Neighboring Extremal Controllers for Singular Problems

S. Gros, B. Srinivasan, D. Bonvin
Laboratoire d'Automatique
Ecole Polytechnique Fédérale de Lausanne
CH-1015 Lausanne

***Abstract*— The dynamic optimization problem in the presence of uncertainty (model mismatch and disturbances) is addressed. It has been recently proposed that this problem can be solved by tracking the necessary conditions of optimality in the various intervals of the solution. In this paper, it is shown that the standard neighboring extremal approach, which uses linearization around the optimal trajectory, drives to zero the first-order variation of the necessary conditions of optimality on the parts of the solution where no constraint is active. This fact is used to extend the neighboring extremal approach to singular problems. In singular problems, the linearization around the optimum lacks the information needed to build a neighboring extremal controller. This paper proposes to use the nonlinear dynamics to provide the lacking information. The theoretical ideas are illustrated for singular problems on a simple semi-batch chemical reactor.**

I. Introduction

Dynamic optimization provides an unified framework for improving process operations while taking into account operational and other types of constraints [1], [8]. Recently, there has been some emphasis on using measurements in the optimization framework in order to handle the uncertainty (model mismatch, process variations and disturbances) that is inevitably present in a real process. Among the various measurement-based optimization methods available in the literature [4], [7], a promising one, labeled NCO tracking, consists of enforcing the Necessary Conditions of Optimality corresponding to the real situation [7].

NCO tracking uses the fact that there are only two types of arcs that can constitute the optimal solution: constraint-seeking and sensitivity-seeking arcs [8]. This distinction depends on whether the solution is determined by the constraints of the optimization problem or forces a sensitivity (gradient) to zero. When the solution is determined by the constraints, tracking the necessary conditions of optimality corresponds to enforcing the corresponding constraints. The other case of sensitivity-seeking arcs, which is typically more involved, will be considered in this paper.

Along a sensitivity-seeking arc, optimization can be treated as the regulation of a sensitivity around zero. The main difficulty arises from the fact that this gradient information depends on the adjoint variables that are typically unknown. Thus, an efficient way of estimating them and their variations is needed. A simple technique that has been used for over four decades is the sweep method where, for a linear (linearized) system, the adjoint variables (or their variations) are considered proportional to the state variables (or their variations). The linearized version of this approach leads to the Neighboring Extremal (NE) controller [2], [3], [6]. It is shown in this paper that the standard NE controller forces the first-order variation of the NCO to zero. Thus, the NE controller can in fact be used for NCO tracking.

The main contribution of this paper lies in the use of the link between NE controllers and NCO tracking to extend the NE controllers to the singular case. The singular case arises when the optimal inputs cannot be computed directly from the NCO, and thus time differentiations of the NCO are required. In such a case, the standard NE controller cannot be used since it calls for inversion of a singular matrix. Time differentiations of the first-order variation of the NCO are used to derive the NE controller in the singular case.

The interesting aspect is that time differentiation of the NCO requires information on the nonlinear dynamics. Thus an important feature of the neighboring extremal controller for singular problem is the interplay between the linearized and nonlinear dynamics.

The paper is organized as follows. In Section 2, the necessary conditions of optimality are derived and the standard neighboring extremal controller is presented. Also, a link between the two is established. In Section 3, the neighboring extremal approach is extended to singular systems. An application example is presented in Section 4, and Section 5 concludes the paper.

II. Neighboring Extremal Approach for Non-singular Problems

A. Dynamic optimization

Consider the following dynamic optimization

$$u^*(t) = \arg\min_{u(t)} J \tag{1}$$

$$s.t. \quad J = \Phi(x(t_f)) + \int_0^{t_f} L(x,u)d\tau \tag{2}$$

$$\dot{x} = F(x,u) \qquad x(0) = x_0 \tag{3}$$

$$S(x,u) \leq 0, \qquad T(x(t_f)) \leq 0 \tag{4}$$

where x represents the n-dimensional state vector of the system, u the m-dimensional input vector, F the system dynamics, J the scalar objective function to be minimized,

t_f the fixed final time, Φ the terminal cost, L the integral cost, S the path constraints, and T the terminal constraints. The solution of the optimization problem (1)-(4) will be referred to as the nominal solution. This solution is typically discontinuous and consists of several intervals with corresponding arcs. Along the various arcs, the solution is either (i) determined by the constraints of the optimization problem, or (ii) inside the feasible region. Only the latter case will be considered here and, thus, the constraints S and T will not be considered.

