Use of Measurements for Enforcing the Necessary Conditions of Optimality in the Presence of Constraints and Uncertainty

G. François, B. Srinivasan, and D. Bonvin
Laboratoire d’Automatique
École Polytechnique Fédérale de Lausanne
CH-1015 Lausanne, Switzerland.

Abstract

Process measurements can be used in an optimization framework to compensate the effects of run-time uncertainty. Among the various options for input adaptation, a promising approach consists of directly enforcing the Necessary Conditions of Optimality (NCO) that include two parts: the active constraints and the sensitivities. In this paper, the variations of the NCO under parametric uncertainty are studied and used to design appropriate adaptation laws. The inputs are separated into constraint-seeking and sensitivity-seeking directions depending on which part of the NCO they enforce. In addition, the directional influence of uncertainty is used to reduce the number of variables to be adapted. The theoretical concepts are illustrated in simulation on the run-to-run optimization of a batch emulsion polymerization reactor.

Keywords Dynamic optimization, Run-to-run optimization, Necessary Conditions of Optimality, NCO tracking, Batch reactor, Emulsion polymerization.

1 Introduction

Optimization has received growing attention in the past years since it represents the natural choice for reducing production costs while guaranteeing quality and safety specifications. However, the main limitation of standard model-based optimization is the difficulty of developing reliable models [21]. Also, there are many process variations in run time that need to be accounted for. Industry typically copes with these uncertainties by adopting a conservative strategy that guarantees constraint satisfaction even in the worst case [21, 14, 3]. However, introducing conservatism is detrimental to the optimization objective. This conservatism can be reduced, and the objective improved, when measurements are used in the optimization framework. This idea has led to a paradigm shift from model-based [10] to measurement-based optimization [18, 16].

In measurement-based optimization, two major approaches can be distinguished:

- Measurement-based optimization based on model refinement: Here, measurements are used to refine the available model, and the updated model is used for optimization [5, 15].
- Measurement-based optimization with direct input update: Here, the measurements are used to directly update the inputs. One of the ideas herein deals with the tracking of the Necessary Conditions of Optimality (NCO) [20, 18], which will be elaborated in this work.

NCO-tracking methods are based on the following idea: Optimal inputs must satisfy the NCO. Due to uncertainty (model mismatch and process disturbances), the inputs computed using a process model may not meet the NCO for the real process. Hence, in this class of methods, the measurements are directly used to correct for uncertainty and enforce the necessary conditions of optimality.

Enforcing NCO is a relatively old idea [4, 13] that has been revisited recently [12]. The main contribution of the present work is to give importance to constraints and perform a parametric sensitivity analysis. There are different types of sensitivity studies depending on what is varied. The classical type considers the first- or second-order variations of the cost and constraints resulting from variations of the manipulated variables [1]. Also, the effect of constraints variations on the cost has been studied [6]. In this work, the variational study will be twofold: (i) With regard to the uncertain parameters as in [2], and (ii) from a measurement perspective in order to be able to implement the necessary input adjustments. Note that the term “measurements” normally represents only what comes directly from a sensor. However in this work, for the sake of simplicity, both measured and the estimated quantities will be referred to as “measurements”.

The variational analysis of the NCO provides analyti-
2 Static Optimization Problem

2.1 Problem Formulation

Consider the following static optimization problem:

\[
\begin{align*}
\min_{\pi} J &= \phi(\theta, \pi) \\
\text{s.t.} & \quad T(\theta, \pi) \leq 0
\end{align*}
\]

where \( J \) is the scalar cost function to be minimized, \( \pi \) the \( n_\pi \)-dimensional vector of inputs, \( \theta \) the \( n_\theta \)-dimensional vector of uncertain parameters, and \( T \) the \( \tau \)-dimensional vector of constraints. This formulation is quite general since both static optimization with equality constraints and dynamic optimization problems can be brought to this form (see Appendix).

By removing the dependent and/or inactive constraints, the problem can be reformulated such that (i) the constraints are linearly independent, and (ii) all constraints are active at the optimum. Such a reformulated problem will be considered in the sequel. This implies that, for feasibility, the number of inputs must be greater than or equal to the number of constraints, i.e. \( n_\pi \geq \tau \).

In addition, it will be assumed in this paper that the active set of constraints does not change with the uncertain parameters \( \theta \), i.e. for all values of \( \theta \) the same constraints are active at the optimum.

2.2 Necessary Conditions of Optimality

With the aforementioned assumptions, the NCO for Problem (1) are:

\[
\begin{align*}
T(\theta, \pi) &= 0 \\
\frac{\partial \phi(\theta, \pi)}{\partial \pi} + \nu^T(\theta) \frac{\partial T(\theta, \pi)}{\partial \pi} &= 0
\end{align*}
\]

where \( \nu \) are the \( \tau \)-dimensional Lagrange multipliers for the constraints. The NCO have two parts: (i) the constraint part (2), and (ii) the sensitivity part (3). The NCO are verified at the optimum corresponding to \( \pi^*(\theta) \).

3 Input Adaptation to Meet Constraints and Sensitivities

3.1 Variations of the NCO with Parametric Uncertainty

Let \( \theta^* \) be the nominal value of the parameter vector \( \theta \). Consider the variation \( \Delta \theta = \theta - \theta^* \) in the neighborhood of \( \theta^* \). This variation causes a corresponding variation in the optimal inputs \( \pi^* \), i.e. \( \Delta \pi^* = \pi^* (\theta + \Delta \theta) - \pi^* (\theta^*) \), and in the Lagrange multipliers \( \nu \), i.e. \( \Delta \nu = \nu (\theta + \Delta \theta) - \nu (\theta) \). For simplicity of notations, the distinction between \( \pi \) and \( \pi^* \) is dropped in the sequel.

