Self-optimizing Control and NCO tracking in the Context of Real-Time Optimization

Johannes Jäschke, Sigurd Skogestad

Department of Chemical Engineering, NTNU Trondheim, Norway
(e-mail: {jaschke,skoge}@chemeng.ntnu.no)

Abstract: This paper reviews the role of self-optimizing control (SOC) and necessary conditions of optimality tracking (NCO tracking). We argue that self-optimizing control is not an alternative to real-time optimization (RTO), NCO tracking or model predictive control (MPC), but is to be seen as complementary. In self-optimizing control we determine controlled variables (CV), that keep the process close to the optimum when a disturbance enters the process. These CVs are controlled at their setpoints using PID or model predictive controllers. Preferably, the setpoints are kept constant, but they may also be adjusted using RTO or NCO tracking. In any case, a good choice of CVs will reduce the frequency of setpoint changes by RTO or NCO tracking. When selecting self-optimizing CVs, a set of disturbances has to be assumed, as unexpected disturbances are not rejected in SOC. On the other hand, RTO and NCO tracking adapt the inputs at given sample times without any assumptions on what disturbances occur. It is only assumed that they occur on a slower time scale than the sampling. Disturbances with high frequencies or which which do not lead to a steady state are not rejected optimally. By using NCO tracking in the optimization layer and SOC in the control layer below, we demonstrate that the advantages of both methods complement each other. This combination allows fast optimal action for the expected disturbances, while other disturbances are compensated by NCO tracking on a slower time scale.

Keywords: Self-optimizing control, Real-time optimization, NCO tracking, Optimal operation,

1. INTRODUCTION

In recent years there has been a plethora of contributions on optimal operation in literature. Several approaches and strategies have been independently of each other. Optimal operation methods may be categorized by how the control strategy is determined:

- Model used on-line (on-line optimization)
- Model used off-line
- Model not used at all

On-line optimization makes use of a model which is usually updated using measurements and which is optimized on-line in real-time to minimize a predefined cost function. This is typically known as real-time optimization (RTO).

In the off-line approach, expensive online computations are avoided, and optimal operation is achieved by designing a "smart" control structure. This controlled variable (CV) selection has the objective to transform the economic objectives into control objectives (Morari et al. (1980)). A process model is used to support decision making in control structure design, but it will not be used for online optimization. Self-optimizing control (Skogestad (2000)) belongs into this category.

A third strategy avoids using a process model at all, but uses measurements in order to obtain gradient information about the process. This information is used to update the inputs directly in order to obtain optimal operation. NCO tracking (François et al. (2005)) and extremum seeking control (Krstic and Wang (2000)) represent this category.

This paper focuses on the two last categories, and is structured as follows: In the next section we present two methods for implementing optimal operation, NCO tracking (François et al. (2005)) which does not rely on a process model, and the null-space (Alstad and Skogestad (2007)) as a representative of self-optimizing control (using the model off-line). We discuss some properties of self-optimizing control and NCO tracking, and their relation to each other. Based on the results we propose to consider the methods *complementary* and to use them together. NCO tracking is used in the RTO layer, while self-optimizing control is used in the lower, dynamic control layer. To illustrate the presented ideas, we use a dynamic model of a CSTR from Economou and Morari (1986) and apply the described methods to it.

2. OPTIMAL OPERATION METHODS

In many cases, steady state operation accounts for the largest part of the operating cost, and for these processes significant economical improvements can be achieved by operating the plant optimally at steady state. Therefore we choose to focus on steady state optimization in this work. We formulate the problem of achieving optimal operation as

$$\min_{\mathbf{u}} J(\mathbf{u}, \mathbf{d}) \quad \text{s.t.} \quad \begin{cases} \text{plant and} \\ \text{constraints} \end{cases} \tag{1}$$

where $\mathbf{u}$ is the vector of adjustable input variables (e.g. valve opening, pump speed or the setpoint signal to the regulatory control system), and $\mathbf{d}$ is a vector of unknown disturbance parameters. For the rest of the paper we assume that the active constraints have been implemented and that (1) can be re-written as an unconstrained optimization problem.

2.1 Self-optimizing control using the null space method

The idea of self-optimizing control has been formulated by Skogestad (2000):

> Self-optimizing control is when we can achieve an acceptable loss with constant setpoint values for the controlled variables (without the need to re-optimize when disturbances occur).