The notation $a_b = \frac{\partial a}{\partial b}$ will be used. When the solution is inside the feasible region, i.e. when no constraints are active, the NCO can be expressed as:

$$H_u = \lambda^T F_u + L_u = 0 \tag{5}$$

where the Hamiltonian is given by $H = \lambda^T F + L$, with the adjoints λ governed by the following equations:

$$\dot{\lambda} = -H_x^T = -F_x^T \lambda - L_x^T, \qquad \lambda(t_f) = \Phi_x^T(t_f) \tag{6}$$

Since along the arcs where none of the constraints are active, the solution seeks to push the sensitivity H_u to zero, such an arc will be referred to as a *sensitivity-seeking arc*.

B. Neighbouring Extremal Approach

Including the dynamic constraints of the optimization problem (1)-(3) in the cost function, the augmented cost function, $\bar{J}$, can be written as [3]:

$$\bar{J} = \Phi(x(t_f)) + \int_0^{t_f} \left(H - \lambda^T \dot{x}\right) dt \tag{7}$$

In the presence of the perturbations δx and δu around the nominal trajectories, the augmented cost becomes $\bar{J} = \bar{J}_{nom} + \delta\bar{J} + \delta^2\bar{J} + O(\delta^3\bar{J})$, where $\bar{J}_{nom}$ is the nominal cost, and $\delta\bar{J}$ and $\delta^2\bar{J}$ are given by [3]:

$$\begin{aligned} \delta\bar{J} &= \left(\Phi_x - \lambda^T\right)\delta x\big|_{t_f} \\ &+ \int_0^{t_f} \left[\left(H_x + \dot{\lambda}^T\right)\delta x + H_u \delta u\right] d\tau \end{aligned} \tag{8}$$

$$\begin{aligned} \delta^2\bar{J} &= \frac{1}{2}\delta x(t_f)^T \Phi_{xx} \delta x(t_f) \\ &+ \frac{1}{2}\int_0^{t_f} \begin{bmatrix} \delta x^T & \delta u^T \end{bmatrix} \begin{bmatrix} H_{xx} & H_{xu} \\ H_{ux} & H_{uu} \end{bmatrix} \begin{bmatrix} \delta x \\ \delta u \end{bmatrix} d\tau \end{aligned} \tag{9}$$

From (5) and (6), at the optimum $\delta\bar{J} = 0$. Ignoring the terms of order 3 and higher, the NE approach minimizes $\delta^2\bar{J} \simeq \bar{J} - \bar{J}_{nom}$ subject to the linearized system equations:

$$\delta u^*(t) = \arg\min_{\delta u(t)} \delta^2\bar{J} \tag{10}$$

$$s.t. \qquad \dot{\delta x} = F_x \delta x + F_u \delta u \tag{11}$$

For this NE problem with the state δx, the inputs δu and the adjoints $\bar{\lambda}$, the corresponding Hamiltonian, $\bar{H}$, the adjoint equations and the necessary condition of optimality are given by:

$$\begin{aligned} \bar{H} &= \bar{\lambda}^T(F_x\delta x + F_u\delta u) \\ &+ \frac{1}{2}\begin{bmatrix} \delta x^T & \delta u^T \end{bmatrix} \begin{bmatrix} H_{xx} & H_{xu} \\ H_{ux} & H_{uu} \end{bmatrix} \begin{bmatrix} \delta x \\ \delta u \end{bmatrix} \end{aligned} \tag{12}$$

$$\bar{H}_{\delta u} = \bar{\lambda}^T F_u + \delta x^T H_{xu} + \delta u^T H_{uu} = 0 \tag{13}$$

$$\begin{aligned} \dot{\bar{\lambda}} &= -(\bar{H}_{\delta x})^T = -F_x^T\bar{\lambda} - H_{xx}\delta x - H_{xu}\delta u, \\ \bar{\lambda}(t_f) &= \Phi_{xx}\delta x(t_f) \end{aligned} \tag{14}$$

Since the equations are linear, the NE solution δu can be written explicitly. The key to the NE solution is the sweep method, where the adjoints are considered as linear functions of the states: $\bar{\lambda} = S\delta x$. From the dynamics of $\bar{\lambda}$, one can compute a differential equation for S [3]. The explicit solution takes the following form:

$$\delta u = -K(t)\delta x \tag{15}$$

$$K(t) = H_{uu}^{-1}(H_{ux} + F_u^T S) \tag{16}$$

$$\begin{aligned} \dot{S}(t) &= -H_{xx} - SF_x - F_x^T S + H_{xu}K \\ &\quad + SF_uK, \quad S(t_f) = \Phi_{xx}(x(t_f)) \end{aligned} \tag{17}$$

It should be emphasized here that S and K are evaluated along the nominal trajectories of the system, i.e. $x(t) = x_{nom}(t)$, $u(t) = u_{nom}(t)$ with the corresponding adjoints $\lambda(t) = \lambda_{nom}(t)$, and therefore are functions of t only.
The problem is non-singular if H_{uu} is invertible, and singular otherwise. Thus, the feedback law (15)-(17) is only defined for non-singular problems.

