Integrating Self-Optimizing Control and Real-Time Optimization using zone control MPC

José Eduardo A. Graciano[1], Johannes Jäschke[2], Galo A. C. Le Roux[1] and Lorenz T. Biegler[3]

[1] Department of Chemical Engineering, Polytechnic School of the University of São Paulo, São Paulo, Brazil,
[2] Department of Chemical Engineering, Norwegian University of Science and Technology (NTNU), Trondheim, Norway,
[3] Chemical Engineering, Carnegie Mellon University, Pittsburgh, PA

Abstract

The combination of real-time optimization (RTO) and model predictive control (MPC) methodologies is widely used in the chemical and petrochemical industry to optimize continuous processes. However, often the setpoint updates computed by the RTO are not frequent enough to capture all disturbances. This leads to suboptimal process operation, because the system is not re-optimized after a disturbance, until it has reached its new (suboptimal) steady state. An efficient way to handle this issue is to include economic considerations in the design of the lower MPC layer. This is the main idea of self-optimizing control (SOC), where controlled variables are selected, such that keeping them constant results in near-optimal operation without requiring an RTO update. Thus, we argue that SOC is complementary to RTO, and we develop a new MPC strategy with zone control and SOC variable targets. In particular, we present an approach towards solving the problem of low-frequency setpoint updates, also when the active set changes. We demonstrate the performance of our approach in steady state and dynamic simulations, and compared it to a classic RTO/MPC combination. The results show that our approach improves the coordination between the RTO and MPC layers. As it gives better performance in between RTO executions, it also leads to a higher overall profit.

Keywords: hierarchical control structure, real-time optimization, self-optimizing control, model predictive control

1. Introduction

The petrochemical industry is a mature business that has two main innovation targets: economic (due to competition) and environmental (due to new and stricter laws)[1]. Process automation is playing a key role to help the petrochemical industry to meet new requirements in energy efficiency and economic performance. The overall control framework in a petrochemical industry typically follows a hierarchical structure characterized either by a functional or a temporal decomposition [2]. Functional decomposition sorts the control objectives in order of decreasing importance (i.e. to maintain the plant in a safe operation mode, to meet product quality and yield demands, and to maximize the plant profit). Temporal decomposition is applied when the control framework is formulated as a dynamic problem, and there are significant changes of fast and slow state variables or disturbances, [2].

This work is focused on functional hierarchical decomposition control, see Fig.1. We consider the optimization and control layers represented by the RTO (Real-time Optimization) and MPC (Model Predictive Control) blocks in this chart. The RTO provides ideal economic target set points for the MPC layer, which is responsible to control the process at this steady state.

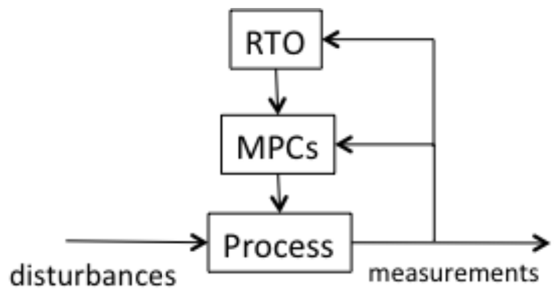


Figure 1. Functional process control hierarchy

A common way to design the RTO layer is to use a first principle steady state model to describe the plant behavior and optimize an economic objective function subject to the model. At each RTO sampling time, key parameters of the steady state model are updated to reduce the plant-model mismatch, using plant measurements [3]. Then, the set point values, obtained by the economic model optimization, are fed to the MPC control layer below. This strategy gained prominence in the late 1980's, when new developments allowed the application of RTO with equation oriented modeling environments, fast computational processing capability and large scale sparse matrix solvers, [4].

It has been pointed out in the literature (e.g. [5]) that this scheme frequently leads to suboptimal operation, because RTO updates only the set points at given sampling times, usually after a new steady state is detected. Consequently, in the presence of disturbances, the plant operates suboptimally until the detection of the next (suboptimal) steady state and the computation of the new optimal operating point. This has a clear disadvantage over other optimization methodologies, such as Economic MPC or Dynamic Real Time Optimization, which rely on dynamic models to update the economic set points promptly without waiting for steady state [6, 7].

On the other hand, the single layer control approach using Dynamic Real Time Optimization (D-RTO) requires quite accurate dynamic process models, and these are not always available. If only linear models are available, the control framework may lead to suboptimal solutions since there is an intrinsic mismatch between plant and model. On the other hand, accurate nonlinear dynamic models lead to large-scale optimization problems, which nowadays are not difficult to solve, but still present some challenges with respect to solution time and synchronization between plant and control. For this reason the two layer control approach with RTO on the top and a fast (MPC) control layer below remains an attractive alternative for many process systems. Moreover, the two-layer approach may have a higher acceptance among plant operators, because if the optimizer fails, there are still the familiar setpoints that can be set by manually by the operators in order to keep the plant at a given operating point.

The approach described in this paper applies directly to two-layer implementations of RTO and is straightforward to implement within commercial software such as Aspen RTO and Invensys ROMeo. The main idea is to include information about the plant economics into the lower layer, such that the MPC control layer is tightly coordinated with the RTO layer. In particular, the MPC layer is designed to reject disturbances in between RTO executions in a near-optimal manner. This may be achieved by using self-optimizing controlled (SOC) variables in the MPC layer [8]. The main idea of the SOC concept is to choose a set of controlled variables that have *optimal* set point values that are insensitive to disturbances (including variables that remain at their active constraints despite the presence of disturbances). In other words, when controlling self-optimizing controlled variables at their constant set points acceptable performance in terms of the economic profit is achieved.

Skogestad and coworkers [8, 9] presents a procedure to find a set of self-optimizing controlled variables using the information provided by a steady state model. This technique was successfully applied to large-scale chemical processes as described in [9, 10]. Another alternative for SOC is to create artificial output variables insensitive to the disturbances, using linear combinations of measured variables [11].

Several methods have been developed in this area that consider the local worst-case loss minimization. Examples include the Exact Local Method [12], which considers a second order approximation around the optimal point to find measurement combinations that are self-optimizing with respect to disturbances changes and implementation error (noise). In addition, the Null Space method in [13] uses the optimum output sensitivity to disturbances to find a matrix of measurement combinations as controlled variables. Locally, this method leads to controlled variables that give zero loss with respect to the analyzed disturbances. The Null Space method was extended in [11] by using extra measurements to reduce the loss caused by measurement noise.

Jäschke and Skogestad [14] developed a method to identify combinations of measured variables using only process data. In this method, plant tests are performed to estimate the measurement gain matrix, while the reduced Hessian matrix is computed from a second order black box model obtained from historical process data. Despite the fact that it does not require a process model, this method demands large amounts of process data, which can make it difficult to implement in practice.

Kariwala and coworkers [15] developed an efficient way to derive an expression for the measurement combination that minimizes the average loss criteria, considering disturbances and implementation error.

Ye et al. [16] combine concepts of necessary condition of optimality (NCO) and SOC to indirectly control the compressed reduced gradient to zero. In their work, the reduced gradients (or controlled variables) are approximated by response surface models over the entire operating space. The method leads to reduced profit loss for a larger disturbance region compared to local methods that are only valid in the vicinity of the nominal operating point.

SOC ideas have also previously been used in an RTO-like framework. In [17], a "necessary conditions of optimality" (NCO) tracking procedure [18] was used in the upper layer, and SOC was applied in the lower layer. Ye et al. [17] developed a new hierarchical control structure, integrating SOC and RTO. They used the NCO as controlled variables and developed a statistical criterion of non-optimality to decide when the controlled variable should be updated..

All the SOC methods described above assume that the active set remains the same when a disturbance occurs. However, one of the main difficulties encountered in practice when designing a self-optimizing control structures is how to handle changing active constraints, because when a constraint becomes active or inactive, the number of degrees of freedom changes. Because of this, it is usually necessary to re-configure the control structure to remain near-optimal. As an example, consider a distillation column, where the boilup is used for controlling an important temperature in the column. Now if the boilup reaches its maximum value, the control structure will have to be reconfigured, such that e.g. the feed rate to the distillation column is lowered in order maintain the important temperature. Alternatively, control of the temperature must be given up to keep the feed rate at its specified value.

Hu et al. [19] propose a way to find a control structure that does not change when the true active set changes. This comes at the expense of generally *not* enforcing control for the active set. This approach results in some loss, and in some cases it may even not be feasible to find controlled variables that can be kept at their constant setpoints over all the disturbance space.

More rigorous approaches first enforce the active constraints in the different regions, and then use the remaining unconstrained degrees of freedom to control self-optimizing variables. Cao [20] presented a method that uses a cascade control approach that can be used when the process is operated with unconstrained degrees of freedom most of the time. Here the inner loop will enforce the constraints, if necessary, while an outer loop modifies the setpoint of the inner loop to control a self-optimizing variable. A saturation block is added to the outer loop to make sure that the setpoint provided does not result in a constraint violation. This approach is somewhat limited, as it requires the number of potentially active constraints to be lower than the number of controlled variables.

Manum and Skogestad [21] propose to handle this problem by using the Null Space method to find a set of controlled variables in each active constraint region. Then in each region a so-called descriptor function is constructed as a function of measurements. The point at which to switch from one region (control structure) to another is detected by monitoring the difference in the signs of the descriptor functions of neighboring regions. Optimal operation is then achieved by using the descriptor functions for identifying active set changes, and by implementing the corresponding control structure in each region. This approach will work well, when the number of regions is small, and the process moves continuously from one operating point to another, without "jumping" over regions.

The challenge faced by this method is the need to identify and store all the different active set regions, the descriptor functions, and the corresponding set of self-optimizing control variables. Although the computations are performed offline, the number of active constraint regions can become very large. In fact, the number of regions can become so large that this approach becomes practically infeasible. Moreover, as the analysis is for optimal steady state optimization, and depends on local information (current and neighboring regions) large dynamic disturbance effects may cause the algorithm to fail to detect active set changes, or suggest wrongly to change the active set. In this case a re-optimization of the overall system must be performed, in order to re-identify the correct active set, and then restart this algorithm. Finally, if the process is very nonlinear, it may be necessary to add further heuristics for region detection, because the method is based on approximating the nonlinear plant by a QP.

