

文章编号: 1001-0920(2013)11-1697-05

一种基于输出反馈的显式实时优化控制方法

叶凌箭¹, 马修水¹, 宋执环²

(1. 浙江大学宁波理工学院, 浙江宁波 315100; 2. 浙江大学控制科学与工程学系, 杭州 310027)

摘要: 针对一阶最优性必要条件跟踪法优化间隔较长的缺点, 提出一种基于输出反馈的显式实时优化方法. 对系统进行不同工况下的离线优化, 经函数拟合得到最优控制输入与输出变量的显式回归模型, 直接应用于实时优化, 避免了在线梯度估计. 研究一个连续搅拌釜式反应器的反应过程, 并对比两种方法的优化效果, 结果验证了所提出方法的实际使用效果.

关键词: 实时优化; 输出反馈; 一阶最优性必要条件跟踪; 优化控制

中图分类号: TP273

文献标志码: A

An explicit optimizing control approach based on output feedback

YE Ling-jian¹, MA Xiu-shui¹, SONG Zhi-huan²

(1. Ningbo Institute of Technology, Zhejiang University, Ningbo 315100, China; 2. Department of Control Science and Engineering, Zhejiang University, Hangzhou 310027, China. Correspondent: YE Ling-jian, E-mail: ljye@nit.zju.edu.cn)

Abstract: To circumvent the shortcoming of necessary conditions of optimality(NCO) tracking, namely that the optimizing period is relatively long, an explicit optimizing control approach based on output feedback is proposed. A number of off-line optimizations are carried out under various operating conditions, and the explicit model between optimal input and output variables is numerically established via function fitting. The model is used for online implementation and thus online gradient estimation is avoided. A continuous stirred tank reactor(CSTR) case is studied to demonstrate the advantages of proposed approach.

Key words: real-time optimization; output feedback; necessary conditions of optimality tracking; optimizing control

0 引言

竞争日益激烈的市场环境下, 如何降低成本、提高生产效益已成为流程工业提高企业竞争力的首要任务, 其中一个重要手段就是对生产过程进行操作实时优化(RTO), 通过RTO不断调整过程的操纵输入以适应不同工况, 使生产过程更好地提高产品质量、降低原材料消耗等.

实时优化控制一般均需要系统模型, 然后通过数值求解确定最优操作点^[1]. 参数估计-重新优化二步法^[2]是目前最为广泛接受的RTO方法, 即在不稳定扰动影响下首先辨识出当前工况下的系统扰动参数, 然后重新优化计算出新的最优操作点. 国内外文献报道的在工程上已成功应用的实时优化系统也大都属于此类, 但是此方法不能从根本上满足实时性的需求^[3], 近年

来兴起的一种直接更新操纵输入的方法^[4-5], 如一阶最优性必要条件(NCO)跟踪法^[6-9]引起了学术界的关注. 不同于传统的参数估计-重新优化二步法, NCO跟踪法通过在线评估约束和目标函数的梯度值, 直接调整操纵输入使其满足新工况下的NCO条件. 由于绕过了数据调和、参数估计等步骤, 实时性能得到一定程度的提高.

现有NCO跟踪法的一个缺点是实时性有限, 原因是在使用有限差分法等对目标函数梯度进行在线估计时, 需要对输入进行反复摄动并等待系统重新达到稳态, 延长了RTO间隔. 针对这一缺点, 本文提出了一种基于输出反馈的显式实时优化方法. 将计算负荷移至离线进行, 得到最优输入与输出变量之间显式模型, 确定控制器更新律后用于在线实施. 新方法不需

收稿日期: 2012-07-25; 修回日期: 2012-12-11.

基金项目: 国家自然科学基金项目(61304081); 国家973计划项目(2012CB720505); 浙江省自然科学基金项目(LQ13F030007); 宁波市创新团队项目(2012B82002).

作者简介: 叶凌箭(1984—), 男, 讲师, 博士, 从事化工过程实时优化的研究; 马修水(1963—), 男, 教授, 博士生导师, 从事检测仪表与自动化等研究.