C. Link between NE approach and NCO tracking

As shown in Section II-B, the neighboring extremal approach minimizes $\delta^2\bar{J}$. In the following, it will be shown that it also forces the first-order variation of the NCO to zero, i.e. $\delta H_u = 0$.

Theorem 1: The NE solution of the variational optimization problem (10)-(11) forces the first-order variation of the NCO to zero, i.e. $\delta H_u = 0$.

Proof: The first-order variation of the adjoints λ obeys the following differential equation:

$$\dot{\delta\lambda} = -F_x^T\delta\lambda - \delta F_x^T\lambda - \delta L_x^T, \quad \delta\lambda(t_f) = \Phi_{xx}\delta x(t_f) \tag{18}$$

Using the fact that $\delta F_x = \sum_{k=1}^{n} \frac{\partial F_x}{\partial x_k}\delta x_k + \sum_{k=1}^{m} \frac{\partial F_x}{\partial u_k}\delta u_k$, noting that the same structure holds for δL_x, and regrouping to be able to introduce the Hamiltonian gives:

$$\dot{\delta\lambda} = -F_x^T\delta\lambda - H_{xx}\delta x - H_{xu}\delta u, \quad \delta\lambda(t_f) = \Phi_{xx}\delta x(t_f) \tag{19}$$

Comparing (14) and (19), it can be seen that $\bar{\lambda} = \delta\lambda$.

The first-order variation of H_u is given by:

$$\delta H_u = \delta(\lambda^T F_u) + \delta L_u = \delta\lambda^T F_u + \lambda^T \delta F_u + \delta L_u \tag{20}$$

Using similar expressions for δF_u and δL_u gives:

$$\delta H_u = \delta\lambda^T F_u + \delta u^T H_{uu} + \delta x^T H_{xu} \tag{21}$$

A comparison of (13) and (21) gives $\bar{H}_{\delta u} = \delta H_u$. Thus, since the NE approach forces $\bar{H}_{\delta u} = 0$, it also forces the first variation of the NCO to be zero, i.e. $\delta H_u = 0$.

□

III. Neighboring Extremal Approach for Singular Problems

A. Computing the singular input

Singular situations, i.e. those with $H_{uu} = 0$, are quite common. An important class of systems with that property is the class of input-affine systems with the integral cost independent of the inputs. Here, $F(x,u) = f(x) + g(x)u$ and $L_u = 0$. So, $H_u = \lambda^T g(x)$ does not depend on the inputs, leading to $H_{uu} = 0$. In such a case, the NE approach presente in the previous section cannot be used directly. To compute the optimal inputs from $H_u = 0$, the NCO need to be differentiated with respect to time along the trajectories of the system until u appears explicitly (the number of differentiations is termed the order of singularity).
In the case of a single-input affine system with the integral cost independent of the input, $H_u = \lambda^T g$ and $\dot{H}_u$ is needed (with $[f,g] = g_x f - f_x g$ noting the Lie bracket [5]):

$$\begin{aligned} \dot{H}_u &= \dot{\lambda}^T g + \lambda^T \dot{g} = -H_x g + \lambda^T g_x F = \\ &\quad -\lambda^T F_x g + \lambda^T g_x F = \lambda^T [F,g] = \\ &\quad \lambda^T([f,g] + [g,g]\,u) = \lambda^T [f,g] \end{aligned} \tag{22}$$

$\dot{H}_u$ being also independent of u, another differentiation is required. As $\dot{H}_u$ inherits the structure of H_u, the computation to get $\ddot{H}_u$ from $\dot{H}_u$ is the same as that for getting $\dot{H}_u$ from H_u :

$$\begin{aligned} \ddot{H}_u &= \dot{\lambda}^T [f,g] + \lambda^T [f,g]_x F = \\ &\quad \lambda^T([f,[f,g]] + [g,[f,g]]\,u) \end{aligned} \tag{23}$$

- If $[g,[f,g]] \neq 0$, the order of singularity of the system is 2, and the conditions $H_u = 0$, $\dot{H}_u = 0$ and $\ddot{H}_u = 0$ are equivalent to:

$$\lambda^T \left[\begin{array}{ccc} g & [f,g] & [f,[f,g]] + [g,[f,g]]\,u \end{array} \right] = 0 \tag{24}$$

  The input satisfying (24) is given by:

$$u_{sing} = -\frac{\lambda^T [f,[f,g]]}{\lambda^T [g,[f,g]]} \tag{25}$$

- If $[g,[f,g]] = 0$, more differentiations are required, until u appears explicitly.

