

基于神经网络的系统优化与参数估计 集成研究 (ISOPE) 方法

郑 磊 万百五

(西安交通大学系统工程研究所 西安 710049)

摘 要 提出了两种基于神经网络 (NN) 改进的系统优化与参数估计集成 (ISOPE) 稳态优化算法, 其中利用动态信息建立动态 NN 模型用于过程稳态优化. 目的是为了克服 ISOPE 算法对真实过程的摄动, 减少 ISOPE 算法设定点变动次数, 充分利用过程动态信息. 仿真结果验证了两种改进算法的优越性和有效性.

关键词 稳态优化, 神经网络 (NN), ISOPE

1 引言

系统优化与参数估计集成 ISOPE (Integrated System Optimization and Parameter Estimation) 方法是由 Roberts 领导的研究小组提出的一种非线性系统稳态优化方法^[1]. 它可以在模型与实际过程存在差异的情况下, 求得系统的最优解 (控制器设定值). 但是, ISOPE 方法需要对实际过程进行摄动以求得优化所需的导数. 而摄动对实际过程有不良影响, 在生产中是不允许的, 同时会显著增加算法的无效等待时间. 为了克服这个缺点, 本文将多层前向神经网络 (Multi-Layer Feedforward Neural Network) 引入 ISOPE 算法, 提出了两种改进算法. 经过仿真表明, 新方法明显减少了设定点变动次数, 加快了收敛速度.

2 多层前向神经网络 (MFNN) 及 BP 算法

多层前向神经网络是一种典型的前馈神经网络, 本文采用 3 层前向神经网络, 用 BP 算法作为 NN 网络的学习算法. 具体算法参考文献 [2].

已经证明, NN 网络能逼近任意函数^[2], 因此可以采用它作为实际过程的近似模型, 求取过程输出对输入的偏导数.

3 ISOPE 方法

ISOPE 方法是在两步法的基础上由 Roberts 等人提出的一种非线性系统稳态优化方法. ISOPE 方法通过引入新变量将系统优化和参数辨识分离, 在上层协调器中完成两部分信息的交换^[1].

通常情况下, 实际过程稳态优化问题 (ROP) 以如下形式

$$\begin{aligned} & \min Q(c, y) \\ & \text{s. t. } y = F^*(c) \\ & G(c, y) \geq 0 \end{aligned}$$

给出. 式中 $Q(c, y)$ 表示优化问题的目标函数; $G(c, y)$ 表示与输出相关的不等式约束; $y = F^*(c)$ 表示优化问题实际过程的等式约束或模型; c, y 分别是优化控制变量和过程输出.

一般情况下, 并不能确切给出实际过程方程, 只能得到过程估计模型 $y = F(c, \alpha)$. 可以证明, 在非常宽松的假设条件下, 上面的 ROP 问题等价于新的优化问题(1)(EOP)^[1]

$$\begin{aligned} & \min_{c, \alpha} q(c, \alpha) \\ \text{s. t. } & F(c, \alpha) = F^*(c) \\ & g(c, \alpha) \geq 0 \end{aligned} \quad (1)$$

式中 $q(c, \alpha) = Q(c, F(c, \alpha))$, $g(c, \alpha) = G(c, F(c, \alpha))$, α 是过程模型参数化变量.

为了将系统优化和参数估计两个子问题分离开, 在 EOP 问题(1)中引入一个附加的等式约束 $v = c$ 变成如下的优化问题(2)

$$\begin{aligned} & \min_{c, \alpha} q(c, \alpha) \\ \text{s. t. } & F(v, \alpha) = F^*(v) \\ & g(c, \alpha) \geq 0 \\ & v = c \end{aligned} \quad (2)$$

引入拉格朗日乘子 (λ, η, ζ) , 根据 Kuhn-Tucker 条件可以将优化问题(2)分解成如下的模型优化子问题、参数估计子问题和协调子问题.

模型优化子问题

给定参数 α 和拉格朗日乘子 λ 求解如下优化问题

$$\begin{aligned} & \min_c \{q(c, \alpha) - \lambda^T c\} \\ \text{s. t. } & g(c, \alpha) \geq 0 \end{aligned}$$

参数估计子问题

$$F(v, \alpha) = F^*(v) \rightarrow \alpha$$

协调子问题

$$\begin{aligned} \lambda = \lambda(c, v, \alpha, \zeta) &= \left[\frac{\partial^T F(v, \alpha)}{\partial v} \quad \frac{\partial^T F^*(v)}{\partial v} \right] \left[\frac{\partial^T F(v, \alpha)}{\partial \alpha} \right]^{-1} \left[\frac{\partial^T g(c, \alpha)}{\partial \alpha} \zeta + \frac{\partial^T q(c, \alpha)}{\partial \alpha} \right] \\ v^t &= \varphi_v(c^{t-1}, v^{t-1}) \\ \zeta^t &= \varphi_\zeta(\zeta^{t-1}, \zeta^{t-1}) \end{aligned}$$

