Anton Friedl, Jiří J. Klemeš, Stefan Radl, Petar S. Varbanov, Thomas Wallek (Eds.)
Proceedings of the 28th European Symposium on Computer Aided Process Engineering
June 10th to 13th, 2018, Graz, Austria. 

# Changing between Active Constraint Regions for Optimal Operation: Classical Advanced Control versus Model Predictive Control

Adriana Reyes-Lúa[a], Cristina Zotică[a], Tamal Das[a], Dinesh Krishnamoorthy[a] and Sigurd Skogestad[a*]

[a]*Norwegian University of Science and Technology, Department of Chemical Engineering, Sem Sælands vei 4, 7491 Trondheim, Norway*
**sigurd.skogestad@ntnu.no*

## Abstract

Control structures must be properly designed and implemented to maintain optimality. The two options for the supervisory control layer are Advanced Control Structures (ACS) and Model Predictive Control (MPC). To systematically design the supervisory layer to maintain optimal operation, the constraints that can be given up when switching active constraint regions should be prioritized. We analyze a case study in which we control the temperature and the flow in a cooler with two degrees of freedom (DOF) represented by two valves, one for each of the two streams. Either valve can saturate and make a constraint active, forcing other constraints to be given-up, and thus changing the set of active constraints. We show that optimal or *near*-optimal operation can be reached with both ACS and MPC. We do a fair comparison of ACS and MPC as candidates for the supervisory layer, and provide some guidelines to help steer the choice.

**Keywords:** Process control, supervisory control, PID control, MPC, optimal control, active constraints

## 1. Introduction

On a time-scale basis, the overall control problem of a process plant can be decomposed into different layers. The upper layers are explicitly related to slow time scale economic optimization, which sends economic setpoints to the lower and faster control layer. The control layer is divided into supervisory layer and regulatory layer. The latter follows the set-points given by the former and stabilizes the plant. Most process are operated under a set of constraints, which can be operational limitations, quality specifications, or safety and environmental requirements. “Active constraints” are related to variables that should be kept at their limiting value to achieve optimality. These can be either Manipulated Variables (MVs) or Controlled Variables (CVs). The MVs correspond to the dynamic (physical) DOF used by the control system, and a typical MV constraint is the maximum opening of a valve. An example of CV constraint is the maximum pressure in a distillation column. Every process is subject to disturbances, such as changes in feed rate or product specification. It is the task of the supervisory or “advanced” control layer to maintain optimal operation despite disturbances. The supervisory control layer has three main tasks (Skogestad, 2012):

1. Switch between the set of CVs and control strategies when active constraint changes occur due to disturbances.

2. Supervise the regulatory layer, avoiding saturation of the MVs used for regulatory control.
3. Follow economic objectives by using the setpoints to the regulatory layer as MVs .

The supervisory control layer could be designed using classical ACS with PID controllers, or using MPC, which achieves optimal operation and handles constraints and interactions by design. With ACS, we refer to PID-based structures such as split range control (SRC), input resetting (valve positioning), and use of selectors, to name a few.

## 2. Changes in active constraint regions and optimal operation

When a disturbance occurs, the process might start operating in a different active constraint region. If the supervisory layer is well-designed, it is possible to maintain optimal operation by using ACS with PID controllers, or by using MPC.

### *2.1. Optimal control in the presence of active constraint changes*

Regardless of whether we choose ACS or MPC, the first step to systematically design the supervisory control layer is to identify and prioritize all constraints. It is useful to visualize how disturbances may cause new constraints to become active. In some cases, we can generate a plot showing the active constraint regions (optimal operation) as a function of variations in important disturbances by solving a series of optimization problems. This may be very time consuming and, in some cases, difficult due to the lack of an appropiate model. Moreover, it can also be difficult to visualize for more than two variables. Alternatively, we can use process knowledge and engineering insight to minimize the need for numerical calculations (Jacobsen and Skogestad, 2011). This information is useful regardless of the type of controller used in the supervisory layer.