As long as u does not appear explicitly, the initial structure of H_u is kept through the differentations, thus the computations are purely inductive, i.e. (noting $ad_f^k g = [f, ad_f^{k-1} g]$) the k^{th} differentiation of H_u is:

$$H_u^{(k)} = \lambda^T (ad_f^k g + \left[g, ad_f^{k-1} g\right] u) \tag{26}$$

The process of differentiation is terminated when $\left[g, ad_f^{\sigma-1} g\right] \neq 0$, where σ is the order of singularity of the system. The singular input is then defined by the condition:

$$\lambda^T \left[\begin{array}{ccccc} g & \ldots & ad_f^{\sigma-1} g & ad_f^{\sigma} g + \left[g, ad_f^{\sigma-1} g\right] u \end{array} \right] = 0 \tag{27}$$

The input satisfying (27) is given by:

$$u_{sing} = -\frac{\lambda^T ad_f^{\sigma} g}{\lambda^T \left[g, ad_f^{\sigma-1} g\right]} \tag{28}$$

A general analysis of the computations associated with singular problems is presented in [8].

B. Computing the NE controller

As the differentiation of H_u is needed for computing the singular input, to set up a NE controller, δH_u will be differentiated with respect to time until δu appears explicitly. This is where an interplay between the nonlinear dynamics and the linearized ones occurs as will be explained next. Since in the singular case $H_{uu} = 0$, δH_u reads:

$$\delta H_u = \delta\lambda^T F_u + \delta x^T H_{xu} \tag{29}$$

The time differentiation of δH_u is given by:

$$\frac{d}{dt}\delta H_u = \dot{\delta\lambda}^T F_u + \delta\lambda^T \frac{d}{dt} F_u + \dot{\delta x}^T H_{xu} + \delta x^T \frac{d}{dt} H_{xu} \tag{30}$$

Among the different terms, $\dot{\delta x}$ and $\dot{\delta\lambda}$ are obtained from the linearized dynamics, while the nonlinear dynamics is used for the time differentiations $\frac{d}{dt}F_u$ and $\frac{d}{dt}H_{xu}$, since the latter cannot be deduced from the linearization alone.
It can be computed from (21) and (30), that δH_u and $\frac{d}{dt}\delta H_u$ can be presented in the following form:

$$\delta H_u = \delta\lambda^T A_0^T + \delta x^T B_0^T + \delta u^T C_0^T \tag{31}$$

$$\frac{d}{dt}\delta H_u = \delta\lambda^T A_1^T + \delta x^T B_1^T + \delta u^T C_1^T \tag{32}$$

where $A_0 = F_u^T$, $B_0 = H_{ux}$, $C_0 = H_{uu}$ and $A_1 = \dot{A}_0 - A_0 F_x^T$, $B_1 = \dot{B}_0 + B_0 F_x - A_0 H_{xx}$, $C_1 = B_0 F_u - A_0 H_{xu}$. From (31) and (32), it can be seen that $\frac{d}{dt}\delta H_u$ inherits the structure of δH_u. Thus, by induction, the first-order variation δH_u and its time derivatives have the generic form:

$$\frac{d^k}{dt^k}\delta H_u = \delta\lambda^T A_k^T + \delta x^T B_k^T + \delta u^T C_k^T \tag{33}$$

where the recursive law to describe the k^{th} differentiation as a function of the $(k-1)^{\text{th}}$ differentiation can be written as :

$$A_k = \dot{A}_{k-1} - A_{k-1} F_x^T \tag{34}$$

$$B_k = \dot{B}_{k-1} + B_{k-1} F_x - A_{k-1} H_{xx} \tag{35}$$

$$C_k = B_{k-1} F_u - A_{k-1} H_{xu} \tag{36}$$

The time differentiation stops at $k = \bar{\sigma}$, when $C_{\bar{\sigma}}$ become invertible. Then, as before, the sweep method with $\delta\lambda = S\delta x$ can be used to compute the feedback law. It can be seen that the parts of the feedback law corresponding to

(15) and (17) remain unaltered, while the gain matrix (16) is replaced by:

$$K(t) = C_{\bar{\sigma}}^{-1}(A_{\bar{\sigma}}S + B_{\bar{\sigma}}) \tag{37}$$

As for equations (16) and (17), it should be emphasized here that K is evaluated along the nominal trajectories of the system, and therefore is a function of t only.
The first difference with non-singular problems is that, instead of inverting $C_0 = H_{uu}$, the equations are differentiated with respect to time until C_k becomes invertible.
It will be shown next that the number of differentiations needed to make δu appear in the differentiations of δH_u is the same as the number of differentiations of H_u needed to make u appear, i.e. $\bar{\sigma} = \sigma$. This is shown by proving the following:

$$\frac{d^k}{dt^k}\delta H_u = \delta\frac{d^k}{dt^k}H_u \tag{38}$$

Since $\forall k < \sigma$, H_u is not a function of u, the proof will be simplified by considering a general function G (representing in our case H_u) of x only.