要对系统重新达到稳态后进行梯度评估,因此更新间隔大大缩短,提高了实时优化性能. 对一个连续搅拌釜式反应器(CSTR)反应过程的研究表明,所提出的方法具有更好的实时优化效果.

1 NCO 跟踪

考虑如下连续过程稳态优化问题:

$$\min_u J(u, d), \text{ s.t. } G(u, d) \leq 0. \quad (1)$$

其中: 标量 J 为目标函数, 通常是经济指标; $u \in R^{n_u}$ 和 $d \in R^{n_d}$ 分别为操纵变量和不可测扰动; $G \in R^{n_g}$ 为约束条件. 假设 u_{opt} 是最优输入, 即式(1)的解, 根据优化理论, u_{opt} 应满足如下 KKT 条件^[1]:

$$\begin{aligned} G &\leq 0, \mu^T G = 0, \\ \frac{\partial \Phi}{\partial u} &\triangleq \frac{\partial J}{\partial u} + \mu^T \frac{\partial G}{\partial u} = 0. \end{aligned} \quad (2)$$

其中: $\mu \in R^{n_g}$ 为拉格朗日乘子向量, 标量 Φ 为拉格朗日函数. 满足上述 KKT 条件的第 1 步是确定积极约束 G_a , 即非零拉格朗日乘子 μ_i 对应的约束, G_a 确定后可由式(2)约去 μ , 并得到

$$\begin{aligned} G_a &= 0, \\ \nabla_r J &\triangleq \frac{\partial J}{\partial u} \left[I - \left(\frac{\partial G_a}{\partial u} \right)^+ \left(\frac{\partial G_a}{\partial u} \right) \right] = 0, \end{aligned} \quad (3)$$

其中 $(\cdot)^+$ 为矩阵的 pseudo 伪逆矩阵. 式(3)包含积极约束 G_a 和简约梯度 $\nabla_r J$ 两部分, 称为连续过程优化问题的 NCO 条件. NCO 跟踪通过直接调整 u 的值, 使其经过若干次迭代运算后重新满足式(3)的 NCO 条件, 重新达到新工况下的最优点. 因为 G_a 一般可测, 较为常见的处理手段是直接使用反馈控制器对其进行卡边控制, 所以后文提到的 NCO 跟踪将只关注如何满足 NCO 条件中梯度部分. 实际上, 积极约束 G_a 也可以通过增加罚函数项等手段转化为梯度形式^[8], 这样只需满足如下不带约束条件的梯度即可:

$$J_u \triangleq \frac{\partial J}{\partial u} = 0. \quad (4)$$

使用如下控制律^[7-8], 经过若干次迭代后可以满足新工况下的 NCO 条件:

$$u_{k+1} = u_k - \alpha J_{uu}^{-1}(u_k) J_u(u_k). \quad (5)$$

其中: u_i 为第 i 次迭代的控制输入, 调节系数 $\alpha \in (0, 1]$ 用于调节收敛速度, J_{uu} 为海森 (Hessian) 矩阵. NCO 跟踪的控制结构如图 1 所示.

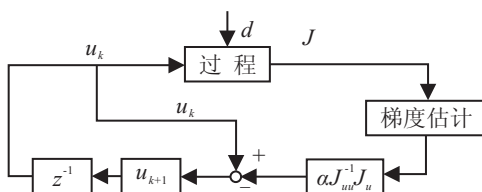


图 1 NCO 跟踪控制结构

执行式(5)的 NCO 跟踪控制律, 需要获取当前输

入位置 u_k 的目标函数二阶梯度 $J_{uu}(u_k)$ 和一阶梯度 $J_u(u_k)$. $J_{uu}(u_k)$ 的计算较为复杂, 可以使用名义工作下的数值代替, 或者由标准 BFGS 更新算法^[10]获得. $J_u(u_k)$ 一般实验性地获得, 现有文献报道的 NCO 跟踪法大多使用有限差分法, 即在 u_k 附近作微小扰动, 等待系统重新到达稳态后使用目标函数变化差值计算梯度. 这一方法有如下缺点:

1) 优化间隔较长. 优化间隔即每次更新式(5)算法的间隔, 由于 $J_u(u_k)$ 是稳态意义下的梯度值, 必须保证系统已重新达到稳态, 这将导致时滞较长的系统不能快速执行优化算法. 另外, 对于控制输入个数较多的系统, 需要对每个控制输入进行依次扰动, 这将进一步导致优化间隔加长.

