On the Equivalence between Neighboring-Extremal Control and Self-Optimizing Control for the Steady-State Optimization of Dynamical Systems

Grégory François[†], B. Srinivasan[+], and Dominique Bonvin[†*]

[†] *Laboratoire d'Automatique, École Polytechnique Fédérale de Lausanne*

CH-1015 Lausanne, Switzerland.

[+] *Department of Chemical Engineering, Ecole Polytechnique de Montreal*

Montreal, Canada, H3C 3A7.

E-mail: dominique.bonvin@epfl.ch

Abstract

The problem of steering a dynamical system toward optimal steady-state performance is considered. For this purpose, a static optimization problem can be formulated and solved. However, because of uncertainty, the optimal steady-state inputs can rarely be applied directly in an open-loop manner. Instead, plant measurements are typically used to help reach the plant optimum. This paper investigates the use of optimizing control techniques for input adaptation. Two apparently different techniques of enforcing steady-state optimality are discussed, namely, neighboring-extremal control and self-optimizing control based on the null-space method. These two techniques are compared for unconstrained real-time optimization in the presence of parametric variations. It is shown that, for the noise-free scenario, the two methods can be made equivalent through appropriate tuning. Note that both approach can use

[*]To whom correspondence should be addressed

measurements that are taken either at successive steady-state operating points or during the transient behavior of the plant. Implementation of optimizing control is illustrated through a simulated CSTR example.

Introduction

Process optimization has received significant attention in the last 30 years. Long considered an appealing research tool for design and operation, optimization has become a credible and viable *technology*[1] that is used extensively and routinely in industry[2]. In practice, optimization is complicated by the presence of uncertainty in the form of plant-model mismatch, parametric uncertainty and unknown disturbances. Uncertainty can be very detrimental to optimality, as any model-based optimization approach tries to push the plant as much as possible based on the available model.

An efficient way to combat the effect of uncertainty is to use plant measurements to either (i) adapt the model parameters and re-optimize on the basis of the updated model (*explicit optimization*)[3,4], or (ii) adapt the plant inputs directly (*implicit optimization*). Implicit optimization typically uses one of the following schemes:

1. Search (zeroth-order) methods – In techniques labeled *evolutionary optimization*[5], a simplex-type algorithm is used to approach the optimum. The cost function is measured experimentally for various combination of the operating conditions.

2. Perturbation (first-order) methods – In techniques labeled *extremum-seeking control*[6,7], the gradients are estimated experimentally using sinusoidal excitation. The excitation frequency has to be sufficiently low for a time-scale separation between the system dynamics and the excitation frequency to exist. Like the techniques of the first type, this scheme uses only cost measurements.

3. Control methods – In techniques such as *NCO tracking*[8] and *self-optimizing control*[9], the optimization problem is recast as a problem of choosing and tracking variables whose optimal values are invariant, or nearly invariant, to uncertainty. If these variables vary, for

example due to disturbances, their measured or estimated values are simply brought back to their invariant set points using feedback control. In contrast to the other schemes, the measurements in the control methods are *not* the cost function but auxiliary measurements such as the process inputs and outputs.

RTO techniques based on "control" will be discussed in this paper. The control action can be applied at discrete points in time based on steady-state measurements and computation of updated inputs. The inputs are then applied to the plant and held constant until the next steady-state measurements become available. Alternatively, the inputs can be updated continuously on the basis of transient measurements. No distinction is made here on whether the transient measurements are available in discrete or continuous time. The important factor is that these measurements are taken before the plant reaches steady state.

Most "control" approaches rely on the necessary conditions of optimality (NCO) as these conditions are invariant to uncertainty. The NCO for a constrained optimization problem has two parts: the feasibility and the sensitivity part. These two parts require different types of measurements (constraint values vs. cost and constraint gradients) and thus are often considered separately[8]. However, the plant inputs typically affect both parts. One solution to this problem consists in using input separation to generate two decoupled problems, namely, a constraint-tracking problem and a sensitivity-reduction problem. The sensitivity of the active constraints with respect to the various inputs can be used to separate the input space in a subspace that affects the active constraints and a complementary subspace that does not. This input separation defines so-called *constraint*- and *sensitivity-seeking* directions[8]. This paper addresses only the sensitivity part of the NCO.

Two methods for enforcing steady-state optimality using measurements will be presented. The first method, neighboring-extremal control (NEC), implements first-order optimality corrections using state feedback[10]. It has been shown that NEC is a first-order approximation to gradient-based optimization[11]. The second method is self-optimizing control (SOC) based on the null-space method, which proposes to determine CVs that ensure marginal optimality loss when maintained at their nominal optimal values[9,12]. It has been shown that a particular choice of CVs leads to an estimation of the cost gradient[13], which is precisely the focus of NEC. Although the two approaches attempt to solve the same static optimization problem, they are still sometimes seen as being different[13], which is probably due to the fact that they were introduced with different types of measurements, namely, at steady state for NEC[11] and in the transient for SOC[9]. By presenting the two approaches on the same footing, this paper shows that they are indeed very similar and, in fact, the two methods can be made strictly equivalent through appropriate tuning. The implementation issue with either steady-state or transient measurements is also discussed. Note that the links between NEC, SOC and other RTO schemes has also been the subject of a recent publication[14].

The paper is organized as follows. The section Preliminaries introduces the dynamical system and the static optimization problem associated with maximizing steady-state performance. The next section describes the NEC and SOC algorithms that are used to solve a static optimization problem and discusses their implementation using either steady-state or transient measurements. The two techniques are formally compared in the section Comparison Between NEC and SOC and illustrated on a simulated CSTR example in the section Illustrative Example. Finally, the last section concludes the paper.

Preliminaries

Optimality can be implemented by enforcing the plant NCO, namely the active constraints and the reduced gradients. This way, the optimization problem is formulated as a multivariable feedback control problem. The focus of this paper is on forcing the cost gradient of the plant to zero, and not on meeting plant constraints. We will therefore assume that the active constraints are known and enforced using feedback control, thus resulting in an *unconstrained* optimization problem.

Dynamical System

We consider the following dynamical system:

$$\dot{x}(t) = F(x(t), u(t), \theta) \tag{1}$$

$$y(t) = H(x(t), u(t), \theta), \tag{2}$$

where $x \in \Re^{n_x}$ represent the states, $u \in \Re^{n_u}$ the inputs, $y \in \Re^{n_y}$ the outputs, and $\theta \in \Re^{n_\theta}$ the vector of uncertain parameters. The time dependency of the variables, that is, $x(t)$, $u(t)$ and $y(t)$, will be used to indicate that the system is in a transient state. In contrast, the steady-state behavior will be expressed by the variable without explicit time dependency, namely, x, u and y. F and H are smooth functions that represent the state and output functions, respectively.

This study assumes no plant-model mismatch and no measurement errors. The emphasis will be on the comparison of the proposed NEC and SOC techniques. Obviously, the effect of plant-model mismatch and measurement errors is of importance and could be the subject of further investigation.

Static Optimization Problem and Optimality Conditions

Consider the following unconstrained static optimization problem:

$$\min_u J(u) = \varphi(x, u, \theta) \tag{3}$$

$$\text{s.}t. \quad F(x, u, \theta) = 0, \tag{4}$$

where J is the cost to be minimized and φ is a smooth function that represents the cost. At steady state, the output equations read:

$$y = H(x, u, \theta). \tag{5}$$

As indicated above, the variables x, u, and y represent the states, inputs and outputs at steady state.

Introducing the Lagrangian $L(x,u,\lambda,\theta) := \varphi + \lambda^T F$, where λ represents the adjoints, and the notation $a_b := \frac{\partial a}{\partial b}$ of dimension (dim a)$\times$(dim b), the NCO for Problem (3)-(4) are:

$$L_u = \varphi_u + \lambda^T F_u = 0_{1\times n_u} \quad (6)$$

$$L_x = \varphi_x + \lambda^T F_x = 0_{1\times n_x} \quad (7)$$

$$L_\lambda = F^T = 0_{1\times n_x}. \quad (8)$$

Note that equation (8) is the same as equation (4). Assuming F_x to be invertible, the adjoint variables can be computed from $L_x = 0$, which gives $\lambda^T = -\varphi_x F_x^{-1}$ and

$$L_u = \varphi_u - \varphi_x F_x^{-1} F_u = \frac{d\varphi}{du} = 0_{1\times n_u}, \quad (9)$$

which simply says that the total derivative of the cost function with respect to u, that is, accounting for the direct effect of u and the effect of u through x, vanishes at the optimum. This total derivative is the *gradient* of the cost function with respect to u, which is denoted as the n_u-dimensional vector $g(x,u,\lambda,\theta) := \left(\frac{d\varphi}{du}\right)^T = J_u^T$.

