A Feedback-based Optimization Method for Uncertain Batch Processes

Jing Zhu, Lingjian Ye*, Wanqing Tao and Xiushui Ma
Ningbo Institute of Technology, Zhejiang University, 315100, Ningbo, China
*Corresponding author

Abstract—Optimal control of batch processes is often implemented in an open-loop manner with an online optimizer. In this paper, a feedback-based optimization method is proposed for batch processes which suffer from parametric uncertainties. Firstly, the optimality conditions and the expression for optimal input are derived based on the uncertain parametric model, then the output measurements and corresponding optimal inputs are collected via off-line simulation. Then, the explicit control law is obtained through regression, which is implemented in a feedback manner for optimization purpose. Case study on a fed-batch reactor indicates that the proposed approach can attain good optimizing performance in a wide range of uncertainties.

Keywords—batch process; dynamic optimization; output feedback; uncertainty

I. INTRODUCTION

Batch process is a time-varying system without steady state points. Typically, optimization of batch processes involves technique of dynamic optimization, which can be classified as the analytical and numerical approaches [1].

In the numerical approach, the infinite dimensional trajectories along the time-axis are discretized into finite dimensional variables, a static nonlinear programming problem is then formulated and solved numerically. Based on the variables discretized, the sequential approach and simultaneous approach can be distinguished [1]. In the former approach, the discretization is only imposed on the input variables, other process variables are computed through forward integration. The latter approach discretizes both the input and state variables, the differential equations are approximated and used as constraints to improve the speed for optimization iterations. Both the two approaches are widely used for optimization of batch processes.

On the other hand, the batch processes suffer from various disturbances [2], including: (1) uncertain initial conditions of state variables; (2) uncertain parameters in the process model. Both the two cases are referred as the parametric uncertainties. One common way to deal with this issue is the two-step approach, that is, the uncertain parameters are firstly estimated (using measurements) and then the optimization problem is resolved on basis of the updated model [3,4]. However, the approach requires sufficient input excitations [5] and the computation burden is high due to the frequent re-optimizations. In the necessary conditions of optimality (NCO) tracking approaches [2,6], the input trajectories are parameterized into a few finite decision variables and sub-arcs, based on the nominal condition of a reference model, then the NCO are enforced using the parameterized variable from batch to batch. In another class of neighboring-extremal control (NEC) [7], the second order variation objective function is minimized, which in essence enforces the first order variations of NCO to zeros.

In this paper, we propose a new feedback-based optimization method for uncertain batch processes. The input adaptation law is established based on the output measurements and optimal input arcs. The input law follows an explicit relationship between the outputs and optimization solutions, which is quick to respond online.

II. OPTIMIZATION OF BATCH PROCESSES

The following class of batch processes with parametric uncertainties is considered

\[
\begin{align*}
\min_{x(t), \theta} & \phi(x(t_f), \theta) \\
\text{s.t.} & \quad \dot{x} = F(x, u, \theta), \quad x(0) = x_0 \\
& \quad y = h(x, \theta) + v
\end{align*}
\]

where \( \phi \) is the cost function, \( u, x, y \) and \( \theta \) are the manipulated variables, states, outputs and uncertain parameters, respectively. \( t_f \) is the batch time, \( F \) and \( h \) are the dynamic model and output model, \( v \) is the measurement noise.

Based on the Pontryagin Maximum Principle [8], the following Hamilton function is minimized

\[
\begin{align*}
\min_{x(t), \theta} & \ H(t) = \lambda^T F(x, u, \theta) \\
\text{s.t.} & \quad \dot{x} = H_x, \quad x(0) = x_0 \\
& \quad \lambda^T = -H_u, \quad \lambda^T(t_f) = \phi(t_f)
\end{align*}
\]

where \( \lambda \) is the co-states, and \((\ast)\) stands for the partial derivative. The necessary conditions of optimality are

\[
H_u = \lambda^T F_u(x, u, \theta) = 0
\]

from which we know that the optimal input can be determined via condition \( \lambda^T F_u = 0 \) when \( u \) is within the feasible region. However, the optimal input are coupled with \( x, \lambda \) and \( \theta \).
To eliminate $\lambda$, differentiating (3) with respect to time gives

$$\frac{dH_u}{dt} = \lambda^T \left( \frac{\partial F}{\partial x} F - \frac{\partial F}{\partial x} F_u + \frac{\partial F}{\partial u} \dot{u} \right)$$  \hspace{1cm} (4)

Defining the operator $\Delta$ as

$$\Delta \phi = \phi_x F - F_x \phi + \phi_u \dot{u}$$  \hspace{1cm} (5)