Consider the case when the self-optimizing variable $\mathbf{c}$ is a linear combination of measurements $\mathbf{y}$,

$$\mathbf{c} = \mathbf{H}\mathbf{y}, \tag{2}$$

where $\mathbf{H}$ is the constant measurement selection matrix. In the null-space method (Alstad and Skogestad, 2007), we approximate the original optimization problem (1) locally by a quadratic optimization problem.

Theorem 1. (Alstad et al. (2009)). Given a sufficient number of measurements ($n_y \geq n_u + n_d$) and no measurement noise, select $\mathbf{H}$ in the null space of the optimal sensitivity matrix $\mathbf{F}$,

$$\mathbf{H}\mathbf{F} = 0, \tag{3}$$

where

$$\mathbf{F} = \frac{\partial \mathbf{y}^{\text{opt}}}{\partial \mathbf{d}}. \tag{4}$$

Controlling $\mathbf{c} = \mathbf{H}\mathbf{y}$ to zero yields locally zero loss from optimal operation.

The optimal sensitivity $\mathbf{F}$ can be obtained numerically or calculated using the shorthand notation $J_u = \partial J/\partial \mathbf{u}$, $\mathbf{J}_{ud} = \partial^2 J/(\partial \mathbf{u} \partial \mathbf{d})$ and $\mathbf{J}_{uu} = \partial^2 J/\partial \mathbf{u}^2$ as

$$\mathbf{F} = -\mathbf{G}^y \mathbf{J}_{uu}^{-1} \mathbf{J}_{ud} + \mathbf{G}_d^y, \tag{5}$$

where we use a linearized process model $\mathbf{y} = \mathbf{G}^y \Delta\mathbf{u} + \mathbf{G}_d^y \Delta\mathbf{d}$ (Alstad et al., 2009).

We sketch the proof of the null space theorem: In a neighbourhood of the nominal point, the optimal change in the measurements can be expressed as

$$\mathbf{y}^{\text{opt}}(\mathbf{d}) - \mathbf{y}^{\text{opt}}(\mathbf{d}^{\text{nom}}) = \mathbf{F}(\mathbf{d} - \mathbf{d}^{\text{nom}}). \tag{6}$$

The optimal variation in the controlled variables $\mathbf{c}$ then becomes

$$\mathbf{c}^{\text{opt}}(\mathbf{d}) - \mathbf{c}^{\text{opt}}(\mathbf{d}^{\text{nom}}) = \mathbf{H}\mathbf{F}(\mathbf{d} - \mathbf{d}^{\text{nom}}), \tag{7}$$

and since $\mathbf{H}$ is chosen in left null space of $\mathbf{F}$, the optimal variation $\mathbf{c}^{\text{opt}}(\mathbf{d}) - \mathbf{c}^{\text{opt}}(\mathbf{d}^{\text{nom}}) = 0$. □

Another approach is to use the insight that at optimal operation the gradient should be zero, $J_u = 0$ (first order necessary optimality condition). Thus, an ideal self-optimizing control variable is to select $\mathbf{c} = J_u$. We show in Appendix A that choosing $\mathbf{H}$ in the null space of $\mathbf{F}$ is in fact identical to selecting $\mathbf{c} = J_u$.

2.2 NCO tracking

A different approach, followed by François et al. (2005) is the NCO tracking scheme. It is well known, that the Karush-Kuhn-Tucker (KKT) necessary conditions for optimality (NCO) of the plant must hold at the optimal operating point. If a disturbance enters the process, the control scheme is used to adapt the inputs stepwise in such a way, that the NCO are satisfied. We do not present the general NCO tracking procedure (with constraints) here, but we rather give a derivation of the special case without constraints, i.e. only the sensitivity seeking directions. Then the optimization problem in consideration is

$$\min J(\mathbf{u}, \mathbf{d}), \tag{8}$$

and the necessary condition for optimality are:

$$J_u(\mathbf{u}) = 0 \tag{9}$$

To achieve optimal operation, we update the input $\mathbf{u}$ at each sample time k using the update equation

$$\mathbf{u}_{k+1} = \mathbf{u}_k + \Delta\mathbf{u}, \tag{10}$$

until (9) is satisfied. To obtain the update term $\Delta\mathbf{u}$, we linearize (9) around the current operating point $\mathbf{u}_k$:

$$J_u(\mathbf{u}_k + \Delta\mathbf{u}) = J_u(\mathbf{u}_k) + \mathbf{J}_{uu}(\mathbf{u}_k)\Delta\mathbf{u} \tag{11}$$