2) 扰动幅度需要合理确定, 使得目标函数变化既不会因为太小淹没在噪声中, 又不会因为太大导致系统过分震荡. 另一个问题是即使系统已处于当前工况的最优点, 算法仍将对操纵输入进行不断扰动, 反而有可能使系统偏离最优位置.

上述分析的显著缺点将很大程度上影响现有 NCO 跟踪算法的实际使用效果.

2 基于输出反馈的显式实时优化方法

本文的目的是寻求一种直接更新操纵输入的实时优化算法. 不同于 NCO 跟踪, 这里的方法不是在线估计梯度值, 而是基于输出反馈离线得到最优输入和输出变量之间的映射关系, 然后实施在线优化算法.

2.1 显式实时优化

将过程输出变量 $y \in R^{n_y}$ 的模型记为

$$y = M(u, d), \quad (6)$$

其线性化模型为

$$y = \frac{\partial M}{\partial u} u + \frac{\partial M}{\partial d} d. \quad (7)$$

假设所有操纵变量可测, 即 u 为输出变量 y 的子集, 且满足 $n_y > n_d$, 则有

$$d = \left(\frac{\partial M}{\partial d} \right)^+ \left(y - \frac{\partial M}{\partial u} u \right) \triangleq K(y), \quad (8)$$

其中 $K(y)$ 定义为 d 与 y 之间的估计模型. 基于输出变量, 通过式(8)可以得到新工况下的扰动变量值. 从实用性的角度出发, 将其直接代入原优化问题, 绕开估计扰动变量的中间步骤. 重新求解式(1), 其最优输入 u_{opt} 为

$$u_{\text{opt}} = \arg \min_u J(u, d) \triangleq O(u, d), \quad (9)$$

其中 $O(u, d)$ 为 u_{opt} 与 u 和 d 之间的映射函数关系. 将式(8)代入(9)可得

$$u_{\text{opt}} = O(u, K(y)) \triangleq \tilde{O}(y), \quad (10)$$

其中 $\tilde{O}(y)$ 为最优输入 u_{opt} 和输出变量 y 之间的显式模型. 从式(10)可以看到, 如果能得到 $\tilde{O}(y)$ 的具体表

达式, 则可以直接基于输出反馈完成过程的优化控制.

为了获取 $\tilde{O}(y)$, 一种办法是先将 J 和 M 等过程模型线性化, 然后求解 $\tilde{O}(y)$ 的解析表达式. 该方法的缺点是其结果只在线性化点附近的较小领域内精确, 对于非线性较强的过程, 当操作点远离线性化点时, 线性化误差带来的影响将显著放大. 为了使方法同样适用于非线性过程, 本文使用数值方法将求解 $\tilde{O}(y)$ 转化为函数拟合问题. 首先离线产生一些 u_{opt} 和 y 的数据, 使用最小二乘或神经网络等函数拟合方法得到 $\tilde{O}(y)$ 的近似模型 $u_o = \hat{O}(y)$, 其优化效果取决于函数拟合的精确程度及其外推能力. 由于最优操纵输入 u_{opt} 可直接根据输出变量 y 得到, 将这种方法称为显式实时优化法.

引理 1 目标函数的损失函数 $L = J - J_{opt}$, 拟合误差 $\xi = \|u_o - u_{opt}\|_2$. 使用函数拟合的显式实时优化方法, 其损失函数 L 的上界 \bar{L} 可用拟合误差 ξ 表达, 具体关系为

$$\bar{L} = 0.5\bar{\lambda}(J_{uu})\xi^2. \quad (11)$$

其中: $\bar{\lambda}(\cdot)$ 为矩阵的最大特征值, J_{uu} 为 u_{opt} 处目标函数的海森矩阵.