The paper is organized as follows. Section 2 presents the NCO for a static optimization problem. In Section 3, the variations of the NCO due to parametric uncertainty are studied. The inputs are separated into constraint- and sensitivity-seeking directions that are adapted independently. In Section 4, the directional influence of uncertainty on the optimal inputs is investigated. Section 5 proposes an optimization scheme that implements the proposed directional adaptations. The methodology is illustrated in Section 6 through the run-to-run optimization of a batch emulsion polymerization reactor, while Section 7 concludes the paper.
The variation of the two parts of the NCO (2)-(3) resulting from changes in \( \theta \), \( \pi \) and \( \nu \) can be written as:

\[
\Delta T = \frac{\partial T}{\partial \pi} \Delta \pi + \frac{\partial T}{\partial \theta} \Delta \theta = 0
\]  

(4)

\[
\Delta \left( \frac{\partial \phi}{\partial \pi} + \nu^T \frac{\partial T}{\partial \pi} \right) = \Delta \pi^T \left( \frac{\partial^2 \phi}{\partial \pi^2} + \nu^T \sum_j \nu_j \frac{\partial^2 T_j}{\partial \pi^2} \right) + (\Delta \nu)^T \frac{\partial T}{\partial \pi} + \Delta \theta^T \left( \frac{\partial^2 \phi}{\partial \pi \partial \theta} + \nu^T \frac{\partial^2 T}{\partial \pi \partial \theta} \right) = 0
\]  

(5)

When the nominal inputs are applied to the perturbed system, i.e. \( \Delta \theta \neq 0 \), and the inputs are not adapted \( \Delta \pi = 0 \), the equations (4)-(5) will not be verified. Hence, the idea of adjusting the inputs \( \pi \) to satisfy the NCO.

### 3.2 Constraint- and Sensitivity-seeking Input Directions

Since the NCO have two parts, adaptation of certain input directions, \( \tilde{\pi} \), can be used to satisfy the constraint part (4), while other directions, \( \bar{\pi} \), can be used for the sensitivity part (5). Consider the \( (\tau \times n_\pi) \) matrix \( G = \partial T / \partial \pi \). This matrix represents the effect of the inputs on the active constraints in the neighborhood of the value of \( \pi \) for which the matrix is computed. From \( n_\pi \geq \tau \), the maximal rank of \( G \) is \( \tau \), if \( n_\pi > \tau \), then there exist some linear combinations \( \tilde{\pi} \) of the inputs that do not have any influence, at least locally, on the constraints and some linear combinations \( \bar{\pi} \) of the inputs that have an influence on the constraints.

With these definitions, \( \tilde{\pi} \) are such that \( \partial T / \partial \tilde{\pi} \) is a full-rank square matrix and \( \bar{\pi} \) are such that \( \partial T / \partial \bar{\pi} = 0 \). It follows that \( \tilde{\pi} \) is the \( \tau \)-dimensional vector of constraint-seeking input directions and \( \bar{\pi} \) the \( (n_\pi - \tau) \)-dimensional vector of sensitivity-seeking input directions. If \( n_\pi = \tau \), there will only be constraint-seeking input directions.

The separation \( \pi^T \rightarrow [\tilde{\pi} \bar{\pi}^T] \) is performed using SVD of the \( (\tau \times n_\pi) \) matrix \( G = \partial T / \partial \pi \). The matrices involved in the decomposition, \( G = G_G V_G^T \), can be partitioned as:

\[
S_G = [S_G 0], \quad V_G = [V_G \bar{V}_G]
\]  

(6)

with the constraint- and sensitivity-seeking input directions defined as follows:

\[
\tilde{\pi} = V_G^T \pi, \quad \bar{\pi} = \bar{V}_G \pi
\]  

(7)

Thus, \( \partial T / \partial \tilde{\pi} = U_G S_G \), is invertible, and \( \partial T / \partial \bar{\pi} = 0 \). Note that \( \pi = V_G \pi + \bar{V}_G \bar{\pi} \). With respect to \( \tilde{\pi} \) and \( \bar{\pi} \), the NCO (2)-(3) become:

\[
T'(\theta, \tilde{\pi}) = 0
\]  

(8)

\[
\frac{\partial \phi(\theta, \tilde{\pi}, \bar{\pi})}{\partial \tilde{\pi}} + \nu^T \frac{\partial T(\theta, \tilde{\pi}, \bar{\pi})}{\partial \tilde{\pi}} = 0
\]  

(9)

\[
\frac{\partial \phi(\theta, \tilde{\pi}, \bar{\pi})}{\partial \bar{\pi}} = 0
\]  

(10)

### 3.3 Adaptation of Input Directions

The adaptation laws for the constraint- and sensitivity-seeking input directions are obtained directly from the variations of the NCO. The results are summarized in the following theorem.