Proposition 1: Consider the vector function $G(x)$, its first-order variation δG and the time differentiations of G along the dynamics $\dot{x} = F(x,u)$. Then, the following equality holds: $\delta\frac{d^k}{dt^k}G = \frac{d^k}{dt^k}\delta G \;\forall\; k$.
Proof: The proof proceeds by induction. For $k = 1$,

$$\begin{aligned}
\delta\frac{d}{dt}G &= \delta(G_x\dot{x}) = \delta(G_xF) \qquad (39)\\
&= \sum_{j=1}^{n}\sum_{k=1}^{n}\delta x_j\frac{\partial^2 G}{\partial x_j\partial x_k}F_k + G_xF_x\delta x\\
&\quad + G_xF_u\delta u\\
\frac{d}{dt}\delta G &= \frac{d}{dt}G_x\delta x = \sum_{j=1}^{n}\sum_{k=1}^{n}\delta x_j\frac{\partial^2 G}{\partial x_j\partial x_k}\dot{x}_k + G_x\dot{\delta x}\\
&= \sum_{j=1}^{n}\sum_{k=1}^{n}\delta x_j\frac{\partial^2 G}{\partial x_j\partial x_k}F_k + G_xF_x\delta x \qquad (40)\\
&\quad + G_xF_u\delta u
\end{aligned}$$

Simple inspection shows that equations (39) and (40) are identical. To continue the induction, suppose $\delta\frac{d^{k-1}}{dt^{k-1}}G = \frac{d^{k-1}}{dt^{k-1}}\delta G$. Then,

$$\begin{aligned}
\frac{d^k}{dt^k}\delta G &= \frac{d}{dt}\frac{d^{k-1}}{dt^{k-1}}\delta G = \frac{d}{dt}\delta\frac{d^{k-1}}{dt^{k-1}}G = \delta\frac{d}{dt}\frac{d^{k-1}}{dt^{k-1}}G\\
&= \delta\frac{d^k}{dt^k}G \qquad (41)
\end{aligned}$$

follows.

□

The above result is fairly intuitive as time differentiation and computation of the first-order variation are linear operators and therefore the order of the operations can be interchanged.

C. Satisfaction of boundary conditions

A second difference arises from the fact that, with the feedback gain (37), only $\frac{d^\sigma}{dt^\sigma}\delta H_u = 0$ is enforced. Thus, this does not guarantee that the lower derivatives $\frac{d^k}{dt^k}\delta H_u$ are 0 for $k < \sigma$. One possibility to guarantee this would be to force all the σ initial conditions to zero, i.e.

$$\left[\frac{d^k}{dt^k}\delta H_u\right]_{t=0} = \delta x(0)^T\left[A_kS + B_k\right]_{t=0}^{T} = 0 \quad \forall k < \sigma \tag{42}$$

This way, the feedback law would ensure $\delta H_u = 0$ for all t. However, in practice, this approach is not efficient since it is often not possible to enforce exactly the σ initial conditions $\delta x(0)^T(A_kS + B_k)_{t=0}^{T} = 0$.

Thus, since the handle for forcing $\delta H_u = 0$ is only through its σ^{th} derivative, fast asymptotically stable σ^{th}-order dynamics are imposed on δH_u so as to drive it quickly to zero:

$$\frac{d^\sigma}{dt^\sigma}\delta H_u = -\sum_{k=0}^{\sigma-1}\gamma_k\frac{d^k}{dt^k}\delta H_u \tag{43}$$

Replacing the terms in (43) by the expression found in (33) gives

$$\delta\lambda^TA_\sigma^T + \delta x^TB_\sigma^T + \delta u^TC_\sigma^T = -\sum_{k=0}^{\sigma-1}\gamma_k(\delta\lambda^TA_k^T + \delta x^TB_k^T) \tag{44}$$

This modifies only the $K(t)$ matrix of the feedbak law (15)-(17):

$$K(t) = C_\sigma^{-1}\left[(A_\sigma + \sum_{k=0}^{\sigma-1}\gamma_kA_k)S + B_\sigma + \sum_{k=0}^{\sigma-1}\gamma_kB_k\right] \tag{45}$$

The poles of δH_u dynamics (determined by the choice of γ_k) have to be chosen carefully. On the one hand, fast poles lead to large corrections that might invalidate the linear approximation. Moreover, fast poles render the feedback highly sensitive to noise and may thus lead to poor results in terms of optimality. On the other hand, the poles have to be fast enough so that the perturbations are rejected sufficiently fast compared to the final time t_f.

