. (c) Constrained optimum $u_0 < \underline{u} < \overline{u}$.

Proof. To remain feasible, we need $u \leqslant \overline{u}$ to satisfy the constraints in $\mathbb{Y}^+$ and $u \geqslant \underline{u}$ to satisfy the constraints in $\mathbb{Y}^-$. Satisfying both these conditions is possible if and only if $\underline{u} \leqslant \overline{u}$. □

Theorem 1 tells when satisfying all constraints is feasible using a single input u, that is, when there is an allowable operating window for $u \in [\underline{u}, \overline{u}]$. If we have infeasibility (conflicting constraints) according to Theorem 1, then we have two possibilities:

1. let another input (MV) v take over the control of one of the constraints which is no longer controlled using u. This will involve MV-MV switching from input u to v (see discussions in Section 6.5).
2. If we dont have another MV v to take control of the conflicting constraints, then one would have to prioritize the constraints and give up the less important constraint (this will be formally stated later in Theorem 3).

Theorem 1 only tells us when the constraints are feasible. To get a unique value for u we need to add another objective. Typically, we have a controlled variable (y_0) with a desired setpoint, and if possible we want to use the corresponding value of input (u_0) which gives $y = y_0$ at steady-state. More generally, the desired input may result from an optimization problem,

$$u^* = \arg\min_u \{J(u,d) \mid \text{s.t. CV and MV constraints}\} \quad (5)$$

with u^* being the unique minimizer of (5). Then u_0 is the value of u^* for the corresponding unconstrained problem;

$$u_0 = \arg\min_u J(u,d) \quad (6)$$

We may also have cases where the objective is to maximize or minimize some variable. This will correspond to having $u_0 = \infty$ or $u_0 = -\infty$ (for the corresponding unconstrained problem).

Theorem 2 (Optimalilty)**.** *Assume the operational objective is to minimize J, and let u_0 denote the value of the input u that minimizes J for the unconstrained case. Assuming feasible operation, so that $\underline{u} < \overline{u}$ (Theorem 1), the optimal input u is given by*

$$u^* = \text{mid}(\overline{u}, u_0, \underline{u}) \quad (7)$$

Proof. If $\underline{u} < u_0 < \overline{u}$ then the unconstrained optimum is within the feasible region, and optimal operation occurs when $u^* = u_0$. This is schematically represented in Fig. 3a.

If $\underline{u} < \overline{u} < u_0$, then u_0 is infeasible. Hence $u^* = \underline{u}$ (see Fig. 3b). Similarly, if $u_0 < \underline{u} < \overline{u}$, then u_0 is infeasible. Hence $u^* = \overline{u}$ (see Fig. 3c). □

For implementation, we assume that we have several SISO controllers that compute the desired input u_i for each CV $y_i \in \mathbb{Y}$. If these controllers have integral action, then we must design a suitable anti-windup scheme [26], which is discussed later in Section 6.4.

There are several ways the result in Theorem 2 can be implemented using selectors. The most obvious is with three selector blocks; a max- and min-selector for $\underline{u}$ and $\overline{u}$, respectively, and a mid-selector for u as shown with the max/min-mid structure in Fig. 4a.

$$u = \text{mid}(\overline{u}, u_0, \underline{u}) \quad (8)$$

However implementations with only two selectors are also generally possible. This is further discussed in the following remark.

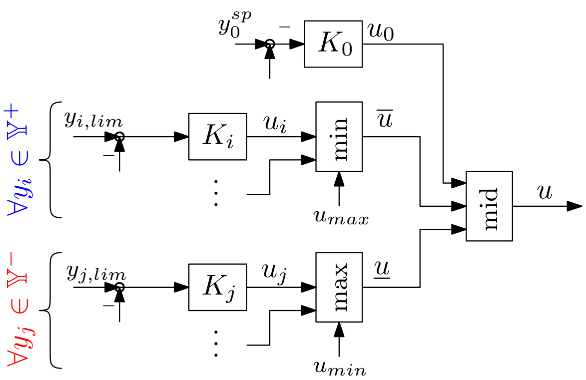
Remark 1 (Two selector blocks)**.** *If we have feasibility, the mid-selector* (8) *is equivalent to using a combined minimum and maximum selector block in series which may be in any order.*
Min-Max structure (Fig. 4b)

$$u = \max_{j \in \mathbb{Y}^-}(\{u_j\}, \underbrace{\min_{i \in \mathbb{Y}^+}(u_0, \{u_i\})}_{=\overline{u}'}) \quad (9)$$

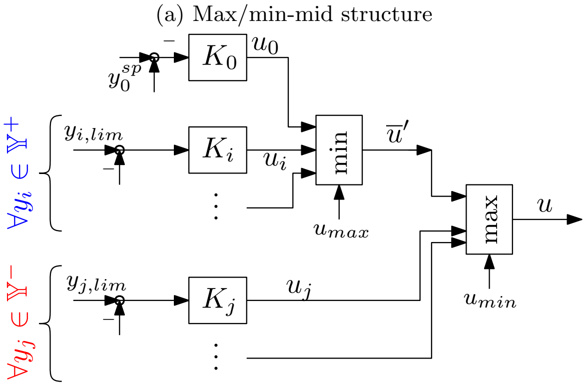
Max-Min structure (Fig. 4c)

$$u = \min_{i \in \mathbb{Y}^+}(\{u_i\}, \underbrace{\max_{j \in \mathbb{Y}^-}(u_0, \{u_j\})}_{=\underline{u}'}) \quad (10)$$

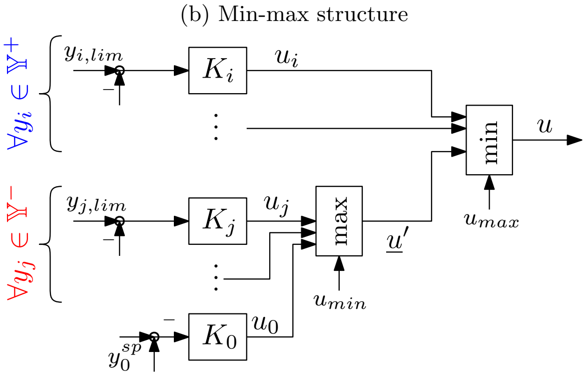
Note that here, the last selector block will always be satisfied. In the min-max structure, the constraints $\mathbb{Y}^+$ are masked and $\mathbb{Y}^-$ will always be satisfied. Similarly, in the max-min structure, the constraints $\mathbb{Y}^-$ are masked and $\mathbb{Y}^+$ will always be satisfied [7]. Table 1 shows a numerical example to illustrate that the three structures studied in Fig. 4 are equivalent when the set of constraints is feasible (cases 1, 2 and 3). It also shows the difference in the different structures in terms of responding to infeasibility (cases 4, 5 and 6 when $\overline{u} < \underline{u}$), which is formalized in the theorem below.



(a) Max/min-mid structure



(b) Min-max structure



(c) Max-min structure

Figure 4: CV-CV and CV-MV switching using selectors. The three alternative structure (a), (b) and (c) are equivalent if we have feasibility, that is, Theorem 1 holds.

Theorem 3 (Selector design for conflicting constraints)**.** *If the constraints are conflicting, that is, $\overline{u} < \underline{u}$ and Theorem 1 does not hold, then*

- *the min-max structure in Fig. 4b always gives up $\overline{u}$ corresponding to the constraints in $\mathbb{Y}^+$*
- *the max-min structure in Fig. 4c always gives up $\underline{u}$ corresponding to the constraints in $\mathbb{Y}^-$*
- *the mid-selector structure in Fig. 4a gives up $\overline{u}$, $\underline{u}$, or both of them, depending on the value of u_0.*

Proof. The result follows quite trivially by considering what happens if we don't have feasibility, that is, when $\overline{u} < \underline{u}$. Since any downstream selector masks all previous selections, a min-max selector always chooses $u = \underline{u}$, and a max-min selector always chooses $u = \overline{u}$. This is also illustrated for a numerical example in Table 1, and by replacing the entries in Table 1 with symbols, we would get a general proof. □

Often, the different constraints have different levels of priority, and a common approach to handle infeasibility is to "give-up" the less important constraints. As seen above, the order of the selector blocks plays an important role in deciding which constraints are given up. Consequently, Theorem 3 can be used to decide which structure to use based on the constraint priority level.

In short, Theorem 3 tells us that for the max-min and min-max structures, the constraint entering the last selector block will be prioritized in case of conflicting constraints. That is, the input u will in cases of conflict "give up" controlling a masked constraint y with "low priority". However, this does not necessarily mean that we need to give up controlling y, because it may be possible to let another input (MV) v take over the task of controlling the masked constraint y. This will involve MV-MV switching from input u to v as discussed in Section 6.5.