**Theorem 1** For the static optimization problem (1), the adaptation of the constraint-seeking input directions needed to meet the NCO in the presence of parametric uncertainty is given by:

\[
\Delta \bar{\pi} = - \left( \frac{\partial^2 \phi}{\partial \bar{\pi}^2} \right)^{-1} \frac{\partial^2 \phi}{\partial \bar{\pi} \partial \theta} \Delta \theta = - \left( \frac{\partial \phi}{\partial \bar{\pi}} \right)^{-1} T_m
\]  

(11)

where \( T_m \) is the measured value of the constrained quantities \( T \) before input adaptation. Similarly, the adaptation of the sensitivity-seeking input directions is given by:

\[
\Delta \tilde{\pi} = - \left( \frac{\partial^2 \phi}{\partial \tilde{\pi}^2} \right)^{-1} \frac{\partial^2 \phi}{\partial \tilde{\pi} \partial \theta} \Delta \theta = - \left( \frac{\partial \phi}{\partial \tilde{\pi}} \right)^{-1} \left( \frac{\partial \phi}{\partial \tilde{\pi}} \right)_m \Delta \theta
\]  

(12)

where \( \left( \frac{\partial \phi}{\partial \tilde{\pi}} \right)_m \) is the measured or estimated value of \( \left( \frac{\partial \phi}{\partial \tilde{\pi}} \right) \) before the adaptation of \( \tilde{\pi} \), but after the adaptation of \( \bar{\pi} \) to meet the constraints.

**Proof:** The variation of (8) can be rewritten as:

\[
\frac{\partial T}{\partial \tilde{\pi}} \Delta \tilde{\pi} + \frac{\partial T}{\partial \bar{\pi}} \Delta \bar{\pi} + \frac{\partial T}{\partial \theta} \Delta \theta = 0
\]  

(13)

Since \( \partial T / \partial \tilde{\pi} = 0 \) and \( \partial T / \partial \bar{\pi} \) is invertible, the first equality of (11) follows. From (4), the term \( \frac{\partial T}{\partial \theta} \Delta \theta \) can be interpreted as the value of \( T \), with no input adaptation (\( \Delta \pi = 0 \)), i.e. \( T_m \). Using \( T_m = \frac{\partial T}{\partial \theta} \Delta \theta \) gives the second equality of (11).

Similarly, the variation of (10) becomes:

\[
\frac{\partial^2 \phi}{\partial \tilde{\pi}^2} \Delta \tilde{\pi} + \frac{\partial^2 \phi}{\partial \tilde{\pi} \partial \theta} \Delta \bar{\pi} + \frac{\partial^2 \phi}{\partial \bar{\pi} \partial \theta} \Delta \theta = 0
\]  

(14)

Assuming \( \frac{\partial^2 \phi}{\partial \bar{\pi}^2} \) to be invertible, which is the case for non-singular problems, and replacing \( \Delta \tilde{\pi} \) by its expression from (11) gives the first equality in (12). The term in rectangular brackets in (12) indicates the sensitivity of \( \partial \phi / \partial \tilde{\pi} \) due to the direct effect of \( \Delta \theta \) and the effect of \( \Delta \bar{\pi} \) on \( \Delta \tilde{\pi} \). In other words, upon multiplication by \( \Delta \theta \), it corresponds to \( \left( \frac{\partial \phi}{\partial \tilde{\pi}} \right)_m \) as described in the theorem statement. This interpretation gives the second equality in (12).

The reason for labeling the constraint-seeking input direction \( \tilde{\pi} \) and the sensitivity-seeking input direction \( \bar{\pi} \) is...
evident from Theorem 1. The constraints $T(\theta, \bar{\pi}) = 0$ and the sensitivities $\partial \phi(\theta, \bar{\pi}, \bar{\pi})/\partial \bar{\pi} = 0$ are enforced using $\bar{\pi}$ and $\bar{\pi}$, respectively.

Note that equation (9) is only there to compute the Lagrange multipliers $\nu$, and does not indulge directly in the adaptation. In fact, making the adaptation independent of the Lagrange multipliers and their variations is the main reason for introducing the separation between constraint- and sensitivity-seeking input directions.

It is interesting to note the two different ways of expressing the adaptation laws (11) and (12). The first expression indicates how $\pi$ should be changed when $\Delta \theta$ is known, while the second expression does the adaptation from measurements of either the constrained quantities or the sensitivities. Thus, in a measurement-based optimization framework, the optimal solution in the presence of uncertainty can be implemented from measurements of $T_m$ and $(\partial \phi/\partial \bar{\pi})_m$, i.e. without having to know the model parameter variation $\Delta \theta$.

4 Input Adaptation with Directional Information on the Influence of Uncertainty

In this section, the effect of uncertainty is supposed to be known, i.e. it is possible to compute off-line the optimal inputs for different values of the uncertain parameters around the nominal parameter vector $\bar{\theta}$. Thus, it is assumed that the $(n_\pi \times n_\theta)$-dimensional sensitivity matrix $D = \partial \pi/\partial \theta$ is known. If $D$ and $\Delta \theta$ are known, the input update is given by:

$$\Delta \pi = D \Delta \theta$$

(15)

It follows from (7) and (15) that $\Delta \bar{\pi} = \bar{D} \Delta \theta$ and $\Delta \bar{\pi} = \bar{D} \Delta \theta$, with $D = V_G^T D$, $\bar{D} = V_G^T \bar{D}$. Comparing these expressions for $\Delta \bar{\pi}$ and $\Delta \pi$ to (11) and (12) gives:

$$\bar{D} = -\left(\frac{\partial T}{\partial \bar{\pi}}\right)^{-1}\frac{\partial T}{\partial \bar{\pi}}$$

(16)

$$\bar{D} = -\left(\frac{\partial^2 \phi}{\partial \pi \partial \theta} \right)^{-1} \left[ \frac{\partial^2 \phi}{\partial \pi \partial \bar{\pi}} - \frac{\partial^2 \phi}{\partial \bar{\pi} \partial \bar{\pi}} \frac{\partial T}{\partial \bar{\pi}} \right]$$

(17)

4.1 Separation of Constraints

In order to write the adaptation laws, it is important to know which constraints are affected by the uncertainty and which are not. This can be obtained from the $D$ matrix as described below.