In contrast, the approach in this paper is different, as we do *not* pre-compute all controlled variables for all regions *a-priori*. Instead we use an RTO that is executed at given sample times, and that computes the nominal setpoints for the controlled variables in the current RTO interval, together with a set of self-optimizing controlled variables. Compared to the parametric programming approach by [21], our approach has several advantages:

- We do not need to solve a large multiparametric program and store all results together with the descriptor functions.
- We do not need to include heuristics to handle nonlinearities.

- By using MPC in the lower layer, we can handle active set changes to some degree without waiting for the next RTO update, by using a concept that we call “zone control.”
- If an active set change is not detected between RTO executions, it will be corrected by the following RTO update.

If the disturbances do not change the active set, the lower layer MPC will control the self-optimizing controlled variables to their setpoints and enforce the correct active set given by the RTO. This is simply the standard self-optimizing control idea, where optimal operation is achieved by controlling the self-optimizing controlled variables and the active constraints to their given setpoints.

When considering active set changes, there are two fundamental cases that must be handled by every method that attempts to give optimal operation:

1. *Detecting when a new constraint becomes active*: Here a degree of freedom is consumed for controlling the new constraint, and control of one of the unconstrained self-optimizing controlled variables must be given up.
2. *Detecting when a previously active constraint becomes inactive:* Here the constraint that was enforced previously must be released, and a new self-optimizing variable must be controlled instead.

The approach described in this paper is designed to handle Case 1 *between RTO updates*, i.e., detecting and enforcing previously inactive constraints. Case 1 is the most important one in terms of plant economics, because the economic cost for not enforcing the correct constraint has a linear bound in the error [22]. That is, the economic loss is proportional to the error.

In our approach, the lower MPC layer (without the need to execute the RTO) uses a “zone control” concept to detect when a new constraint becomes active, and controls it to its new optimal value.

On the other hand, our approach does *not handle Case 2 in between RTO updates*. That is, if a constraint becomes inactive in between RTO runs, it will be kept at its bound until the next RTO execution. However, handling Case 2 correctly in between RTO updates is not as critical in terms of economic loss, as implementing Case 1. That is, because the cost for controlling a "wrong" active constraint has a quadratic bound in the error [22]. So when a disturbance causes the constraint to become inactive, keeping it at its bound has only a minor effect on the economic loss, until it is relaxed at the next RTO execution.

In summary, the contribution of this paper is to develop steps towar solving the problem that occurs frequently in RTO, where a long time delay between RTO updates can lead to economic loss [5]. The RTO computes new SOC variables and their optimal setpoints at each RTO execution, and these are implemented by the MPC layer below. The zone control concept makes it possible to handle situations where disturbances cause previously inactive constraints to become active. In contrast to [21], where all regions and controlled variables are generated off-line, in this work the required information is generated online by the RTO, and only for the required region. This simplifies the implementation significantly, because:

- It is not necessary to keep track of all the regions and implement a logic to determine active set changes.
- No logic is required to detect region changes and switch the control structures.
- No extra measures have to be taken to change from one control structure to another in a smooth way (bumpless transfer).
- No additional mechanisms are required to recover from wrongly identified active set changes. This will be taken care of at the next RTO execution.

Although Case 2 is not addressed in this paper, we believe that our approach is an important step toward improving suboptimal performance due to low frequency RTO updates. In particular, it handles most disturbances that occur in between RTO executions in a near-optimal manner, especially when the active set does not change, or when a new constraint becomes active.

The paper is organized as follows. Section 2 describes the integration between the RTO and SOC methodologies. Section 3 presents the development of MPC with zone control and SOC targets, which is capable of handling active set changes. The first case study in section 4 deals with the production of ammonia. Section 5 presents a second case study using a

Benzene/Toluene/Xylene (BTX) separation plant, which is analyzed at steady state in section 5.1 and dynamically in Section 5.2. Final discussions and conclusions are given in section 6.

2. RTO framework implementation with SOC

We assume that the problem of operating a process optimally at steady state can be formulated in terms of the following optimization problem:

$$\begin{aligned} \min_{y,u} \quad & \varphi(z,u,d) \\ \textit{subject to} \quad & f(z,u,d)=0 \\ & g(z,u,d)\leq 0 \\ & y=h(z,u,d) \end{aligned} \tag{1}$$

where $\phi: \mathbb{R}^{n_z}\times\mathbb{R}^{n_u}\times\mathbb{R}^{n_d}\rightarrow\mathbb{R}$ is the economic objective function, $z\in\mathbb{R}^{n_z}$ is the vector of model states, $u\in\mathbb{R}^{n_u}$ the vector of model inputs and $d\in\mathbb{R}^{n_d}$ is the vector of disturbances. Further, $f: \mathbb{R}^{n_z}\times\mathbb{R}^{n_u}\times\mathbb{R}^{n_d}\rightarrow\mathbb{R}^{n_f}$ denotes the model equations and $g: \mathbb{R}^{n_z}\times\mathbb{R}^{n_u}\times\mathbb{R}^{n_d}\rightarrow\mathbb{R}^{n_g}$ denotes the inequality constraints. Finally, $y\in\mathbb{R}^{n_y}$ denotes the process measurements and h: $\mathbb{R}^{n_z}\times\mathbb{R}^{n_u}\times\mathbb{R}^{n_d}\rightarrow\mathbb{R}^{n_y}$ the corresponding mapping from process variables and disturbances to the measurements. To implement the optimal solution of (1) in the real plant, the hierarchical structure shown in Fig.2 is often used in industry, and it is also adopted in the present work.

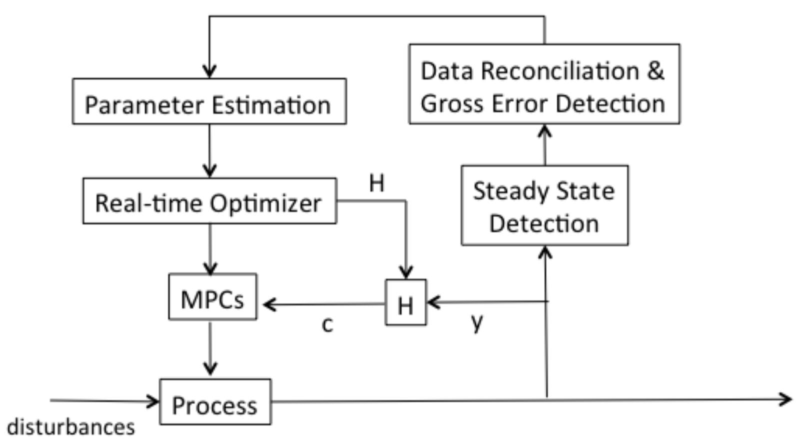


Figure 2: Proposed framework for the implementation of SOC in the RTO

Our RTO cycle starts with the detection of plant steady state conditions. Then the process measurements are screened for consistency and presence of gross errors in the reconciliation module. After that, the reconciled data are used by the parameter estimation module to update the process model. The next stage of our approach comprises the optimization of an economic objective function subject to the updated model, which provides the optimal set points for the controlled variables $c^* \in \mathbb{R}^{n_c}$ as well as a sensitivity analysis of the optimal solution. This sensitivity information is used as in Alstad et al. [13] for finding self-optimizing controlled variables that are linear combinations of measurements, $c = Hy$. In particular, the optimal sensitivity is needed to construct the measurement combination matrix $H \in \mathbb{R}^{n_c \times n_y}$. These controlled variables c, together with the currently active constraints are then passed down to the MPC layer, which computes the plant inputs u.

The idea is that when disturbances change, controlling the self-optimizing controlled variables will lead to near-optimal adjustments of the inputs also in between the RTO runs, and will thus improve the economic performance of the plant, without waiting for the next RTO execution.

The zone-control MPC layer enforces the currently active constraints and tries to adjust the inputs u to control the self-optimizing variables to their setpoints. If a disturbance causes a previously inactive constraint to become active, then the MPC will no longer be able to maintain controlled variables at their setpoints, because one or more of the zone constraints has become active. In this case the MPC will give up on perfectly controlling the self-optimizing controlled variables to their setpoints, and instead enforce the constraints.

The measurement combination matrix H is determined by the optimal sensitivity of economic objective function of NLP (1). In particular, the self-optimizing control variables $c = Hy$ are calculated as linear combinations of measured variables by the Null Space method [13], where H is selected to be in the left null space of the optimal sensitivity matrix $F \in \mathbb{R}^{n_y \times n_d}$, such that $HF = 0$. Here F is defined as:

$$F = \frac{\partial y^{opt}}{\partial d} = \begin{bmatrix} \frac{\partial y_1^{opt}}{\partial d_1} & \cdots & \frac{\partial y_1^{opt}}{\partial d_{n_d}} \\ \vdots & \ddots & \vdots \\ \frac{\partial y_{n_y}^{opt}}{\partial d_1} & \cdots & \frac{\partial y_{n_y}^{opt}}{\partial d_{n_d}} \end{bmatrix} \quad (2)$$

The NLP sensitivity matrix F describes the *optimal* changes of the (unconstrained) outputs y with respect to the disturbances, i.e., $F = \frac{\partial y^{opt}}{\partial d}$ is the sensitivity of the optimal steady-state process output y for a given disturbance (d). Note that the rows in F must be linearly independent in order to result in a full rank matrix for F . Based on (2), the number of artificial self-optimizing variables is equal to $n_c = n_y - n_d$, which should be selected to fill the number of controlled variables.

Remark: The condition for existence of a non-trivial null-space is that $rank(F) < n_y$. This implies in general that there are more measurements y than there are disturbances d .

In this study the NLP sensitivity matrix F is computed by re-solving the optimization problem for each disturbance, and applying finite differences. Alternatively, F may be calculated based on the implicit function theorem applied to the optimality conditions, as is done in the software sIPOPT [23].

3. Development of an MPC with zone control and artificial SOC variables targets for RTO implementation

As mentioned above, keeping the self-optimizing variables at their setpoints when the active set changes, can lead to violation of other process constraints. This section proposes an algorithm capable of respecting the process constraints and assuring an acceptable profit loss by integrating both RTO and SOC methodologies. Here, the self-optimizing control methodology is implemented through MPC with zone control. The MPC guarantees the feasibility of constrained inputs and output variables, while it controls the set of self-optimizing controlled variables to the best possible degree. Fig.3 illustrates this idea, where the output variables are divided in two classes: constrained variables (e.g. product properties specification or safe operation temperatures) and other controlled variables (artificial self-optimizing output variables).