D. Summary of the development

It was shown in the previous section that the standard NE controller (for non-singular systems) forces the first variation of H_u to zero, i.e. enforces $\delta H_u = 0$.
In this section, it was established that for an optimization problem which is singular (with an order of singularity of σ) and thus for which the standard NE controller is not defined, the feedback control law defined by:

$$\begin{aligned}
\delta u &= -K\delta x \qquad (46)\\
K &= C_\sigma^{-1}\left[(A_\sigma + \sum_{k=0}^{\sigma-1}\gamma_kA_k)S + B_\sigma + \sum_{k=0}^{\sigma-1}\gamma_kB_k\right] \qquad (47)\\
\dot{S} &= -H_{xx} - SF_x - F_x^TS + H_{xu}K\\
&\quad + SF_uK, \quad S(t_f) = \Phi_{xx}(x(t_f)) \qquad (48)
\end{aligned}$$

forces δH_u to have the dynamics:

$$\frac{d^\sigma}{dt^\sigma}\delta H_u + \sum_{k=0}^{\sigma-1} \gamma_k \frac{d^k}{dt^k}\delta H_u = 0$$

where $\gamma_0, ..., \gamma_{\sigma-1}$ can be chosen arbitrarily.

IV. Illustrative example

A. Reaction system

The reaction system $A + B \rightarrow C$ and $2B \rightarrow D$ is considered in a semi-batch chemical reactor with the reactant B being fed, where C is the desired product and D an undesired side product [8]. The dynamics of the system can be described using the following equations:

$$\begin{aligned} \dot{c_A} &= -k_1 c_A c_B - u\frac{c_A}{V}, & c_A(0) = c_{A0} \quad (49)\\ \dot{c_B} &= -k_1 c_A c_B - 2k_2 c_B^2 & (50)\\ &\quad -u\frac{(c_B - c_{Bin})}{V}, & c_B(0) = c_{B0}\\ \dot{V} &= u, & V(0) = V_0 \quad (51) \end{aligned}$$

where c_A and c_B are the concentrations ($\frac{mol}{l}$) of the species A and B, respectively, V the volume (l), k_1 and k_2 the kinetic coefficients ($\frac{l}{mol\ h}$), u the inlet feed rate ($\frac{l}{h}$), c_{Bin} the inlet concentration ($\frac{mol}{l}$) and c_{A0}, c_{B0}, and V_0 the initial conditions with $c_{A0} = 0.72\ \frac{mol}{l}$, $c_{B0} = 0.0614\ \frac{mol}{l}$ and $V_0 = 1\ l$. The numerical values of the parameters are: $k_1 = 0.053\ \frac{l}{mol\ h}$, $k_2 = 0.128\ \frac{l}{mol\ h}$, $c_{Bin} = 5\ \frac{mol}{l}$. The final time is $t_f = 250\ s$.

B. Optimization problem

The optimization problem consists of maximizing $n_C(t_f) - n_D(t_f)$, i.e. the difference between the numbers of moles of C and D at final time. It can be shown that $n_C(t) = c_{A0}V_0 - c_A(t)V(t)$ and $n_D(t) = \frac{1}{2}(c_{B0}V_0 + c_{Bin}(V(t) - V_0) - c_B(t)V(t) - c_{A0}V_0 + c_A(t)V(t))$. Removing the constant terms in $n_C(t_f) - n_D(t_f)$, the optimization problem can be formulated as the following minimization problem:

$$\begin{aligned} \min_{u(t)} J &= (3c_A(t_f) + c_{Bin} - c_B(t_f))V(t_f) \quad (52)\\ &\quad s.t.\ \text{Dynamic equations } (49) - (51)\\ &\quad 0 \le u \le u_{max} \end{aligned}$$

The necessary conditions of optimality read:

$$H_u = -\lambda_1\frac{c_A}{V} - \lambda_2\frac{(c_B - c_{Bin})}{V} + \lambda_3 = 0 \quad (53)$$

Since H_u is independent of u, the problem is singular. The time derivatives of H_u take the form:

$$\begin{aligned} \dot{H}_u &= -\lambda_1\frac{k_1 c_A(c_B - c_{Bin})}{V} & (54)\\ &\quad -\lambda_2\frac{k_1 c_A(c_B - c_{Bin}) + 2k_2 c_B(c_B - 2c_{Bin})}{V} = 0 \end{aligned}$$

$$\begin{aligned} \ddot{H}_u &= \lambda_1(\frac{c_{Bin}k_1 c_A(k_1 c_A + 4k_2 c_B)}{V} & (55)\\ &\quad +2\frac{uk_1 c_A(c_B - c_{Bin})}{V^2})\\ &\quad +\lambda_2(\frac{c_{Bin}(4k_1 c_A k_2 c_B + 8k_2^2 c_B^2 + k_1^2 c_A^2)}{V} +\\ &\quad 2\frac{u(c_B - c_{Bin})(k_1 c_A + 2k_2(c_B - c_{Bin}))}{V^2}) = 0 \end{aligned}$$