First-Order Variations of the Necessary Conditions of Optimality

The two methodologies discussed in this paper rely on linear approximations around the nominal optimum, namely the first-order variations of the NCO for NEC and the sensitivity of the outputs and inputs with respect to parametric variations for SOC. The goal of this subsection is to establish preliminary results through the analysis of the first-order variations of the NCO.

Consider the parametric variations $\delta\theta$ around the nominal values of the parameters, θ_{nom}. The

NCO equations (6)-(8) can be linearized with respect to x, u, λ and θ:

$$
\begin{aligned}
\delta L_u^T &\simeq L_{ux}\,\delta x + L_{uu}\,\delta u + F_u^T\,\delta\lambda + L_{u\theta}\,\delta\theta = 0_{n_u \times 1} && (10)\\
\delta L_x^T &\simeq L_{xx}\,\delta x + L_{xu}\,\delta u + F_x^T\,\delta\lambda + L_{x\theta}\,\delta\theta = 0_{n_x \times 1} && (11)\\
\delta L_\lambda^T &\simeq F_x\,\delta x + F_u\,\delta u + F_\theta\,\delta\theta = 0_{n_x \times 1}\,, && (12)
\end{aligned}
$$

where $\delta x = x - x_{nom}$, $\delta u = u - u_{nom}$, $\delta\lambda = \lambda - \lambda_{nom}$ and $\delta\theta_{nom} = \theta - \theta_{nom}$, with x_{nom}, u_{nom} and λ_{nom} representing the states, inputs and adjoints that correspond to θ_{nom}.

The system of linear equations (10)-(12), which contains $(2n_x + n_u)$ equations for the $(2n_x + n_u + n_\theta)$ unknowns δx, $\delta\lambda$, δu and $\delta\theta$, can be solved for given values of $\delta\theta$. Indeed, δx and $\delta\lambda$ can be expressed in terms of δu and $\delta\theta$ from equations (12) and (11) as:

$$
\begin{aligned}
\delta x &= -F_x^{-1}F_u\,\delta u - F_x^{-1}F_\theta\,\delta\theta && (13)\\
\delta\lambda &= -F_x^{-T}L_{xx}\,\delta x - F_x^{-T}L_{xu}\,\delta u - F_x^{-T}L_{x\theta}\,\delta\theta\,. && (14)
\end{aligned}
$$

The cost φ is a function of x, u and θ. From the first-order variations of the NCO, δx can be expressed in terms of δu and $\delta\theta$ as in equation (13), which allows expressing the cost variation in terms of δu and $\delta\theta$ as the function $\delta\phi(\delta u, \delta\theta)$.

Optimal Gradient

Equation (9) indicates that the gradient vanishes at the optimum. Equation (10) expresses that the gradient needs to be kept at zero to maintain (first-order) optimality. Upon inserting the expressions for δx and $\delta\lambda$ given in (13) and (14) into equation (10), the gradient condition for optimality reads:

$$
g_{opt}(\delta u, \delta\theta) = \mathscr{A}\;\delta u + \mathscr{B}\;\delta\theta = 0_{n_u \times 1}\,, \qquad (15)
$$

with

$$\mathscr{A} \;:=\; L_{uu} - L_{ux}F_x^{-1}F_u - F_u^T F_x^{-T} L_{xu} + F_u^T F_x^{-T} L_{xx} F_x^{-1} F_u = \frac{d^2\phi}{du^2}\,, \tag{16}$$

$$\mathscr{B} \;:=\; L_{u\theta} - L_{ux}F_x^{-1}F_\theta - F_u^T F_x^{-T} L_{x\theta} + F_u^T F_x^{-T} L_{xx} F_x^{-1} F_\theta = \frac{d^2\phi}{du\;d\theta}\,, \tag{17}$$

with the $(n_u \times n_u)$ Hessian matrix $\mathscr{A}$, assumed here to be regular, and the $(n_u \times n_\theta)$ matrix $\mathscr{B}$.

Equation (15) can be used to express the variation δu that is necessary to offset the effect of the disturbance $\delta\theta$, namely:

$$\delta u = \mathscr{C}\;\delta\theta, \tag{18}$$

with the $(n_u \times n_\theta)$ matrix $\mathscr{C} := -\mathscr{A}^{-1}\,\mathscr{B}$. Hence, if the parametric variations $\delta\theta$ were known, it would be straightforward to compute the input corrections δu to keep the gradient equal to zero despite parametric disturbances. However, since $\delta\theta$ is typically unknown, the challenge will be to infer it from the known and measured quantities δu and δy. NEC and SOC differ in the way this is done.

Static Real-Time Optimization via NEC and SOC

Neighboring-Extremal Control

NEC attempts to maintain process optimality in the presence of disturbances through appropriate state feedback[10,15]. The technique, which has been revisited recently to handle parametric uncertainty and output feedback[11], uses the first-order variations of the NCO to compute δu in terms of the parametric disturbances $\delta\theta$ as given in equation (18). More specifically, NEC relies on the *implicit* estimation of $\delta\theta$ from δy and δu, which is described next. The approach is illustrated here for solving the unconstrained static optimization problem (3).

For this, the output equations (5) are linearized with respect to x, u and θ:

$$\delta y = H_x \delta x + H_u \delta u + H_\theta \delta\theta \,, \tag{19}$$

where $\delta y = y - y_{nom}$, with y_{nom} representing the outputs that correspond to θ_{nom}.

Using δx from (13) gives:

$$\delta y \;=\; \left(H_u - H_x F_x^{-1} F_u\right)\delta u + \left(H_\theta - H_x F_x^{-1} F_\theta\right)\delta\theta = \mathscr{Q}\delta u + \mathscr{P}\delta\theta, \tag{20}$$

with the $(n_y \times n_u)$ matrix $\mathscr{Q} := \frac{dH}{du} = \left(H_u - H_x F_x^{-1} F_u\right)$ and the $(n_y \times n_\theta)$ matrix $\mathscr{P} := \frac{dH}{d\theta} = \left(H_\theta - H_x F_x^{-1} F_\theta\right)$. Note that equation (20) verifies the first-order variations of the NCO.

Let us assume $n_y \geq n_\theta$, that is, there are at least as many output measurements as there are uncertain parameters. Using (20), the parametric variations$\delta\theta$ can be inferred from δy and δu as follows:

$$\delta\theta \;=\; \mathscr{D}\left(\delta y - \mathscr{Q}\delta u\right), \tag{21}$$

where $\mathscr{D}$ is a $(n_\theta \times n_y)$ pseudoinverse of $\mathscr{P}$, that is, $\mathscr{D}\mathscr{P} = I_{n_\theta}$.[1] The feasibility of this estimation is crucial and requires $\text{rank}(\mathscr{P}) = n_\theta$, which corresponds to all uncertain parameters having a noticeable and distinct effect on the outputs y.

Equation (15) provides a first-order approximation to the cost gradient, which can be estimated from δy and δu upon using equation (21) to eliminate $\delta\theta$:

$$g = G^y \delta y + G^u \delta u \,, \tag{22}$$

with the $(n_u \times n_y)$ matrix $G^y := \mathscr{B}\mathscr{D}$ and the $(n_u \times n_u)$ matrix $G^u := \mathscr{A} - \mathscr{B}\mathscr{D}\mathscr{Q}$.

The gradient can be controlled to zero in basically two different ways, as shown next.

[1]We purposely do not choose $\mathscr{D}$ to be the unique Moore-Penrose pseudoinverse of $\mathscr{P}$ as we are interested in generating the maximum number of degrees of freedom that will be used in the comparison of NEC and SOC.