Since we want the update $\Delta\mathbf{u}$ to force the sensitivity to zero, we set the left hand side of (11) to zero and solve for $\Delta\mathbf{u}$ (François et al., 2005).

$$\Delta\mathbf{u} = -\mathbf{J}_{uu}^{-1}(\mathbf{u}_k) J_u(\mathbf{u}_k) \tag{12}$$

This Newton update step is exact for a quadratic approximation of the system (8), in the sense that the NCO (9)are satisfied after one iteration. In practice we do not apply the full update step $\Delta\mathbf{u}$, because this may lead to feasibility and convergence problems as the process can move outside the region where the quadratic approximation is valid. To avoid this, the update term $\Delta\mathbf{u}$ is multiplied by some tuning parameter $\beta \in [0\,1]$, such that $\mathbf{u}_{k+1} = \mathbf{u}_k + \beta\Delta\mathbf{u}$.

To evaluate (12) we need the derivative $J_u(\mathbf{u}_k)$ for a given input $\mathbf{u}_k$. In this work it is chosen to make a small perturbation in the input and to run the process for a given time to estimate the gradient by finite differences. The magnitude of the perturbation is desired to be small in order not to upset the process excessively. At the same time it has to be larger than the process noise to yield sufficient information about the descent direction.

Since the Hessian $\mathbf{J}_{uu}(\mathbf{u}_k)$ is difficult to obtain, it is often determined once at the nominal operating point. Alternatively, as we choose to do in this work, an approximation of the inverse of the Hessian can be obtained by a BFGS update scheme. The NCO tracking algorithm is summarized in Fig. 1. This procedure is analog to a Newton(like) method in optimization. In the analogy, the steady state operating periods correspond to function evaluations in the newton procedure, and the solution is found when the NCO hold.

3. RELATIONSHIP BETWEEN SELF-OPTIMIZING CONTROL AND NCO TRACKING

Both methods pursue the same goal, minimization of the operating cost. However, in NCO tracking, we focus on manipulating the *input values* (at given sample times) to force the sensitivities to zero. Here it should be noted that the NCO tracking is more general as described above, since the gradient may be obtained by several different (statistical) methods, such as correlation methods

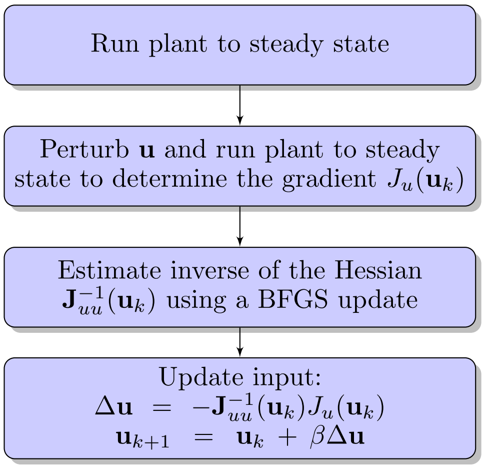
Run plant to steady state
Perturb $\mathbf{u}$ and run plant to steady state to determine the gradient $J_u(\mathbf{u}_k)$
Estimate inverse of the Hessian $\mathbf{J}_{uu}^{-1}(\mathbf{u}_k)$ using a BFGS update
Update input: $\Delta\mathbf{u} = -\mathbf{J}_{uu}^{-1}(\mathbf{u}_k)J_u(\mathbf{u}_k)$ $\mathbf{u}_{k+1} = \mathbf{u}_k + \beta\Delta\mathbf{u}$

Fig. 1. Simple NCO tracking procedure

or black-box models. In some cases, the gradient may even be measured directly. Once the gradient is available, it is used to calculate the input change needed to push the sensitivities to zero.

In self-optimizing control, on the other hand, we focus on finding *controlled variables* which give "good" performance in practice. The system is "solved" for the correct inputs indirectly, by feedback and PID controllers. Controlling $\mathbf{c} = \mathbf{H}\mathbf{y}$ using $\mathbf{H}$ from the null-space method, as described above, is only one particular example of self-optimizing control. In general we may impose structural constraints on $\mathbf{H}$ yielding $\mathbf{H}\mathbf{F} \neq 0$. Then the self-optimizing variable $\mathbf{c}$ is no longer identical to J_u. However, from a dynamic point of view it may be desirable to compromise on the steady state optimality in favor of good controllability.