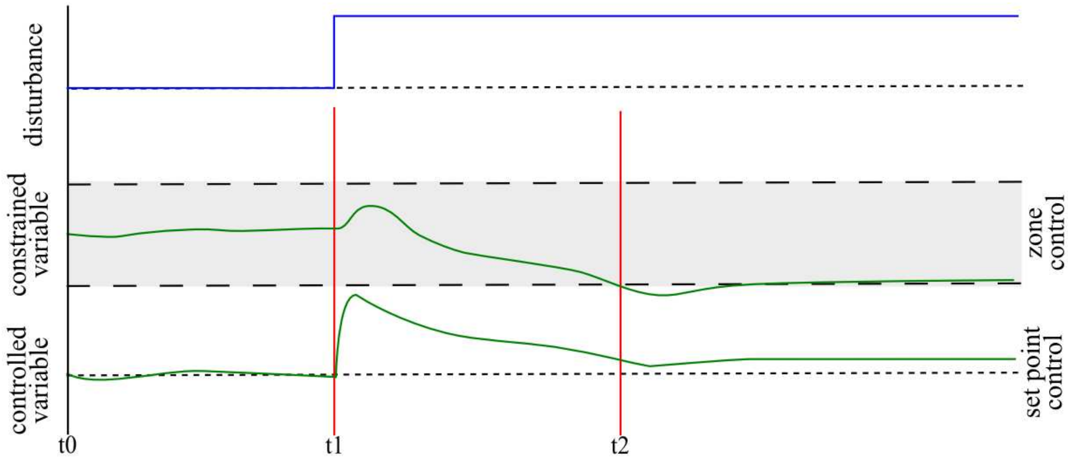


Figure 3. MPC with zone control and SOC

At $t0$, the controller keeps the controlled variable at its set point and the constrained variable does not have any influence on the MPC objective function (since it is within its zone). When a disturbance affects the system at $t1$, the controller tries to drive the controlled variable to its set point, but at the expense of moving the constrained variable toward one of its zone bounds. At $t2$, the constrained variable reaches its lower bound; then, the controller is forced to keep the constrained variable inside its zone, leading to an offset in the previously controlled variable.

This strategy allows for the imposition of the constraint satisfaction within an RTO cycle, while keeping the controlled variables as close to their optimal set points as possible, until a new RTO cycle updates the controlled variable set points to values compatible with the actual set of disturbances. While the classic MPC with zone control uses the input (manipulated variables) targets as controlled variables [24], this study uses a set of self-optimizing control (output) variables as controlled variables in order to assure acceptable loss in case of known disturbances.

This zone control MPC approach will detect when new constraints become active, because it will not be possible to maintain the self-optimizing controlled variables at their setpoints without one of the zone constraints becoming active. However when a disturbance causes a constraint to become *inactive* between RTO executions, this will not be detected by the zone control MPC. The reason for this is that the controlled variable $c=Hy$ was designed for the active set determined by the previous RTO execution; it does not contain any information about how the cost is affected when a disturbance causes one of the constraints to become inactive. In this case the old active set will be controlled until a new H matrix is obtained at the next RTO run.

Modified dynamic model

The formulation of MPC with zone control and SOC targets considers a linear dynamic model with n_u inputs, n_x states, and n_y outputs.

$$\begin{aligned} x(k+1) &= A\,x(k) + B\,u(k) \\ y(k) &= C\,x(k) \end{aligned} \tag{3}$$

The model (3) is rearranged in an incremental form in order to eliminate the output offset as described in [25],

$$\begin{aligned} \begin{bmatrix} x(k+1) \\ u(k) \end{bmatrix} &= \begin{bmatrix} A & B \\ 0 & I_{nu} \end{bmatrix} \begin{bmatrix} x(k) \\ u(k-1) \end{bmatrix} + \begin{bmatrix} B \\ I_{nu} \end{bmatrix} \Delta u(k) \\ y(k) &= \begin{bmatrix} C & 0 \end{bmatrix} \begin{bmatrix} x(k) \\ u(k-1) \end{bmatrix} \end{aligned} \tag{4}$$

which we write in compact form as

$$\begin{aligned} \bar{x}(k+1) &= D\,\bar{x}(k) + E\,\Delta u(k) \\ y(k) &= M\,\bar{x}(k) \end{aligned} \tag{5}$$

here $y(k) \in \mathbb{R}^{n_y}$ is the vector of measured output variables (including the constrained and unconstrained measured variables) and $\bar{x}(k) \in \mathbb{R}^{n_x+n_u}$. The dynamic model in (5) is the basis of a new dynamic model with two output vectors, namely, a vector of artificial SOC variables $c(k) \in \mathbb{R}^{n_c}$ and a vector of constrained variables $r(k) \in \mathbb{R}^{n_r}$:

$$
\begin{aligned}
\bar{x}(k+1) &= D\,\bar{x}(k) + E\,\Delta u(k) \\
c(k) &= H\,M\,\bar{x}(k) \\
r(k) &= W\,M\,\bar{x}(k)
\end{aligned}
\tag{6}
$$
[1]

The vector of artificial SOC variables $c(k)$ is selected from $y(k)$ using the map H (matrix of measurements combination obtained by the null space method), while $r(k)$ is the vector of constrained variables (e.g. product composition) selected using the matrix *W*, comprised of zeros and ones. After that, the dynamic model used here is simplified to:

$$
\begin{aligned}
\bar{x}(k+1) &= D\,\bar{x}(k) + E\,\Delta u(k) \\
c(k) &= U\,\bar{x}(k) \\
r(k) &= V\,\bar{x}(k)
\end{aligned}
\tag{7}
$$

here $U=HM$ and $V=WM$.

Both vectors of predicted outputs, $\bar{c}(k) \in \mathbb{R}^{(n_c \cdot p)}$ and $\bar{r}(k) \in \mathbb{R}^{(n_r \cdot p)}$, the controlled and constrained variables respectively, are extended over *p* intervals, and the input movements are extended over *m* intervals, i.e., $\Delta\bar{u}_k \in \mathbb{R}^{(n_u \cdot m)}$ (vector of input movements starting in *k*), according to (8) and (9); Using the matrices Ψ, Θ, Ω and T are defined in equations (A1) and (A2) of Appendix A, we write the controlled variables over the horizon as

$$\bar{c}(k) = \Psi\,\bar{x}(k) + \Theta\Delta\bar{u}_k \tag{8}$$

$$\bar{r}(k) = \Omega\,\bar{x}(k) + \mathrm{T}\,\Delta\bar{u}_k \tag{9}$$

We assume that the input values are constant after *m*, i.e.

$$u(k+m) = u(k+m+1) = \ldots = u(k+m+(p-m)) \Rightarrow \Delta u(k+m+i) = 0,\ i = 0,\ldots,(p-m).$$

The reason we distinguish between the controlled outputs *c* and the constrained outputs *r* is because this makes it possible to immediately assign different weightings to them in the MPC controller.

[1] This model may be generated by the RTO at each sampling time, while respecting the current active constraints. As none of the constraints will be relaxed between the RTO runs, this model is well suited for use in the MPC layer.

Zone constraints

Using the vector of constrained predicted variables $\bar{r}(k)$, it is possible to determine a set of inequality constraints that implement the zone control strategy shown in Fig.3. Equation (10) describes this set of inequality constraints.

$$-\mathrm{T}\Delta\bar{u}_k \le \Omega\,\bar{x}(k) - b_{\min} \quad ; \quad \mathrm{T}\Delta\bar{u}_k \le b_{\max} - \Omega\,\bar{x}(k) \tag{10}$$

Furthermore, the input (manipulated) variables should also be constrained due to physical limits imposed by the plant equipment, for instance, maximum or minimum flow rate for a particular stream. The nominal values of the input variables, with respect to the input increments, are given by (11) and the set of their inequality constraints are given by (12).

$$\bar{u}(k) = \tilde{M}\Delta\bar{u}_k + \tilde{I}u(k-1); \qquad \tilde{M} = \begin{bmatrix} I_{n_u} & 0 & \cdots & 0 \\ I_{n_u} & I_{n_u} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ I_{n_u} & I_{n_u} & \cdots & I_{n_u} \end{bmatrix} \in \mathbb{R}^{(n_u.m)\times(n_u.m)}$$
$$\tilde{I} = \begin{bmatrix} I_{n_u} \\ I_{n_u} \\ \vdots \\ I_{n_u} \end{bmatrix} \in \mathbb{R}^{(n_u.m)\times n_u} \tag{11}$$

$$\tilde{M}\Delta\bar{u}_k \le \bar{u}_{\max} - \tilde{I}u(k-1) \quad ; \quad -\tilde{M}\Delta\bar{u}_k \le \tilde{I}u(k-1) - \bar{u}_{\min} \tag{12}$$

By grouping all the inequality constraints ((10) and (12)) and including l_1 penalty variables to create soft constraints (and thus avoid infeasibilities in the optimization step of the control problem) the set of inequality constraints in (13) and (14) is obtained, which implements the zone control policy (see Fig.3).

$$\begin{bmatrix} T & -I_{n_r.p} & 0 & 0 & 0 \\ -T & 0 & -I_{n_r.p} & 0 & 0 \\ \tilde{M} & 0 & 0 & -I_{n_u.m} & 0 \\ -M & 0 & 0 & 0 & -I_{n_u.m} \end{bmatrix} \begin{bmatrix} \Delta\bar{u}_k \\ s_U^O \\ s_L^O \\ s_U^I \\ s_U^I \end{bmatrix} \le \begin{bmatrix} b_{\max} - \Omega\bar{x}(k) \\ \Omega\bar{x}(k) - b_{\min} \\ \bar{u}_{\max} - \tilde{I}u(k-1) \\ \tilde{I}u(k-1) - \bar{u}_{\min} \end{bmatrix} \tag{13}$$

$$\tilde{A}\begin{bmatrix}\Delta u_k \\ s_U^O \\ s_L^O \\ s_U^I \\ s_L^I\end{bmatrix} \le \tilde{b} \qquad (14)$$

here s are the penalty variables, the superscripts O and I indicate constrained outputs and input variables respectively, and the subscripts U and L indicate upper and lower bounds. These l_1 penalties (soft constraints) are included to promote numerical robustness of the resulting MPC controller. If a solution to the hard constrained problem exists then, for a sufficiently large value of M, the solutions of the hard and soft constrained problems are identical.