One advantage of using the mid-selector in Fig. 4a is that we have explicit calculation of both $\overline{u}$ (the largest allowed input) and $\underline{u}$ (the smallest allowed input). This is useful, since it can be used to verify feasibility by checking whether $\overline{u} > \underline{u}$. If this is not satisfied, then we may use the value of u_0 to tell which constraint to give up, or even to give up both constraints in a desired manner by selecting an appropriate value for u_0 (see Table 1). The latter,

Table 1: Numerical example illustrating that the three structures in Fig. 4 are equivalent when the set of constraints are feasible (Cases 1-3), but when $\overline{u} < \underline{u}$ (infeasibility), they differ (cases 4-6).

cases	$\underline{u}$	$\overline{u}$	u_0	mid (Fig. 4a)	min-max (Fig. 4b)	max-min (Fig. 4c)
1	1	10	5	5	5	5
2	1	10	12	10	10	10
3	1	10	0	1	1	1
4	10	1	5	5	10	1
5	10	1	12	10	10	1
6	10	1	0	1	10	1

that is, giving up both of the conflicting constraints, is not possible with the max-min or min-max structures.

There are also special cases where only a single selector block is sufficient.

Remark 2 (Single selector block)**.** *If $\mathbb{Y}^- = \emptyset$ (i.e. we only have constraints where reducing the input u is better in terms of satisfying the constraints), then the solution is always feasible and the optimal input u is given using a minimum selector*

$$u^* = \min(u_0, \overline{u}) \tag{11}$$

This is a special case of Fig. 4b without the max-selector.

Similarly, if $\mathbb{Y}^+ = \emptyset$ (i.e. we only have constraints where increasing the input u is better in terms of satisfying the constraints), then the solution is always feasible and the optimal input u is given using a max-selector

$$u^* = \max(u_0, \underline{u}) \tag{12}$$

This is a special case of Fig. 4c without the min-selector.

Finally, if there is one constraint in the set $\mathbb{Y}^+$ and one constraint in the set $\mathbb{Y}^-$, we can use a single mid-selector. This is a special case of Fig. 4a without the min- and max-selectors.

The systematic design procedure of the selectors for the case with a single MV (u) can be summarized by the following steps:

Step 1. Group the list of constraints into two sets, namely the set $\mathbb{Y}^+$ and the set $\mathbb{Y}^-$.

Step 2. Design individual SISO controllers to compute the input for each CV constraint (u_i) and for the CV setpoint controller (u_0).

Step 3. Use a minimum selector block to choose the largest allowed input $\overline{u}$ that satisfies all the constraints in the set $\mathbb{Y}^+$, and use a maximum selector block to choose the smallest allowed input $\underline{u}$ that satisfies all the constraints in the set $\mathbb{Y}^-$. The problem is feasible, that is, the set of constraints are not conflicting if $\overline{u} > \underline{u}$ (Theorem 1).

Step 4. When the problem is feasible, the optimal input is given by $u = mid(\underline{u}, u_0, \overline{u})$ where u_0 is the control input computed by the controller that controls the CV with a setpoint that can be given up (Theorem 2).

Step 5. To handle infeasibility, use a max-min structure (Fig. 4c) if the constraints in $\mathbb{Y}^-$ can be given up, or use a min-max structure (Fig. 4b) if the constraints in $\mathbb{Y}^+$ can be given up (Theorem 3).

4. Active constraint control using back-off

Due to imperfect control or measurement noise, sometimes it may be desirable to add a safety margin, known as *back-off*, where the setpoint is offset by a constant value from its limit. Back-off may also be used to simplify the control structure design since the same controller may be used in the different active constraint regions. However, there is an economic loss introduced by back-off, and quantifying this loss can help the designer in deciding whether tight control is required for some CV constraints.

Consider the steady-state optimization problem (1), which is now rewritten as

$$\min_{\mathbf{w}} J(\mathbf{w}) \tag{13a}$$

$$\text{s.t.} \quad \mathbf{g}_{\mathbb{A}}(\mathbf{w}) + \boldsymbol{\varepsilon} = 0 \tag{13b}$$

where $\mathbf{w} := [\mathbf{x}^\mathsf{T}, \mathbf{u}^\mathsf{T}]^\mathsf{T}$ denotes the combined input and state variables, and $\boldsymbol{\varepsilon} \geqslant 0$ is the back-off (safety margin) for the set of active constraints $\mathbf{g}_{\mathbb{A}}$. Note that we only consider the active constraints and hence $\boldsymbol{\varepsilon} \in \mathbb{R}^{n_a}$. We have also eliminated $\mathbf{d}$ for the sake of simplicity, since we are interested in quantifying the loss w.r.t the back-off $\boldsymbol{\varepsilon}$ for a given disturbance. This can be seen as a parametric optimization problem, where we are interested in analyzing the effect of the back-off parameter $\boldsymbol{\varepsilon}$ on the cost. The following theorem says that the loss is given by the value of the corresponding Lagrange multipliers, λ.

Theorem 4 (Loss of back-off)**.** *Assume that the optimization problem* (13) *has an unique primal and dual solution $\mathbf{w}^*(\boldsymbol{\varepsilon})$ and $\boldsymbol{\lambda}^*(\boldsymbol{\varepsilon})$ respectively. Then the steady-state loss due to a small non-zero back-off $\boldsymbol{\varepsilon}$ is given by*

$$Loss = J^*(0) - J^*(\boldsymbol{\varepsilon}) = -\boldsymbol{\lambda}^*(0)^\mathsf{T}\boldsymbol{\varepsilon} \tag{14}$$

Proof. See Appendix A □

The main implications of Theorem 4 for control structure design are:

1. Determine which CVs need to be tightly controlled: If the Lagrange multiplier λ_i for a given constraint is large, then this constraint must be controlled tightly in order to minimize the losses.

2. Simplify control structure design: If the Lagrange multiplier λ_i for a given constraint is sufficiently small such that the loss is negligible/acceptable, then one can allow a large back-off, thus simplifying the control structure design [27].

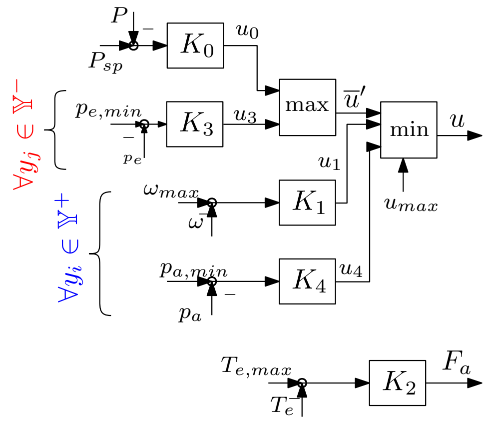


Figure 5: Example 1: Proposed max-min control structure design for gas turbine control.

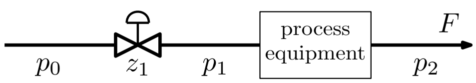


Figure 6: Example 2: Flow through a pipe with one MV ($u = z_1$)

5. Illustrative Examples

5.1. Revisiting example 1 (Gas turbine)

We now apply the systematic approach presented in the previous section to check for consistency and design a control structure for the motivating example introduced in Section 2.

Since the engine temperature $y_2 = T_e$ is controlled to its limit of $T_{e,max}$ using the air inflow rate F_a, we need to design control structures for the rest of the CVs using the fuel u as the MV. For this MV u, the constraints can be grouped into

$$\mathbb{Y}^+ = \{\omega_{max}, p_{a,min}, u_{max}\}$$
$$\mathbb{Y}^- = \{p_{e,min}\}$$

Since we have both set $\mathbb{Y}^+$ and $\mathbb{Y}^-$, we do not have the special case where we can use a single selector block. The problem is feasible only if $\underline{u} > \overline{u}$ (Theorem 1) where

$$\underline{u} = \min(u_1, u_4, u_{max})$$
$$\overline{u} = u_3$$

In the case of infeasibility, we will give up the constraint on $\mathbb{Y}^- = \{p_{e,min}\}$. From Theorem 3, the resulting control structure is using a maximum followed by minimum (max-min) selector block as shown in Fig. 5.

This example demonstrates that the use of the proposed systematic procedure to design selector blocks is straightforward and leads to simple design.

5.2. Example 2: Flow through a pipe

Consider a simple example of flow through a pipe with a valve placed upstream of some processing equipment as

Table 2: Example 2: Controller gains.

	FC	PC for $y_1 = p_{1,max}$	PC for $y_2 = p_{1,min}$
K_P	0.2314	1.1091×10^{-5}	1.1091×10^{-5}
K_I	0.0231	1.1091×10^{-6}	1.1091×10^{-6}
K_{aw}	0.1	0.1	0.1

shown in Fig. 6. There is no accumulation in the processing equipment (i.e. inflow = outflow = F), but there is a pressure drop across the equipment from p_1 to p_2. This could be for example, a filter, a long pipe section, or some kind of a flow restriction or an orifice, the details of which are not important, since the models nor information about the equipment are not used in the control structure design. The valve z_1 is a control valve which is assumed to be the only available degree of freedom. The objective is to maximize the flow rate F by manipulating the valve position $u = z_1$, subject to constraints on the flow rate F, downstream choke pressure p_1 and valve opening z_1. The boundary pressures p_0 and p_2 are disturbances. The optimization problem is:

$$\begin{aligned} &\max_{z_1} F \\ &\text{s.t.} \\ &\quad F \leqslant F_{max} \\ &\quad p_1 \leqslant p_{1,max} \\ &\quad p_1 \geqslant p_{1,min} \\ &\quad z_1 \leqslant z_{1,max} \end{aligned} \tag{15}$$

where F_{max} =10 kg/s, $z_{1,max} = 1$, $p_{1,max} = 2.5$ bar, and $p_{1,min} = 1.5$ bar. Note that there are both max and min-constraints on p_1. Depending on the disturbances, one of these constraints may be optimally active at any given time, and the CV that needs to be controlled using z_1 needs to be switched to remain optimal and feasible.