It is seen from (4) that the sensitivity of $T$ with respect to the uncertain parameters contains two effects, i.e. the effect of $\Delta \theta$ over $\Delta \pi$ and the direct effect of $\Delta \theta$:

$$\frac{\partial T}{\partial \pi} \frac{\partial \pi}{\partial \theta} + \frac{\partial T}{\partial \theta} = G_D + E = 0$$

(18)

where $G = \partial T/\partial \pi$, $D = \partial \pi/\partial \theta$ and $E = \partial T/\partial \theta$. This equation allows computing $E$ from $G$ and $D$, where $E$ contains the information of how the constraints vary with uncertainty.

If $\tau \leq n_\theta$, all constraints are affected by the uncertainty. However, if $\tau > n_\theta$, there exist some combinations of constraints that are not affected by uncertainty. The separation between uncertainty-dependent and uncertainty-independent constraints is done via SVD of the $(\tau \times n_\theta)$-dimensional matrix $E = U_E S_E V_E^T$. Assuming $E$ to be of full rank, the matrices $U_E$ and $S_E$ can be partitioned as:

$$U_E = [\bar{U}_E \hat{U}_E], S_E = \begin{bmatrix} \hat{S}_E & 0 \\ 0 & 0 \end{bmatrix}$$

(19)

The uncertainty-dependent and uncertainty-independent constraints can be defined as:

$$\hat{T} = \bar{U}_E^T \bar{T}, \quad \hat{\bar{T}} = \bar{U}_E^T \hat{T}$$

(20)

Using these definitions results in $\frac{\partial T}{\partial \pi} = \bar{U}_E^T \frac{\partial \bar{T}}{\partial \pi}$, $\bar{D} = \bar{U}_E^T \bar{D} V_E^T = \hat{S}_E V_E^T$ and $\frac{\partial E}{\partial \pi} = U_E^T \frac{\partial E}{\partial \pi} = U_E^T E = \bar{U}_E^T \hat{S}_E V_E^T \bar{U}_E$. Hence, $\frac{\partial \hat{E}}{\partial \pi}$ is invertible from the non singularity of $S_E$ and $V_E$ (properties of SVD), and $\partial T/\partial \theta = 0$ since $\hat{U}_E$ and $\bar{U}_E$ are orthogonal. Note that $\hat{T}$ and $\hat{T}$ are of dimension $n_\theta$ and $(\tau - n_\theta)$, respectively.

4.2 Separation of the Input Directions

Using the directional information on uncertainty, an additional separation can be done that leads to the definition of four classes of input directions. The interest of this separation is obvious since uncertainty-independent input directions will not be affected by uncertainty and thus need not be adapted.

The separation is similar to that proposed in Section 4.1. It is based on the matrices $U_D$, $\bar{U}_D$, $\hat{U}_D$ and $\bar{U}_D$ which are defined from $D$ and $\bar{D}$ the same way that $\hat{U}_E$ and $\bar{U}_E$ are defined from $E$ in (19).

Figure 1 indicates the various directions in input space and the adaptations that are needed for meeting the
NCO in the presence of parametric uncertainty. It will be seen that an update is necessary only for $\hat{\pi}$ and $\hat{\pi}$. The former is adapted using information on the constraints $T$, while the latter uses sensitivity information on $\partial \phi/\partial \pi$ for adaptation.

Thus, the number of directions that are adapted is $\dim(\hat{\pi}) + \dim(\hat{\pi}) = 2n_\theta$. Note that, if the adaptation were not via the measurements but directly from knowledge of $\Delta \theta$, it would be sufficient to adapt only $n_\theta$ directions. Thus, using the measurements for adaptation increases the number of directions to be adapted by a factor 2. Furthermore, directional information is useful only when $n_\theta < \max(\tau, n_e - \tau)$, i.e. a no-update generation can be generated.

4.3 Adaptation of Input Directions

When information on $\Delta \theta$ is available, the adaptation along the four directions is given by:

$$\Delta \hat{\pi} = U_D^T D \Delta \theta \neq 0, \quad \Delta \hat{\pi} = U_D^T D \Delta \theta = 0 \quad (21)$$

$$\Delta \hat{\pi} = U_D^T D \Delta \theta \neq 0, \quad \Delta \hat{\pi} = U_D^T D \Delta \theta = 0 \quad (22)$$

However, the goal here is to express the adaptation laws in terms of constraints and sensitivities rather than $\Delta \theta$, which is given in the following theorem.

**Theorem 2** For the static optimization problem (1), the adaptation of the uncertainty-dependent input directions are given by:

$$\Delta \hat{\pi} = -\left(\frac{\partial \hat{\pi}^T \hat{T}}{\partial \hat{\pi}}\right)^{-1} \frac{\partial \hat{T}}{\partial \theta} \Delta \theta = -\left(\frac{\partial \hat{\pi}^T \hat{T}}{\partial \hat{\pi}}\right)^{-1} \hat{T}_m \quad (23)$$

$$\Delta \hat{\pi} = -\left(\frac{\partial \hat{\pi}^T \hat{T}}{\partial \hat{\pi}}\right)^{-1} \left[ \frac{\partial \hat{\pi}^T \hat{T}}{\partial \hat{\pi} \partial \theta} \right] \Delta \theta = -\left(\frac{\partial \hat{\pi}^T \hat{T}}{\partial \hat{\pi} \partial \theta} \right)^{-1} \left(\frac{\partial \hat{\pi}}{\partial \theta}\right)_m \quad (24)$$