In table 1 we summarize the main differences between the null space method and NCO tracking as presented in the previous section. These differences lead us to propose to use the different methods in different layers of the overall plant control structure.

Table 1. Summary of properties

null space method	NCO tracking
• locally valid (linearization) • model used off-line • continuous input change by PID controller • fast reaction (due to instantaneous PID feedback) • only rejects expected disturbances	• globally valid (on convergence) • no model used, gradient measured experimentally • discrete updates at sample times • slow reaction (only at sample times)

The control structure of a chemical plant is usually divided into different layers, according to the different time scales the control operate in. At the top of the hierarchical structure we have a real time optimization system (RTO), which updates the setpoint values for the layer below at given sample times. Usually the RTO system optimizes a large nonlinear steady state model. Because of the steady state solution of the RTO and the infrequent updates, the output of the RTO system is generally not directly applied on the process, but is implemented by a dynamical control system, such as a model predictive controller (MPC) or a decentralized PID control system, Fig. 2.

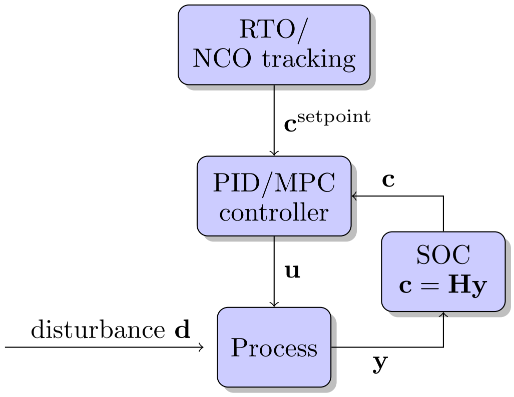

Fig. 2. Relation between NCO tracking and self-optimizing control

In this dynamic control layer below the RTO, we see the place of self-optimizing control. All plant measurements are sent to a "selection unit", where the controlled variables are calculated according to $\mathbf{c} = \mathbf{H}\mathbf{y}$. These controlled variables $\mathbf{c}$ enter the dynamic PID or MPC controller, which manipulates the plant inputs $\mathbf{u}$ to keep them at their set-points (given from the RTO layer above).

This type of control structure is very common in industry, although usually with some modifications. Many plants do not have a RTO layer, and the set-points for PID/MPC controllers are set from experience or engineering insight. A common realization of this structure is to choose $\mathbf{H}$ such that every row contains only a single "1". This is the special case of conventional decentralized control.

In many cases, the self-optimizing control structure needs a RTO system above. The reason for this is the limitation imposed by the local nature of obtaining the measurement selection matrix. When selecting $\mathbf{H}$ in the null space of $\mathbf{F}$, a local approximation of the process is obtained based on expected disturbances. Unexpected disturbances and disturbances that move the process far away from the linearization point, cannot be compensated by the null-space method. They have to be counteracted by re-optimization of the system.

It may be argued that if an RTO system is in place, there is no need to select $\mathbf{H}$ to give a self-optimizing control structure because the setpoints are updated by the RTO system. However there are at least two reasons to choose $\mathbf{H}$ as a self optimizing variable combination: 1. Choosing a self-optimizing variable combination enables a faster optimal reaction to assumed (main) disturbances, not only at sample times. 2. RTO has to change the setpoints less frequently, that is only in case of large disturbances (outside region of linearization), and for unknown disturbances.

These considerations lead us to see self-optimizing control and NCO tracking as *complementary*, and to use them together. NCO tracking is used in the optimization layer and is thus an alternative to real-time optimization (RTO), while self-optimizing control is located in the lower layer, as in Fig. 2. In this configuration the manipulated variables for the NCO tracking layer are the setpoints for the self-optimizing control layer.

4. SIMULATIONS

4.1 Model

To illustrate the statements above, we present simulation results for a dynamic CSTR with a feed stream F containing mainly the component an and a reversible chemical reaction $A \rightleftharpoons B$, Fig. 3. The process model is taken from Economou and Morari (1986), and the dynamics of the system are described by following set of equations:

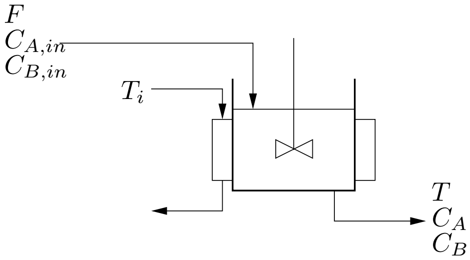

Fig. 3. Schematic diagram of a CSTR

$$\frac{dC_A}{dt} = \frac{1}{\tau}(C_{A,in} - C_A) - r \tag{13}$$

$$\frac{dC_B}{dt} = \frac{1}{\tau}(C_{B,in} - C_B) + r \tag{14}$$

$$\frac{dT}{dt} = \frac{1}{\tau}(T_i - T) + \frac{-\Delta H_{rx}}{\rho C_p} r \tag{15}$$

The reaction rate r is defined by

$$r = k_1 C_A - k_2 C_B \tag{16}$$

where

$$k_1 = C_1 e^{\frac{-E_1}{RT}} \quad \text{and} k_2 = C_2 e^{\frac{-E_2}{RT}}. \tag{17}$$

This process has one manipulated input ($\mathbf{u}$), the jacket temperature T_i. The expected disturbances enter the process as variations in the feed concentrations $C_{A,in}, C_{B,in}$, and the measured variables are C_A, C_B, T, and the input T_i.

The objective is to maximize the profit function which is a trade-off between heating cost and income from selling product B:

$$P = [p_{C_B} C_B - (p_{T_i} T_i)^2], \tag{18}$$

Here p_{C_B} is the price of the desired product B and p_{T_i} is the cost for heating the reactor. The parameter values are given in table 2, and the nominal operation values for all variables are listed in table 3.

Parameter	Value
p_{C_B}	2.009
p_{T_i}	$1.657 \cdot 10^{-3}$

Table 2. Objective function parameters

4.2 Simulations

First, we control the process for the expected disturbances using direct NCO tracking. Next we use the null space method and compare the results with direct NCO tracking. After comparing both control structures for an unexpected disturbance we finally combine the methods as in Fig.2.

Variable	Value	Unit	Description
F	1	holdup min^{-1}	Flow rate
C_1	5000	s^{-1}	Arrhenius factor 1
C_2	10^6	s^{-1}	Arrhenius factor 2
C_p	1000	cal kg^{-1}K^{-1}	Heat capacity
E_1	10^4	cal mol^{-1}	Activation energy 1
E_2	15000	cal mol^{-1}	Activation energy 2
R	1.987	cal mol^{-1}K^{-1}	Ideal gas constant
T_i	–	K	Input
$C_{A,in}$	1	mol/l	Conc. A in feed
$C_{B,in}$	0	mol/l	Conc. B in feed
$-\Delta H_{rx}$	5000	cal mol^{-1}	Heat of reaction
ρ	1	kg/l	Density
τ	1	min	Time constant

Table 3. Nominal values for the CSTR model

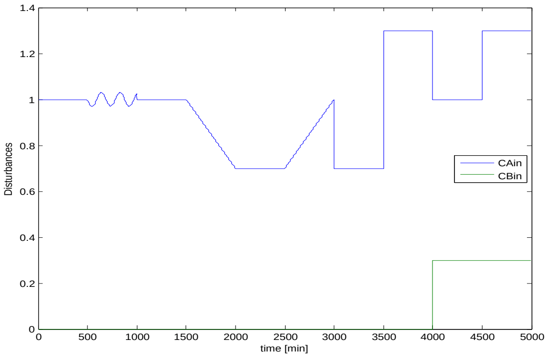

Fig. 4. Disturbance trajectories $C_{A,in}, C_{b,in}$

The expected disturbance scenario is given in Fig. 4. After 500 minutes at the nominal value, the concentration $C_{A,in}$ varies sinusoidal before returning to the nominal value. Then ramp disturbances are introduced, followed by large step disturbances. At 4000 minutes, the concentration $C_{B,in}$ makes a step change of 0.3 mole/l. The non-steady state periods (sinusoid and ramp) are included to test how the controller behaves in this case, which is likely to happen in reality. Note that strictly speaking, the gradient is not defined, as the process is not at steady state.

Direct NCO tracking To obtain the gradient information, the input T_i is perturbed with a step of size 1 K. Starting with a positive value, the sign is altered every forth NCO iteration. Changing the sign of the perturbation was found to give better overall performance of the NCO procedure. No steady state detection is implemented in the NCO tracking procedure. Instead, a step test is used to determine the approximate time for the system to settle down to a new steady state. At the nominal point, the system has a time constant of less than two minutes for an input step of $\Delta T_i = 5$ K. To let the system settle down far from the nominal point, where the system dynamics are different, a sample time of 10 minutes is chosen for the direct NCO tracking procedure. The step size parameter β is set to 0.4.