Implementation using steady-state measurements

Equations (18) and (21) can be combined to eliminate $\delta\theta$ and written in an iterative manner as:

$$\delta u_{k+1} \quad = \quad K^{y}_{NEC}\, \delta y_k + K^{u}_{NEC}\, \delta u_k, \tag{23}$$

with the $(n_u \times n_y)$ matrix $K^{y}_{NEC} := \mathscr{C}\,\mathscr{D}$ and the $(n_u \times n_u)$ matrix $K^{u}_{NEC} := -\mathscr{C}\,\mathscr{D}\mathscr{Q}$. Here, the index k indicates the k^{th} steady-state iteration, with the measurements δy_k and δu_k taken at steady state.

By combining equation (22) written for the k^{th} iteration and equation (23), the "steady-state" NEC law can be written generically as:[2]

$$\delta u_{k+1} \quad = \quad \delta u_k + K_{NEC}\, g_k, \tag{24}$$

where K_{NEC} is the $(n_u \times n_u)$ controller gain matrix. This equation is a first-order approximation to the gradient-based optimization scheme as was shown in[11]. Interestingly, equation (24) indicates that the NEC law has an integral term and is therefore able to force the estimated gradient to zero.

Remark 1

It is possible to define the generalized gradient, $\mathscr{R}g$*, by multiplying the gradient with a* $(n_u \times n_u)$ *regular matrix* $\mathscr{R}$*. This will not affect the scheme* (24) *as long as* $\mathscr{R}$ *is considered in designing the controller, that is,* $K_{NEC} := -\Gamma\mathscr{A}^{-1}\mathscr{R}^{-1}$.

[2]Formally, $K_{NEC} := -\mathscr{A}^{-1}$, where $\mathscr{A}$ represents the Hessian of the cost function at the nominal optimum. Note that using the inverse of the Hessian enforces decoupling but corresponds to dead-beat control, which may not be advisable under noise. The gain matrix is often taken as $K_{NEC} := -\kappa\mathscr{A}^{-1}$, since the adaptation gain $\kappa \in (0,1]$ helps enforce convergence by ensuring that the step is not too large. Here, the general formulation with the $(n_u \times n_u)$ gain matrix K_{NEC} is considered in order to have as many tuning parameters as possible in investigating the equivalence between NEC and SOC. This can be interpreted as $K_{NEC} := -\Gamma\mathscr{A}^{-1}$, with the $(n_u \times n_u)$ matrix Γ.

Implementation using transient measurements

If transient measurements are available online, the NEC integral control law (24) can be rewritten in the equivalent continuous-time formulation:

$$\delta\dot{u}(t) = K_{NEC}\, g(t), \tag{25}$$

where $g(t)$ is the online *estimate* of the steady-state gradient. This control law will be called "transient" NEC. Because of the presence of an integral term, NEC will drive the dynamical system to optimal steady-state performance with $g(\infty) = 0$.

It is clear that using NEC with transient measurements has the potential of being faster than with steady-state measurements[16]. However, it all boils down to the accuracy with which the steady-state gradient can be estimated. This topic is outside the scope of this paper.

Self-Optimizing Control

The original approach to determine the CVs has been through the so-called null-space approach[12] that uses a model of the plant to compute the optimal inputs and outputs for specific (parametric) disturbances. Other approaches have also been presented, which use either minimization of an appropriate loss function[17] or measured data directly[18].

The approach based on the null-space approach proceeds as follows: (i) calculate the sensitivity of the optimal outputs and inputs with respect to disturbances and/or parametric variations (as in this study),

$$\mathscr{S} \;:=\; \begin{bmatrix} \frac{dy}{d\theta} \\ \frac{du}{d\theta} \end{bmatrix}_{opt}, \tag{26}$$

where $\mathscr{S}$ is the $[(n_y + n_u) \times n_\theta]$ sensitivity matrix of rank n_θ, (ii) compute the $[\overline{n} \times (n_y + n_u)]$ matrix $\mathscr{N}$ that spans the left null space of $\mathscr{S}$, that is, $\mathscr{N}\,\mathscr{S} = 0_{\overline{n} \times n_\theta}$ with $\overline{n} = n_y + n_u - n_\theta$, and

(iii) select n_u CVs in $\mathcal{N}$.

The reason for using the left null space of $\mathcal{S}$ is very intuitive. If $(n_y + n_u)$ output and input measurements are available and n_θ parameters vary, $\mathcal{N}$ contains the $\overline{n}$ combinations of measurements that are insensitive to parametric variations and thus must remain unchanged to enforce optimality. With n_u inputs, we need n_u CVs to generate a square control system. If $n_y \geq n_\theta$, that is $\overline{n} \geq n_u$, it is always possible to select n_u CVs in the left null space of $\mathcal{S}$ as follows:

$$c = N \begin{bmatrix} y \\ u \end{bmatrix}, \tag{27}$$

with the $[n_u \times (n_y + n_u)]$ matrix $N := M\mathcal{N}$, where M is an arbitrary full-rank $[n_u \times \overline{n}]$ matrix. These CVs are kept at their nominal setpoints c_{sp} to enforce optimality despite the presence of disturbances (of known identity). Equivalently, the variations of the CVs,

$$\delta c := c - c_{sp} = N^y\, \delta y + N^u\, \delta u, \tag{28}$$

are kept at zero, where N^y is the $(n_u \times n_y)$ matrix including the first n_y columns of N and N^u the $(n_u \times n_u)$ matrix including the last n_u columns of N. In contrast, for $n_y < n_\theta$, that is $\overline{n} < n_u$, there are too few combinations of measurements (CVs) that are insensitive to the disturbances to bring the n_u elements of g to zero.

Remark 2

Controllability plays an important role in the choice of the null space. The total derivatives of the CVs with respect to the inputs are given by

$$\frac{d\,\delta c}{d\,\delta u} = N^y \frac{dH}{du} + N^u = N^y \mathcal{Q} + N^u.$$

The matrix $N^y \mathcal{Q} + N^u$ *needs to be invertible to have controllable CVs.*

Remark 3

There are several ways of choosing the CVs through the choice of the arbitrary $(n_u \times \overline{n})$ *matrix* M.

Each choice results in no performance loss for variations of the parameters θ. In practice, however, one may favor certain choices for ease and accuracy of measurement. Note that, for $n_y = n_\theta$, the matrix M is a $[n_u \times n_u]$ regular matrix, and the CVs span the entire null space $\mathcal{N}$.

Remark 4

The original formulation of SOC based on the null-space method used the sensitivity of the optimal values of measured quantities in the broad sense[12]. No distinction were made between inputs and outputs, though the authors mentioned that the inputs are often included. The minimal number of measurements was indicated as $n_\theta + n_u$, which makes sense as n_θ measurements are needed to estimate the parametric uncertainty and n_u measurements are necessary to compute the input updates. Since the inputs are typically known and available, the term $\left(\frac{du}{d\theta}\right)_{opt}$ should always be included in $\mathcal{S}$, thereby increasing the dimension of the null space $\mathcal{N}$.

As for gradient control discussed above, driving δc to zero can also be done in two different ways.

Implementation using steady-state measurements

The CVs variations can be driven to zero iteratively, using for example the discrete integral control law:

$$\delta u_{k+1} \quad = \quad \delta u_k + K_{SOC}\, \delta c_k, \tag{29}$$

where K_{SOC} is the $(n_u \times n_u)$ controller gain matrix and δc_k are the variations of the CVs observed at the k^{th} steady-state iteration.

Combining the last two equations and using the notations $K^y_{SOC} := K_{SOC}\, N^y$ and $K^u_{SOC} := (I_{n_u} + K_{SOC}\, N^u)$ gives:

$$\delta u_{k+1} \quad = \quad K^y_{SOC} \delta y_k + K^u_{SOC} \delta u_k\,, \tag{30}$$

with the measurements δy_k and δu_k taken at steady state. Equations (29) and (30) represent the "steady-state" SOC laws.

Implementation using transient measurements

If transient measurements are available online, the SOC integral control law (29) can be rewritten in the equivalent continuous-time formulation:

$$\delta \dot{u}(t) = K_{SOC}\, \delta c(t), \tag{31}$$

where $\delta c(t)$ is an online *estimate* of the steady-state gradient. This control law will be called "transient" SOC. Because of the presence of an integral term, SOC will drive the dynamical system to optimal steady-state performance with $\delta c(\infty) = 0$.