证明 将 J 在 u_{opt} 点处泰勒展开到 2 阶, 有

$$J = J^* + J_u^*(u - u_{opt}) + 0.5(u - u_{opt})^T J_{uu}^*(u - u_{opt}), \quad (12)$$

其中 $(\cdot)^*$ 为在 u_{opt} 处的取值. 注意到, 在 u_{opt} 处一阶导数 J_u^* 应为 0, 以满足最优性必要条件, 因此有

$$J = J^* + 0.5(u - u_{opt})^T J_{uu}^*(u - u_{opt}). \quad (13)$$

将 $u = u_o$ 代入式 (13) 可得

$$J = J^* + 0.5(u_o - u_{opt})^T J_{uu}^*(u_o - u_{opt}). \quad (14)$$

因此显式实时优化方法的损失函数为

$$L = J - J^* = 0.5(u_o - u_{opt})^T J_{uu}^*(u_o - u_{opt}) \leq 0.5\bar{\lambda}(J_{uu}^*) \|u_o - u_{opt}\|_2^2 = 0.5\bar{\lambda}(J_{uu}^*)\xi^2. \quad (15)$$

由此引理 1 得证. \square

由引理 1 可知, 若要取得更好的优化效果, 则应尽量减小拟合误差 ξ , 即提高函数拟合精度. 另外注意到, 引理 1 只保证了在拟合数据点处有效, 为了在所有的操作范围内具有良好的优化效果, 应采用合适的函数拟合方法保证 $\hat{O}(y)$ 的外推能力, 避免过拟合现象.

得到 $\hat{O}(y)$ 后, 可以得到实时优化的控制律. 为了使过程避免过度震荡, 采用如下控制律:

$$u_{k+1} = (1 - \beta)u_k + \beta\hat{O}(y), \quad (16)$$

其中 $0 < \beta \leq 1$ 为松弛因子, 可根据实际情况调节. 式 (16) 经过若干个迭代步骤后, 当前的控制输入 u_k 将收敛到 u_o , 其控制结构如图 2 所示.

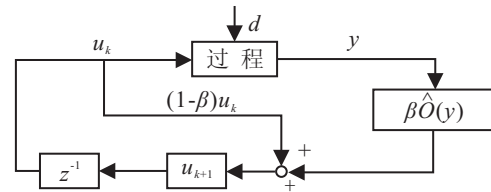


图 2 基于输出反馈的显式实时优化法控制结构

2.2 与 NCO 跟踪法的对比

与 NCO 跟踪法对比结果如下:

1) 由第 1 节分析可知, NCO 跟踪法的缺点是在线使用实验法获取梯度信息. 由于梯度信息需要过程达到稳态后才能正确取得, 式 (5) 的控制更新律必须大于过程本身的稳定时间, 导致 RTO 效果有相当程度的滞后. 同时, 如果扰动的产生频率高于此时间间隔, 则不能获取正确的梯度, 显式实时优化方法的控制律可以通过离线计算获得, 因此式 (16) 的更新间隔不依赖于过程的稳定时间, 效果更加迅速, 并且可以抑制高频扰动.

2) NCO 方法的另一个缺点是使用有限差分法获取梯度信息, 需要不停摄动操纵变量 u , 从控制角度而言这是不利的, 因为控制的目的是使过程稳定, 而不是使过程不断被激励.

3) NCO 方法只测量目标函数 J , 而本文方法是基于输出反馈的, 因此利用了过程更多的有用信息. 另一方面, 由式 (8) 可知, 本文要求输出变量的个数不少于独立扰动变量个数.

4) 本文使用函数拟合的方法获取显式控制律, 由于函数拟合本身存在误差, 本文方法的缺点是有一定程度的目标函数损失, 而 NCO 跟踪法在理论上最终可以收敛到最优点.