where $\hat{T}_m$ is the value of $\hat{T}$ before input adaptation, and $(\frac{\partial \hat{\pi}}{\partial \theta})_m$ is the value of $(\frac{\partial \hat{\pi}}{\partial \theta})$ with $\hat{\pi}$ adapted to meet $\hat{T}$. No adaptation is needed for uncertainty-independent input directions:

$$\Delta \hat{\pi} = 0 \quad (25)$$

$$\Delta \hat{\pi} = 0 \quad (26)$$

**Proof:** Equations (25)-(26) are taken directly from (21)-(22).

Equation (4) can be written with respect to $\hat{T}$ as:

$$\frac{\partial \hat{T}}{\partial \pi} \Delta \hat{\pi} + \frac{\partial \hat{T}}{\partial \pi} \Delta \hat{\pi} + \frac{\partial \hat{T}}{\partial \theta} \Delta \pi + \frac{\partial \hat{T}}{\partial \theta} \Delta \theta = 0 \quad (27)$$

or, with $\Delta \pi = 0$ and $\frac{\partial \hat{T}}{\partial \theta} = 0$:

$$\frac{\partial \hat{T}}{\partial \pi} \Delta \hat{\pi} + \frac{\partial \hat{T}}{\partial \pi} \Delta \hat{\pi} = 0 \quad (28)$$

Also, it follows from (28):

$$\frac{\partial \hat{T}}{\partial \pi} \Delta \hat{\pi} + \frac{\partial \hat{T}}{\partial \pi} \Delta \hat{\pi} = 0 \quad (29)$$

Since $\partial \hat{T}/\partial \theta$ and $\partial \hat{\pi}/\partial \theta$ are invertible (the proof of the latter being rigorously similar to the one for $\partial \hat{T}/\partial \theta$ in Section 4.1, replacing $E$ by $D$, $T$ by $\pi$ and $T$ by $\hat{\pi}$), so is $\partial \hat{\pi}/\partial \hat{\pi}$. Due to the invertibility of $\partial \hat{T}/\partial \pi$, the first equality in (23) follows directly from (28). The second equality in (23) can be obtained by interpreting $\partial \hat{T}/\partial \pi$ as the value of $\hat{T}$ in the absence of input adaptation, i.e. $\hat{T}_m$.

For proving (24), (14) can be rewritten with respect $\hat{\pi}$:

$$\frac{\partial \hat{\pi}}{\partial \hat{\pi}} \partial \hat{\pi} \partial \theta \Delta \theta = \Delta \hat{\pi}$$

Since $\Delta \hat{\pi} = 0$ and $\Delta \hat{\pi} = 0$, (30) becomes:

$$\frac{\partial \hat{\pi}}{\partial \hat{\pi}} \partial \hat{\pi} \partial \theta \Delta \theta = \Delta \hat{\pi}$$

Using (23) for $\Delta \hat{\pi}$ leads to the first equality in (24). Using the same argument as in the proof of Theorem 1 for interpreting the sensitivity $(\frac{\partial \hat{\pi}}{\partial \theta})_m$ gives the second equality in (24).

5 Run-to-run Optimization Scheme

The previous section has shown that the optimal operation can be reached by adapting $\hat{\pi}$ and $\hat{\pi}$ to enforce $\hat{T} = 0$ and $\partial \phi/\partial \hat{\pi} = 0$, respectively. The use of integral control laws for this adaptation will be discussed in this section.

Introducing the diagonal gain matrices $K_1$ and $K_2$ of dimension $n_\theta \times n_\theta$, with $0 < K_1, K_2 < I$, the integral adaptation laws for constraint-seeking and sensitivity-seeking uncertainty-dependent input parameters can be written for the $k$th iteration as:

$$\dot{\hat{\pi}}(k) = \hat{\pi}(k-1) - K_1 \left(\frac{\partial \hat{T}}{\partial \pi}\right)^{-1} \hat{T}_m(k-1) \quad (32)$$

$$\dot{\hat{\pi}}(k) = \hat{\pi}(k-1) - K_2 \left(\frac{\partial \hat{\pi}}{\partial \theta}\right)_m (k-1) \quad (33)$$

Equations (32)-(33) are in fact (23)-(24) written in integral form. The diagonal matrices $K_1$ and $K_2$ are introduced to make the step cautious.
Several remarks are in order:

1. \( P_1 \) transforms \( T \) to \( \tilde{T} \) and \( P_2 \) transforms \( \tilde{\tau} \) to \( \tau \).

2. If the directional effect of uncertainty is unknown the projections based on \( D \) or \( E \) have no effect.

So, \( \tilde{U}_E \tilde{U}_E^T = I_r \) and \( \tilde{U}_D \tilde{U}_D^T = I_r \). This leads to
\( P_1 = I_r, \) \( P_2 = \tilde{V}_G \tilde{V}_G^T, \) and \( \tilde{P}_2 G^+ = G^+ \). Thus, in this case, \( P_1 \) and \( P_2 \) could be removed from the scheme.

3. Though the linearization of the system is only locally valid, the scheme has been shown to converge for a class of nonlinear systems [8].

4. The gradient is computed experimentally using finite differences, by perturbing \( \tilde{\tau} \) with a square wave.

5. As pointed out in Theorems, \( \left( \frac{\partial \phi}{\partial \tilde{\tau}} \right)_m \) needs to be evaluated under the condition \( T_m = 0 \) and after the adaptation of \( \tilde{\tau} \) has converged. This means that a time-scale separation is needed. Hence, \( \tilde{T}_m \) is enforced (and so is \( T_m \)) since \( \tilde{T}_m \) is uncertainty-independent and so is equal to zero) at a faster rate than \( \left( \frac{\partial \phi}{\partial \tilde{\tau}} \right)_m = 0 \).