Target control

MPC setpoint control (see Fig.3) is implemented using a classic quadratic control objective function with a finite prediction horizon of p intervals, m input movements, and using the vector of predicted artificial SOC variables $\overline{c}(k)$:

$$J_k = (\overline{c}(k) - \overline{c}^{sp})^T \overline{Q}(\overline{c}(k) - \overline{c}^{sp}) + \Delta\overline{u}_k^T \overline{R} \Delta\overline{u}_k \qquad (15)$$

where $\overline{c}^{sp} = \begin{bmatrix}\underbrace{c^{sp^T} \ldots c^{sp^T}}_{p}\end{bmatrix}^T$ is the self-optimizing control variables setpoint vector. As noted in [26] the matrices Q and R can be determined from Hessian information from the RTO problem. Here we choose $\overline{Q} = diag\begin{bmatrix}\underbrace{Q \ \ldots \ Q}_{p}\end{bmatrix}$ as a diagonal weighting matrix on the differences between the controlled variables and their setpoints and $\overline{R} = diag\begin{bmatrix}\underbrace{R \ \ldots \ R}_{m}\end{bmatrix}$ as a diagonal weighting matrix on the input movements.

Equation (15) is written in terms of the input movements by substituting (8) into (15). This leads to the control objective function in terms of $\Delta\overline{u}_k$ in (16) and its quadratic form in (17).

$$J_k = (\Psi\overline{x}(k) + \Theta\Delta\overline{u}_k - \overline{c}^{sp})^T \overline{Q}(\Psi\overline{x}(k) + \Theta\Delta\overline{u}_k - \overline{c}^{sp}) + \Delta\overline{u}_k^T \overline{R} \Delta\overline{u}_k \qquad (16)$$

$$J_k = \Delta\bar{u}_k^{\,T} Y \Delta\bar{u}_k + 2\bar{a}_f^{\,T} \Delta\bar{u}_k + \bar{a} \tag{17}$$

where:

$$Y = \Theta^T \overline{Q} \Theta + \overline{R}$$
$$\bar{a}_f^{\,T} = (\Psi x(k) - \bar{c}^{sp})^T \overline{Q} \Theta$$
$$\bar{a} = (\Psi x(k) - \bar{c}^{sp})^T \overline{Q} \, (\Psi x(k) - \bar{c}^{sp})$$

MPC with zone control and artificial SOC variables targets

Now, it is necessary to combine the setpoint and zone control approaches into the same objective function. For this reason, the set of penalty variables is included into (17) as an l_1 penalty function, leading to (18).

$$\begin{aligned} J_k &= \Delta\bar{u}_k^{\,T} Y \Delta\bar{u}_k + 2\bar{a}_f^{\,T} \Delta\bar{u}_k + [s_U^{O\,T} \;\; s_L^{O\,T} \;\; s_U^{I\,T} \;\; s_L^{I\,T}]\, M\, e \\ e &= [1,1,1,\ldots]^T \end{aligned} \tag{18}$$

where M is a sufficiently large diagonal weighting matrix to make sure that the input movements will maintain the feasibility of the constrained variables. It is important to note that the input constraints are frequently modeled as hard constraints, because they represent physical limits. In the present work they are formulated as soft constraints with l_1 penalty functions for the sake of simplicity and numerical robustness. This was confirmed in the numerical results below, where we never experienced violation of input constraints.

Finally, the control problem is defined by (19) subject to the set of inequality constraints from (14).

$$\begin{aligned} & \min_{\Delta\bar{u}_k,\, s} J_k \\ \textit{subject to:} \; & \\ & \Delta\bar{u}_{\min} \le \Delta\bar{u} \le \Delta\bar{u}_{\max} \\ & s_U^O \ge 0,\; s_L^O \ge 0,\; s_U^I \ge 0,\; s_L^I \ge 0 \\ & \tilde{A} \begin{bmatrix} \Delta\bar{u}_k \\ s_U^O \\ s_L^O \\ s_U^I \\ s_L^I \end{bmatrix} \le \tilde{b} \end{aligned} \tag{19}$$

The above formulation is carried out on two nonlinear case studies presented in the next sections.

4. Case Study 1: Ammonia production

The MPC with zone control and SOC targets developed in the previous section is implemented in a case study of ammonia production, defined in [21], and presented in Fig.4. In this process, the feed stream (composed of hydrogen H_2 and nitrogen N_2) is compressed and mixed with the recycle stream to generate stream S2. Then, this mixture reacts at pressure P_{reac} to produce ammonia, which is cooled to temperature T_{flash} and then separated from the light components (H_2 and N_2). The recycle stream (S5) is split, generating stream S7 that is purged and stream S6 that is compressed and mixed with the feed stream (S1).

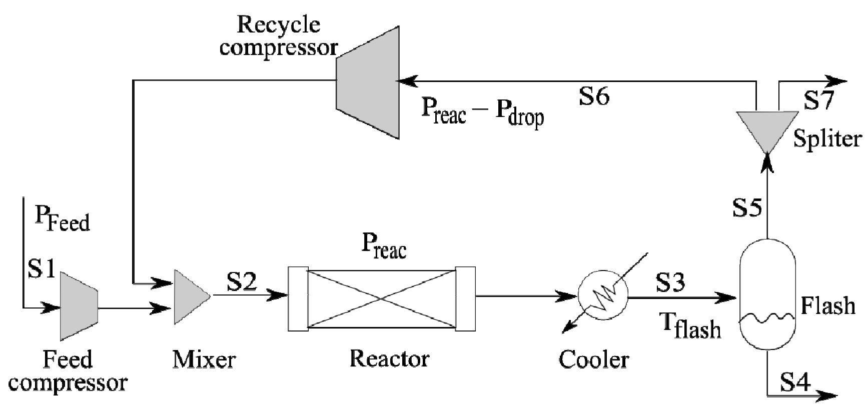


Figure 4. Schematic representation of ammonia production process

The ammonia production process is modeled at steady state by a set of mass and energy balances and equilibrium equations, simulated in the software AMPL® (the complete model is given in Appendix B). Equation (20) describes the optimization problem with the economic objective function composed by the production costs (compressors and cooling work) and profit (ammonia stream, S4).

$$
\begin{aligned}
&\min_{u} \quad Cost^{opt} = P_{feed}W_{feed} + P_{recy}W_{recy} + P_{cool}W_{cool} - P_{NH3}S4_{NH3} \\
&\text{subject to}: \ Steady\ \ State\ \ Model \\
&\qquad 266 \le T_{flash} \le 288 \qquad [K] \\
&\qquad 0 \ \ \le S6 \ \ \le 3.5 \qquad [mol/time]
\end{aligned}
\tag{20}
$$

where u=[P_{reac}, T_{flash}, sf] is the vector of decision variables (reactor pressure, flash temperature and split fraction); W_{feed}, W_{recy} and W_{cool} are respectively the work performed by the feed compressor, recycle compressor and cooler; $S4_{NH3}$ is the molar flow rate of ammonia in the product stream; P are the prices given in Table B2. The operational regions of this process are defined by a set of inequality constraints in flash temperature (T_{flash}) and recycle molar flow rate (S6).

Two disturbances are considered in the present analysis, $d1$ in the feed flow rate and $d2$ in the feed composition. These disturbances can be mathematically represented by (21). Fig.5 depicts the process profit function with respect to the disturbances and Fig.6 shows the active set map for the operating space, including the minimum cooler temperature (T_{flash}) and the maximum flow rate S6 and split fraction (sf).

$$S1 = 5.1 + d1 \quad \text{[mol/time]}$$
$$x^{S1} = \begin{bmatrix} 0.8 + d2 \\ (1-0.8) - d2 \\ 0 \end{bmatrix} \begin{matrix} \rightarrow H_2 \\ \rightarrow N_2 \\ \rightarrow NH_3 \end{matrix} \tag{21}$$

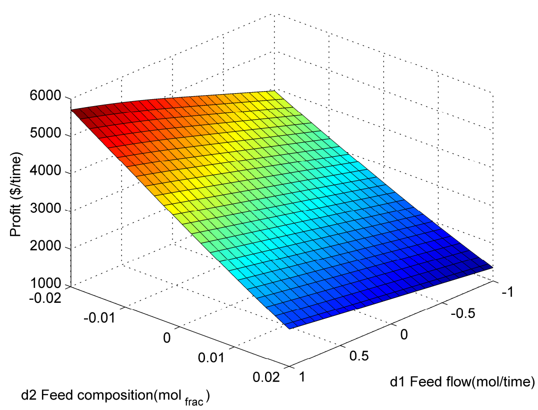


Figure 5. Profit of ammonia plant with respect to disturbances (This surface would be the cost if there were no active set changes)

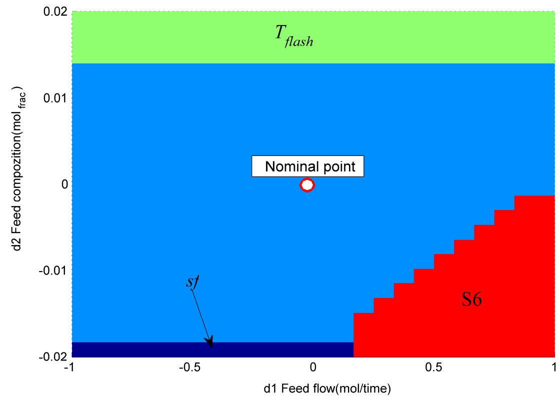


Figure 6. Active set map for the disturbance region, ammonia production case study. Each color denotes a region where the active set does not change. The variable names within the regions denote the constraints that are active

4.1. Steady state analysis

Three steady state cases are carried out to compare the economic performance of different MPC approaches under presence of disturbances. Case A simulates the "classical" control approach, where three process variables, flash temperature (T_{fash}) reactor pressure (P_{reac}) and recycle flow rate (S6), are controlled variables. Case B applies the control of three SOC variables computed as the linear combination of 6 measurements y=[$S2_{H2}$, $S2_{N2}$, S6, P_{reac}, T_{flash}, sf]^T. In this case, zone control of the constrained variables (molar flow rate, S6) is not considered. Case C simulates our new MPC with zone control and SOC targets; in this case, the same artificial SOC variables are controlled as in Case B. However, the zone constraints are enforced to respect the upper bounds on S6 stream.