综上所述, 本文方法的最大特点是优化效果更加“实时”, 针对第 4) 个缺点, 具体应用时可以根据拟合效果预先估计损失范围, 如果能保证损失在可以接受的小范围内, 则使用本文方法大有裨益.

3 CSTR 实例研究

3.1 过程描述

反应原料通入 CSTR 进行一级可逆放热反应. 反应物中 A 和 B 的浓度分别为 C_{Ai} 和 C_{Bi} , 进料温度和出料温度分别为 T_i 和 T , 未反应的 A 和产物 B 的浓度分别为 C_A 和 C_B , 如图 3 所示.

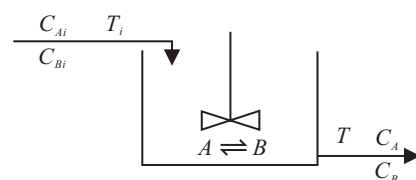


图 3 CSTR 放热反应过程

一级可逆放热反应式为



此 CSTR 过程可用如下物料和能量平衡微分方程描述:

$$\frac{dC_A}{dt} = \frac{1}{\tau}(C_{Ai} - C_A) - r, \quad (18)$$

$$\frac{dC_B}{dt} = \frac{1}{\tau}(C_{Bi} - C_B) + r, \quad (19)$$

$$\frac{dT}{dt} = \frac{1}{\tau}(T_i - T) + 5r. \quad (20)$$

其中: $\tau = 60$ s 为停留时间, 反应速率 r 表示为

$$r = 5000e^{-\frac{10000}{1987T}}C_A - 10^6e^{-\frac{15000}{1987T}}C_B. \quad (21)$$

操纵输入 u 、输入变量 y 和扰动变量 d 分别为

$$u = [T_i], \quad (22)$$

$$y = [y_1 \ y_2 \ y_3]^T = [C_A \ C_B \ T]^T, \quad (23)$$

$$d = [C_{Ai} \ C_{Bi}]^T. \quad (24)$$

扰动变量的变化区域 D 为 $\{0.5 \text{ mol/L} \leq C_{Ai} \leq 1.5 \text{ mol/L}, 0 \leq C_{Bi} \leq 0.5 \text{ mol/L}\}$. 目标函数 J 为系统的操作成本, 有

$$J = -20090C_B + (0.1657T_i)^2. \quad (25)$$

其中: 等式右边第 1 项为产物 B 收益的负数, 第 2 项为加热物料的能量成本. 名义工作点的过程变量数据列于表 1.

表 1 CSTR 过程名义工作点各变量数值

变量	物理含义	名义点数据	单位
C_A	出料 A 浓度	0.498	mol·L ⁻¹
C_B	出料 B 浓度	0.502	mol·L ⁻¹
T	物料出口温度	426.803	K
T_i	反应器温度	424.292	K
C_{Ai}	进料 A 浓度	1.0	mol·L ⁻¹
C_{Bi}	进料 B 浓度	0	mol·L ⁻¹
J	目标函数	-5 149.3	\$.h ⁻¹

3.2 显式实时优化控制律

使用本文方法产生用于函数拟合的样本数据, 将扰动变量 C_{Ai} , C_{Bi} (D 空间) 和操纵变量 T_i ([380 K, 450 K]) 在范围内各自 10 等分, 得到 $11^3 = 1331$ 个操作点并分别计算出最优操纵输入 u_{opt} . 使用最小二乘法拟合, 线性回归和二阶多项式的回归结果分别为

$$\hat{O}_1(y) = 342.26 + 63.00y_1 - 45.25y_2 + 0.15y_3, \quad (26)$$

$$\begin{aligned} \hat{O}_2(y) = & -762.6 - 394.2y_1 + 419.5y_2 + 5.4y_3 + \\ & 9.6y_1y_2 + 1.44y_1y_3 - 1.38y_2y_3 - 84.5y_1^2 + \\ & 49.5y_2^2 - 0.006y_3^2, \end{aligned} \quad (27)$$