6. This time-scale separation is realized by trying to enforce the constraints every run and sensitivities only every (say) ten runs (represented in dotted lines in Figure 2).

### 6 Run-to-run Optimization of a Batch Polymerization Process

The case study presented in this paper concerns the emulsion copolymerization of styrene/\( \alpha \)-methylstyrene in a batch reactor. The optimization objective is to minimize the batch time necessary to meet the performance specifications by manipulating the reactor temperature. The optimization strategy presented in the previous sections is implemented to handle parametric uncertainties.

#### 6.1 Tendency Model for an Emulsion Copolymerization Process

A tendency model that describes the main phenomena of the process is used in this study in order to determine the qualitative structure of the optimal solution. The assumptions made are the same as in [9]. The following equations are used:

\[
\begin{align*}
\dot{M} &= -R_p - k_p M p \frac{N_i}{N_p} \bar{n}, \\
\dot{N}_p &= \frac{1 + \frac{N_i}{N_p}}{N_p + \frac{N_i}{N_p}} + k_{irM} M p \frac{N_i}{N_p}, \\
\dot{Q}_0 &= \frac{\bar{V}_d (T_f - T)}{N_p}, \\
\dot{T} &= \frac{\bar{V}_d (T_f - T)}{N_p}, \\
\dot{T}_j &= \frac{\bar{V}_d (T_j - T)}{N_p} + \frac{U A (T_j - T)}{\rho \nu \bar{v}_j}, \\
\end{align*}
\]

where \( M, M_0, M_p \) are respectively the overall monomer concentration, the initial monomer concentration (styrene and 10% in mass of \( \alpha \)-methylstyrene) and the concentration of monomer in the particles. \( N_p, Q_0, N_i, c, \) and \( \bar{n} \) are the number of particles, the zeroth-order moment of the molecular weight distribution, the Avogadro number, the capturing efficiency of the particles with respect to micelles [11], and the average number of radicals per particle. \( T, T_j \) and \( T_m \) are the reactor, the jacket and the jacket inlet temperatures. \( V \) and \( V_j \) the reactor contents and the jacket volumes, \( \Delta H \) the polymerization reaction enthalpy. \( m_C \) and \( C_j \) the reactor and the cooling fluid heat capacities, \( \rho \) and \( F_l \) the cooling fluid density and the flowrate, \( U \) the heat transfer coefficient, and \( A \) the heat exchange surface. \( R_p \) and \( R_i \) represent the monomer consumption and the initiator decompositions rates. \( R_i = 2 k_i k_d I_0 \), where \( f_i \) is the efficiency factor of initiator decomposition and \( I_0 \) the initiator concentration. \( k_p, k_{irM} \) and \( k_d \) are the propagation, transfer to monomer, and initiation rate constants and are assumed to follow Arrhenius law. The concentration of the emulsifier is \( S = S_0 - k_e (X M_0) \bar{N} \frac{1 + X \bar{m}}{\bar{m}} \) where \( S_0 \) is the initial emulsifier concentration and \( k_e \) a constant. \( M_p \) has an expression that depends on conversion:

\[
M_p = \begin{cases} 
M_p \frac{1}{1 - X} & \text{if } X \leq X_c \\
M_p \frac{1}{1 - X} \frac{1 - X \bar{m}}{\bar{m}} & \text{if } X > X_c 
\end{cases}
\]

where \( X(t) = 1 - \frac{M(t)}{M_0} \) is the conversion, \( \rho_m \) and \( \rho_p \) are the monomer and polymer densities, \( M_M \) the monomer
molecular weight, and $X_c$ the critical conversion. The number average molecular weight $M_n(t)$ can be defined as $M_n(t) = M_M \frac{M(0) - M(t)}{Q_0(t)}$. The values of the parameters are given in Table 6.1.

### 6.2 Nominal Optimization

#### 6.2.1 Problem formulation

The optimization problem consists of determining the reactor temperature policy that minimizes the batch time subject to bounds on the reactor temperature and the jacket inlet temperature, and terminal constraints on conversion and the number average molecular weight. The batch time represents another decision variable. The problem can be expressed mathematically as follows:

$$
\min_{T(t), t_f} J = t_f 
$$

subject to

- dynamic system (35)
- $T_{\min} \leq T(t) \leq T_{\max} \quad T_{jmin_{\min}} \leq T_{jmin}(t)$
- $X(t_f) \geq X_{fa}$
- $M_n(t_f) \geq M_{nfa}$

where $T_{\min}$ and $T_{\max}$ are the bounds on the reactor temperature, $T_{jmin_{\min}}$ a lower bound on the jacket inlet temperature, $X_{fa}$ and $M_{nfa}$ minimal desired values at final time for the conversion and the number average molecular weight, respectively. Table 2 gives the values of the constraints.

<table>
<thead>
<tr>
<th>Constraint</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>$T_{\min}$</td>
<td>313 K</td>
</tr>
<tr>
<td>$T_{\max}$</td>
<td>343 K</td>
</tr>
<tr>
<td>$T_{jmin_{\min}}$</td>
<td>293 K</td>
</tr>
</tbody>
</table>

Table 2: Values of constraints

This dynamic optimization problem can be transformed into a static one (1). The details of this transformation are given in the Appendix.

#### 6.2.2 Sequence of arcs and input parameterization

The reactor temperature has two effects on the process. Increasing the reactor temperature accelerates the reaction but leads to shorter polymer chains and lower molecular weights. Since terminal constraints on both conversion and molecular weight are imposed, the optimal temperature profile represents an intrinsic compromise between these two effects.