These cases are conducted by the solution of the optimization problem in (22) (which corresponds to the steady state solution for a nonlinear model predictive controller), for a given disturbance value, $-1.00 \le d1 \le 1.00, -0.02 \le d2 \le 0.02$:

$$
\begin{aligned}
&\min_{u} \quad Obj = \left(c1 - c1^{sp}\right)^2 + \left(c2 - c2^{sp}\right)^2 + \left(c3 - c3^{sp}\right)^2 \\
&subject to: \quad \text{Steady State Model} \\
&\qquad Cost^{\exp} = P_{feed}W_{feed} + P_{recy}W_{recy} + P_{cool}W_{cool} + P_{NH3}S4_{NH3} \\
&\qquad 266 \le T_{flash} \le 288 \qquad [K] \\
&\qquad 0.4 \le sf \le 0.8
\end{aligned} \tag{22}
$$

where c are the controlled variables defined in Table 1 and c^{sp} are their setpoint values computed at the nominal point ($d1 = 0$ and $d2 = 0$). The sensitivity information required for the null space matrix, F, is also computed at this point, by solving problem (20), nested within finite difference perturbations. Additionally, Case C, considers the constraint in (23):

$$S6 \leq 3.5 \qquad [mol / time] \tag{23}$$

The performance of the three Cases is compared through the loss function, computed by the difference between the cost (negative of profit) achieved by the solution of the optimization problem in (22) ($Cost^{\exp}$) and the optimum cost ($Cost^{opt}$, solution of (20)). The results can be observed in Fig.7, which shows the loss profile with respect to the disturbances for each Case.

Table 1. Set of controlled variables for each Case (AV: artificial variable)

Case	c1	c2	c3
A	P_{reac}	T_{flash}	S6
B	AV	AV	AV
C	AV	AV	AV

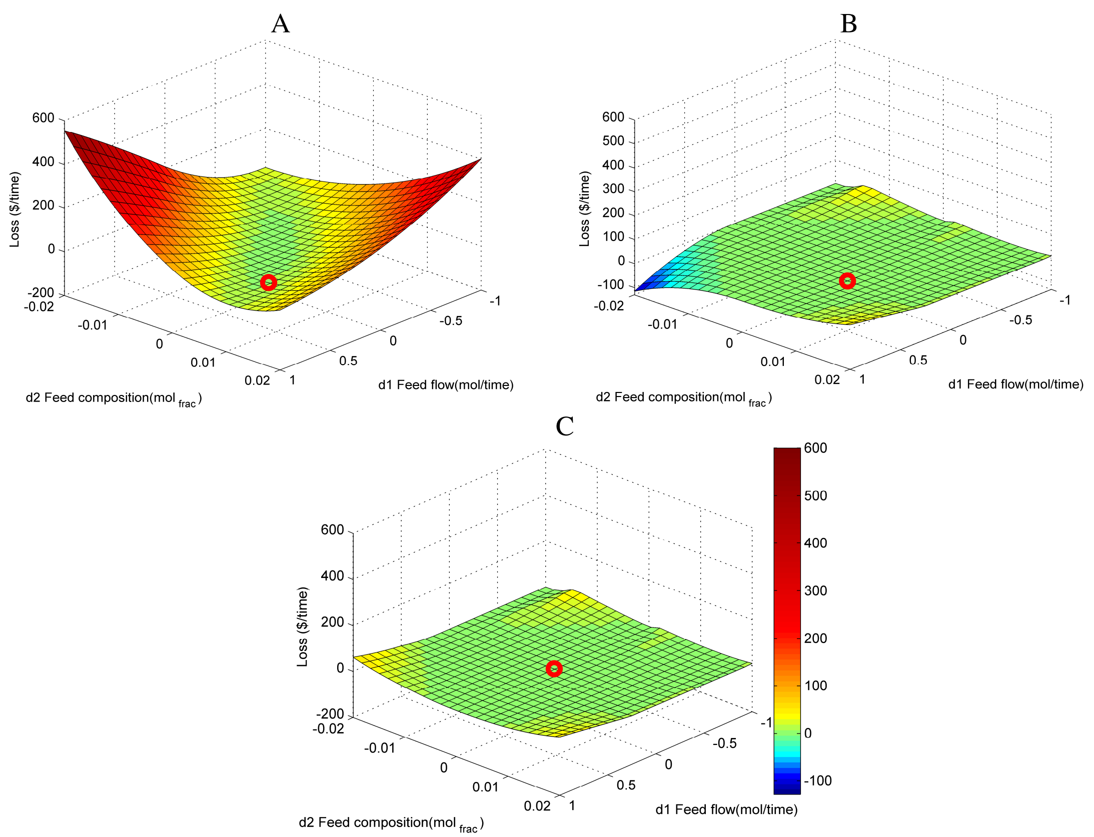


Figure 7. Steady state analysis results: (A) “classic” MPC, (B) MPC with artificial SOC variables and (C) MPC with zone control and SOC targets

The results observed in Fig.7 show that the classical MPC approach (Case A) obtained the worst steady state performance with maximum profit loss value of approximately 10 percent. In comparison, the control of SOC variables (Cases B and C) presented only 1 percent of loss in the worst case. It is important to notice the difference between the performances of Cases B and C. As long as Case B does not consider the constraint satisfaction of all process variables, this case obtained a negative loss region where constraint (23) is violated (it represents a better economic performance, but for an infeasible process). On the other hand, Case C enforces constraint (23), since the controlled variable S6 in handled in the zone control approach. For this reason, Case C does not present the same behavior (negative loss region) observed in Case B.

5. Case Study 2: BTX separation

In this section, the novel MPC with zone control and SOC targets is implemented in the second case study, a BTX (Benzene, Toluene and p-Xylene) separation by a multi-column distillation process, is described in [27] and depicted in Fig.8. In addition to providing a steady state analysis, we also demonstrate the dynamic performance of our suggested method.

A BTX mixture feeds the first column, where benzene is removed from the top flow rate. The mixture, rich in Toluene and p-Xylene, feeds the second column where Toluene is removed in the top flow rate and Xylene from the bottom.

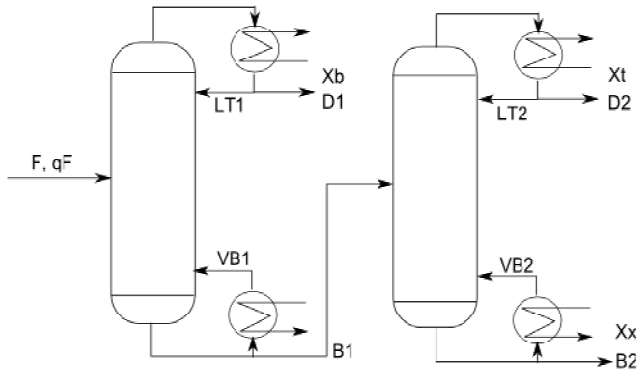


Figure 8. BTX process schematic representation

The columns are modeled as a sequence of ideal equilibrium stages, with constant relative volatility and vapor flow rate, through all the stages. The liquid flow rate is given by the Francis weir formula. Both columns have 41 theoretical equilibrium stages, including the total condenser and the partial reboiler, and the feed trays correspond to stage 21, counting from the bottom to the top. More details about this model, as well as the model built in AMPL® can be found in [27].

In the first column, seven states are considered as measured variables, including the distillate Benzene composition Xb, 3 stage temperatures in the rectification section and 3 in the stripping section. The second column has eight measured variables, including the Toluene and p-Xylene molar compositions (Xt and Xx), and equivalent stage temperatures of first column. All of these states comprise the vector of measured variables, $y(k)$ in (5).

It is assumed that the molar holdups in the condenser drums and reboilers are controlled by the distillate and bottom flow rates, respectively. Thus, the problem has four manipulated variables, which are chosen as $u = [LT1, VB1, LT2, VB2]$, and the economic optimization problem given by (24).

$$\begin{aligned} \min_{u} \quad & Cost^{opt} = p_F F + p_V (VB1 + VB2) - p_B D1 - p_T D2 - p_X B2 \\ & Xb \geq 0.95 \\ & Xt \geq 0.95 \\ & Xx \geq 0.95 \\ & VB1 \leq 4.080 \quad [kmol/\min] \\ & VB2 \leq 2.405 \quad [kmol/\min] \end{aligned} \tag{24}$$

where D and B are the respective distillate and bottom flow rates, F is the Feed flow rate, and p_F , p_V , p_B , p_T and p_X are respectively the prices of feed, vapor, benzene, toluene and xylene streams. Relevant model parameters are given in Table 2.

Table 2. Parameters values for BTX process model

Feed F [kmol/min]	Liquid fraction qF	Vapor price p_V [$/kmol]	Feed price p_F [$/kmol]	Benzene price p_B [$/kmol]	Toluene price p_T [$/kmol]	p-Xylene price p_X [$/kmol]
1.41	1.00	0.035	1.00	1.00	3.00	2.00

The two disturbances acting on the system are the molar fraction of benzene and toluene in the feed stream, which can be described by (25).

$$\begin{aligned} z_{ben}^F &= 0.40 + d1 \\ z_{tol}^F &= 0.20 + d2 \\ z_{xyl}^F &= 1 - z_{ben}^F - z_{tol}^F \end{aligned} \tag{25}$$

Fig.9 depicts the cost profile with respect to the disturbances $d1, d2 \in [-0.015, 0.045]$. The active set map for this region is shown in Fig.10, which presents four different active sets, including the minimum toluene product concentration Xt , and the maximum boil up rates for the reboilers $VB1$ and $VB2$.