Prioritization of constraints has been implemented in a few industrial MPC applications (Qin and Badgwell, 2003). Reyes-Lúa et al. (2018) propose a guideline to generate a priority list of constraints that can be used also for ACS. Under this scheme, the constraints with the lowest priority should be the first given-up when it is not feasible to fulfill all constraints. This way, controlling a high priority constraint will never be sacrificed in order to fulfill a low priority constraint.

### *2.2. Advanced control structures in the supervisory layer*

ACS requires a choice of pairings, which can become challenging with changing active constraints. When implementing ACS, Reyes-Lúa et al. (2018) propose to start designing the control system for the nominal point, with few active constraints and with most of the priorities satisfied. Then, to minimize the need for reassignment of pairings when there are changes in active constraints, we should pair MVs with CVs according to the *Pairing Rule* (Minasidis et al., 2015): *An important controlled variable (CV) (which cannot be given up) should be paired with a manipulated variables (MV) that is not likely to saturate*.

When a disturbance occurs and the process starts operating in a different active constraint region, two types of constraints might be reached:

- MV constraint: we must give up controlling the corresponding CV. If the pairing rule is followed, this MV is paired with a low priority CV, which can be given-up. However, if it is not possible to follow the *pairing* rule, the high priority CV must be reassigned to an MV which is controlling a low priority CV. This requires the use of ACS such as input resetting (valve position control) or SRC combined with a selector block.
- CV constraint: we should give up controlling a CV with a lower priority. We can do this using a *min/max* selector.

*2.3. Model predictive control in the supervisory layer*

MPC uses an explicit process model to predict the future response of the plant and, by computing a sequence of future MV adjustments, optimizes the plant behavior. The first input of the sequence is applied to the plant, and the entire calculation is repeated at every sampling time (Qin and Badgwell, 2003).

The main challenge when using MPC is that expertise and a good model is required. This is either difficult to have ready at startup, or the modelling effort is too expensive. To achieve a truly optimal operation, the model would need to be perfect, and all the measurements would need to be available and reliable, which is unrealistic from a practical point of view. There are methods to circumvent this, but there is no universal solution and this analysis is out of the scope of this paper.

When an application lacks DOF to meet all control specifications, standard text-book MPC does not handle changes in active constraints effectively. The standard approach is to use weights in the objective function to assign the priorities. Having weights in the objectives function implies a trade-off between the control objectives. An optimal weights selection can assure that a CV is completely given-up, or that the solution will lie at the constraint, as explained in Section 3.4.2. However, there is no systematic way of choosing the weights, as there are no tuning rules for MPC, and this has to be done by trial and error.

An alternative approach consists of implementing a two-stage MPC with a priority list. The first stage has the purpose of finding the solution of a sequence of local steady-state optimization problems (LPs and/or QPs). In this sequence constraints are added in order of priority. The resulting information regarding feasibility is used in the formulation of the dynamic optimization problem for the MPC in the second stage (Qin and Badgwell, 2003).

## 3. Case study

We study a cooler in which the main control objective is to keep the outlet temperature in the hot stream to a desired setpoint ($T_H = T_H^{sp}$) by using cooling water ($F_C$). Additionally, the setpoint for the flow of the hot stream ($F_H^{sp}$) can be changed.

There are two MVs, one corresponding to the cooling water ($F_C$) and another to the hot stream ($F_H$). Desired operation is at maximum throughput, with $F_H^{sp} = F_H^{max}$. The primary input ($F_C$) may saturate for a large disturbance ($T_c^{in}$). This case is an extension of what is presented by Reyes-Lúa et al. (2018).