其拟合 R^2 值分别为 0.609 和 0.926. 为了进一步提高拟合效果, 使用径向基函数 (RBF) 神经网络得到另一结果, 记为

$$\hat{O}_3(y) = Z(y). \quad (28)$$

使用蒙特卡罗实验方法对 $\hat{O}_1(y)$, $\hat{O}_2(y)$ 和 $\hat{O}_3(y)$ 进行效果验证, 产生 100 组随机扰动, 并通过非线性模型计算目标函数损失. 结果显示, $\hat{O}_1(y)$ 的平均损失为 31.815, $\hat{O}_2(y)$ 的平均损失为 1.463, $\hat{O}_3(y)$ 的平均损失为 0.845, 使用 RBF 神经网络后进一步提高了优化效果, 几乎完成了理想的最优控制. 因此, 使用式 (16) 的显式实时优化控制律为

$$u_{k+1} = (1 - \beta)u_k + \beta\hat{O}_i(y), \quad i \in \{1, 2, 3\}. \quad (29)$$

3.3 动态仿真

为了进一步验证本文方法的实际使用效果, 利用 Matlab Simulink 软件进行动态仿真. 仿真安排如下: 初始时刻系统各变量均处于名义点, 扰动 C_{Bi} 在 $t = 20 \text{ min}$ 时刻产生值为 0.5 的阶跃变化, C_{Ai} 在 $t = 40 \text{ min}$ 和 $t = 60 \text{ min}$ 分别产生值为 0.5 和 1.5 的阶跃变化, 如图 4 所示.

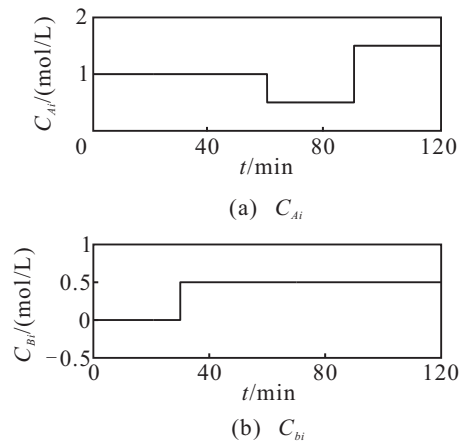


图 4 扰动变量变化曲线

在 NCO 跟踪 (SimA) 方面, 由于 NCO 跟踪算法需要在线获取梯度信息, 算法的 RTO 周期必须大于系统本身的时间常数. 通过观察发现, 此 CSTR 过程在扰动影响下从一个稳态到另一个稳态的经历时间约为 5 min, 因此 NCO 跟踪算法的 RTO 周期定为 5 min. 仿真运行期间, 摄动 T_i (摄动幅度为 0.5) 使用有限差分法计算梯度, 然后利用式 (5) 执行 NCO 跟踪 ($\alpha = 0.8$), 两个步骤交替进行, 实现过程的实时优化控制. 在显式实时优化方面, 由于不需要等待系统重新到达稳态即可执行实时优化, RTO 频率大大加快. 以 $\hat{O}_3(y)$ 为例, 分别设定优化间隔为 10 s, 1 min 和 5 min (记为 SimB10, SimB60 和 SimB300) 进行仿真验证 ($\beta = 0.8$).

NCO 跟踪和显式实时优化的仿真结果如图 5 所示, 其中虚线是各个时间段的实际最优输入. 从图中 SimA 曲线可以看到, 采用 NCO 跟踪经过若干次调整后, T_i 均能收敛到最优输入值, 但是有相当程度的滞后. 扰动发生时首先需要摄动 T_i , 等待目标函数变化并稳定后再更新 T_i . 可以预见, 如果是高频扰动, 则

NCO 方法将不能有效完成优化任务. 另一方面, 即使当前输入已处于最优点, 算法仍将不停扰动操纵输入, 以探测是否有扰动发生, 因此系统没有“控制”意义上的稳定点. 从 SimB10, SimB60 和 SimB300 可以看到, 由于无需等待系统达到稳定点即可及时更新 T_i , 实时性大大提高, 其中 SimB10 的效果最好. 同时注意到, 所有方法最终并没有收敛到最优点, 而是有一定程度的偏差, 这是因为使用回归法拟合并不能完全精确, 其误差幅度取决于在某个特定点上的拟合效果.