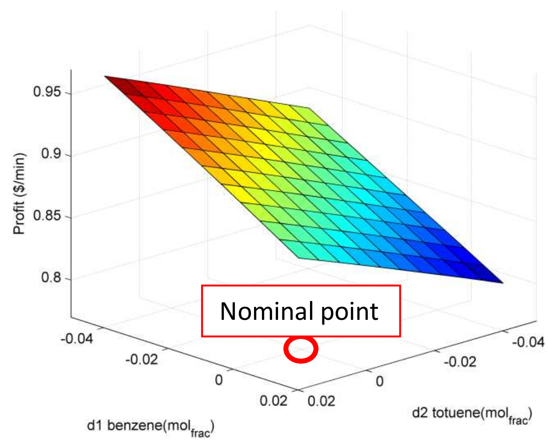


Figure 9. Cost profile with respect to disturbances of BTX process (this surface shows how the cost would change if there were no active set changes)

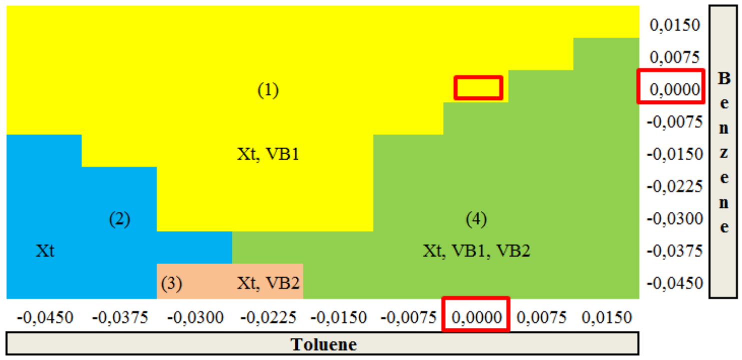


Figure 10. Active set map of BTX process (The variables in each colored region denote the constraints that are active)

5.1 Steady state analysis

In this section, three steady state cases are carried out to compare the economic performance of different MPC approaches, as performed in the ammonia production case study (see section 4.1). Once more Case A simulates the classical MPC approach, where the three product concentrations and one tray temperature in the first column (TC1_10) are controlled variables. Case B applies the control of four SOC variables computed as the linear combination of 12 measured tray temperatures defined in section 5. In this case zone control of the constrained variables (products concentration) is not considered. Case C represents the new MPC with zone control and SOC targets developed in the present work, controlling the same artificial SOC variables as Case B. However, the zone constraints are enforced to respect the lower bounds on product concentrations, simulating the zone control approach.

Mathematically, the cases are conducted by the solution of the optimization problem in (26) (again, this problem corresponds to the steady state solution obtained with a nonlinear model predictive controller), for a given disturbance value $-0.015 \le d1 \le 0.045$, $-0.015 \le d2 \le 0.045$ $[mol\ frac]$

$$
\begin{aligned}
\min_{u} \quad & Obj = \left(c1 - c1^{sp}\right)^2 + \left(c2 - c2^{sp}\right)^2 + \left(c3 - c3^{sp}\right)^2 + \left(c4 - c4^{sp}\right)^2 \\
\text{subject to:} \quad & \text{Steady State Model} \\
& \text{Cost}^{\exp} = p_F F + p_V (VB1 + VB2) - p_B D1 - p_T D2 - p_X B2 \\
& VB1 \le 4.080 \qquad \text{[kmol/min]} \\
& VB2 \le 2.405 \qquad \text{[kmol/min]}
\end{aligned}
\tag{26}
$$

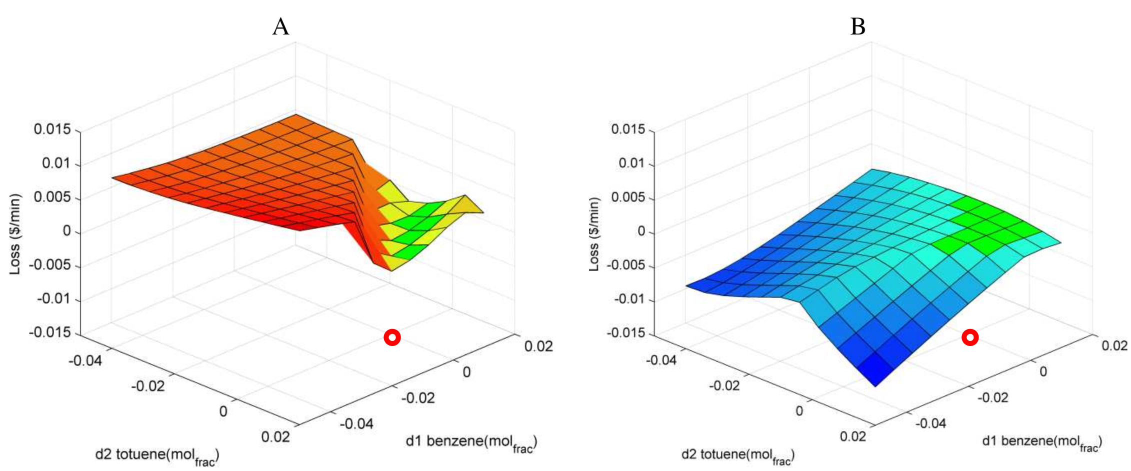
where c are the controlled variables summarized in Table 3 and c^{sp} are their setpoint values computed at the nominal point ($d1 = 0$ and $d2 = 0$). As carried out in the first case study, the sensitivity information required for the null space matrix, *F,* is computed at nominal point, by solving problem (24), nested within finite difference perturbations. Additionally, Case C, considers the constraints in (27).

Table 3. Set of controlled variables for each Case, BTX case study(AV: Artificial variable computed by the null space method)

Case	c1	c2	c3	c4
A	Xb	TC1_10	Xt	Xx
B	AV*	AV	AV	AV
C	AV	AV	AV	AV

$$\begin{aligned} Xb &\geq 0.95 \\ Xt &\geq 0.95 \\ Xx &\geq 0.95 \end{aligned} \tag{27}$$

Fig.11 presents the profit loss function with respect to the disturbances, computed by the difference between cost achieved by the solution of the optimization problem in (26) ($Cost^{exp}$) and the optimum cost ($Cost^{opt}$) from the solution of (24).



C

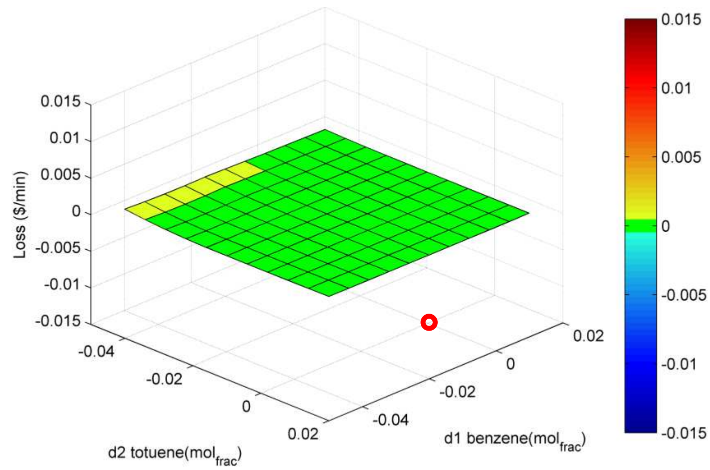


Figure 11. Steady state analysis results: (A) "classic" MPC, (B) MPC with artificial SOC variables and (C) MPC with zone control and SOC targets (red marks are the nominal point). BTX case study

The results show that the loss function is significantly influenced by the choice of the control structure (see Fig.11). Note that Case A, the "classical" control approach, has the worst performance regarding this set of disturbances, yielding lower values of profit loss close to the nominal point, while the largest part of its area presents losses greater than 0.005 $/min with a maximum of 0.01173 $/min. The behavior of the classical MPC approach can be compared to the illustrative example given in the second paragraph of section 3, where suboptimal operation is expected after a determined disturbance, at least until the RTO module updates the setpoint values.

On the other hand, Case B simulates the control of the artificial SOC variables without enforcing the product constraints. This case shows a loss close to zero around the nominal point and negative loss in the remaining area. This behavior is explained by violation of the product concentration constraints. In other words, in the presence of the analyzed disturbances, control of these setpoints without a policy of constraint satisfaction (zone control) leads to an increased profit because the product stream does not satisfy specifications. (Here the toluene concentration is less than 95% at top of the second column). This was also discussed in the first paragraph of section 3. However, such performance is not feasible, as it violates the constraints on the products, which cannot be sold anymore at the desired price.

Finally, Case C shows the best performance among the analyzed approaches, with a flat loss profile surface close to zero, and maximum loss value of 0.00076 \$/min. In this case, the constraints in (27) enforce the minimum product concentration values, at the expense of yielding offsets in the controlled variables, as expected in the development of this new method.

5.2. Dynamic analysis

The BTX plant described above is now modeled dynamically in MATLAB® and simulated as a system of 246 nonlinear ordinary differential equations, to represent the process. The MPC controller uses linear models identified by transfer functions in step response cases at the nominal point, and then, converted to a state-space model. The "classical" MPC is implemented in the case study through the MATLAB® MPC Toolbox 4.1.2, using the controlled variables defined in Case A of section 5.1. On the other hand, the MPC with zone control and SOC target (Case C) is applied by solving the optimization problem defined in (18) using the interior-point algorithm implemented in the MATLAB® function "*quadprog*". Nonlinear dynamic simulations were performed for Cases A and C. Case B was not considered further, because product specifications were already violated at steady state.

The dynamic cases comprise the simulation of both MPC approaches (Cases A and C) starting from the nominal optimal operating point. At time zero, the RTO layer computes the optimum economic setpoints and the sensitivity analysis (optimal sensitivity matrix F), and the controlled variable $c=Hy$. At time 8 min, a disturbance is introduced ($d1 = -0.04$) and the MPC drives the process towards a new operational steady state point. Finally, at time 50 min the RTO layer updates the setpoint values to the actual economical optimum. Fig.12 depicts the profit obtained by approaches A, C and the optimum steady-state condition in these settings.