The constraints can be grouped into

$$\mathbb{Y}^+ = \{F_{max}, p_{1,max}, z_{1,max}\}$$
$$\mathbb{Y}^- = \{p_{1,min}\}$$

Since we have both sets $\mathbb{Y}^+$ and $\mathbb{Y}^-$, we need at least two selector blocks, and we consider the max-min and min-max structures in Fig. 7. Note that since the objective is to maximize production, we have $u_0 = \infty$ (for the imaginary unconstrained case). This value will not affect the operation, except that it will lead to maximizing u.

We use three SISO controllers to control the three CV constraints, namely,

1. Pressure controller that uses z_1 to control $y_1 = p_1$ to $p_{1,max}$. The output from this controller is u_1.
2. Pressure controller that uses z_1 to control $y_2 = p_1$ to $p_{1,min}$. The output from this controller is u_2.
3. Flow controller that uses z_1 to control $y_3 = F$ to F_{max}. The output from this controller is u_3.

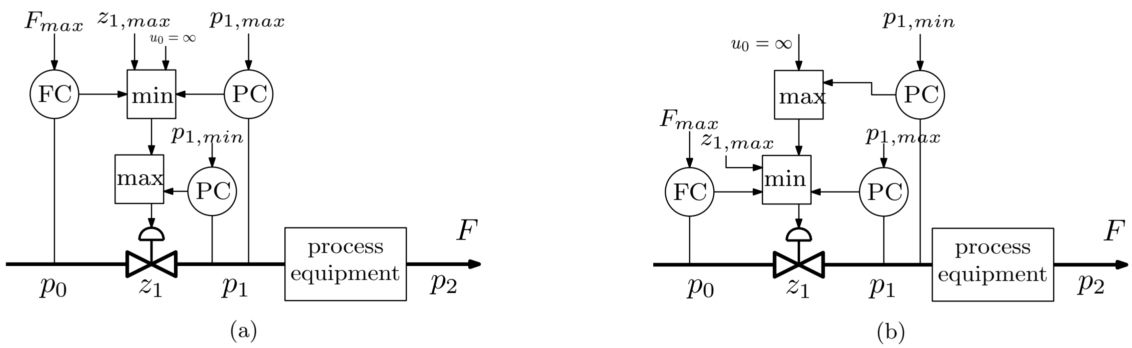


Figure 7: Example 2: Two alternative control structures for constraint switching using selectors. (a) Min-max structure, see (9) (b) Max-min structure, see (10).

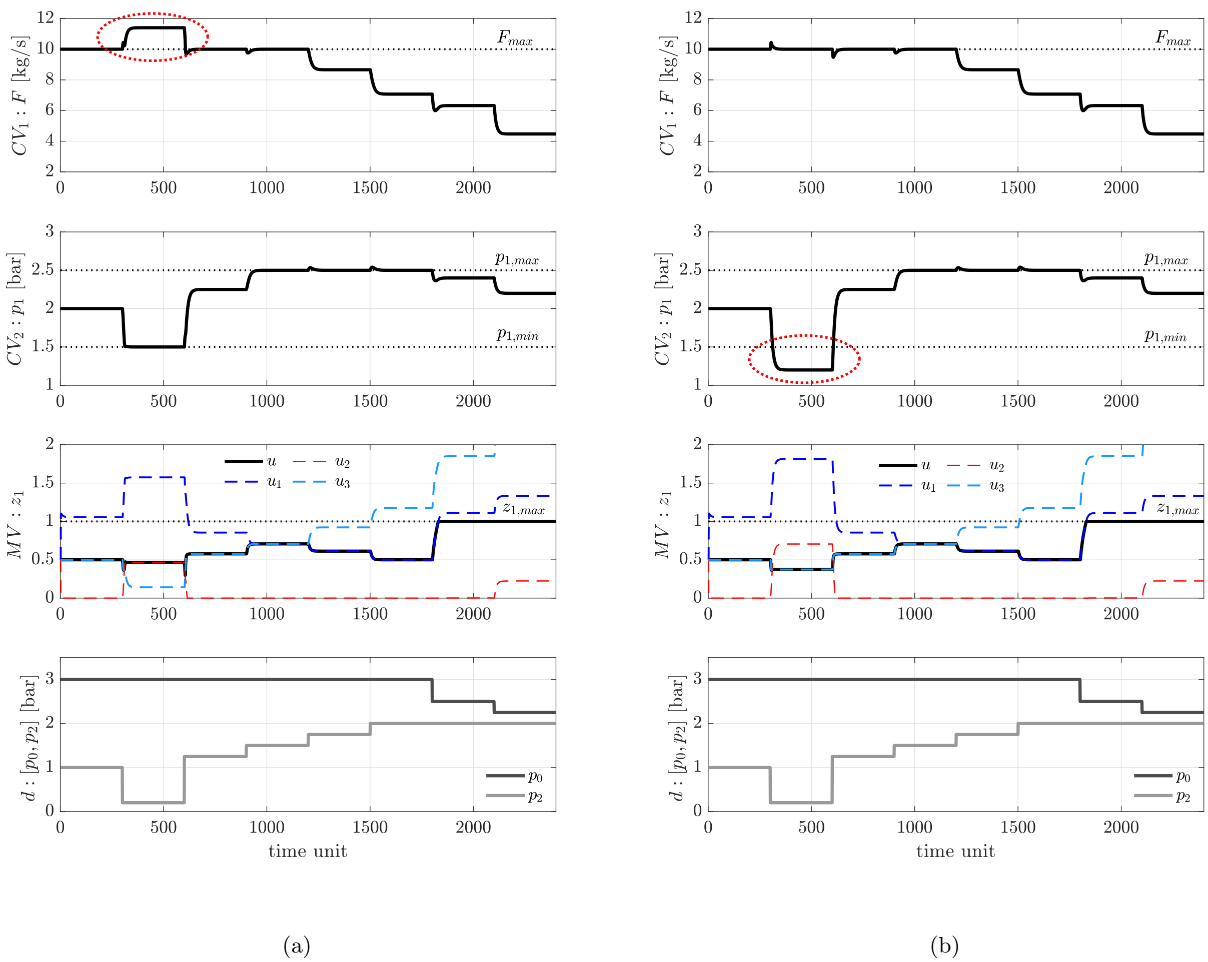


Figure 8: Example 2: Simulation results (a) using the min-max control structure in Fig. 7a (b) using the max-min control structure in Fig. 7b . The control structures behave identical except between time $t = 300$ s and $t = 600$ s when operation is infeasible, and the structures violates different constraints (indicated by red dotted ellipses).

In addition, we have that the manipulated variable is constrained. Therefore, the input to one of the blocks is

4. $u_{max} = z_{1,max}$

At any given time the control action $u = z_1$ computed by one of these four controllers is implemented on the plant. We have that for this example,

$$\overline{u} = \min(u_1, u_3, u_{max})$$
$$\underline{u} = u_2$$

The controllers have integral action, so an anti-windup scheme is required to avoid integral windup when the controller is not selected. We use the back-calculation scheme, which is discussed in Section 6.4. The controller tuning parameters are shown in Table 2.

For disturbances in the boundary pressures p_0 and p_2, simulation results with the two structures in Fig. 7 are shown in Fig. 8a and Fig. 8b, respectively [1]. We see that the responses are good and the simulation results are identical, except in the time period between $t = 300$ s and $t = 600$ s, where operation with all constraints being satisfied is infeasible, since $\underline{u} > \overline{u}$ which violates Theorem 1.

For the min-max structure in Fig. 7a, we violate the maximum flow limit F_{max} which belongs to $\mathbb{Y}^+$ (marked by a red dotted ellipse in Fig. 8a).

For the max-min structure in Fig. 7b, we instead violate the minimum pressure limit $p_{1,min}$ which belongs to $\mathbb{Y}^-$ (marked by a red dotted ellipse in Fig. 8b).

In order to say which structure is the best, this would depend on whether it is acceptable to violate the constraint on F_{max} or $p_{1,min}$ (cf. Theorem 3). If its not possible to violate any of these constraints, then the safety system will need to be activated. Of course, it will be not activated immediately, because some back-off to the hard constraint has most likely been used.