Fig. 5 shows the concentration and temperature trajectories for the NCO tracking procedure. The control strategy enables stable control. It is furthermore found that the step disturbances are very well handled. Since the method assumes steady state after 10 minutes, and uses the results at each sample time for calculating the input update, it has difficulties handling disturbances which do not lead to a steady state (sinusoidal and ramp). However, the controller manages to keep the system stable during these periods. It is found that the performance of NCO tracking algorithm is very sensitive to the tuning parameter β, the sample time, and the timing and kind of the disturbance.

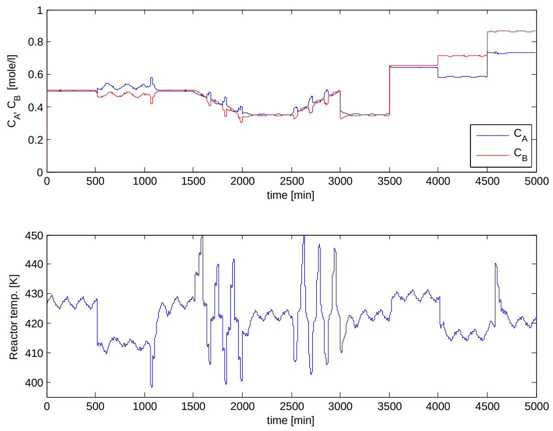

Fig. 5. NCO tracking, concentrations and temperature

Self-optimizing control using the null-space method Next, the process is controlled using the null space method from section 2.1. Since we have one input and 2 disturbances to compensate for, we need three measurements for the invariant variable combination. We choose two concentrations and the reactor temperature, so $\mathbf{y} = [\,C_A\ C_B\ T\,]^{\mathrm{T}}$. We optimize the steady state system at the nominal operating point and then introduce small perturbations in the disturbance variables $\mathbf{d} = [C_{A,in}\, C_{B,in}]^{\mathrm{T}}$. After re-optimizing we calculate

$$\mathbf{F} = \frac{\partial \mathbf{y}^{\mathrm{opt}}}{\partial \mathbf{d}} = \begin{bmatrix} -0.4862 & -0.3223 \\ -0.5138 & -0.6777 \\ -9.9043 & 40.5807 \end{bmatrix}. \qquad (19)$$

Then $\mathbf{H} = [\,-0.7688\ 0.6394\ 0.0046\,]$, and $\mathbf{HF} = 0$. Using a PID controller, the self-optimizing variable $\mathbf{c} = \mathbf{Hy}$ is controlled to zero. The concentration and temperature trajectories with self-optimizing control are plotted in Fig. 6.

Comparing inputs and profit for NCO tracking and self-optimizing control The input usage for the two cases described above is quite differently, Fig. 7. While the NCO tracking procedure needs large input variations to control the process, the input usage of the self-optimizing control structure is very moderate and smooth.

Comparing the profits, Fig. 8, shows that both systems perform quite similar to each other at steady state periods, but for disturbances, where no steady state is reached within one sample time, NCO tracking is not performing as good as the self-optimizing control policy using the null-space method.

Using NCO tracking as RTO and self-optimizing control in the lower layer If it can be guaranteed that the disturbances in the feed concentration are the only ones entering the process, then using only self-optimizing control is sufficient, and a RTO layer is not necessary. However, the situation changes for disturbances not anticipated in the control structure design. Consider a positive step change in the activation energy E_2 of 10% after 3200 min. This disturbance hinders the reverse reaction. Comparing the profits using the two control structures, Fig. 9, shows that the self-optimizing control system cannot exploit the improved conditions caused by the unexpected disturbance.