![Figure 3: Nominal optimal temperature profile](image)

It is important to note that the mechanism of emulsion polymerization can be divided into three steps [17]: Particle nucleation, particle growth with monomer saturation and particle growth without monomer supply. These will be reflected in the optimal solution discussed below. The optimal solution obtained numerically and shown in Figure 3 consists of four arcs that can be interpreted as follows:

- **Arc 1**: Since polymer particles do not grow during the nucleation step, it is possible to accelerate the nucleation without affecting the molecular weight by increasing the temperature. As a result, the optimal reactor temperature is at $T_{\max}$.

- **Arc 2**: As soon as the particles start to grow, the optimal reactor temperature is somewhere between $T_{\min}$ and $T_{\max}$ due to the intrinsic compromise. Arc 2 implements the transition between Arc 1 and this intrinsic compromise in minimum time, by imposing $T_{jmin_{\min}}$, the lowest $T_{jmin}$ possible.

- **Arc 3**: During particle growth with monomer saturation, the reaction rate is nearly constant. Any increase in temperature will favor conversion at the expense of a smaller average molecular weight, and vice versa. It is thus best to keep the temperature constant at some optimal value.

- **Arc 4**: During particle growth without monomer supply, the reaction rate decreases with $M_p$. Hence,
optimal operation consists of increasing the reactor temperature to compensate for the increase in reaction rate. Using tools from differential geometry, it was shown that in this arc $T \propto T^2$. However, for the range of conversions considered, a linear temperature profile is sufficient, with its initial temperature fixed at the temperature of Arc 3 and its slope, $\alpha$, considered as an optimization variable.

In addition, the two terminal constraints regarding conversion and molecular weight must be active for the batch to be optimal. Thus, the optimal temperature profile can be parameterized using the five parameters $t_{sw1}$, $t_{sw2}$, $t_{sw3}$, $t_f$ and $\alpha$ shown in Figure 3 and the two active terminal constraints.

6.3 Optimization in the Presence of Uncertainty

6.3.1 Optimization without directional information regarding the uncertainty: Parametric uncertainty on the rate constants for both propagation and transfer to monomer is considered (±15% variation in $k_{p0}$ and ±10% variation in $k_{trM0}$). Also, 5% zero-mean gaussian measurement noise is added. As a result, a backoff (i.e. conservatism) is introduced in order not to violate the terminal constraints. This conservatism is introduced via the set points $X^*_{f_d} = 63\%$ and $M^*_{nf_d} = 2.1 \times 10^6 \text{ g/mol}$ provided to the controllers.

Adaptation starts from the conservative worst-case strategy, which will serve as the reference for comparison. The worst case occurs when $k_{p0}$ is at $-15\%$ and $k_{trM0}$ at $+10\%$ of their nominal values. Among the five parameters, the slope $\alpha$ is kept at its conservative value because of its low sensitivity. The other four parameters are adjusted on a run-to-run basis using batch-end measurements in order to satisfy the two active terminal constraints on conversion and number average molecular weight.

Note that, for this example, $\tau = 2$, $n_x = 4$, and $n_\theta = 2$. In this case, the $D$ matrix does not help reduce the number of variables to adapt. With the approach proposed in Section 3, two constraint-seeking input directions can be defined. Only the adaptation of the constraint-seeking directions is investigated in simulation. The sensitivity-seeking directions are not adapted since the associated benefit is small and thus buried into the measurement noise.

Three uncertain cases and the corresponding measurement-based optimization results are presented in Table 3. In Case (a), the uncertainty is relatively large and the improvement through adaptation is over 30%. In Cases (b) and (c), though the uncertainty is smaller, the improvement is still significant.

<table>
<thead>
<tr>
<th>Case</th>
<th>$k_{p0}$</th>
<th>$k_{trM0}$</th>
<th>5th batch</th>
<th>30th batch</th>
</tr>
</thead>
<tbody>
<tr>
<td>(a)</td>
<td>+15%</td>
<td>-10%</td>
<td>1.32 (28.9)</td>
<td>1.12 (39.6)</td>
</tr>
<tr>
<td>(b)</td>
<td>+5%</td>
<td>-5%</td>
<td>1.48 (20.5)</td>
<td>1.30 (30.0)</td>
</tr>
<tr>
<td>(c)</td>
<td>-5%</td>
<td>+5%</td>
<td>1.66 (10.5)</td>
<td>1.53 (17.5)</td>
</tr>
</tbody>
</table>

Table 3: Values of the uncertain parameters and cost $t_f$ in hours (with improvement in % from the conservative solution) after 5 and 30 batches

The run-to-run evolution of the cost function is shown in Figure 4. Though the scheme takes about 12 runs to converge to the optimal values, the major part of the improvement is done in the first few batches. With the backoff used, the controller gains are tuned such that no constraint violation occurs, as shown in Figure 4.

Figure 4: Evolution of the cost, the average molecular weight and the conversion: Case (a) (o, solid), Case (b) (+, dotted), Case (c) (△, dashed)

6.3.2 Optimization with directional information regarding the uncertainty: In this section, unidimensional parametric uncertainty (±15% variation in $k_{p0}$) is considered. With $n_\theta = 1$, $\tau = 2$ and $(n_x - \tau) = 2$, it is possible to illustrate how the directional information regarding the uncertainty can be used in the adaptation framework. Also, no measurement noise is considered since, with measurement noise, the adaptation of the sensitivity-seeking input parameters does not lead to distinguishable improvement.

Adaptation starts from the conservative worst-case
strategy, which will serve as the reference for comparison. The worst case occurs for \( k_{p0} - 15\% \). As before, the slope \( \alpha \) is kept at its conservative value and is not adapted.