Comparison Between NEC and SOC

A nice feature of both NEC and SOC compared to other RTO techniques lies in their ability to compute the gradient information from δy and δu at a *single operation point*, while other techniques, such as the search and perturbation methods, require several operating points. The interested reader is referred to[16,19] for a detailed comparison of gradient-based RTO schemes.

Another feature of the NEC and SOC laws is that, upon convergence, either the plant gradient g or the CV variations δc vanish, as per equation (24) and (29). The optimizing ideas of NEC and SOC are illustrated in Figure 1.

Although the two ideas of NEC and SOC seem quite different, there are many similarities between the two schemes. It has already been pointed out that the CVs ideally represent the gradient g.[13] The current paper goes a step further to show that the two methods can be made equivalent through appropriate tuning.

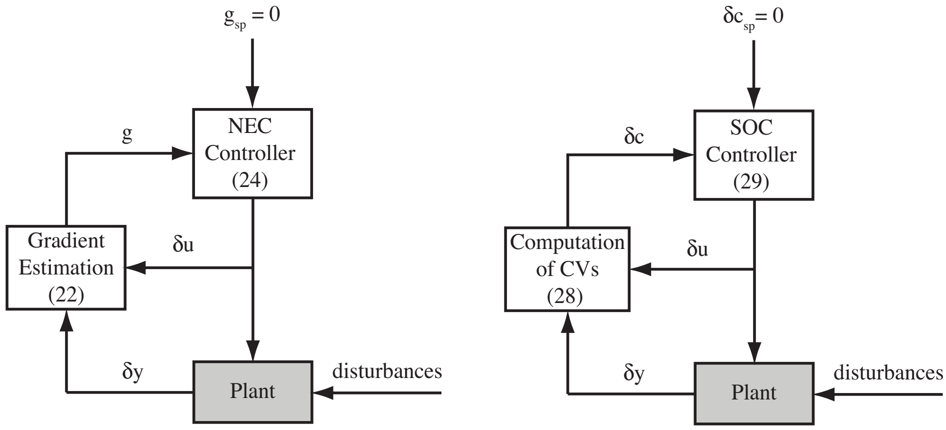

Figure 1: Optimizing ideas of NEC (left-hand side) and SOC (right-hand side).

Equivalence Between NEC and SOC

On the outset, NEC seems to have no degrees of freedom at all, while SOC has quite a few associated with the choice of the CVs in the null space and the choice of the controller. However, if NEC uses the degrees of freedom available in the choice of $\mathscr{D}$, a pseudoinverse of $\mathscr{P}$, and in the transformation matrix $\mathscr{R}$ introduced to define a generalized gradient, then the next theorem states that the two methods are strictly equivalent.

Theorem 1

[Equivalence Between NEC and SOC] Consider the optimization problem (3)-(5) with n_u inputs, n_y outputs and n_θ uncertain parameters, with $n_y \geq n_\theta$. Let F_x be invertible and the matrices $\mathscr{A}$, $\mathscr{B}$, $\mathscr{P}$ and $\mathscr{Q}$ be full rank. Let the controlled variables g be given by (22) for NEC and δc given by (28) for SOC.

1. *Any generalized gradient $\mathscr{R}g$ computed in NEC can be interpreted as CVs in SOC, that is, for any transformation $\mathscr{R}$ and matrix $\mathscr{D}$ used to compute g, with $\mathscr{D}\mathscr{P} = I_{n_\theta}$, the generalized gradient is insensitive to parametric variations.*

2. *Any controllable vector δc in SOC is a generalized gradient in NEC, that is, for any choice of N^y and N^u in the left null space of $\mathscr{S}$ such that $(N^y\mathscr{Q} + N^u)$ is invertible, there exists a matrix $\mathscr{D}$ satisfying $\mathscr{D}\mathscr{P} = I_{n_\theta}$ and a regular matrix $\mathscr{R}$ such that $N^y = \mathscr{R}G^y$ and $N^u = \mathscr{R}G^u$.*

Proof: Part 1. Using $\mathscr{R}g = \mathscr{R}G^y\,\delta y + \mathscr{R}G^u\,\delta u$, the generalized gradient can be expressed as $\delta c = N^y\,\delta y + N^u\,\delta u$, with $N^y = \mathscr{R}G^y$ and $N^u = \mathscr{R}G^u$.

We show next that the $[n_u \times (n_y + n_u)]$ matrix $N = \mathscr{R}[G^y \;\; G^u]$ lies in the left null space of $\mathscr{S}$, that is, $N\mathscr{S} = 0$. Using the definitions of G^y and G^u, the identities $\mathscr{D}\mathscr{P} = I_{n_\theta}$ and $\mathscr{A}\mathscr{C} = -\mathscr{B}$, and the expressions $\frac{dy}{d\theta} = \frac{dH}{du}\frac{du}{d\theta} + \frac{dH}{d\theta} = \mathscr{Q}\mathscr{C} + \mathscr{P}$ and $\frac{du}{d\theta} = \mathscr{C}$, one can write:

$$
\begin{aligned}
\mathscr{R}[G^y\,G^u]\mathscr{S} &= \mathscr{R}[\mathscr{B}\mathscr{D} \quad (\mathscr{A} - \mathscr{B}\mathscr{D}\mathscr{Q})]\begin{bmatrix}\mathscr{Q}\mathscr{C} + \mathscr{P}\\ \mathscr{C}\end{bmatrix}\\
&= \mathscr{R}(\mathscr{B}\mathscr{D}\mathscr{P} + \mathscr{A}\mathscr{C}) = 0_{n_u \times n_\theta}\,. \qquad (32)
\end{aligned}
$$

Part 2. Given any N that lies in the left null space of $\mathscr{S}$, we need to show that it fits the structure $N^y = \mathscr{R}\mathscr{B}\mathscr{D}$ and $N^u = \mathscr{R}\mathscr{A} - \mathscr{R}\mathscr{B}\mathscr{D}\mathscr{Q} = \mathscr{R}\mathscr{A} - N^y\mathscr{Q}$ for some $\mathscr{R}$ and $\mathscr{D}$, with $\mathscr{D}\mathscr{P} = I_{n_\theta}$. In other words, one needs to show the existence of the two matrices $\mathscr{R}$ and $\mathscr{D}$ such that the following three conditions are satisfied: (i) $N^y = \mathscr{R}\mathscr{B}\mathscr{D}$, (ii) $N^u = \mathscr{R}\mathscr{A} - N^y\mathscr{Q}$, and (iii) $\mathscr{D}\mathscr{P} = I_{n_\theta}$.

Among the three conditions, condition (ii) can be used to calculate $\mathscr{R}$, namely, $\mathscr{R} = (N^y\mathscr{Q} + N^u)\,\mathscr{A}^{-1}$. Since $\mathscr{A}$ is a positive-definite Hessian, its inverse exists. The term $(N^y\mathscr{Q} + N^u)$ is regular from the assumption that the CVs are controllable. Hence, $\mathscr{R}$ exists and is regular.

The condition that $\mathscr{N}$ lies in the left null space of $\mathscr{S}$ can be written as

$$
\begin{aligned}
[N^y \;\; N^u]\mathscr{S} &= [N^y \;\; N^u]\begin{bmatrix}\mathscr{Q}\mathscr{C} + \mathscr{P}\\ \mathscr{C}\end{bmatrix}\\
&= N^y\mathscr{Q}\mathscr{C} + N^y\mathscr{P} + N^u\mathscr{C}\\
&= N^y\mathscr{P} + \mathscr{R}\mathscr{A}\mathscr{C} = N^y\mathscr{P} - \mathscr{R}\mathscr{B} = 0_{n_u \times n_\theta}\,. \qquad (33)
\end{aligned}
$$

In what follows, we first choose $\mathscr{D}$ that satisfies condition (i). Then, we will show that the choice proposed also satisfies condition (iii). We will need to distinguish two cases, namely, $n_u \geq n_\theta$ and $n_u < n_\theta$.