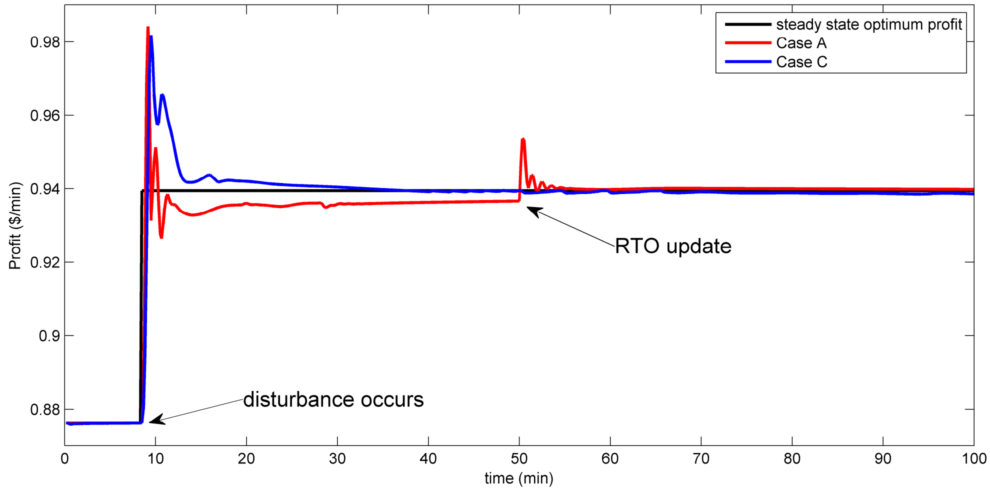


Figure 12. Comparison of the profit obtained by each MPC approach

Note in Fig. 12 that MPC with zone control and SOC targets (Case C) has better economic performance compared to conventional MPC (Case A). After introducing the disturbance, our new MPC approach settles to the economic optimum, whereas the classical approach maintains the process at a suboptimal operating point. At 50 minutes, when a new RTO cycle is performed, we observe a system upset with the classical approach, in order to drive the process to the optimum economic setpoint. Since Case C already operates at the optimum setpoint, this upset does not appear in the implementation of zone control MPC.

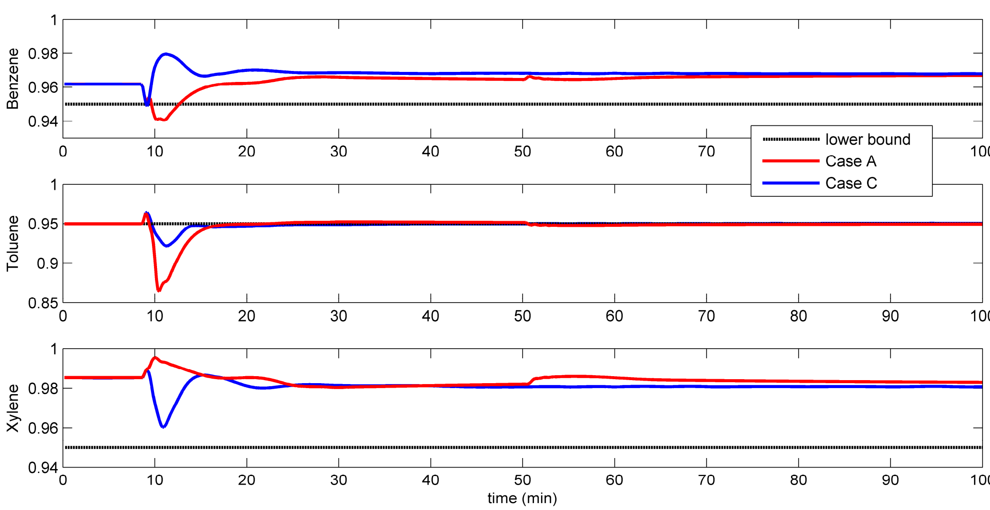


Figure 13. Constrained variable profiles

Fig.13 shows the concentration profile of each product stream. It can be observed that, after the disturbance, the benzene concentration starts to decrease for both cases; however, only Case A yields an out of specification stream of benzene. Observing the concentration profile of the toluene stream, we observe better performance of the zone control policy, which obtain out of specification products over a shorter period of time. The constraint violation comes from the implementation as soft constraints. No constraint violation occurs for the (inactive) constraint on xylene.

However, when the RTO re-optimizes with constant disturbance, we see that Case A purifies more than required at time 50 until it reaches the new optimal steady-state, while Case C minimizes the “product give-away”. With Case C, concentrations move toward their targets faster. In particular, the product stream has a minimum toluene purity of 92.2%, compared to 86.6 % with the classical approach.

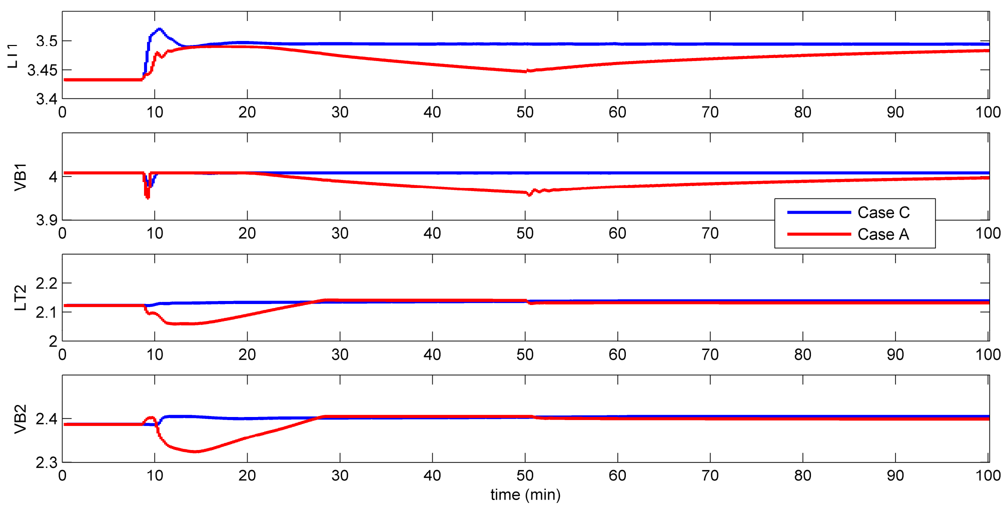


Figure 14. Manipulated variables

Finally, manipulated variable profiles are depicted in Figure 14. In this case, we also observe that the MPC with zone control and SOC targets yields smaller control actions than the classical approach, which may be assigned to the control of the SOC variables. This is mainly observed in the manipulated variable behavior of the second column. Another important consideration is the ability of the new approach to stabilize the system after the disturbance, which is not observed in the response of the classical approach.

6. Conclusions

This study presents a new RTO algorithm with zone control and SOC variable targets. The novel approach is demonstrated on two case studies: a plant of ammonia production and a detailed simulation of a multi-column distillation process. The results at steady state and dynamic operation show better economic performance of the new approach in comparison with classical RTO/MPC, and require less effort from the manipulated variables to keep the process under control. This characteristic improves process dynamics, since it requires less process change when the RTO re-optimizes. Moreover, the zone constraint policy outperforms the classical target approach regarding constraint satisfaction, showing faster responses to drive the concentration profile back to their zones or targets. These facts indicate that the integration of RTO and SOC can be a good solution to the problem of low frequency updates in RTO. Moreover, the zone control policy presented in this work is an important step towards handling the problem of active set changes observed in the SOC methodology. Although constraints that become inactive are not relaxed in between RTO executions, the results presented in this paper show that our SOC-based MPC approach is able to do what was proposed, which is to decrease the profit loss between two RTO performances.

For future work we plan to improve our MPC approach by using additional sensitivity information to relax constraints that become inactive in between RTO runs. Moreover, we will consider improved methodologies to select the matrix H, such as the "exact local method" [11], that can take measurement noise into account. . Finally, another interesting development is to offer an objective procedure to find the weighting matrix $\overline{Q}$ that guarantees the existence of a more consistent MPC scheme, as in Zanon et al. [26].

Acknowledgments

The authors would like to thank CAPES (Brazil) for the financial support.

References

[1] T. Ren, "Barriers and drivers for process innovation in the petrochemical industry: A case study," *J. Eng. and Technol. Manage* 26 (4) (2009) 285–304.

[2] M. A. Brdys and P. Tatjewski, *Iterative Algorithms for Multilayer Optimization Control*. Imperial College Press, London, 2005.

[3] I. P. Miletic and T. E. Marlin, "On-line Statistical Results Analysis in Real-Time Operations Optimization," *Ind. Eng. Chem. Res.* 37 (9) (1998) 3670–3684.

[4] M. L. Darby, M. Nikolaou, J. Jones, and D. Nicholson, "RTO: An overview and assessment of current practice," *J. Process Contr.* 21 (6) (2011) 874–884.

[5] S. Engell, "Feedback control for optimal process operation," *J. Process Contr* 17 (3) (2007) 203–219.

[6] J. Jäschke, X. Yang, and L. T. Biegler, "Fast economic model predictive control based on NLP-sensitivities," *J. Process Contr.* 24 (8) (2014) 1260–1272.

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

[8] S. Skogestad, "Plantwide control: the search for the self-optimizing control structure," *J. Process Contr* 10 (5) (2000) 487–507.

[9] A. C. B. de Araújo, M. Govatsmark, and S. Skogestad, "Application of plantwide control to the HDA process. I—steady-state optimization and self-optimizing control," *Control Eng. Pract.* 15 (10) (2007) 1222–1237.

[10] T. Larsson, K. Hestetun, E. Hovland, and S. Skogestad, "Self-Optimizing Control of a Large-Scale Plant: The Tennessee Eastman Process," *Ind. Eng. Chem. Res.* 40 (22) (2001) 4889–4901.

[11] V. Alstad, S. Skogestad, and E. S. Hori, "Optimal measurement combinations as controlled variables," *J. Process Contr.* 19 (1) (2009) 138–148.

[12] I. J. Halvorsen, S. Skogestad, J. C. Morud, and V. Alstad, "Optimal Selection of Controlled Variables" *Ind. Eng. Chem. Res.* 42 (14) (2003) 3273–3284.

[13] V. Alstad and S. Skogestad, "Null Space Method for Selecting Optimal Measurement Combinations as Controlled Variables," *Ind. Eng. Chem. Res.* 46 (3) (2007) 846–853.

[14] J. Jaschke and S. Skogestad, "Using Process Data for Finding Self-optimizing Controlled Variables," 10th IFAC *International Symposium on Dynamics and Control of Process Systems*, Mumbai, December, 2013.