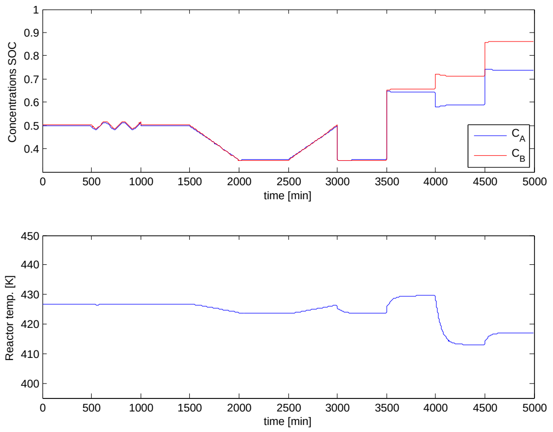

Fig. 6. SOC, concentrations and temperature

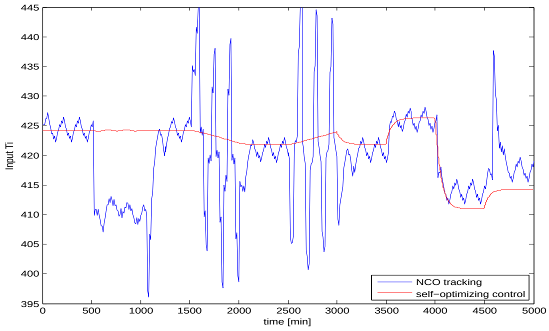

Fig. 7. Input usage for SOC and NCO tracking

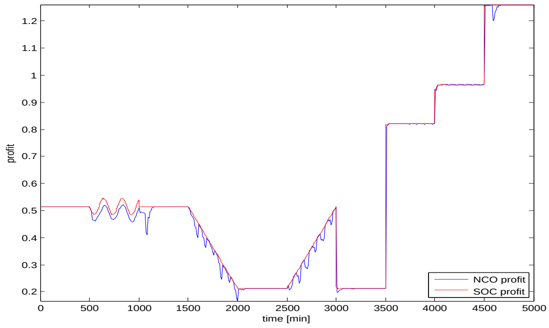

Fig. 8. Profit for SOC and NCO tracking

Adapting the self-optimizing control setpoints by RTO or NCO can solve this problem, and at the same time reduce RTO/NCO sample time. In Fig. 10 the instantaneous profit for direct NCO tracking (sample time: 10 min) and the combined system with a sample time of 25 min is shown. The combined system operates smoother than the

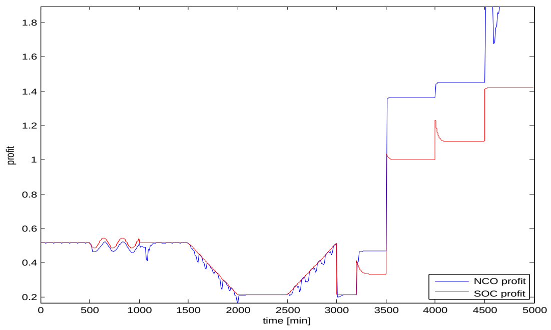

Fig. 9. Profit, direct NCO tracking and SOC. Note: unexpected disturbance occurs after 3200 min

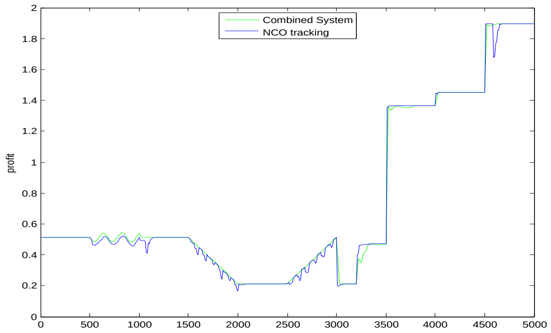

Fig. 10. Profit, combined NCO/SOC and direct NCO tracking

pure NCO system while giving similar performance. In particular, the input is used less aggressive, Fig. 11.

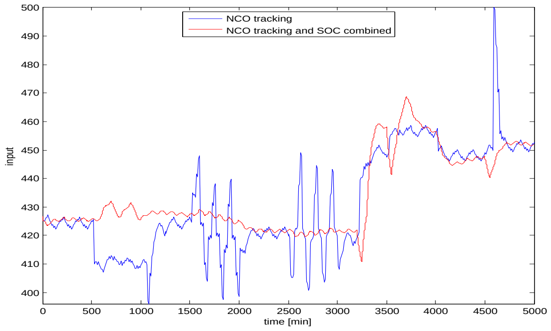

Fig. 11. Input, combined NCO/SOC and direct NCO tracking

Using online RTO the performance could be improved even further because the setpoints move directly to the optimal values instead of iteratively approaching them.

5. CONCLUSION

Due to the limitations of self-optimizing control it can generally not replace an RTO system. However, combining RTO (NCO tracking) and self-optimizing control overcomes the limitations of the local nature of self-optimizing control. At the same time we have fast, dynamic steady state optimal control for the expected disturbances.