Since u appears in the second time derivative of H_u, the order of singularity is $\sigma = 2$. Eliminating the adjoint variables from (53)-(55), the singular input reads:

$$u_{sing} = \frac{c_B V(k_1 c_A c_B - 2k_1 c_A c_{Bin} - 4k_2 c_B c_{Bin})}{2c_{Bin}(c_B - c_{Bin})} \quad (56)$$

The initial conditions are chosen so that the optimal solution consists only of a singular arc. The optimal state and input trajectories are shown in Figure 1.

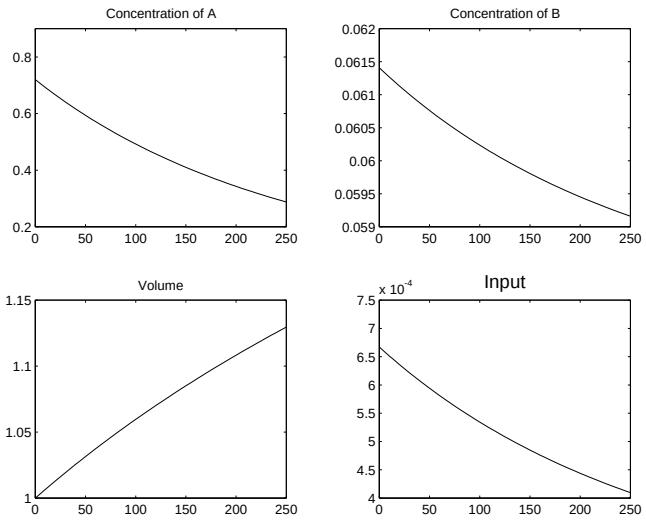

Fig. 1. Optimal state and input trajectories.

C. NE controller

The computation of the NE controller starts with the (symbolic) computation of the matrix A_k, B_k, C_k for $k = 1, 2$. For this purpose, equations (34)-(36) are used with A_0 and B_0 given by:

$$A_0 = \begin{bmatrix} -\frac{c_A}{V} & -\frac{c_B - c_{Bin}}{V} & 1 \end{bmatrix}$$

$$B_0 = \begin{bmatrix} -\frac{\lambda_1}{V} & -\frac{\lambda_2}{V} & \frac{\lambda_1 c_A + \lambda_2(c_B - c_{Bin})}{V^2} \end{bmatrix}$$

The expressions for C_k are:

$$C_0 = 0, \qquad C_1 = 0$$
$$C_2 = 2\frac{(c_B - c_{Bin})((\lambda_1 + \lambda_2)k_1 c_A + 2\lambda_2 k_2(c_B - c_{Bin}))}{V^2}$$

Note that, since C_2 is non-zero, its inverse can be used for calculating K in (45). The numerical values are computed along the nominal solution using the corresponding adjoints. The values of γ chosen to determine the dynamics of H_u are: $\gamma_0 = 0.0016$ and $\gamma_1 = 0.1$, corresponding to the two stable poles -0.02 and -0.08.

D. Simulation with perturbations

A perturbation of $\pm$ 10 % is introduced in the initial conditions. First, the nominal input is applied to the perturbed system. Then, the nominal input is used as a feedforward term with the NE controller providing the feedback term. The NE controller uses the numerical values of $K(t)$ and the nominal trajectories of the states. The following table displays the numerical results for four different simulations.

c_{A0}	c_{B0}	V_0	J_{opt}	J_{OL}	J_{FB}	Recovery (%)
0.720	0.0614	1	6.5556	6.5556	6.5556	-
0.648	0.0553	0.9	5.7801	5.7938	5.7802	99.78
0.648	0.0675	0.9	5.7686	5.7843	5.7687	99.30
0.792	0.0553	1.1	7.3685	7.3851	7.3686	99.27
0.792	0.0675	1.1	7.3542	7.3684	7.3542	99.86

TABLE I

RESULTS FOR DIFFERENT PERTURBATIONS OF THE INITIAL CONDITIONS. J_{opt} = OPTIMAL COST FOR THE PERTURBED SYSTEM, J_{OL} = COST OBTAINED BY APPLYING THE NOMINAL INPUT OPEN LOOP TO THE PERTURBED SYSTEM, J_{FB} = COST OBTAINED BY APPLYING THE NOMINAL INPUT ALONG WITH THE NE FEEDBACK TO THE PERTURBED SYSTEM.