If $n_u \geq n_\theta$, the pseudoinverse of $\mathscr{R}\mathscr{B}$ can be applied to condition (i) to obtain $\mathscr{D} = (\mathscr{R}\mathscr{B})^+\,N^y$, with the superscript $(.)^{+}$ denoting the Moore-Penrose pseudoinverse. Looking into condition (iii), it can be seen with the help of (33) that

$$\mathscr{D}\mathscr{P} = (\mathscr{R}\mathscr{B})^{+} N^{y} \mathscr{P} = (\mathscr{R}\mathscr{B})^{+} (\mathscr{R}\mathscr{B}) = I_{n_\theta}. \tag{34}$$

If $n_u < n_\theta$, then the following procedure is followed to choose $\mathscr{D}$. (a) Compute the null space of the $(n_u \times n_\theta)$ matrix $\mathscr{R}\mathscr{B}$ and append the $(n_\theta - n_u)$ rows termed $\mathscr{N}_{\mathscr{R}\mathscr{B}}$ such that $\begin{bmatrix} \mathscr{R}\,\mathscr{B} \\ \mathscr{N}_{\mathscr{R}\mathscr{B}} \end{bmatrix}$ has rank n_θ. (b) Choose $\mathscr{D} = \begin{bmatrix} \mathscr{R}\,\mathscr{B} \\ \mathscr{N}_{\mathscr{R}\mathscr{B}} \end{bmatrix}^{-1} \begin{bmatrix} N^{y} \\ \mathscr{N}_{\mathscr{R}\mathscr{B}} \mathscr{P}^{+} \end{bmatrix}$, which obviously satisfies the condition (i). It can be verified that condition (iii) is verified:

$$\mathscr{D}\mathscr{P} = \begin{bmatrix} \mathscr{R}\,\mathscr{B} \\ \mathscr{N}_{\mathscr{R}\mathscr{B}} \end{bmatrix}^{-1} \begin{bmatrix} N^{y} \mathscr{P} \\ \mathscr{N}_{\mathscr{R}\mathscr{B}} \mathscr{P}^{+} \mathscr{P} \end{bmatrix} = \begin{bmatrix} \mathscr{R}\,\mathscr{B} \\ \mathscr{N}_{\mathscr{R}\mathscr{B}} \end{bmatrix}^{-1} \begin{bmatrix} \mathscr{R}\,\mathscr{B} \\ \mathscr{N}_{\mathscr{R}\mathscr{B}} \end{bmatrix} = I_{n_\theta}. \tag{35}$$

Thus, the existence of the two matrices $\mathscr{R}$ and $\mathscr{D}$ has been shown. □

Remark 5

The extra degrees of freedom one has in choosing the CVs in the null space $\mathscr{N}$ are translated into the extra degrees of freedom that exist in the selection of the pseudoinverse of $\mathscr{P}$ in NEC. In the presence of noise, these extra degrees of freedom could become very handy.

Corollary 1

[Controller Design] Consider the optimization problem (3)-(5) with n_u inputs, n_y outputs and n_θ uncertain parameters, with $n_y \geq n_\theta$. Let F_x be invertible, the matrices $\mathscr{A}$, $\mathscr{B}$, $\mathscr{P}$ and $\mathscr{Q}$ be full rank, and the matrix $\mathscr{D}$ be such that $\mathscr{D}\mathscr{P} = I_{n_\theta}$. Let the controlled variables g be given by (22) for NEC and δc given by (28) for SOC.

1. *Given a NEC controller with the gain matrix K_{NEC}, an equivalent SOC control law can be obtained by choosing $N^{y} = \mathscr{B}\,\mathscr{D}$, $N^{u} = \mathscr{A} - \mathscr{B}\,\mathscr{D}\,\mathscr{Q}$, and $K_{SOC} = K_{NEC}$.*

2. *Given a SOC controller with $\delta c = N^{y}\delta y + N^{u}\delta u$ and the gain matrix K_{SOC}, an equivalent*

NEC law can be obtained by choosing $K_{NEC} = K_{SOC}\mathscr{R}$ *and* $g = G^y\,\delta y + G^u\,\delta u$*, with* $\mathscr{R} = (N^y\mathscr{Q} + N^u)\,\mathscr{A}^{-1}$*,* $G^y = \mathscr{B}\,\mathscr{D}$*,* $G^u = \mathscr{A} - \mathscr{B}\,\mathscr{D}\,\mathscr{Q}$*, and either* $\mathscr{D} = (\mathscr{R}\mathscr{B})^+ N^y$ *if* $n_u \geq n_\theta$ *or* $\mathscr{D} = \begin{bmatrix} \mathscr{R}\,\mathscr{B} \\ \mathscr{N}_{\mathscr{R}\mathscr{B}} \end{bmatrix}^{-1} \begin{bmatrix} N^y \\ \mathscr{N}_{\mathscr{R}\mathscr{B}}\mathscr{P}^+ \end{bmatrix}$ *otherwise.*

Proof: The proof is straightforward and is thus omitted here.

Information Required for Control Design

NEC uses the steady-state models $F(x,u,\theta) = 0$ and $y = H(x,u,\theta)$. Furthermore, the identity of the disturbances (in this study, the uncertain parameters θ) need to be known to compute the corresponding partial derivatives that enter in the computation of most control matrices. However, the actual sizes of the parametric variations need not be known as they are inferred from the measurements δy and δu as per equation (21). The condition $n_y \geq n_\theta$ suffices to reconstruct $\delta\theta$ from δy.

The same information allows designing a SOC law based on the null-space method. The steady-state models and the identity of the disturbances are needed to compute the sensitivity matrix $\mathscr{S}$. $\mathscr{S}$ can be either obtained via model-based optimization (to determine the optimal outputs and inputs for the perturbed model), or from $\mathscr{C}$, $\mathscr{P}$ and $\mathscr{Q}$ using (33). The condition $n_y \geq n_\theta$ allows selecting n_u CVs in the null space $\mathscr{N}$. Furthermore, owing to the equivalence between NEC and SOC, it is no longer necessary to evaluate the sensitivity matrix (26). Instead, one can compute $N = M_1 \begin{bmatrix} -\mathscr{A}^{-1}G^y & -\mathscr{A}^{-1}G^u \end{bmatrix}$, with M_1 chosen arbitrarily. Note that, if $M_1 = -\mathscr{A}$, then δc represents an estimate of the gradient g. Alternatively, M_1 can be chosen to optimize some other criterion.

The static models $F(x,u,\theta) = 0$ and $y = H(x,u,\theta)$ are typically identified from steady-state data. An interesting topic regards the possibility of designing a self-optimizing controller directly from data, that is, without expliciting the static models $F(x,u,\theta) = 0$ and $y = H(x,u,\theta)$. One such attempt was presented recently by Jaeschke and Skogestad[18] in the context of SOC. The idea implies (i) estimating the output sensitivities $\frac{dy}{du}$ via step changes of the inputs, and (ii) estimating the cost gradient from open-loop process data by fitting a quadratic function to the measured cost. This allows inferring the matrix N needed to select the CVs.

Illustrative Example

The illustrative example is taken from[16,19], where it has been used for comparing gradient estimation techniques. Steady-state optimization of an isothermal CSTR is investigated, with the reactions $A+B \rightarrow C$ and $2B \rightarrow D$. There are two manipulated variables, the feed rates of A and B. The goal is to maximize the productivity of C at steady state. The problem can be formulated mathematically as follows:

$$\max_{u_A, u_B} \qquad J(u_A, u_B) = \frac{c_C^2 \, (u_A + u_B)^2}{u_A c_{Ain}} - w(u_A^2 + u_B^2) \tag{36}$$

$$\dot{c_A} = -k_1 \; c_A \; c_B + \frac{u_A}{V} c_{Ain} - \frac{u_A + u_B}{V} c_A \qquad c_A(0) = c_{A,s} \tag{37}$$

$$\dot{c_B} = -k_1 \; c_A \; c_B - 2 \; k_2 \; c_B^2 + \frac{u_B}{V} c_{Bin} - \frac{u_A + u_B}{V} c_B \qquad c_B(0) = c_{B,s} \tag{38}$$

$$\dot{c_C} = k_1 \; c_A \; c_B - \frac{u_A + u_B}{V} c_C \qquad c_C(0) = c_{C,s} \tag{39}$$

$$\dot{c_D} = 2 \; k_2 \; c_B^2 - \frac{u_A + u_B}{V} c_D \qquad c_D(0) = c_{D,s} \tag{40}$$

where c_X denotes the concentration of species X and $c_{X,s}$ the corresponding steady-state values, V is the reactor volume, u_A and u_B are the feed rates of A and B, c_{Ain} and c_{Bin} are the inlet concentrations, k_1 and k_2 are the rate constants of the two chemical reactions, and w a weighting parameter.