Since almost every RTO system has a dynamic control system in the layer below, using a self-optimizing control structure in the lower layer, improves performance and can significantly reduce the RTO updates. For NCO tracking this means less perturbations for gradient estimation. For an online RTO, this means more time for complex, time intensive, computations.

REFERENCES

Alstad, V. and Skogestad, S. (2007). Null space method for selecting optimal measurement combinations as controlled variables. *Ind. Eng. Chem. Res.*, 46, 846–853.

Alstad, V., Skogestad, S., and Hori, E. (2009). Optimal measurement combinations as controlled variables. *Journal of Process Control*, 19(1), 138–148.

Economou, C.G. and Morari, M. (1986). Internal model control. 5. extension to nonlinear systems. *Ind. Eng. Chem. Process Des. Dev.*, 25, 403–411.

François, G., Srinivasan, B., and Bonvin, D. (2005). Use of measurements for enforcing the necessary conditions of optimality in the presence of constraints and uncertainty. *Journal of Process Control*, 15(6), 701 – 712. doi:DOI: 10.1016/j.jprocont.2004.11.006.

Krstic, M. and Wang, H.H. (2000). Stability of extremum seeking feedback for general nonlinear dynamic systems. *Automatica*, 36(4), 595 – 601. doi:DOI: 10.1016/S0005-1098(99)00183-1.

Morari, M., Stephanopoulos, G., and Arkun, Y. (1980). Studies in the synthesis of control structures for chemical processes. part i: formulation of the problem. process decomposition and the classification of the control task. analsysis of the optimizing control structures. *AIChE Journal*, 26(2), 220–232.

Skogestad, S. (2000). Plantwide control: The search for the self-optimizing control structure. *Journal of Process Control*, 10, 487–507.

Appendix A. CONNECTION BETWEEN THE GRADIENT AND THE NULL SPACE METHOD

Consider the unconstrained optimization problem

$$\min_{\mathbf{u}} J(\mathbf{u},\mathbf{d}) = \min_{\mathbf{u}} [\mathbf{u}^T\ \mathbf{d}^T] \begin{bmatrix} \mathbf{J}_{uu} & \mathbf{J}_{ud} \\ \mathbf{J}_{du} & \mathbf{J}_{dd} \end{bmatrix} \begin{bmatrix} \mathbf{u} \\ \mathbf{d} \end{bmatrix}. \qquad \text{(A.1)}$$

In Alstad et al. (2009) it is shown, that if the number of measurements is equal to the sum of inputs and disturbances,$n_y = n_u + n_d$, and the measurement map

$$\mathbf{y} = \tilde{\mathbf{G}}^y \, [\, \mathbf{u}\ \mathbf{d} \,]^T, \qquad \text{(A.2)}$$

is invertible, then we can express $\mathbf{H}$ as

$$\mathbf{H} = \mathbf{M}_n^{-1} \left[\mathbf{J}_{uu}^{\frac{1}{2}} \quad \mathbf{J}_{uu}^{1/2} \mathbf{J}_{uu}^{-1} \mathbf{J}_{ud} \right] \left[\tilde{\mathbf{G}}^y \right]^{-1}, \qquad \text{(A.3)}$$

where $\mathbf{M}_n$ can be freely chosen and is usually set to the identity matrix, $\mathbf{M}_n = \mathbf{I}$. In this case, however, we choose to set $\mathbf{M}_n = \mathbf{J}_{uu}^{\frac{1}{2}}$.

This gives

$$\mathbf{H} = [\mathbf{J}_{uu} \quad \mathbf{J}_{ud}] [\tilde{\mathbf{G}}^y]^{-1}, \qquad \text{(A.4)}$$

Calculating the gradient of problem (A.1),

$$\mathbf{J}_u = [\, \mathbf{J}_{uu}\ \mathbf{J}_{ud} \,] \begin{bmatrix} \mathbf{u} \\ \mathbf{d} \end{bmatrix}, \qquad \text{(A.5)}$$

and inserting (A.2), we obtain

$$\mathbf{J}_u = [\, \mathbf{J}_{uu}\ \mathbf{J}_{ud} \,] \, [\tilde{\mathbf{G}}^y]^{-1} \mathbf{y} = \mathbf{H}\mathbf{y} = \mathbf{c}. \qquad \text{(A.6)}$$

Comparing with (A.4), we see that the null space method is identical to controlling the gradient.