[15] V. Kariwala, Y. Cao, and S. Janardhanan, "Local Self-Optimizing Control with Average Loss Minimization," *Ind. Eng. Chem. Res.* 47 (4) (2008) 1150–1158.

[16] L. Ye, Y. Cao, Y. Li, and Z. Song, "Approximating Necessary Conditions of Optimality as Controlled Variables," *Ind. Eng. Chem. Res.* 52 (2) (2013) 798–808.

[17] L. Ye, Y. Cao, X. Ma, and Z. Song, “A Novel Hierarchical Control Structure with Controlled Variable Adaptation,” *Ind. Eng. Chem. Res.* 53 (38) (2014) 14695–14711.

[18] G. François, B. Srinivasan, and D. Bonvin, “Use of measurements for enforcing the necessary conditions of optimality in the presence of constraints and uncertainty,” *J. Process Contr.* 15 (6) (2005) 701–712.

[19] W. Hu, L. M. Umar, G. Xiao, and V. Kariwala, “Local self-optimizing control of constrained processes,” *J. Process Contr* 22 (2) (2012) 488–493.

[20] Y. Cao, “Constrained self-optimizing control via differentiation,” *7th International Symposium on Advanced Control of Chemical Processes (ADCHEM)*, Hong Kong, January, 2004.

[21] H. Manum and S. Skogestad, “Self-optimizing control with active set changes,” *J. Process Contr.* 22 (5) (2012) 873–883.

[22] D. P. Bertsekas, A. Nediâc, and A. E. Ozdaglar, *Convex analysis and optimization*, e.1 ed., Athena Scientific, Belmont, 2003.

[23] H. Pirnay, R. López-Negrete, and L. T. Biegler, “Optimal sensitivity based on IPOPT,” *Math.Prog. Comp.* 4 (4) (2012) 307–331.

[24] A. H. González and D. Odloak, “A stable MPC with zone control,” *J. Process Contr.* 19 (1) (2009) 110–122.

[25] U. Maeder, F. Borrelli, and M. Morari, “Linear offset-free Model Predictive Control,” *Automatica* 45 (10) (2009) 2214–2222.

[26] M. Zanon, S. Gros, and M. Diehl, “Indefinite linear MPC and approximated economic MPC for nonlinear systems,” *J. Process Contr.* 24 (8) (2014) 1273–1281.

[27] R. B. Leer, “Self-optimizing control structures for active constraint regions of a sequence of distillation columns,” Norwegian University of Science and Technology, 2012.

Appendix A

Equations A1 and A2 describe the computation of the output predicted vectors with p predicted intervals and m control actions, where $\bar{c}$ is the predicted vector of artificial SOC variables and $\bar{r}$ is the vector of constrained output variables. Matrices U, V, D and E are given in (7).

$$\begin{bmatrix} c(k+1|k) \\ c(k+2|k) \\ \vdots \\ c(k+m|k) \\ c(k+m+1|k) \\ \vdots \\ c(k+p|k) \end{bmatrix} = \begin{bmatrix} UD \\ UD^2 \\ \vdots \\ UD^m \\ UD^{m+1} \\ \vdots \\ UD^p \end{bmatrix} \bar{x}(k) + \begin{bmatrix} UE & 0 & \cdots & 0 \\ UDE & UE & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ UD^{m-1}E & UD^{m-2}E & \cdots & UE \\ UD^{m}E & UD^{m-1}E & \cdots & UDE \\ \vdots & \vdots & \ddots & \vdots \\ UD^{p-1}E & UD^{p-2}E & \cdots & UD^{p-m}B \end{bmatrix} \begin{bmatrix} \Delta u(k|k) \\ \Delta u(k+1|k) \\ \vdots \\ \Delta u(k+m-1|k) \end{bmatrix} \quad \text{(A1)}$$

$$\bar{c}(k) \quad = \quad \Psi \quad \bar{x}(k) + \quad \Theta \quad \Delta\bar{u}_k$$

$$\begin{bmatrix} r(k+1|k) \\ r(k+2|k) \\ \vdots \\ r(k+m|k) \\ r(k+m+1|k) \\ \vdots \\ r(k+p|k) \end{bmatrix} = \begin{bmatrix} VD \\ VD^2 \\ \vdots \\ VD^m \\ VD^{m+1} \\ \vdots \\ VD^p \end{bmatrix} \bar{x}(k) + \begin{bmatrix} VE & 0 & \cdots & 0 \\ VDE & VE & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ VD^{m-1}E & VD^{m-2}E & \cdots & VE \\ VD^{m}E & VD^{m-1}E & \cdots & VDE \\ \vdots & \vdots & \ddots & \vdots \\ VD^{p-1}E & VD^{p-2}E & \cdots & VD^{p-m}B \end{bmatrix} \begin{bmatrix} \Delta u(k|k) \\ \Delta u(k+1|k) \\ \vdots \\ \Delta u(k+m-1|k) \end{bmatrix} \quad \text{(A2)}$$

$$\bar{r}(k) \quad = \quad \Omega \quad \bar{x}(k) + \quad \mathrm{T} \quad \Delta\bar{u}_k$$

where $\Psi \in \mathbb{R}^{(n_c \cdot p)\times(n_x+n_u)}$, $\Theta \in \mathbb{R}^{(n_c \cdot p)\times n_u}$, $\Omega \in \mathbb{R}^{(n_r \cdot p)\times(n_x+n_u)}$ and $\mathrm{T} \in \mathbb{R}^{(n_c \cdot p)\times n_u}$

Appendix B

The ammonia production case study is based on conversion of hydrogen and nitrogen in ammonia by the stoichiometric equation given in (B1). This process is composed by 8 equipments, which are modeled as a set of mass, energy and equilibrium equations, as follows:

$$3\,H_2 + N_2 \Leftrightarrow 2\,NH_3 \tag{B2}$$

Mixer

$$\begin{aligned} S2 &= S1 + S6 \\ S2\,x2 &= S1\;x1 + S6\;x6 \end{aligned} \tag{B2}$$

where S is the molar flow rate for the respective stream (e.g. stream 1, 2 and 6); *x* is the vector of molar fractions of the respective stream, sorted by H_2, N_2 and NH_3.

Reactor

$$\begin{aligned} S3\,x3 &= S2\,x2 - \xi E \\ K_{eq} &= \frac{\left(P_{reac} x3_{NH3}\right)^2}{\left(P_{reac} x3_{H2}\right)^3\left(P_{reac} x3_{N2}\right)} \\ 1 &= x3_{H2} + x3_{N2} + x3_{NH3} \end{aligned} \tag{B2}$$

The reactor is modeled as an equilibrium reactor and its output stream S3 is calculated by the extent reaction (ξ). *E* is a vector of stoichiometric coefficients $\left[-3,-1,2\right]^T$. K_{eq} is the equilibrium constant given in Table B1 and P_{reac} is the reactor pressure in bar.

Flash

$$\begin{aligned} k_{H2} &= \frac{H^0_{H2} + H_{H2} T_{flash}}{P_{reac}} \\ k_{N2} &= \frac{H^0_{N2} + H_{N2} T_{flash}}{P_{reac}} \\ k_{NH3} &= \frac{10^{A - \frac{B}{(Tflash + C)}}}{P_{reac}} \end{aligned} \tag{B3}$$

$$
\begin{aligned}
&x5 = diag([k_{H2}, k_{N2}, k_{NH3}])\, x4\\
&\sum_{i=\{H2,N2,NH3\}} \frac{x3_i (k_i - 1)}{1 + vf\,(k_i - 1)} = 0\\
&S4 = S3(1 - vf)\\
&S5 = S3\, vf\\
&S3\, x3 = S4\ x4 + S5\ x5
\end{aligned}
\tag{B4}
$$

The constants used to compute the K-values (k_{H2} , k_{N2} and k_{NH3}) are displayed in Table B1, and the vapor fraction *vf* is calculated by the Rachford-Rice equation.

Splitter

$$
\begin{aligned}
&S5\ sf = S6\\
&S5\ (1 - sf) = S7\\
&x5 = x6 = x7
\end{aligned}
\tag{B6}
$$

sf is the splitter fraction used as manipulated variable in Case A of section 4.1.

Feed compressor

$$
W_{feed} = S1 R T_0 \log\left(\frac{P_{reac}}{P_0}\right)\frac{1}{\eta_{fc}}
\tag{B5}
$$

W_{feed} is the compressor work used in the feed stream, η_{fc} is the compressor efficiency, R is the gas constant and T_0 and P_0 are the initial conditions of feed stream (see Table B1)

Recycle compressor

$$
W_{recy} = S6 R T_{flash} \log\left(\frac{P_{reac}}{P_{reac} - \Delta P}\right)\frac{1}{\eta_{rc}}
\tag{B6}
$$

W_{recy} is the compressor work used in the recycle stream S6, η_{rc} is the compressor efficiency and ΔP is the system pressure drop.

Cooler

$$
\begin{aligned}
&Tc = \frac{288 - T_{flash}}{\log\left(\dfrac{288}{T_{flash}}\right)}\\
&W_{cool} = \sum_{i=\{H2,N2,NH3\}} S3\ x3_i\, Cp_i \left(288 - Tflash\right)\left(\frac{288}{Tc} - 1\right)\frac{1}{\eta_{cool}}
\end{aligned}
\tag{B7}
$$

W_{cool} is the cooler work spent in the system, η_{cool} is the cooler efficiency, Cp_i are the heat capacity of each component. The present cooler model considers only the energy used to bring the temperature down to 288K, for higher temperatures the cooler work is considered zero.

Table B1. Constant values

Parameter	Value	Unit
K_{eq}	6.36e-5	
H^0_{H2}	210688	
H_{H2}	-656	
H^0_{N2}	110816	
H_{N2}	-342	
A	4.4854	
B	926.132	
C	-32.98	
Cp_{H2}	28.82	J/mol.K
Cp_{N2}	29.13	J/mol.K
Cp_{NH3}	35.06	J/mol.K
T_0	298.15	K
η_{rc}	1	
η_{fc}	1	
η_{cool}	1	
P_0	50	Bar
ΔP	15	Bar

Table B2. Costs for ammonia production case study

Parameter	Value	Unit
P_{feed}	0.5	$/time
P_{recy}	10	
P_{cool}	1.3	
P_{NH3}	1e4	