5.3. Example 3: Distillation column

We consider a standard two-product distillation column, which is based on the "Column A" model used in [11]. The distillation column has 41 stages, including the reboiler and the total condenser, with the feed entering at stage 21. The column splits a binary mixture with relative volatility of α= 1.5 into a top product D and a bottom product B as shown in Fig. 10. The model assumes constant relative volatility, constant molar flows, no vapour holdup, linearized liquid dynamics and equilibrium on all stages[2].

For a given feed rate of F, the distillation column has five dynamic degrees of freedom, namely, the reflux L, boilup V, overhead vapour V_D, distillate D and the bottom flow B. However, stable operation of the column requires control of the two levels M_B and M_D and the column pressure, which does not have any affect on the steady-state economics. The distillate D, bottom flow B and overhead vapour V_D are thus used in the regulatory layer to tightly control the levels M_D, M_B, and the column pressure respectively as shown in Fig. 10 (in green). This leaves us with two steady-state degrees of freedom, namely, the reflux L and boilup V that can be used to optimize the process.

The objective is to minimize the operating costs and maximize revenue from the products. In addition, there are purity constraints on the top and bottom products, and constraints on the boilup V. The steady-state opti-

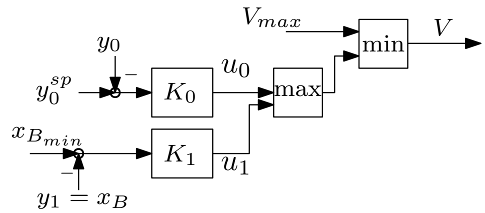


Figure 9: Example 3: Feasible control structure to switch between the three active constraint regions R-I, R-II and R-III.

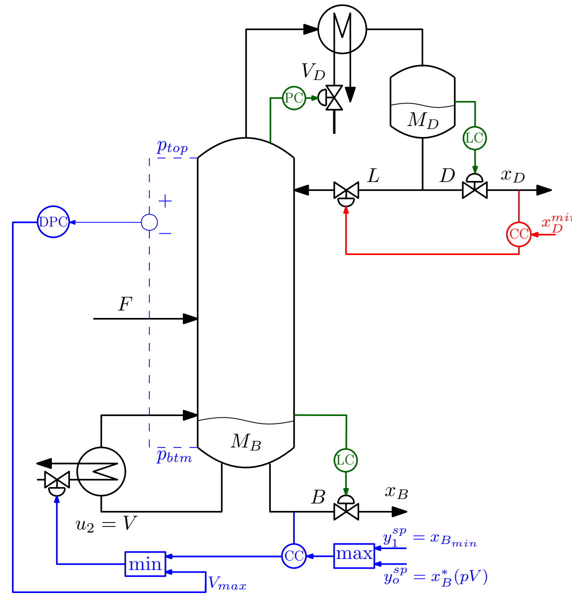


Figure 10: Example 3: Proposed max-min control structure design for optimal operation of a two product distillation column over regions R-I, R-II, and R-III. Control loops that does not have an impact on the steady-state economics are shown in green. The the most valuable product that will always be active is shown in red. The control structure with the max and min-selectors to switch between R-I,R-II and R-III is shown in blue.

[1] For the plant simulator, the model for the flow F is given by, $F = c_{v1} z_1 \sqrt{(p_0 - p_1)/\rho}$ where $c_{v1} = 2 \times 10^{-3}$ m^2, $\rho = 1000$ kg/m^3 and the processing equipment is modeled as a flow through a restriction with $c_{v2} = 10^{-3}$ m^2.

[2] Detailed model description and the MATLAB codes can be found in http://folk.ntnu.no/skoge/distillation/

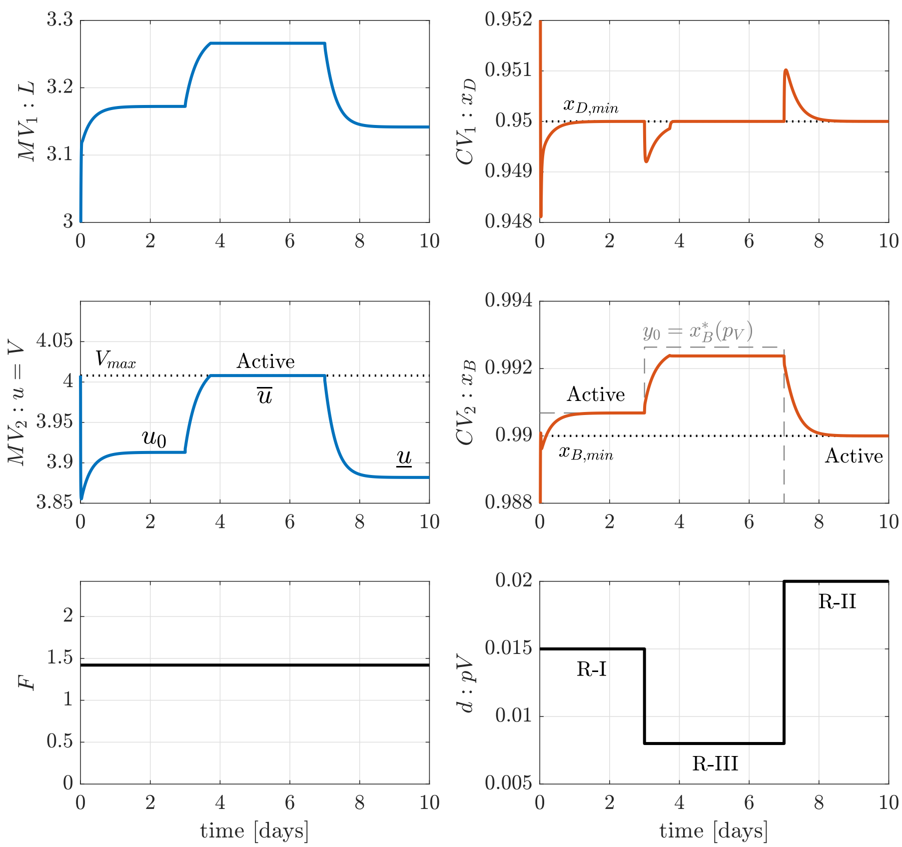


Figure 11: Example 3: Closed loop simulation results for the distillation column with varying energy prices ($d = p_V$). The control scheme in Fig. 10 achieves optimal steady-state operation in all the three constraint regions. The initial response is the start-up period because the column is not operated optimally at $t = 0$.

mization problem is formulated as,

$$\begin{aligned} \min_{L,V} \; & J = p_F F + p_V V - p_D D - p_B B \\ \text{s.t.} \; & \\ & x_D \geqslant x_{D,min} \text{ (always active)} \\ & x_B \geqslant x_{B,min} \\ & V \leqslant V_{max} \end{aligned} \tag{16}$$

where p_F, p_V, p_D, and p_B are the prices for the feed, energy, top product, and bottom product respectively. We assume that the energy price is a disturbance and varies between $p_V \in [0.007, 0.02]\$/mol$, whereas the other prices are constant at $p_F = 1\$/mol$, $p_B = 1\$/mol$ and $p_D = 2\$/mol$.

The most valuable product constraint $x_{D,min}$ will always be active at the optimum, because this avoids product giveaway[11]. Thus we always have $x_D = x_{D,min}$. Therefore the relevant active constraint combinations are

- only x_D active (R-I)
- x_D and $y_c = x_B$ active (R-II)
- x_D and $u = V$ active (R-III)

Due to the pair-close rule, the reflux L is used to control x_D to its limit of $x_{D,min} = 0.95$ as shown in Fig. 10 (in red).

This leaves one degree of freedom, namely the boilup $u = V$. For this MV, we may need to control the concentration x_B at its limit of $x_{B_{min}} = 0.99$ (R-II), or control a self-optimizing variable y_0 to a desired setpoint (R-I). In this example, we consider $y_0 = x_B$ controlled to an economically optimal setpoint given as a function of the energy price $x_{B,sp}(p_V)$ which is determined offline.

We use two SISO controllers to control the two CVs, namely,

1. Concentration controller that uses V to control $y_1 = x_B$ to $x_{B,min}$. The output from this controller is $u_1(x_B = x_{B,min})$. (Active in R-II)
2. Concentration controller that uses V to control $y_0 = x_B$ to $x_{B,sp}$ (unconstrained case). The output from this controller is $u_0(x_B = x_{B,sp})$. (Active in R-I)

In addition, we have that the manipulated variable is constrained. Therefore, we also have

3. $u_{max} = V_{max}$. (Active in R-III)

Note that the limit V_{max} could be to avoid flooding in the column, and may, for example, be computed by a pressure drop (DP) controllers as shown in Fig. 10.

In this case,

$$\mathbb{Y}^+ = \{V_{max}\}$$
$$\mathbb{Y}^- = \{x_{B,min}\}$$

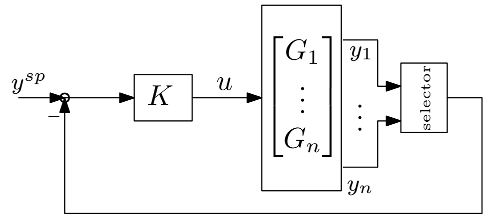


Figure 12: Auctioneering as a special case of CV-CV switching, where the CVs $y_1 \ldots y_n$ all have the same units and same setpoint y^{sp}.