It can be seen from Table 1 that the loss in optimality ($J_{OL} - J_{opt}$) is small (less than 0.5 % for 10 % variation in all initial conditions) since the problem is singular, i.e. $H_{uu} = 0$. So, only higher-order time derivatives contribute to the cost deviation. Furthermore, it can be seen that the proposed NE feedback is able to almost completely recover the loss in optimality ($J_{FB} \approx J_{opt}$).

The simulation results for the initial conditions $c_{A0} = 0.792\ \frac{mol}{l}$, $c_{B0} = 0.0553\ \frac{mol}{l}$, and $V_0 = 1.1\ l$ are depicted in Figure 2. It is clearly seen that the input and states of the perturbed system under NE feedback catch up quickly with the optimal trajectories calculated for the perturbed system. The true optimum for the perturbed system consists of a short constraint-seeking arc, $u = u_{max}$, so as to arrive at the sensitivity-seeking arc in minimum time. In the feedback solution, this arc is absent because the dynamics of δH_u (43) are chosen rather slow. However, it can be seen that the loss in optimality due to the time necessary to catch up is only of the order of 0.01%. Simulations with noise are not shown since the desired effects would be buried in noise.

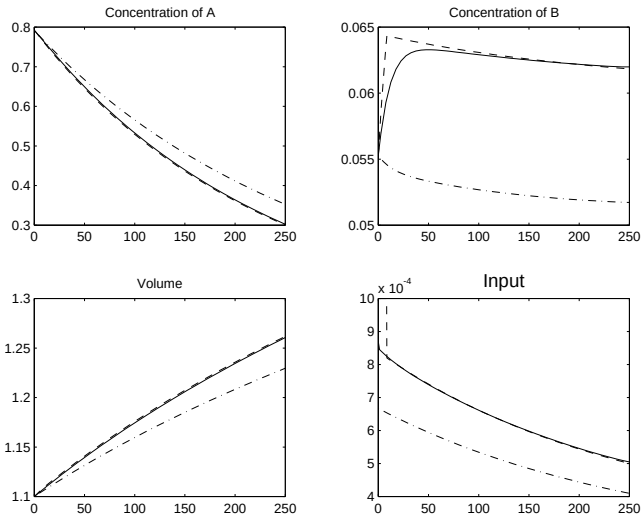

Fig. 2. Simulation with perturbed initial conditions ($c_{A0} = 0.792\ \frac{mol}{l}$, $c_{B0} = 0.0553\ \frac{mol}{l}$, and $V_0 = 1.1\ l$) (dashed line: optimal solution for the perturbed system; dotted-dashed line: perturbed system running open-loop with the nominal solution; solid line: system with NE feedback)

V. CONCLUSION

The problem of tracking the conditions of optimality on the various arcs of an optimal solution has been addressed recently. Towards this end, the current paper revisits the neighboring extremal approach for sensitivity-seeking arcs. A formal connection between this approach and NCO-tracking is established. From this connection, the NE approach is extended to the singular problems. An example has been used to illustrate the application of the method.

Despite the possibly heavy symbolic and numerical computations required to compute the NE feedback, it should emphasized that this effort is done off-line, with the on-line calculation corresponding to simple state feedback. Thus, this approach can also be applied to fast dynamic systems, independent of whether the system is singular or not.

REFERENCES

[1] L.T. Biegler. Solution of dynamic optimization problems by successive quadratic programming and orthogonal collocation. *Comp. Chem. Eng.*, 8(5):243–248, 1984.

[2] A.E. Breakwell, J.L. Speyer, and A.E. Bryson. Optimization and control of nonlinear systems using the second variation. *SIAM Journal on Control and Optimization*, 1:193–217, 1963.

[3] A.E. Bryson and Y.C. Ho. *Applied Optimal Control*. Ginn and Company, 1969.

[4] A. Cruse, W. Marquardt, A. Helbig, and J.-S. Kussi. Optimizing adaptive calorimetric model predictive control of a benchmark semi-batch reaction process. *IFAC World Congress, Barcelona, 2002*, 2002.

[5] A. Isidori. *Nonlinear Control Systems*. Springler-Verlag, 1989.

[6] A.Y. Lee and L. Markus. *Foundations of Optimal Control Theory*. The SIAM series, 1967.

[7] B. Srinivasan, D. Bonvin, E. Visser, and S. Palanki. Dynamic optimization of batch processes - II: Role of measurements in handling uncertainty. *Computers and Chemical Engineering*, 27:27–44, 2003.

[8] B. Srinivasan, S. Palanki, and D. Bonvin. Dynamic optimization of batch processes - I: Characterization of the nominal solution. *Computers and Chemical Engineering*, 27:1–26, 2003.