Imposing $\Delta$ on (4), we have

$$\frac{dH_u}{dt} = \lambda^T \Delta F_u$$  \hspace{1cm} (6)

The time derivative is continued in a similar way, which follows

$$\frac{d^k H_u}{dt^k} = \lambda^T \Delta^k F_u$$  \hspace{1cm} (7)

Since $H_u=0$ should be always satisfied, time derivatives to any order of $H_u$ should be 0. Using this condition and differentiating to order $n_u-1$, these equations are collectively arranged, the following holds

$$\lambda^T \left[ F_u \quad \Delta F_u \quad \cdots \quad \Delta^{n_u-1} F_u \right] = \lambda^T \Gamma = 0$$  \hspace{1cm} (8)

Since $\lambda \neq 0$, $det(\Gamma)=0$ follows to satisfy (8), which is used to get a decoupled optimal input by eliminating the co-states $\lambda$.

### III. FEEDBACK-BASED OPTIMIZATION

#### A. Mapping Function of Optimal Inputs

As has been mentioned, $\lambda$ can be decoupled from the optimal input expressions, which however still contains the unknown $\theta$. Let the expression be

$$u_{opt}(t) = g(x, \theta)$$  \hspace{1cm} (9)

where $g$ is the nonlinear mapping obtained from $det(\Gamma)=0$, which is assumed to be differentiable.

In the conventional two-step approach, $\theta$ is first estimated and then re-optimization is carried out. However, here we consider using off-line calculations in prior, and derive state/output feedback law for optimization purpose. Basically, the feedback law is established as follows. The output model of $y$ is utilized to estimate both $x$ and $\theta$

$$\hat{x} = h^{-1}_x(y - \nu)$$  \hspace{1cm} (10)

where (10) is a typical state estimation problem. Inserting (10) and (11) into (9) yields

$$u_{opt}(t) = g\left(h^{-1}_x(y - \nu), h^{-1}_u(y - \nu)\right)$$  \hspace{1cm} (12)

The shortcoming of above analytic way is that the nonlinear equations are coupled and the inverse function $h^{-1}$ is complicated to derive. Commonly, linearization at a particular point is considered, which may however lead to bad performance when the process is strongly nonlinear. To this end, in this paper we consider the numerical approach, which builds a feedback law covering across the whole operating space. The feedback law follows a form of

$$\hat{u}_{opt} = \hat{g}(y, \alpha)$$  \hspace{1cm} (13)

where $\alpha$ is a vector to be determined in the feedback law. The rationale of this idea lies in the fact that the outputs $y$ reflects the information of unknown $\theta$ and the feedback law avoids online estimates.

#### B. Regression Method

Denote the sampling data as $\{y_i, u_{opt,i}\}$, $i=1, \ldots, N$, where $N$ is the number for data samples. In the following, the polynomial regression is considered as an example. When $\hat{g}(y, \alpha)$ is taken as the first order polynomial, i.e. a linear function, define

$$Y = \begin{bmatrix} y_1^T \\ \vdots \\ y_N^T \end{bmatrix}, \quad U = \begin{bmatrix} u_{opt,1}^T \\ \vdots \\ u_{opt,N}^T \end{bmatrix}$$  \hspace{1cm} (14)

The least square criterion solves the optimal parameters $\alpha$ as

$$\alpha = (Y^T Y)^{-1} Y U$$  \hspace{1cm} (15)

When the second order polynomial is considered, $Y$ is defined as

$$Y = \begin{bmatrix} 1 & y_1^T & \text{vec}^T(y_1, y_1^T) \\ \vdots \\ 1 & y_N^T & \text{vec}^T(y_N, y_N^T) \end{bmatrix}$$  \hspace{1cm} (16)

where vec stands for vectorizing a matrix into a vector. (15) still follows for optimal $\alpha$. Since the loss of objective function can be calculated as

$$\delta \phi = 0.5 \delta u^T H_u \delta u$$  \hspace{1cm} (17)
where $\delta$ denotes deviations from the optimum and $H_{uu}$ is the Hessian of Hamilton function, the better the regression accuracy, the better optimizing performance.

C. Implementation Procedure

In summary, the proposed method is carried out as follows. (1) the optimality conditions are solved based on the process model, the co-states $\lambda$ are eliminated in the optimal inputs $u_{opt}$, which contain the states $x$ and uncertain parameters $\theta$; (2) set different values of $\theta$ in the variations ranges, run simulations to obtain simulations $y$, which are collected in $Y$; (3) Select a particular regression structure and determine the parameters $\alpha$, the regression model is used as the feedback-based control law. During this step, the R$^2$ regression index can be used for determining the model structure, an R$^2$ close to 1 is favored.