The first term of J corresponds to the amount of C produced, $c_C \, (u_A + u_B)$, multiplied by the yield factor, $\frac{c_C (u_A + u_B)}{u_A c_{Ain}}$, while the second term penalizes the control effort.

Two different scenarios are considered throughout this section.

1. **Scenario 1:** We start by considering that the plant differs from the model only by the values

of the rate constants, k_{1p} and k_{2p}. Hence, the vector of uncertain parameters $\theta = [k_1 \;\; k_2]^T$ is of dimension $n_\theta = 2$. Since it is assumed – although not necessary – that the concentrations of the four species are measured, we have $n_y > n_\theta$ and $n_u = n_\theta$.

2. **Scenario 2:** In addition to the uncertainty on the values of the rate constants, the second scenario also considers that the inlet concentration of A is underestimated by the model. Hence, the vector of uncertain parameters $\theta = [k_1 \;\; k_2 \;\; c_{Ain}]^T$ is thus of dimension $n_\theta = 3$. Since it is again assumed – although not necessary – that the concentrations of the four species are measured, we have $n_y > n_\theta$ and $n_u < n_\theta$.

The values of the uncertain parameters are unknown to the RTO schemes. The plant settling time is about 50 min, which corresponds to a dominant time constant of about 12 min. The numerical values of the model and plant parameters are given in Table 1.

Table 1: Model and plant parameters

Model and plant parameters					
k_1	0.75	$\frac{\mathrm{L}}{\mathrm{mol\,min}}$	k_{1p}	1.4	$\frac{\mathrm{L}}{\mathrm{mol\,min}}$
k_2	1.5	$\frac{\mathrm{L}}{\mathrm{mol\,min}}$	k_{2p}	0.4	$\frac{\mathrm{L}}{\mathrm{mol\,min}}$
c_{Ain}	2	$\frac{\mathrm{mol}}{\mathrm{L}}$	$c_{Ain,p}$	2.5	$\frac{\mathrm{mol}}{\mathrm{L}}$
c_{Bin}	1.5	$\frac{\mathrm{mol}}{\mathrm{L}}$	V	500	L
w	0.004	$\frac{\mathrm{mol\,min}}{\mathrm{L}^2}$			

The normalized cost is $J_p(t)/J_{p,opt}$, where $J_{p,opt}$ is the optimal cost of the plant at steady state. This value is of course different for the two scenarios. Hereafter, “transient” SOC corresponds to using SOC with transient measurements for implementation. We will use the label “steady-state” SOC for the case where only steady-state measurements are used for SOC, while “adjusted” SOC will be used when SOC is tuned to match the performance of NEC. Conversely, “adjusted” NEC will be used when a NEC controller is tuned to match the performance of a given SOC law.

Scenario 1

Use of steady-state measurements. We first illustrate the implementation of both NEC and SOC with steady-state measurements. By default K_{SOC} is computed as $(N^y\mathscr{Q}+N^u)^{-1}$ since, as discussed in [16,19], this choice allows local decoupling of the CVs.

Figure 2 compares the normalized costs for "steady-state" NEC and SOC. NEC uses the Moore-Penrose pseudoinverse of $\mathscr{P}$, and the submatrix N for SOC is arbitrarily chosen as the last two rows of $\mathscr{N}$. As seen, it takes 3 iterations to converge close to the plant optimal performance. Both methods perform well and converge in the neighborhood of the plant optimum. The initial value of about 0.8 corresponds to the cost resulting from using the model optimal inputs. The difference of about 20% is what is gained via real-time optimization. Note that the transient cost can be larger than 1 before steady state is reached.

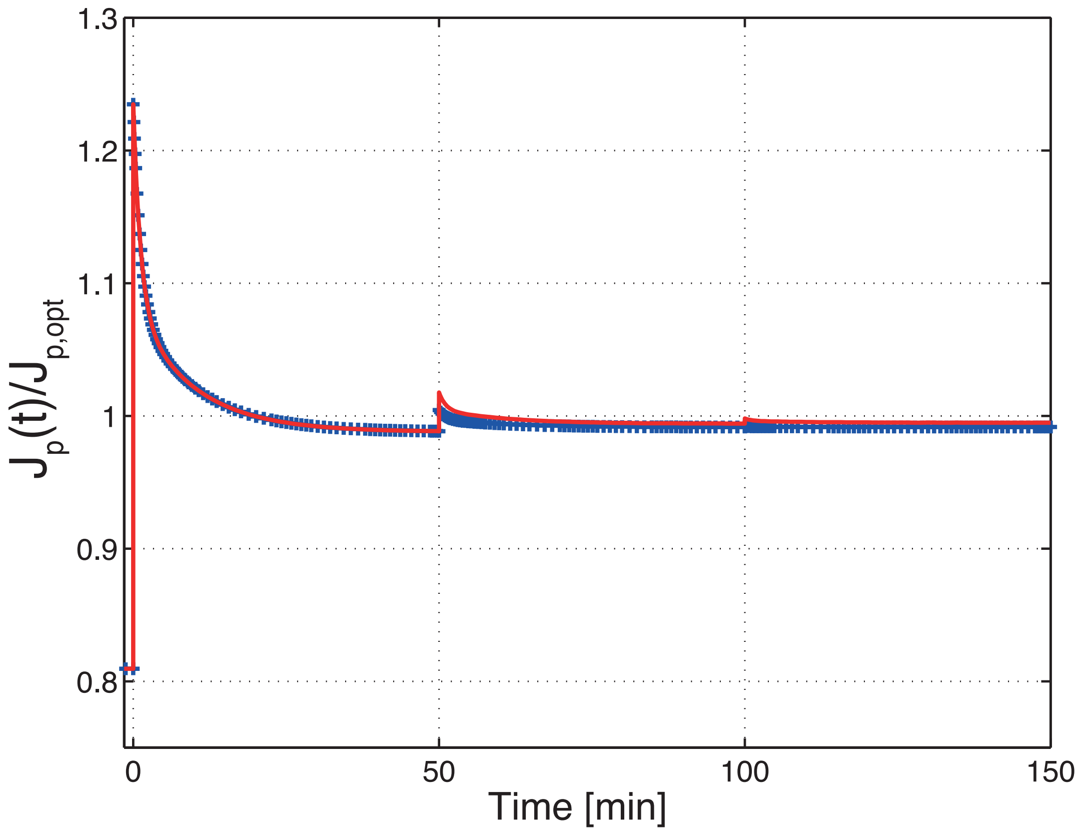

Figure 2: Performance of "steady-state" NEC (solid red line) and SOC (blue crosses). Convergence in 100 minutes.

Use of transient measurements. We illustrate next the application of NEC and SOC with transient measurements. Figure 3 compares the normalized cost for NEC and for SOC for two different choices of CVs. Here, NEC uses $\mathscr{P}^+$, and the submatrix N is chosen as the first two rows and as the last two rows of $\mathscr{N}$. Both methods perform well and converge in the neighborhood of the plant optimum, with only marginal differences between the converged performances. Note that convergence to the neighborhood of the plant optimum is achieved within a single iteration to steady state.