Optimal operation can be achieved using a mid-selector, or a combination of maximum and minimum selectors. Since we cannot give-up on the constraint $V_{max} \in \mathbb{Y}^+$, we choose to use a max-min selector as shown in Fig. 10 (Theorem 3). Furthermore, since the max-selector is used to switch between x_B being controlled to $x_{B,min}$ and x_B being controlled to $x_{B,sp}(p_V)$, we can instead move the max-selector to the setpoint of the composition controller (CC), as shown in Fig. 10.

The simulation results for varying energy price p_V using the proposed control structure is show in Fig. 11. It can be clearly seen that the proposed control structure is able to handle the active constraint switching as the disturbance changes. The concentration controller for x_B was implemented with anti-windup using input resetting to avoid integral windup. The detailed model and control structure design procedure for this example can be found in the supplementary information.

5.4. Example 4: Williams-Otto reactor

The use of selector to switch between different active constraint regions is also tested on a benchmark Williams-Otto reactor example [28], which can be found in the attached supplementary information.

6. Discussion

Some switching logic blocks commonly used in practice can be seen as a special case of the CV-CV and CV-MV switching structures presented above. The developed framework is also applicable to such special cases. In this section, we first point out some of these special cases, and also discuss dynamic implementation aspects.

6.1. Mid-selector for zone control

Mid-selector block is often used in zone control (also known as range control), where the same output has both an upper and lower limit, y_{max} and y_{min}, respectively, and we have a desired value for the input u_0 which may be varying. In this case, $\mathbb{Y}^+ = \{y_{max}\}$ and $\mathbb{Y}^- = \{y_{min}\}$, and we have two different controllers to control y to y_{max}, and y to y_{min}, respectively. Since the limits are on the same output y, Theorem 1 is always satisfied. For example, y may be the temperature in a room, which we want to

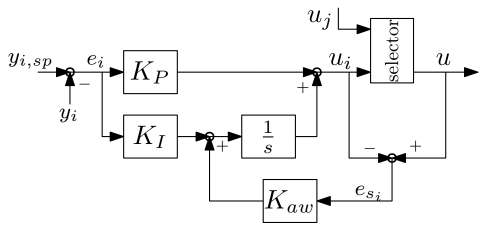


Figure 13: PI controller with anti-windup.

maintain between T_{min} and T_{max}, with a desired value for the input u_0. It is also possible to let u_0 be the output of a third controller that controls y to a variable setpoint, which may be the optimal value for y when the constraints are not active.

6.2. Auctioneering

The proposed framework can also be used in *autioneering* control [2], which is a special case of CV-CV switching where we have similar CVs, i.e. the CVs have the same units and same setpoint. For example, we want to control the maximum temperature along a reactor or control the maximum opening for two valves. Then the selector may be on the output instead, i.e. $y = \max(y_i)$ or $y = \min(y_i)$. In this case, we only need one controller, and a selector block chooses the lowest or highest measurement as feedback, as shown in Fig. 12. Another special case is when one needs to switch between a setpoint control and constraint control on the same variable. In this case, the selector may be used on the setpoint such that we only need one controller. This is also shown for the distillation column example in Fig. 10.

6.3. Input saturation

Typically in process control, the MV represents the actual physical manipulated variable. In this case, one does not even need the selector block for u_{max} or u_{min}, since the MV will physically saturate, e.g. max opening of a valve. However, when we use cascade control, the MV may be the setpoint to another controller. In this case, we need to include the constraint (u_{max} or u_{min}) explicitly as shown in Fig. 4. This is sometimes referred to as clipping the controller output.

6.4. Dynamic implementation and anti wind-up

The theorems presented in Section 3 are based on steady-state analysis. Recall that u_i was defined as the value of the input u which at steady-state satisfies the constraint $y_i = y_{i,lim}$. However, the values for u_i computed by the controllers not selected are not equal to the correct values for u_i even at steady-state, but this is not a problem in practice.

When using selectors, only one of the control actions computed by a plurality of controllers is implemented on the plant at any given time. For the controllers that are not selected, the feedback loop is "broken" and the integral term may build up (known as *windup*), since the tracking error $e_i = y_{i,sp} - y_i$ is non-zero[3].

This windup can be avoided using the back-calculation scheme where an additional feedback path is generated by using the difference between the output of the controller u_i, and the actual output u implemented on the plant u [29, 30]. This signal, denoted by $e_{s_i} := u - u_i$ is fed back to the integrator with gain K_{aw_i} (as shown in Fig. 13) such that e_{s_i} goes towards zero when the controller is deselected. The PI controller with feedback anti-windup can be expressed as:

$$u_i(t) = K_{P_i} e_i(t) + \int_0^t \left(K_{I_i} e_i(\tau) + K_{aw_i} e_{s_i}(\tau) \right) \mathrm{d}\tau \quad (17)$$

where $e_i = y_{i,sp} - y_i$ is the tracking error, u_i is the control action computed by the i^{th} controller and u is the actual control action implemented on the plant, K_{P_i} and K_{I_i} are the proportional and integral gains respectively, and K_{aw_i} is the anti-windup feedback gain.

The integral action will drive the term in the integral to zero, so that at steady state we have

$$K_{I_i} e_i + K_{aw_i} e_{s_i} = 0 \quad (18)$$

In other words, at steady-state we have

$$u_i - u = \frac{K_{I_i}}{K_{aw_i}} e_i$$

If the controller is selected, then at steady-state $e_i = 0$ and the feedback controller will generate the correct steady-state value for u_i. However, the steady-state value of u_i computed by the controller when u_i is not selected, depends on the parameter K_{aw_i}. So, it is clearly not the steady-state value that would give $e_i = 0$. Here, K_{aw_i} is a tuning parameter, and a large value of K_{aw_i} means that $u_i(t)$ is close to u. A too large value of K_{aw_i} may activate u_i when its not necessary, for example, due to measurement noise for y_i or a change in y_i or $y_{i,sp}$, since changes in $e_i = y_{i,sp} - y_i$ will affect u_i through the proportional term $K_{P_i} e_i(t)$.

A reasonable value for the anti-windup gain to avoid unnecessary activation for small change in y_i or $y_{i,sp}$ is

$$K_{aw_i} = \frac{K_{I_i}}{K_{P_i}} \quad (19)$$

which means at steady-state we have

$$u_i - u = K_{P_i} e_i$$

In this case the proportional action $K_{P_i} e_i$ will activate u_i only if $e_i(t) = y_{i,sp} - y_i$ crosses zero, i.e. if y_i reaches it's setpoint/constraint value.

To illustrate this, consider the flow example (Example 2 from Section 5.2). For $p_0 = 1.75$ bar and $p_2 = 3$ bar,

[3]Note that for the active constraint controllers $y_{i,sp} = y_{i,lim}$

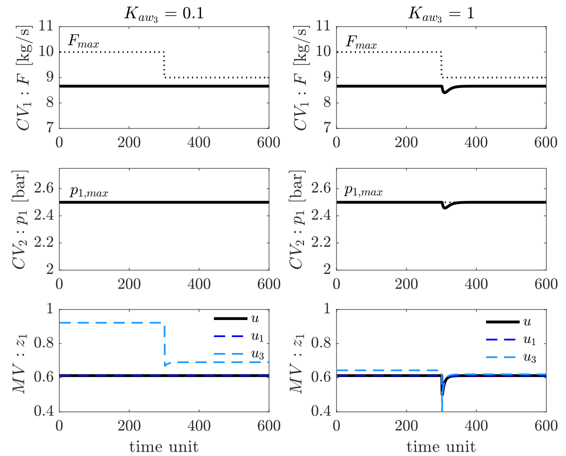


Figure 14: Illustrative example showing that too large value of the anti-windup gain K_{aw} may lead to unnecessary switching of the controllers. The left had side suplot shows the performance with the feedback gain chosen according to (19), and the right had side subplot shows the same with K_{aw} 10 times larger.

the optimal operation occurs when the $p_{1,max}$ constraint is active, with a flow rate of $F = 8.66$ kg/s, as shown in Fig. 14. At time $t = 300$ s, the maximum flow rate F_{max} is reduced from $F_{max} = 10$ kg/s to $F_{max} = 9$ kg/s. This, in principle, should not affect the operation, since $F_{max} = 9$ kg/s will not be active. Indeed, with the chosen feedback gain of $K_{aw} = 0.1$, given by (19), we see from Fig. 14 (left hand side subplots) that the operation remains unchanged as one would expect. However, if we use the larger value of $K_{aw} = 1$ we get unnecessary switching, with the flow controller becoming dynamically active before switching back to the pressure controller as shown in Fig. 14 (right hand side subplots).