*3.1. Process model*

We consider a countercurrent cooler, represented by the dynamic lumped model in Eq. (1). The cooler is discretized in space into a series of $n = 10$ cells, as depicted in Fig. 1. Incompressible fluids and constant heat capacities are assumed. The boundary conditions are: $T_{H_0} = T_{H_{in}}$ for cell $i = 1$ (inlet), and $T_{C_{11}} = T_{C_{in}}$ for cells $i = 10$ (outlet). The energy balance for cell $i = 1 \ldots n$ is:

$$\frac{dT_{C_i}}{dt} = \frac{F_C}{\rho_C V_{C_i}}(T_{C_{i+1}} - T_{C_i}) + \frac{UA_i(T_{H_i} - T_{C_i})}{\rho_C V_{C_i} c_{p_C}} \tag{1a}$$

$$\frac{dT_{H_i}}{dt} = \frac{F_H}{\rho_H V_{H_i}}(T_{H_{i-1}} - T_{H_i}) + \frac{UA_i(T_{H_i} - T_{C_i})}{\rho_H V_{H_i} c_{p_H}} \tag{1b}$$

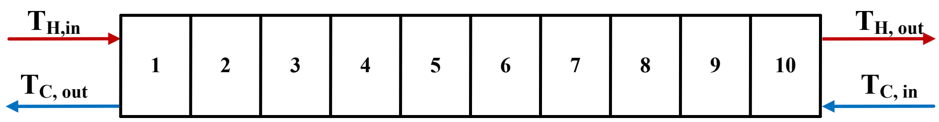



Figure 1: Lumped model for the studied cooler.

*3.2. List of priorities*

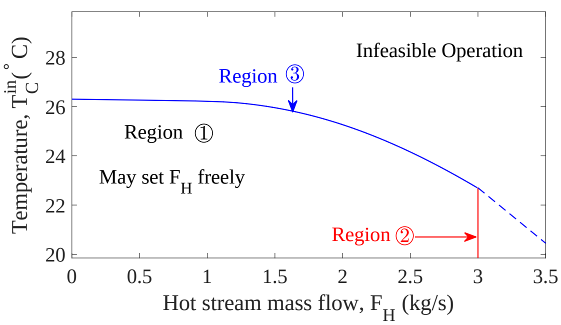
Table 1 shows the priority list of constraints for the cooler we are analyzing.

Table 1: Constraints for the studied cooler.

| Priority level | Description | Constraints |
|---|---|---|
| 1 | MV inequality constraints which define the feasibility region | $F_H \leq F_H^{max}$ <br> $F_C \leq F_C^{max}$ |
| 2 | MV or CV equality constraints, which is the control objective | $T_H = T_H^{sp}$ |
| 3 | Desired throughput | $F_H = F_H^{sp}$ |
| 4 | CV inequality constraints or self optimizing variables | *none* |

*3.3. Active constraint regions for cooler*

As we want to keep $T_H = T_H^{sp}$, this constraint is always active and only one DOF remains. With one DOF and three potential constraints we have three possible active constraint regions, which are shown as a function of the throughput ($F_H$) and the disturbance ($T_c^{in}$) in Fig. 2.




Active constraint in each region:

- Region 1: $F_H = F_H^{sp}$
- Region 2: $F_H = F_H^{max}$
- Region 3: $F_C = F_C^{max}$

Figure 2: Active constraint regions for the cooler

*3.4. Design of the supervisory layer for the cooler*

We consider nominal operation in Region 2 ($F_H^{sp} = F_H^{max}$). According to the priority list, when $T_c^{in}$ is so high that $F_C = F_C^{max}$, the controller should give up controlling $F_H = F_H^{max}$ and reduce $F_H$ to keep $T_H = T_H^{sp}$, thus switching to Region 1. We design both an ACS and an MPC for this case.

*3.4.1. Classical advanced control structures for optimal operation*

To design the supervisory layer using ACS, we implement SRC with a *min* selector block, as in Fig. 3. The controller is tuned by fitting a first order plus delay model obtained from the open-loop step response of the process, and applying the SIMC rule (Skogestad, 2003) with $\tau_c = 80\ s$ and $K_c = -0.06$. To account for the different gains that $F_C$ (negative) and $F_H$ (positive) have on $T_H$, the MVs were respectively multiplied with a gain of 1 and $-2$.