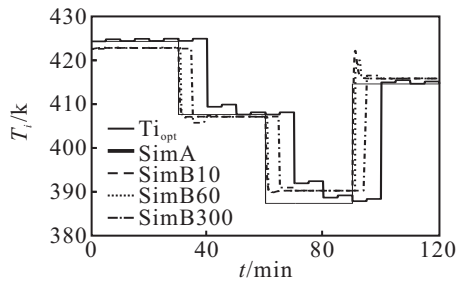


图 5 不同优化方法的操纵变量响应曲线

在 Simulink 中使用积分器对瞬时目标函数进行积分运算, 以 SimA 为基准计算相对差值, 结果如表 2 所示. 由表 2 可见, 历时 120 min 的操作时间, 使用显式实时优化方法取得的额外经济效益分别为 131 \$, 123 \$ 和 77 \$.

表 2 不同优化方法 120 min 内的操作总成本

No	优化方法	$\sum J/\$$	相对结果/\$
1	SimA	-7715	0
2	SimB10	-7846	131
3	SimB60	-7838	123
4	SimB300	-7792	77

4 结 论

NCO 跟踪方法是近年来兴起的一种新型优化控制策略, 与传统的 RTO 实时优化方法相比, 由于不需要估计不确定扰动, 直接调整操纵输入使其逼近最优点, 在一定程度上提高了优化实时性. 但是目标函数梯度仍需在线实验性获得, 因此对于稳定时间较长的过程, 优化效果不能令人满意. 鉴于此, 本文提出了一种基于输出反馈的显式实时优化方法, 同样直接调整操纵输入完成实时优化过程. 首先通过离线计算, 运用

函数拟合方法得到最优输入与输出变量之间的函数关系, 然后在线实行控制律. 新方法无需等待系统重新到达稳定, 因此优化间隔可大大缩短. 着重阐述了 NCO 跟踪与本文方法的联系和区别, 通过对一个 CSTR 放热反应过程进行研究并对比两种优化方法的效果, 验证了所提出方法的有效性.

参考文献(References)

- [1] Edgar T F, Himmelblau D M, Lasdon L S. Optimization of chemical processes[M]. 2nd ed. New York: McGraw-Hill, 2001: 267-280.
- [2] Chen C Y, Joseph B. On-line optimization using a two-phase approach: An application study[J]. Industrial and Engineering Chemistry Research, 1987, 26(9): 1924-1930.
- [3] Engell S. Feedback control for optimal process operation[J]. J of Process Control, 2007, 17(3): 203-219.
- [4] Chachuat B, Srinivasan B, Bonvin D. Adaptation strategies for real-time optimization[J]. Computers and Chemical Engineering, 2009, 33(10): 1557-1567.
- [5] Skogestad S. Plantwide control: The search for the self-optimizing control structure[J]. J of Process Control, 2000, 10(5): 487-507.
- [6] Srinivasan B, Bonvin D, Visser E, et al. Dynamic optimization of batch processes II: Role of measurements in handling uncertainty[J]. Computers and Chemical Engineering, 2003, 27(1): 27-44.
- [7] Francois G, Srinivasan B, Bonvin D. Use of measurements for enforcing the necessary conditions of optimality in the presence of constraints and uncertainty[J]. J of Process Control, 2005, 15(6): 701-712.
- [8] Srinivasan B, Biegler L T, Bonvin D. Tracking the necessary conditions of optimality with changing set of active constraints using a barrier-penalty function[J]. Computers and Chemical Engineering, 2008, 32(3): 572-579.
- [9] Gros S, Srinivasan B, Bonvin D. Optimizing control based on output feedback[J]. Computers and Chemical Engineering, 2009, 33(1): 191-198.
- [10] Nocedal J, Wright S J. Numerical optimization[M]. 2nd ed. New York: Springer, 2006: 136-142.