6.5. Multiple inputs and conflicting constraints

This paper has considered the important case where all the constraints can be associated with a single input u. It is then possible to divide the constraints into the two sets $\mathbb{Y}^+$ and $\mathbb{Y}^-$, which can be associated with min- and max-selectors, respectively.

In the case of multiple inputs, the proposed systematic design of selectors can be applied if each MV is paired with a set of constraints *a-priori*. For example, consider a process with n inputs v_i. If each MV v_i is paired with a set of constraints $\mathbb{Y}_i$, then this can be divided into two subsets $\mathbb{Y}_i^+$ and $\mathbb{Y}_i^-$ for all $i = 1, \ldots, n$, which can each be associated with min- and max-selectors, respectively in a decentralized fashion. This was also seen in the gas turbine example in Fig. 5, where the air inlet flow was paired with $T_{e,max}$, and the fuel u was paired with the other remaining CVs.

In the case where two hard constraints from the set $\mathbb{Y}_i$ cannot be satisfied at the same time, that is, we have infeasibility according to Theorem 1, then one needs to find some other input to take over one of the control tasks. Usually, this means that we need to give up some other control objective. For example, we may no longer be able to set the throughput freely, since we have reached a bottleneck for the process. In either case, this involves an MV-MV switching, which is not the scope of this paper. MV-MV switching may be achieved using split range control, input position control, and controllers with different setpoints. This has been studied in detail in [17, 31, 32]. For example, in the distillation column example, when V_{max} is active, control of $x_{B,min}$ is lost. If this is a hard constraint, then we would need to find another MV, such as the feed rate F, to control x_B to $x_{B,min}$. This is an MV-MV switching and this example is shown in detail in [17] using a combination of split range control and controllers with different setpoints.

A similar "override" distillation example with MV-MV switching from V to F using two controllers with different setpoints is given in [33, Fig. 5] for a reactor-separator-recycle process. Note that since F is already used to control another task (reactor level), the MV-MV switching from V to F has to be combined with a CV-CV switching (min-selector) for F. However, since also this task (reactor level) cannot be given up, this has to be combined with yet another override, MV-MV switch from F to FB (the throughput manipulator) which will reduce the feed FB to the reactor system.

In large multivariable systems with a lot of such CV-CV, CV-MV and MV-MV switchings, the control structure can quickly become complex, and perhaps one would then be better off with multivariable controllers such as model predictive control (MPC).

7. Conclusion

In this paper, we have presented a systematic procedure for designing selectors for CV-CV switching. Theorem 1 establishes the condition under which the constraints are feasible, and Theorem 2 shows that optimal operation can be achieved using minimum and maximum selector blocks in series. Theorem 3 tells us how the max-min and min-max selectors behave when used at conditions where satisfaction of the constraint is infeasible, and Theorem 3 tells us which structure to use based on the constraint priority list. The proposed systematic design framework does not require detailed process models, making it easily applicable and usable in industrial applications. The proposed framework was successfully demonstrated using illustrative examples.

References

[1] B. G. Liptak, Instrument Engineers' Handbook, Volume One: Process Measurement and Analysis, CRC press, 2003.

[2] F. G. Shinskey, Process control systems: application, design, and tuning, Vol. 4, McGraw-Hill New York, 1996.

[3] H. L. Wade, Basic and advanced regulatory control: system design and application, ISA, 2004.

[4] D. E. Seborg, D. A. Mellichamp, T. F. Edgar, F. J. Doyle III, Process dynamics and control, John Wiley & Sons, 2010.
[5] C. A. Smith, A. B. Corripio, Principles and practice of automatic process control, 3rd Edition, Wiley New York, 2006.
[6] T. E. Marlin, Process Control, Designing Processes and Control Systems for Dynamic Performance, 2nd Edition, McGraw Hill, New York, 2000.
[7] A. H. Glattfelder, W. Schaufelberger, Control systems with input and output constraints, Springer Science & Business Media, 2012.
[8] A. M. Foss, Criterion to assess stability of a 'lowest wins' control strategy, 1981.
[9] A. Imani, M. Montazeri-Gh, Stability analysis of override logic system containing state feedback regulators and its application to gas turbine engines, European Journal of Control 52 (2020) 97–107.
[10] A. Glattfelder, W. Schaufelberger, Stability analysis of single loop control systems with saturation and antireset-windup circuits, IEEE Transactions on Automatic Control 28 (12) (1983) 1074–1081.
[11] M. G. Jacobsen, S. Skogestad, Active constraint regions for optimal operation of chemical processes, Industrial & Engineering Chemistry Research 50 (19) (2011) 11226–11236.
[12] B. Srinivasan, D. Bonvin, 110th anniversary: A feature-based analysis of static real-time optimization schemes, Industrial & Engineering Chemistry Research 58 (31) (2019) 14227–14238.
[13] B. Chachuat, B. Srinivasan, D. Bonvin, Adaptation strategies for real-time optimization, Computers & Chemical Engineering 33 (10) (2009) 1557–1567.
[14] S. Engell, Feedback control for optimal process operation, Journal of process control 17 (3) (2007) 203–219.
[15] D. Krishnamoorthy, S. Skogestad, Linear gradient combination as self-optimizing variables, Computer Aided Chemical Engineering In-Press.
[16] D. Krishnamoorthy, S. Skogestad, Online process optimization with active constraint set changes using simple control structures, Industrial & Engineering Chemistry Research.
[17] A. Reyes-Lúa, S. Skogestad, Systematic design of active constraint switching using classical advanced control structures, Industrial & Engineering Chemistry Research 59 (2020) 2229–2241.
[18] R. Jagtap, N. Kaistha, W. L. Luyben, External reset feedback for constrained economic process operation, Industrial & Engineering Chemistry Research 52 (28) (2013) 9654–9664.
[19] S. Skogestad, Plantwide control: the search for the self-optimizing control structure, Journal of process control 10 (5) (2000) 487–507.
[20] 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 Journal 26 (2) (1980) 220–232.
[21] J. Jäschke, Y. Cao, V. Kariwala, Self-optimizing control–a survey, Annual Reviews in Control.
[22] K. B. Ariyur, M. Krstic, Real-time optimization by extremum-seeking control, John Wiley & Sons, 2003.
[23] G. François, B. Srinivasan, D. Bonvin, Use of measurements for enforcing the necessary conditions of optimality in the presence of constraints and uncertainty, Journal of Process Control 15 (6) (2005) 701–712.
[24] V. Kumar, N. Kaistha, Hill-climbing for plantwide control to economic optimum, Industrial & Engineering Chemistry Research 53 (42) (2014) 16465–16475.
[25] S. Gros, B. Srinivasan, D. Bonvin, Optimizing control based on output feedback, Computers & Chemical Engineering 33 (1) (2009) 191–198.
[26] K. J. Åström, Advanced control methods: Survey and assessment of possibilities.
[27] M. S. Govatsmark, S. Skogestad, Selection of controlled variables and robust setpoints, Industrial & engineering chemistry research 44 (7) (2005) 2207–2217.
[28] 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.
[29] H. A. Fertik, C. W. Ross, Direct digital control algorithm with anti-windup feature, ISA transactions 6 (4) (1967) 317.
[30] K. J. Åström, R. M. Murray, Feedback systems: an introduction for scientists and engineers, Princeton university press, 2010.
[31] A. Reyes-Lúa, S. Skogestad, Multiple-input single-output control for extending the steady-state operating range—use of controllers with different setpoints, Processes 7 (12) (2019) 941.
[32] A. Reyes-Lúa, C. Zotică, S. Skogestad, Optimal operation with changing active constraint regions using classical advanced control, IFAC-PapersOnLine 51 (18) (2018) 440–445.
[33] V. Kumar, N. Kaistha, Real-Time Optimization of a Reactor–Separator–Recycle Process II: Dynamic Evaluation, Industrial & Engineering Chemistry Research 58 (5) (2019) 1966–1977.

Appendix A. Proof of Theorem 4

The Lagrangian function of (13) is given as

$$\mathcal{L}(\mathbf{w}, \boldsymbol{\varepsilon}, \boldsymbol{\lambda}) = J(\mathbf{w}) + \boldsymbol{\lambda}^{\mathsf{T}}(\mathbf{g}_{\mathbb{A}}(\mathbf{w}) + \boldsymbol{\varepsilon}) \tag{A.1}$$

The necessary conditions of optimality

$$\frac{\partial \mathcal{L}}{\partial \mathbf{w}} = \frac{\partial J}{\partial \mathbf{w}} + \boldsymbol{\lambda}^{\mathsf{T}} \frac{\partial \mathbf{g}_{\mathbb{A}}}{\partial \mathbf{w}} = 0 \tag{A.2a}$$

$$\mathbf{g}_{\mathbb{A}} + \boldsymbol{\varepsilon} = 0 \tag{A.2b}$$

determines the optimal primal and dual variables $\mathbf{w}^*(\boldsymbol{\varepsilon})$ and $\boldsymbol{\lambda}^*(\boldsymbol{\varepsilon})$ respectively, as a function of the back-off parameter $\boldsymbol{\varepsilon}$, assuming there exists a unique solution for each $\boldsymbol{\varepsilon}$.