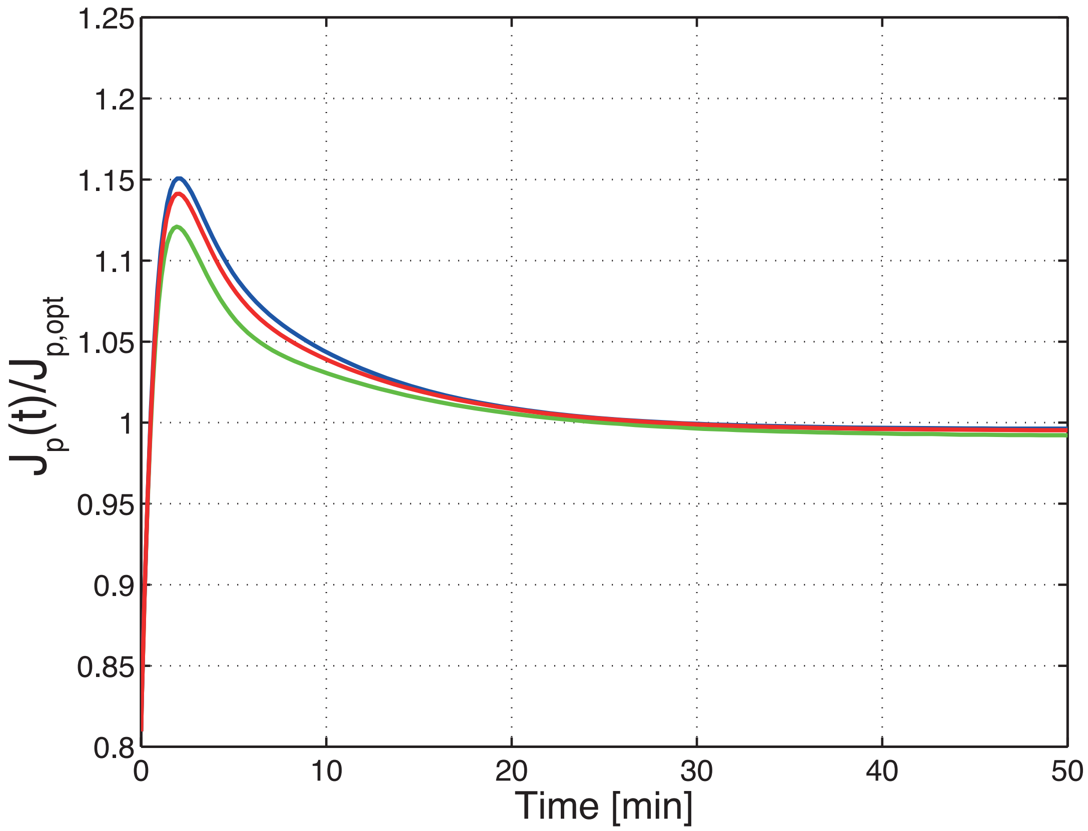

Figure 3: Performance of "transient" NEC (solid red line) and SOC for two different choices of N. The solid blue and green lines are obtained when N corresponds to the two first and the two last rows of $\mathscr{N}$, respectively. Convergence in 30 minutes.

Illustration of Theorem 1. Next, we illustrate Theorem 1. That is, we illustrate first that NEC corresponds to SOC, for which the CVs are the gradient terms. For any regular K_{SOC}, $N_{ideal} := K_{SOC}^{-1}[G^y \;\; G^u]$ leads to strict equivalence between NEC and "adjusted" SOC. We also illustrate the implication of the second part of Theorem 1 – here for $n_u \geq n_\theta$ – and show that a NEC controller can be modified to exactly match SOC, for any choice of controllable CVs. As suggested by Theorem 1, NEC is adjusted via the two matrices $\mathscr{R} = (N^y \mathscr{Q} + N^u)\,\mathscr{A}^{-1}$ and $\mathscr{D} = (\mathscr{R}\mathscr{B})^{+} N^y$. We limit the analysis to the two cases for which N is chosen as the two first rows and the two last rows of $\mathscr{N}$ (other choices would lead to the same conclusions).

Figure 4 compares the performances of “adjusted” NEC and “adjusted” SOC to the corresponding cost profiles obtained with the standard tunings. The comparison of Figure 4 and Figure 3 clearly shows that: (i) “adjusted” NEC matches null-space SOC for both choices of N and (ii) “adjusted” SOC matches “transient” NEC, since all the three corresponding pairs of curves are superimposed.

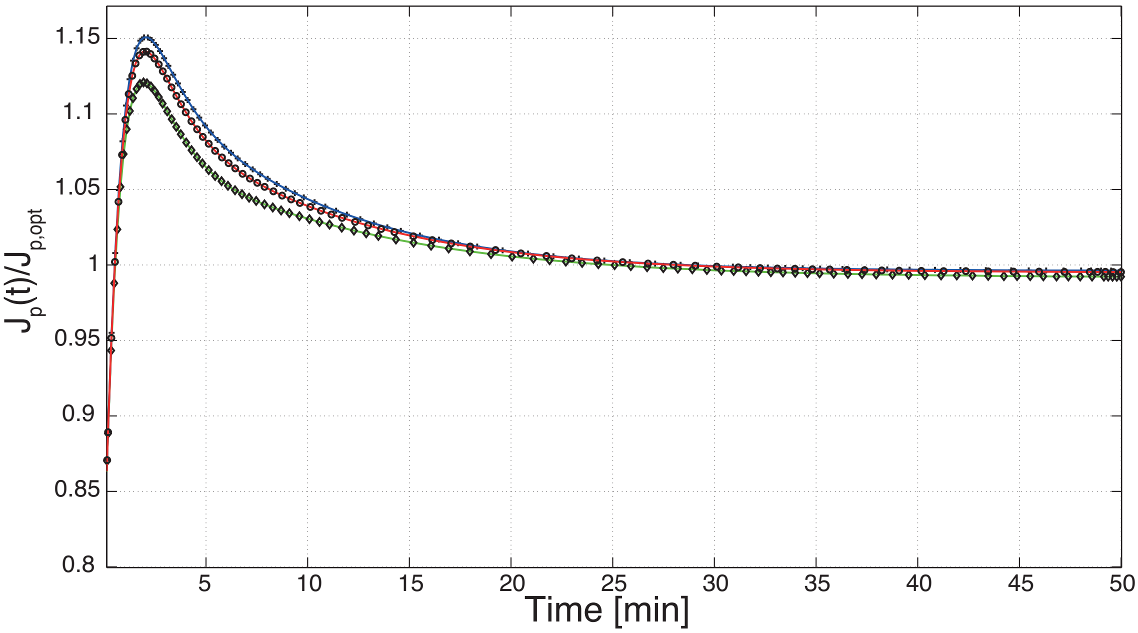

Figure 4: Performance of “adjusted” SOC and “adjusted” NEC. The curve obtained with “adjusted” SOC (black circles) lies on top of the solid red line, which corresponds to the “transient” NEC. The two curves obtained with “adjusted” NEC (black crosses and diamonds) are superimposed to the solid blue and solid green curves, which correspond to SOC when N is chosen as the two first and two last rows of $\mathscr{N}$, repsectively.

Scenario 2

The goal of this subsection is to illustrate the second part of Theorem 1 when $n_u < n_\theta$. As suggested by Theorem 1, when $n_u < n_\theta$, the tuning of “adjusted” NEC requires the use of the same $\mathscr{R}$, i.e.

$\mathscr{R} = (N^y\mathscr{Q} + N^u)\,\mathscr{A}^{-1}$ but of a different $\mathscr{D}$, i.e. $\mathscr{D} = \begin{bmatrix} \mathscr{R}\mathscr{B} \\ \mathscr{N}_{\mathscr{R}\mathscr{B}} \end{bmatrix}^{-1} \begin{bmatrix} N^y \\ \mathscr{N}_{\mathscr{R}\mathscr{B}}\mathscr{P}^+ \end{bmatrix}$ for "adjusted" NEC to be strictly equivalent to SOC.

With $\theta = [k_1 \;\; k_2 \;\; c_{Ain}]$, $\mathscr{S}$ is now of dimension $[6 \times 3]$ and $\mathscr{N}$ is of dimension 3×6. Again, we consider two possible choices for N, that is, the first two rows and the two last rows of $\mathscr{N}$. Since $n_u < n_\theta$, $\text{rank}(\mathscr{R}\mathscr{B}) < n_\theta$ and, thus, $\mathscr{R}\mathscr{B}$ does not have a left pseudoinverse. It is therefore necessary to use the null space $\mathscr{N}_{\mathscr{R}\mathscr{B}}$ of $\mathscr{R}\mathscr{B}$ to construct $\mathscr{D}$. Figure 5 illustrates that, also when $n_u < n_\theta$, there exists an "adjusted" NEC that is strictly equivalent to SOC, for the two different choices of N. Again, the corresponding pairs of curves are superimposed. For the rest, the results are qualitatively similar to those of the case $n_u \geq n_\theta$.