IV. CASE STUDY ON A FED-BATCH REACTOR

A. Process Description

The fed-batch reactor involves the following reactions

\[
A + B \rightarrow C \\
2B \rightarrow D
\]  

(18)

where A and B are reactants, C is the product and D is the byproduct. B is fed during the batch time the feedrate as $u(t)$.

The model equations are

\[
\frac{dx}{dt} = F(x) - g(x) + \mu
\]

(19)

\[
\dot{V} = u, \quad V(0) = V_0
\]

(20)

\[
\dot{c}_C = (c_{ Ao} V_0 - c_D V) / V
\]

(21)

\[
c_D = ((c_A + c_{ Bin} - c_B)V - (c_{ Ao} + c_{ Bin} - c_{ Bo})V_0) / (2V)
\]

(22)

The symbols are explained in Table I. Among these parameters, the reaction coefficient $k_1$ is uncertain, which may vary between 0.02 and 0.08.

B. Optimality Conditions

The dynamic model for the reactor is arranged as

\[
\frac{dx}{dt} = F(x) - g(x) + \mu
\]

(23)

The optimality condition $H_u$ is then

\[
H_u = \lambda^T F_u, \quad F_u = \begin{bmatrix} -k_1 c_c c_B \\ -k_1 c_c c_B - 2k_2 c_c^2 + u(c_{ Bin} - c_B) / V, c_B(0) = c_{ Bin} \\ 0 \end{bmatrix}
\]

(24)

\[
\Delta F_u = \begin{bmatrix} k_1 c_c (c_{ Bin} - c_B) / V \\ k_1 c_c (c_{ Bin} - c_B) + 2k_2 c_c (2c_{ Bin} - c_B) / V \\ 0 \end{bmatrix}
\]

(25)

\[
\Delta^2 F_u = \begin{bmatrix} c_{ Bin} (k_1 c_c + 4k_2 c_c^2) / V \\ c_{ Bin} (4k_1 c_c + 8k_2 c_c^2) + 8k_2 c_c^2 + k_2 c_c^2 / V \\ 0 \end{bmatrix}
\]

\[
-2(c_{ Bin} - c_B) (k_1 c_c - 2k_2 (c_{ Bin} - c_B) u) / V
\]

(26)

\[
\min_{u(t)} \phi = (c_D(t_f) - c_c(t_f)) V(t_f)
\]

which implies maximizing the product C and meanwhile minimizing the byproduct D. The available measurements are

\[
y = \begin{bmatrix} c_A \\ c_B \\ V \end{bmatrix}
\]

(27)

which are all of which are sampled for every 0.1 min.

The cost function is expressed as

\[
\min_{u(t)} \phi = (c_D(t_f) - c_c(t_f)) V(t_f)
\]

(28)

\[
\Delta F_u = \begin{bmatrix} k_1 c_c (c_{ Bin} - c_B) / V \\ k_1 c_c (c_{ Bin} - c_B) + 2k_2 c_c (2c_{ Bin} - c_B) / V \\ 0 \end{bmatrix}
\]

(29)

\[
\Delta^2 F_u = \begin{bmatrix} c_{ Bin} (k_1 c_c + 4k_2 c_c^2) / V \\ c_{ Bin} (4k_1 c_c + 8k_2 c_c^2) + 8k_2 c_c^2 + k_2 c_c^2 / V \\ 0 \end{bmatrix}
\]

(30)

\[
-2(c_{ Bin} - c_B) (k_1 c_c - 2k_2 (c_{ Bin} - c_B) u) / V
\]
Therefore, the optimality condition becomes
\[
\det[\Gamma] = \begin{bmatrix} F_a & \Delta F_a & \Delta^2 F_a \end{bmatrix} = 0
\] (30)

The following optimal input is derived
\[
u_{\text{opt}} = \frac{4k_c c_B^2 c_{\text{Bin}} V - k_c c^2 c_B V + 2k_c c^2 c_{\text{Bin}} V}{2c_{\text{Bin}}(c_{\text{Bin}} - c_B)}
\] (31)

which contains the uncertain \(k_1\). Since \(k_1\) is unknown online, above expression cannot be used directly.

C. Regression Model for Feedback-based Optimization

To obtain the data samples for regression, the variation range of \(k_1\) is divided with an interval of 0.005, each of which is implemented with ramp \(u(t)\), whose initial value and terminal values are randomly chosen between 0 and 0.001. Dynamic simulations are run to generate data sample for measurements. The associated optimal inputs are also stored, which are calculated using (31).