Since $\mathcal{L}$ and $\mathbf{g}$ depends on $\boldsymbol{\varepsilon}$ through $\mathbf{w}$, $\boldsymbol{\lambda}$ and $\boldsymbol{\varepsilon}$, differentiating (A.2) gives

$$\frac{\partial^2 \mathcal{L}}{\partial \mathbf{w}^2} \frac{\partial \mathbf{w}^{\mathsf{T}}}{\partial \boldsymbol{\varepsilon}} + \frac{\partial^2 \mathcal{L}}{\partial \mathbf{w} \partial \boldsymbol{\varepsilon}} + \frac{\partial \mathbf{g}_{\mathbb{A}}^{\mathsf{T}}}{\partial \mathbf{w}} \frac{\partial \boldsymbol{\lambda}}{\partial \boldsymbol{\varepsilon}} = 0 \tag{A.3a}$$

$$\frac{\partial \mathbf{g}_{\mathbb{A}}}{\partial \mathbf{w}} \frac{\partial \mathbf{w}^{\mathsf{T}}}{\partial \boldsymbol{\varepsilon}} + 1 = 0 \tag{A.3b}$$

Let the optimal value function be denoted as $J^*(\boldsymbol{\varepsilon}) = J(\mathbf{w}^*(\boldsymbol{\varepsilon}))$, and the sensitivity of the optimal value function w.r.t $\boldsymbol{\varepsilon}$ can be expressed as

$$\frac{\partial J^*(\boldsymbol{\varepsilon})}{\partial \boldsymbol{\varepsilon}} = \frac{\partial J}{\partial \mathbf{w}} \frac{\partial \mathbf{w}^{\mathsf{T}}}{\partial \boldsymbol{\varepsilon}} \tag{A.4}$$

From (A.2a) and (A.3b), this can be rewritten as,

$$\frac{\partial J^*}{\partial \boldsymbol{\varepsilon}} = -\boldsymbol{\lambda}^*(\boldsymbol{\varepsilon})^{\mathsf{T}} \frac{\partial \mathbf{g}_{\mathbb{A}}}{\partial \mathbf{w}} \frac{\partial \mathbf{w}^{\mathsf{T}}}{\partial \boldsymbol{\varepsilon}} = \boldsymbol{\lambda}^*(\boldsymbol{\varepsilon})^{\mathsf{T}} \tag{A.5}$$

The loss due to a back-off of $\boldsymbol{\varepsilon} > 0$ is the where $\boldsymbol{\lambda}^*(0)$ is the Lagrange multiplier for the active constraints without back-off. Therefore it can be seen that the loss scales linearly with the back-off.

Supplementary information for the article: Systematic design of active constraint switching using selectors

Dinesh Krishnamoorthy, Sigurd Skogestad

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

---

A. Example 3: Distillation column

A.1. Simulator Model

We consider a two-product distillation column with N_T stages as shown in Fig. 1 . The following assumptions are made about the model:

- Binary mixture
- constant pressure, relative volatility and molar flows
- no vapor holdup
- linear liquid dynamics
- equilibrium on all stages

The total mass balance and the mass balance for the light component on stage i, except in the condenser ($i = N_T$), feed stage ($i = N_f$) and reboiler ($i = 1$) is given by:

$$\frac{dM_i}{dt} = L_{i+1} - L_i + V_{i-1} - V_i \tag{1}$$

$$\frac{dM_i x_i}{dt} = L_{i+1}x_{i+1} + V_{i-1}y_{i-1} - L_i x_i - V_i y_i \tag{2}$$

$$\forall i \in \{2, \ldots, N_T - 1\} \setminus \{N_f\}$$

where L_i and V_i are the liquid and vapor flows from the i^{th} stage (in kmol/min), respectively, and M_i is the liquid holdup in the i^{th} stage (in kmol). x_i and y_i

*Corresponding author

Email address: `dinesh.krishnamoorthy@ntnu.no, skoge@ntnu.no` (Dinesh Krishnamoorthy, Sigurd Skogestad)

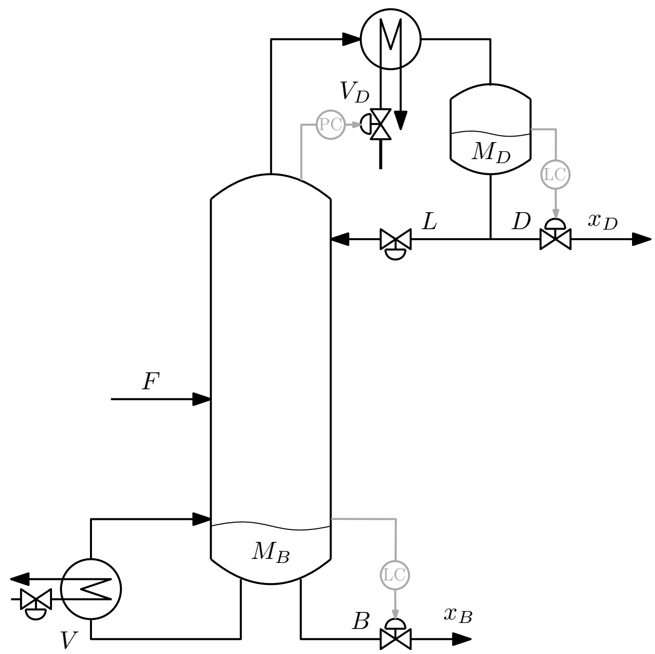


Figure 1: Schematic representation of the distillation column

are the liquid and vapor mole fractions of light component on the i^{th} stage, respectively.

The mass balance on the feed stage ($i = N_f$) is given by,

$$\frac{dM_{N_f}}{dt} = L_{N_f+1} - L_{N_f} + V_{N_f-1} - V_{N_f} + F \tag{3}$$

$$\frac{dM_{N_f}x_{N_f}}{dt} = L_{N_f+1}x_{N_f+1} + V_{N_f-1}y_{N_f-1} - L_{N_f}x_{N_f} - V_{N_f}y_{N_f} + Fz_F \tag{4}$$

The mass balance on the reboiler ($i = 1$) is given by,

$$\frac{dM_B}{dt} = L_2 - V - B \tag{5}$$

$$\frac{dM_Bx_1}{dt} = L_2x_2 - Vy_1 - Bx_1 \tag{6}$$

where B is the bottom flow rate and V is the boilup as shown in Fig. 1.

The mass balance on the condenser ($i = N_T$) is given by,

$$\frac{dM_D}{dt} = V_{N_T-1} - L - D \tag{7}$$

$$\frac{dM_Dx_{N_T}}{dt} = V_{N_T-1}y_{N_T-1} - Lx_{N_T} - Dx_{N_T} \tag{8}$$

where D is the distillate flow rate and L is the reflux as shown in Fig. 1.

From this, we get the expression for the rate of change of liquid mole fraction

$$\frac{dx_i}{dt} = \frac{1}{M_i}\left(\frac{dM_i x_i}{dt} - x_i \frac{dM_i}{dt}\right) \quad \forall i \in \{1, \ldots, N_T\} \tag{9}$$

The model therefore has $2N_T$ differential states denoted by $[\{x_i\}_{i=1}^{N_T}, \{M_i\}_{i=1}^{N_T}]^\mathsf{T}$.

The liquid flows depend on the liquid holdup on the stage above and the vapor flow as follows

$$L_i = L0_i + \frac{1}{\tau_i}(M_i - M0_i) + (V - V0)_{i-1}\lambda \tag{10}$$

where $L0_i$ (in kmol/min) and $M0_i$ (in kmol) are the nominal values for the liquid flow and holdup on stage i. The effect of vapor flow on the liquid flow is captured by λ.

The vapor composition can then be computed from the vapor-liquid equilibrium

$$y_i = \frac{\alpha x_i}{1 + (\alpha - 1)x_i} \tag{11}$$

where α is the constant relative volatility.

A.2. Controller design

We assume that the overhead vapour V_D is used to maintain a constant pressure. Stable operation of the column requires the levels M_B and M_D to be controlled. In this model, the column is stabilized using the LV-configuration where we use D to control M_D, and B to control M_B as shown in Fig. 1. We use a P-controllers for each level control loop, with the controller gain $K_P = 10$ for both the loops.

As mentioned in the manuscript, the purity constraint on x_D will always be active, since this is the most valuable product. The x_D composition is controlled using the reflux L using a PI controller that is tuned using the SIMC tuning rules. For a desired closed loop time constant of $\tau_c = 10$, this results in the proportional gain $K_P = 7.8947$ and $K_I = 0.2193$.

The composition control for the bottom product x_B is also achieved using a PI controller that is tuned using the SIMC rules. For a desired closed loop time constant of $\tau_c = 10$, this results in the proportional gain $K_P = 2.2140$ and $K_I = 0.123$.