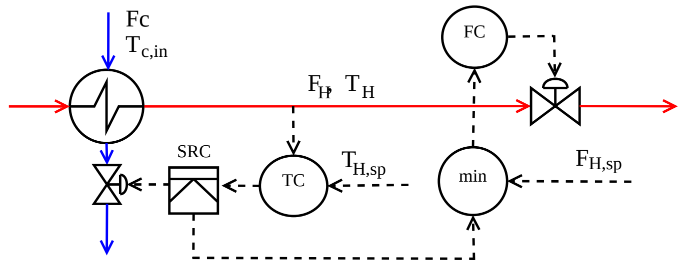



Figure 3: Split range control structure for cooler.

#### *3.4.2. Model predictive control for optimal operation*

The optimal control problem is discretized into a finite dimensional optimization problem divided into $N = 40$ control intervals. We use a third order direct collocation scheme for a polynomial approximation of the system dynamics for each time interval.

The dynamic optimization problem is setup in CasADi (Andersson, 2013), which is an algoritmic differentiation tool. According to Eq. 1, the dynamic model is non-linear. The resulting NLP problem is thus solved using IPOPT (Wächter and Biegler, 2005). The prediction horizon is set to 400 s with a sampling time of $\Delta t = 10$ s. We assume we have full state feedback and the disturbance, $T_c^{in}$, is measured.

In this paper, we chose to implement the standard MPC formulation given by Eq. 2, and to assign different weights for the two control objectives. A high weight is assigned to the high priority CV ($T_H$) and a low weight is assigned to the low priority CV ($F_H$). The values $\omega_1 = 3$ and $\omega_2 = 0.1$ are used. These were found by trial and error. In addition, the MVs are restricted to a rate of change of 10 % of $F_H^{max}$ and $F_C^{max}$ respectively.

$$
\begin{aligned}
&\min \sum_{k=1}^{N} \left( \omega_1 \left\| \left( T_{H_k} - T_H^{sp} \right) \right\|^2 + \omega_2 \left\| \left( F_{H_k}^{max} - F_{H_k} \right) \right\|^2 \right) \\
&\text{s.t.} \\
&\left.\begin{aligned}
&T_{k,i} = f(T_{H_k,i}, T_{H_k,i-1}, T_{C_k,i}, T_{C_k,i+1}, F_{H_k}, F_{C_k}) \\
&0 \leq F_{H_k} \leq F_H^{max} \\
&0 \leq F_{C_k} \leq F_C^{max}
\end{aligned}\right\} \quad \forall k \in \{1, \ldots, N\} \\
&\left.\begin{aligned}
&0 \leq \Delta F_{H_k} \leq 0.1 F_H^{max} \\
&0 \leq \Delta F_{C_k} \leq 0.1 F_C^{max}
\end{aligned}\right\} \quad \forall k \in \{1, \ldots, N-1\}
\end{aligned}
\tag{2}
$$

where $\Delta F_k = F_k - F_{k-1}, \forall k \in \{1, \ldots, N-1\}$. For $k = 1$, $F_{k-1}$ represents the flow at the nominal operation point.

#### *3.4.3. Simulation results*

Fig. 4 shows the simulation results for the case study. $T_H^{sp}$ is $26.3°C$. MPC and SRC structures are tested for the same step disturbances in $T_C^{in}$: $+2°C$ at $t = 10s$, and an additional $+4°C$ at $t = 1000s$. Both MPC and SRC follow the priority list and reach optimal operation at steady state. Once $F_C = F_C^{max}$, the control structure gives-up controlling $F_H = F_H^{max}$, and $F_H$ is used as MV to maintain $T_H = T_H^{sp}$.