In this case, \( G \) is of dimension \( 2 \times 4 \), \( D \) is of dimension \( 4 \times 1 \), \( \tau = 2 \), \( n_\pi = 4 \) and \( n_\theta = 1 \). So, the approach proposed in Section 4 leads to the definition of one direction of each type in Figure 1. For the simulation, + 15\% variation in \( k_{p0} \) and three adaptation scenarios are considered: (i) Adaptation of \( \hat{\pi} \) (ii) Adaptation of \( \hat{\pi} \) and \( \hat{\theta} \) and (iii) Adaptation of \( \hat{\pi}, \hat{\pi} \) and \( \hat{\theta} \). The set points for the various scenarios are so chosen that the results are comparable with those in the previous section, in the sense that the convergence is to a point around \( X_{f_d}^* = 63\% \) and \( M_{n_{fa}}^* = 2.1 \times 10^6 \text{ g/mol} \).

The optimization results are presented in Figure 5 and tabulated in Table 4. Several remarks are in order:

- As expected, most of the optimization is done through \( \hat{\pi} \) adaptation.
- Due to the only local nature of the decoupling between \( \hat{\pi} \) and \( \hat{\theta} \), full constraint satisfaction requires adaptation of \( \hat{\theta} \) as well. Improvement of the objective function through \( \hat{\theta} \) adaptation is negligible compared to what can be done via \( \hat{\pi} \) adaptation alone.
- Adaptation of \( \hat{\pi} \) is slow and leads to only very marginal cost improvement.
- Adaptation of \( \hat{\theta} \) is even slower and leads to even smaller cost improvement (the results are not presented in this paper).
- Though true convergence (in the average) requires a large number of batches, the approach of the optimum is fast, and most of the gain is achieved in just a few runs.
- Though the technique can be used in a noisy situation as well, it will no longer be possible to distinguish between 1.173h and 1.172h!

<table>
<thead>
<tr>
<th>Case</th>
<th>Adaptation</th>
<th>Batches</th>
<th>( t_f )</th>
</tr>
</thead>
<tbody>
<tr>
<td>(i)</td>
<td>( \hat{\pi} )</td>
<td>20</td>
<td>1.197 (32.24)</td>
</tr>
<tr>
<td>(ii)</td>
<td>( \hat{\pi}, \hat{\theta} )</td>
<td>20</td>
<td>1.173 (33.62)</td>
</tr>
<tr>
<td>(iii)</td>
<td>( \hat{\pi}, \hat{\theta}, \hat{\pi} )</td>
<td>300</td>
<td>1.172 (33.68)</td>
</tr>
</tbody>
</table>

Table 4: Adaptation strategies and cost \( t_f \) [hours] (improvement in % from the conservative solution)

In Case (iii), the two-time-scale adaptation is used, with the adaptation of the sensitivities starting only after the constraints have converged. The first batch where the sensitivities are adapted is the 10th one. It is interesting to note in Figure 5 the change away from the constraints caused by the perturbation of \( \hat{\pi} \), which illustrates that the decoupling is not perfect and the two-time-scale strategy is indeed necessary for convergence. Though, in Case (iii), the algorithm takes about 300 batches to converge, only the first 20 batches are represented in the corresponding figures.

7 Conclusions

The variations of the NCO to parametric uncertainty have been investigated. This analysis, along with the directional influence of uncertainty, leads to the definition of four types of input directions. A methodology based on singular value decomposition has been proposed to generate these directions. The applicability and performance of the proposed optimization approach are illustrated through the dynamic optimization of a batch emulsion polymerization process.

The use of directional information regarding uncertainty to perform input separation looks appealing. However, it can only be used when the number of uncertain parameters is small relative to the number of input parameters. Furthermore, it is important to notice that the methodology proposed to generate the input directions is only locally valid. A perspective for future research would be to use the directional information differently, for instance by defining nonlinear transformations so as to extend the applicability domain. Another interesting extension would be to investigate the variations of the NCO to parametric uncertainty for the case of dynamic optimization problems with infinite-dimensional inputs, i.e. without imposing a parameterization.
Appendix - Transformation of a Dynamic Optimization Problem to a Static Optimization Problem in Reduced Form

Consider the following free-terminal time dynamic optimization problem with terminal cost and path and terminal inequality constraints:

\[
\min_{t_f, u(t)} J = \phi(x(\theta), t_f) \\
\text{s.t.} \quad \dot{x} = F(x, \theta, u), \quad x(0) = x_0 \\
S(x, \theta, u) \leq 0 \\
T(x(\theta, t_f)) \leq 0
\]  

where \(J\) is the performance index, \(x\) the \(n\)-dimensional state vector, \(u\) the \(m\)-dimensional input vector, \(F\) the system equations, \(x_0\) the initial conditions, \(\theta\) the \(n_\theta\)-dimensional vector of uncertain parameters, \(\phi\) the terminal cost function, \(S\) the \(\zeta\)-dimensional vector of path constraints, and \(T\) the \(\tau\)-dimensional vector of terminal constraints.

The terminal-cost dynamic optimization problem (38) can be transformed into a static one that reads as in (1) upon:

- assuming that the path constraints are satisfied using on-line control [19]
- considering the input parameterization \(u(t) = U(x(t), \pi)\),
- noting that the states dynamics can be seen as an equality constraint: \(x(\theta, t_f) = x_0 + \int_0^{t_f} F(x, \theta, U(x, \pi))dt\) (which is always satisfied, and then can be removed from the optimization problem)
- noting that \(t_f\) can be included in the vector of parameters.

References