Linear regression is first considered, the following is obtained
\[
\hat{u}_i = 0.0086 - 0.0020c_d + 0.0107c_B - 0.0073V
\] (32)
with \(R^2=0.79\). The second order regression is further considered
\[
\hat{u}_i = 0.26 - 0.066c_d - 0.1c_B - 0.45V + 0.002c_d^2 + 0.027c_c c_B + 0.056e_V - 0.1c_d^2 + 0.1c_B V + 0.19V^2
\] (33)
with \(R^2=0.88\), which improves the regression quality as compared to the linear case. Alternatively, we further consider
\[
\mathcal{F} = \begin{bmatrix} c_d & c_B & V & c_d(t-1) & c_B(t-1) & V(t-1) \end{bmatrix}
\] (34)
by using the old measurements. A linear regression on (34) gives
\[
\hat{u}_i = -0.0001 - 0.098c_d - 1.9c_B + 9.4V + 0.1c_d(t-1) + 1.9c_B(t-1) - 9.4V(t-1)
\] (35)
with \(R^2=0.99\), which is very satisfactory.

D. Validations

The derived feedback laws of \(\hat{u}_i, \hat{u}_s\) and \(\hat{u}_3\) are applied to the fed-reactor and their performances are compared for different \(k_i\). The results are summarized in Table II. It turns out that when \(k_i\) is between 0.03 and 0.04, the performances are similar and all close to the true optimum. However, when \(k_i\) is distant, the situation is different. \(\hat{u}_i\) is the worst, for which when \(k_i\) is 0.02 and 0.08, the loss of cost function is 0.0344 and 0.0744, respectively. For \(\hat{u}_s\), the loss is 0.0157 and 0.012 for the mentioned two cases, respectively. \(\hat{u}_3\) is the best whose performance is almost close to the true optimum for all \(k_i\) in the variation range.

TABLE II. OPTIMIZATION PERFORMANCE OF DIFFERENT LAWS

<table>
<thead>
<tr>
<th>(k_i)</th>
<th>(\hat{u}_{\text{opt}})</th>
<th>(\hat{u}_1)</th>
<th>(\hat{u}_2)</th>
<th>(\hat{u}_3)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.02</td>
<td>-0.0651</td>
<td>-0.0305</td>
<td>-0.0492</td>
<td>-0.0648</td>
</tr>
<tr>
<td>0.03</td>
<td>-0.1392</td>
<td>-0.1266</td>
<td>-0.1332</td>
<td>-0.1392</td>
</tr>
<tr>
<td>0.04</td>
<td>-0.2026</td>
<td>-0.1999</td>
<td>-0.2009</td>
<td>-0.2027</td>
</tr>
<tr>
<td>0.05</td>
<td>-0.2569</td>
<td>-0.2547</td>
<td>-0.2566</td>
<td>-0.2570</td>
</tr>
<tr>
<td>0.06</td>
<td>-0.3036</td>
<td>-0.2918</td>
<td>-0.3070</td>
<td>-0.3035</td>
</tr>
<tr>
<td>0.07</td>
<td>-0.3436</td>
<td>-0.3090</td>
<td>-0.3378</td>
<td>-0.3435</td>
</tr>
<tr>
<td>0.08</td>
<td>-0.3781</td>
<td>-0.3037</td>
<td>-0.3661</td>
<td>-0.3780</td>
</tr>
<tr>
<td>0.02</td>
<td>-0.0651</td>
<td>-0.0305</td>
<td>-0.0492</td>
<td>-0.0648</td>
</tr>
</tbody>
</table>

FIGURE I. INPUT RESPONSE FOR DIFFERENT FEEDBACK LAWS

The response of input trajectories for \(k_i=0.02, 0.04, 0.06\) and 0.08 are shown in Fig. 1. It is evident that although \(\hat{u}_s\) and \(\hat{u}_3\) reasonably work, the inputs resulted from the feedback laws deviate from the optimal arcs. However, \(\hat{u}_3\) is almost identical to the true optimal trajectory, for all the cases shown in Fig.1.

V. CONCLUSIONS

In this paper, we proposed a feedback-based optimization method for batch processes with parametric uncertainties. The optimality conditions and the expression for optimal input are derived based on the uncertain parametric model, then the output measurements and corresponding optimal inputs are collected via off-line simulation. The feedback law is obtained through regression and used for online optimization. The fed-batch reactor case confirms the effectiveness of the proposed scheme.
ACKNOWLEDGMENT

This work was supported by Zhejiang Xinmiao Talent Program, Ningbo Natural Science Foundation (2015A610151), Ningbo Education Planning Project (2016YGH032) and Talent project of Zhejiang Association of Science and Technology (2017YCGC014).

REFERENCES