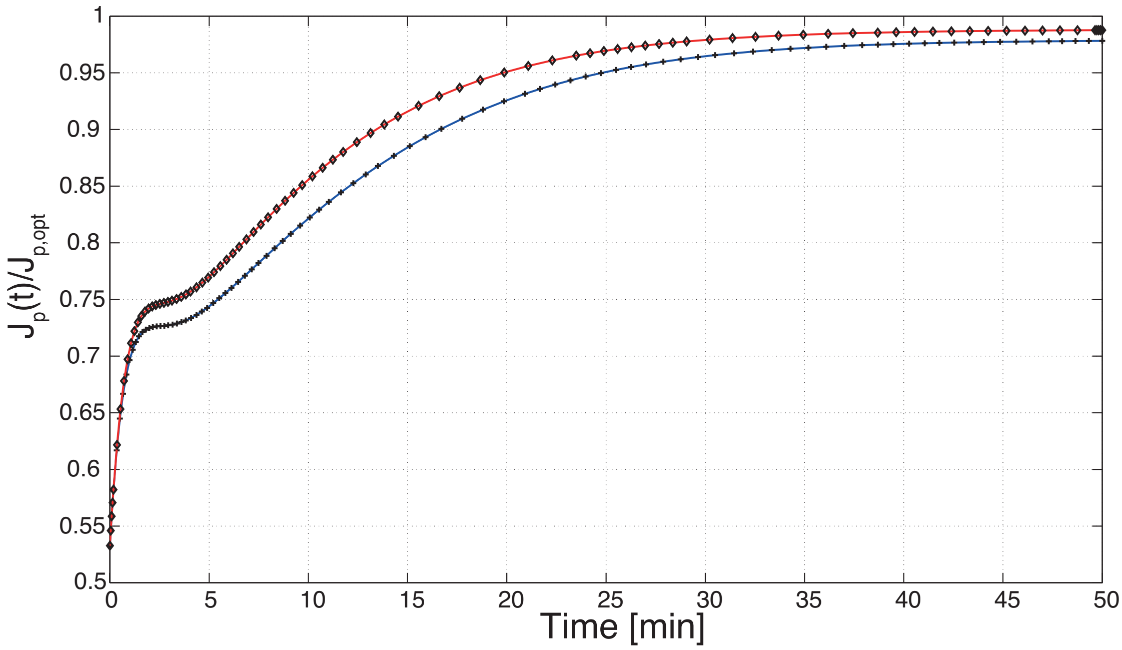

Figure 5: Performance of "adjusted" NEC (black crosses and black diamonds) compared to that of SOC (solid red and blue lines for the cases for which N is chosen as the two first and the two last rows of $\mathscr{N}$, respectively) for the case $n_u < n_\theta$.

Finally, note that, although these techniques are linearization-based and thus are only guaranteed to perform well for small perturbations, large parametric variations were successfully handled in this example, as the plant and model kinetic parameters varied by factors of 2 and 4.

Conclusions

This paper has investigated the equivalence between neighboring-extremal control and self-optimizing control for unconstrained optimization problems. Only the self-optimizing control scheme based on the null space of the sensitivity matrix has been considered. Conditions under which the two techniques are equivalent have been proposed. The key point is that both methods estimate the total derivatives of the cost function from input and output measurements, directly for NEC and indirectly through the choice of CVs for SOC. This work confirms the recent results suggesting that the CVs can be chosen to estimate the cost gradient[13]. Furthermore, it has been argued that the matrix $\mathscr{S}$ used in SOC should include information regarding input sensitivities with respect to uncertain parameters.

Both optimizing schemes are set up on the basis of purely static considerations, namely, the steady-state gradient equal to zero for NEC, and the CVs in the null space of a steady-state sensitivity matrix for SOC. It may seem rather wishful thinking to want to estimate *steady-state* values using transient measurements, without any dynamic consideration. However, this is supported by the fact that the estimated signals tend toward the sought steady-state values when the dynamic system approaches steady state.

This paper has shown that both NEC and SOC can be used with either steady-state or transient measurements. The difference that persists in the literature is artificial and is probably due to the way the implementation was done in the original publications. This means that NEC can also be used as an online optimizing multivariable controller similar to SOC. In this case, the CVs and their setpoints are defined offline and correspond to the cost gradient and its desired value of zero. On the other hand, this also means that SOC can be implemented iteratively using steady-state measurements, which helps exploit the measured CVs in terms of steady-state values.

References

1. Boyd, S.; Vandenberghe, L. *Convex Optimization*; Cambridge University Press, 2004.

2. Rotava, O.; Zanin, A. C. Multivariable control and real-time optimization – An industrial practical view. *Hydrocarbon Processing* **2005**, *84*, 61–71.

3. Marlin, T.; Hrymak, A. Real-time Operations Optimization of Continuous Processes. *In AIChE Symp. Series - CPC-V* **1997**, *93*, 156–164.

4. Zhang, Y.; Monder, D.; Forbes, J. F. Real-Time Optimization under Parametric Uncertainty: A Probability Constrained Approach. *J. Process Control* **2002**, *12*, 373–389.

5. Box, G. E. P.; Draper, N. R. *Empirical Model-building and Response Surfaces*; John Wiley, New York, 1987.

6. Ariyur, K.; Krstic, M. *Real-Time Optimization by Extremum-Seeking Control*; John Wiley, New York, 2003.

7. DeHaan, D.; Guay, M. Extremum-seeking control of state-constrained nonlinear systems. *Automatica* **2005**, *41*, 1567–1574.

8. Francois, G.; Srinivasan, B.; Bonvin, D. Use of measurements for enforcing the necessary conditions of optimality in the presence of constraints and uncertainty. *J. Process Contr.* **2005**, *15*, 701–712.

9. Skogestad, S. Plantwide Control: The Search for the Self-Optimizing Control Structure. *J. Process Control* **2000**, *10*, 487–507.

10. Bryson, A.; Ho, Y.-C. *Applied Optimal Control : Optimization, Estimation, and Control*; Hemisphere Publ. Corp., Washington, USA, 1975.

11. Gros, S.; Srinivasan, B.; Bonvin, D. Optimizing Control Based on Output Feedback. *Comput. Chem. Engng.* **2009**, *33*, 191–198.

12. Alstad, V.; Skogestad, S. Null Space Method for Selecting Optimal Measurement Combinations as Controlled Variables. *Ind. Eng. Chem. Res.* **2007**, *46*, 846–853.

13. Jaeschke, J.; Skogestad, S. Optimal Operation by Controlling the Gradient to Zero. *In Proc. of the 18th IFAC World Congress, Milano, Italy* **2011**, 593–598.

14. Marchetti, A.; Zumoffen, D. On the Links Between Real-Time Optimization, Neighboring-Extremal Control and Self-Optimizing Control. *In Proc. of the European Control Conference, Zurich, Switzerland* **2013**, 4466–4471.

15. Bryson, A. *Dynamic Optimization*; Addison-Wesley, Menlo Park CA, USA, 1999.

16. Francois, G.; Srinivasan, B.; Bonvin, D. Comparison of Six Implicit Real-time Optimization Schemes. *J. Européen des Systèmes Automatisés* **2012**, *46*, 291–305.

17. Kariwala, V.; Cao, Y.; Janardhanan, S. Local Self-Optimizing Control with Average Loss Minimization. *Ind. Eng. Chem. Res.* **2008**, *47*, 1150–1158.

18. Jaeschke, J.; Skogestad, S. Using Process Data for Finding Self-Optimizing Controlled Variables. *In Proc. of the 10th Int. Symp. on Dynamics and Control of Process Systems, DYCOPS 2013), Mumbai, India* **2013**, 451–456.

19. Srinivasan, B.; Francois, G.; Bonvin, D. Comparison of Gradient Estimation Methods for Real-Time Optimization. *Computer-Aided Chemical Engineering* **2011**, *29*, 607–611.