The `MATLAB` scripts for the distillation column example is given below or can be found in `https://github.com/dinesh-krishnamoorthy/Feedback-based-RTO/tree/master/Selectors/ColA`.

B. Example 4: Williams-Otto reactor

B.1. Simulator model

The benchmark Williams-Otto reactor converts the raw materials A and B into useful products P and E along with a byproduct G via a series of reactions,

$$
\begin{aligned}
&A + B \rightarrow C && k_1 = 1.6599 \times 10^6 e^{-6666.7/T_r} \\
&B + C \rightarrow P + E && k_2 = 7.2177 \times 10^8 e^{-8333.3/T_r} \\
&C + P \rightarrow G && k_3 = 2.6745 \times 10^{12} e^{-11111/T_r}
\end{aligned}
$$

The reactor is modeled as,

$$
\begin{aligned}
\frac{\mathrm{d}x_A}{\mathrm{d}t} &= \frac{1}{W}(F_A - F x_A) - k_1 x_A x_B \\
\frac{\mathrm{d}x_B}{\mathrm{d}t} &= \frac{1}{W}(F_B - F x_B) - k_1 x_A x_B - k_2 x_B x_C \\
\frac{\mathrm{d}x_C}{\mathrm{d}t} &= \frac{-F x_C}{W} + 2k_1 x_A x_B - 2k_2 x_B x_C - k_3 x_C x_P \\
\frac{\mathrm{d}x_P}{\mathrm{d}t} &= \frac{-F x_P}{W} + k_2 x_B x_C - 0.5 k_3 x_C x_P \\
\frac{\mathrm{d}x_E}{\mathrm{d}t} &= \frac{-F x_E}{W} + 2k_2 x_B x_C \\
\frac{\mathrm{d}x_G}{\mathrm{d}t} &= \frac{-F x_G}{W} + 1.5 k_3 x_C x_P
\end{aligned}
$$

where the mass holdup $W = 2105$ kg. The reactor is controlled using the reactor temperature $MV_1 := T_r$ and the feed rate $MV_2 := F_B$ with pure B component. Feed rate F_A with pure A component is a disturbance and we assume that it is expected to vary between $1kg/s$ and $2kg/s$. We assume perfect level control such that the outflow $F = F_A + F_B$.

B.2. Controller design

The objective is to maximize the production of useful products P and E. In addition, there are purity constraints on G and A on the product stream, and a minimum outflow rate F_{out}. The steady-state optimization problem is formulated as,

$$
\begin{aligned}
\min_{T_r, F_B} \quad & -1043.38 x_P F - 20.92 x_E F \\
& \qquad + 79.23 F_A + 118.34 F_B \\
\text{s.t.} \quad & \\
& x_G \leqslant x_{G_{max}} \\
& x_A \leqslant x_{A_{max}} \\
& F \geqslant F_{min}
\end{aligned}
\tag{12}
$$

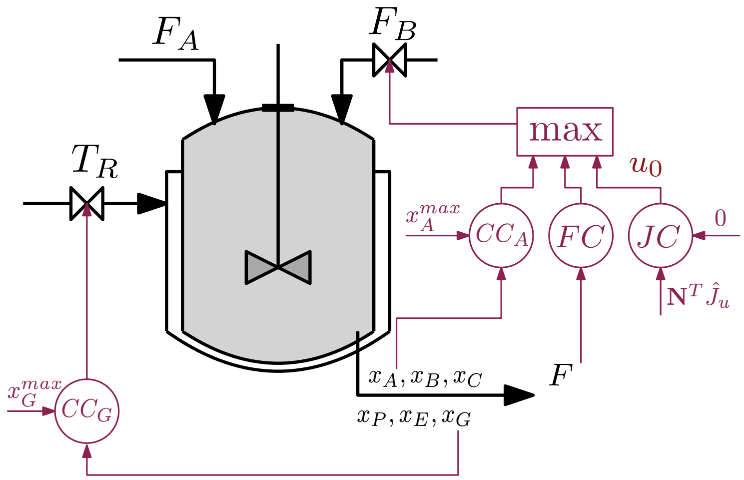


Figure 2: Example 4: Proposed control structure design for optimal operation of a Williams-Otto reactor over regions R-I and R-II.

where $x_{G,max} = 0.08$, $x_{A,max} = 0.12$, and $F_{min} = 4.4\ kg/s$.

Since the purity constraint on x_G is very low, it will always be active for the assumed disturbance range of $F_A \in [1, 2] kg/s$. Therefore the relevant active constraint combinations are

- only x_G active (R-I)
- x_G and x_A active (R-II)
- x_G and F active (R-III)

Since we have two MVs, we first pair the reactor temperature $u_1 = T_r$ to tightly control x_G to its limit of $x_{G,max} = 0.08$ using a PI control. This leaves us with one degree of freedom, namely $u_2 = F_B$, which will be used to control either the self-optimizing CV y_0 to a desired setpoint in region R-I, or control x_A to its limit of $x_{A,max} = 0.12$ in region R-II, or control F to its minimum limit of $F_{min} = 4.4\ kg/s$ in region R-III. In this case,

$$\mathbb{Y}^- = \{x_{A,max}, F_{min}\}$$

and since we have only $\mathbb{Y}^-$, a single maximum selector block can be used to switch between the different active constraint regions (cf. Remark **??**).

Therefore, we use four SISO controllers to control the CVs in each active constraint region, namely,

1. Composition controller denoted by CC_G that uses T_r to control x_G to $x_{G,max}$ (in regions R-I, RII, and R-III).
2. Composition controller denoted by CC_A that uses F_B to control x_A to $x_{A,max}$ (in region R-II).

Table 1: Example 4: Controller gains.

	CC_G	CC_A	JC	FC
K_P	193.4236	-66.2252	0.1251	0
K_I	1.2895	-0.1104	1.5640×10^{-4}	1
K_{aw}	-	0.0017	0.0013	0.005

3. Flow controller denoted by FC that uses F_B to control F to F_{min} (in region R-III).

4. Self-optimizing controller denoted by JC that uses F_B to control the self-optimizing CV y_0 to $y_{0,sp}$ (in region R-I).

In region R-I, we have used a linear gradient combination as the self-optimizing CV $y_0 = \mathbf{N}^{\mathsf{T}}\nabla_u J$ controlled to a constant setpoint of $y_{0,sp} = 0$. In this example, the linear gradient combination is given by $y_0 = 0.9959\nabla_{F_B}J + 0.0906\nabla_{T_r}J$.

The control structure design is shown in Fig. 2. The controller tuning parameters are shown in Table 1.

Fig. 3 shows the simulation results using the proposed control structure design. It can be clearly seen that as the disturbance changes, the max selector block is able to automatically handle the CV-CV switching between R-I, R-II and R-III. The `MATLAB` scripts for the distillation column example is given below or can be found in `https://github.com/dinesh-krishnamoorthy/Feedback-based-RTO/tree/master/Selectors/WilliamsOtto`.

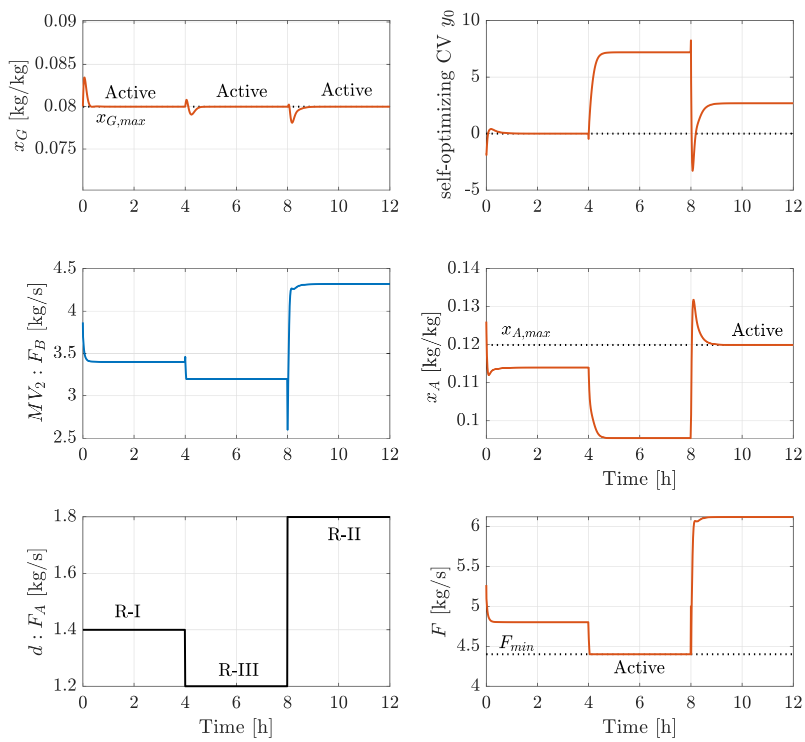


Figure 3: Example 4: Simulation results showing the automatic switching between the active constraint regions as the disturbance changes using selectors designed according to Theorem 1.