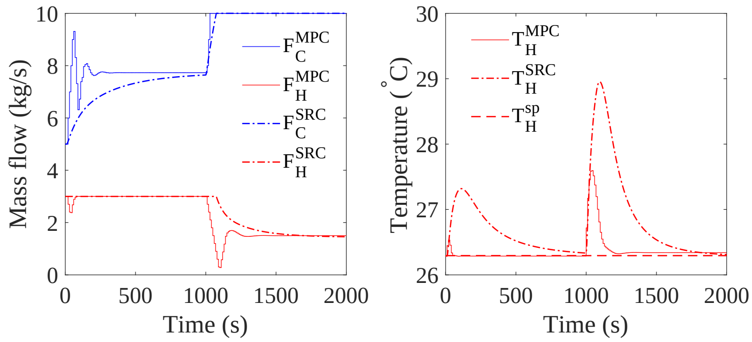



Figure 4: Simulation results for MPC and SRC.

## 4. Discussion and conclusion

The supervisory control layer can be designed using MPC or ACS. MPC uses the manipulated variables to achieve optimal operation by design, but it requires expertise and a model, which may be difficult to obtain. Well designed ACS can also maintain optimal operation, require much less model information, and are usually easier to implement and tune. In our example, SRC efficiently switches the MVs, achieving optimal operation. Compared to ACS, MPC implementation requires more effort as the tuning of weights in the objective function is more challenging because it is done by trial and error. As it is seen in Fig 4, for a short transient time during the first disturbance in the MPC implementation, $F_H \neq F_H^{max}$. This could be improved by increasing $\omega_2$ relative to $\omega_1$. This would however be at the expense of having an offset for $T_H$ from $T_H^{sp}$, as its weight in the objective function would be smaller. Therefore, we should point out that a different MPC implementation or tuning could have better performance, especially on the input usage.

We recommend to use priority lists as a tool for analyzing and designing the supervisory layer. Understanding the process is an important step to decide which controller should be implemented. Both ACS and MPC have advantages and disadvantages, and the designer of the control layer should be aware of these. While in simple cases such as the presented case study, ACS seems better fitted due to achieving optimality with less implementation effort, in multivariable systems with more interactions, MPC should be considered as the most convenient alternative.

## 5. Acknowledgements

This work was partly supported by the Norwegian Research Council under HighEFF (Energy Efficient and Competitive Industry for the Future) and SUBPRO (Subsea Production and Processing).

## References

J. Andersson, 2013. A General Purpose Software Framework for Dynamic Optimization. Phd thesis, KU Leuven.

M. G. Jacobsen, S. Skogestad, 2011. Active Constraint Regions for Optimal Operation of Chemical Processes. Industrial & Engineering Chemistry Research 50 (19), 11226–11236.

V. Minasidis, S. Skogestad, N. Kaistha, 2015. Simple Rules for Economic Plantwide Control. In: K. V. Gernaey, J. K. Huusom, R. Gani (Eds.), PSE 2015 and ESCAPE-25. Elsevier Science Direct, pp. 101–108.

S. Qin, T. A. Badgwell, 2003. A survey of industrial model predictive control technology. Control Engineering Practice 11 (7), 733–764.

A. Reyes-Lúa, C. Zotica, S. Skogestad, 2018. Optimal Operation with Changing Active Constraint Regions using Classical Advanced Control. In: 10th ADCHEM. IFAC, Shenyang, China.

S. Skogestad, 2003. Simple analytic rules for model reduction and PID controller tuning. Journal of Process Control 13, 291–309.

S. Skogestad, 2012. Economic Plantwide Control. In: G. P. Kariwala, Vinay Rangaiah (Ed.), Plantwide Control. Wiley, Ch. 11, pp. 229–251.

A. Wächter, L. T. Biegler, Apr. 2005. On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming. Mathematical Programming 106 (1), 25–57.