4 基于神经网络改进的 ISOPE 方法

上面描述的 ISOPE 算法中, 有两个缺点: (1) 利用摄动法求得实际过程输出对输入(设定点)的偏导数阵 $(\partial F^*(v)/\partial v)$; (2) 算法中根本没有利用实际过程的动态信息. 本文将用实际过程的动态信息训练得到动态 NN 模型, 在此动态 NN 模型上代替摄动法求取所需的偏导数阵, 完成稳态优化, 改进 ISOPE 算法.

4.1 神经网络模型的获取

通常, 稳态 NN 模型是通过对大量的稳态输入、输出样本训练得到的, 取得稳态训练样本的方法一般有两种: (1) 采用历史稳态数据; (2) 对实际过程的设定点进行多次的摄动. 但这两种方法都有不足之处, 甚至不可行, 因为: (1) 由于工业过程往往都是慢时变过程, 历史数据不

能确切地反映过程当前的特性;(2)实际工业过程要求尽可能少地变动稳态设定点。

为此,考虑利用系统的动态信息建立系统的动态神经网络模型,由动态模型导出稳态模型⁽³⁾或者直接代替稳态模型⁽⁴⁾进行稳态优化。

动态 NN 模型的获得,可以利用某一次设定点变化时的动态过程采样信息 $(c(k), y(k), y(k-1), \dots, y(k-N))$, N 为过程阶次)构造一个动态训练样本集,采用合适的神经网络训练方法(本文采用 BP 算法)离线训练得到,训练动态 NN 模型框图如图 1。

按照系统辨识理论,在上述动态模型的求取方法中,要求辨识输入信号对动态系统而言是充分激励的。但本文探讨的是稳态优化算法,仅需要知道稳态模型。文献[5]已经证明,为求得一个稳态模型,阶跃输入信号是充分激励的。所以,本文算法利用阶跃型信号作为辨识输入。同时,充分考虑到训练动态 NN 模型的特殊性和实际过程的要求,我们利用阶梯形输入信号代替阶跃信号,这样既可以为动态 NN 模型的训练提供足够的设定点和过程信息,又可以避免对实际过程产生较大的冲击,符合实际应用的要求。

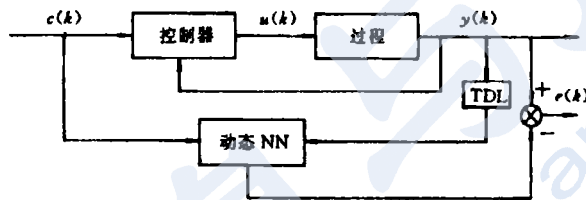


图 1 训练动态 NN 模型框图

得到动态 NN 模型后,可以利用此动态 NN 模型仿真得出稳态 NN 模型,然后以稳态 NN 模型进行优化;也可以直接用动态 NN 模型进行优化,只是当优化需要过程的稳态信息时,对动态 NN 模型进行运行仿真,仿真计算出所需的稳态信息。后一种方法有一个显著的优点:可以直接进行动态过程采样来修正动态 NN 模型;而利用稳态 NN 模型时则必须等待实际过程达到稳态时采样稳态信息,在漫长的动态过渡过程时间里,算法处于等待状态;而且用动态 NN 模型不必使算法等待过渡过程结束就可以开始新一次的修正、优化,充分利用动态过渡过程的信息和时间,大大提高算法的连续性和迭代效率。本文仿真中采用此方法。

4.2 改进的 ISOPE 算法 1(NN-ISOPE1)

由于改进的 ISOPE 算法是将 MLFNN 求取的偏导数引入原 ISOPE 算法中,所以将它称为 NN-ISOPE1 算法。NN-ISOPE1 算法的迭代结构框图如图 2 所示。

NN-ISOPE1 算法与 ISOPE 算法相比较,迭代结构基本相似,只是 NN-ISOPE1 算法中多一个模块:NN 模型的在线修正。NN 模型的在线修正既是神经网络本身的需要,又是实际工业过程稳态优化控制的需要。一方面,用神经网络辨识实际过程的模型存在误差;另一方面,由于生产原料及生产环境变化的影响,实际工业过程一般都是慢时变过程,需要不断的在线辨识更新模型,使之更符合实际过程。

4.3 改进的 ISOPE 算法 2(NN-ISOPE2)

NN-ISOPE1 算法中仅利用神经网络来求得所需的偏导数,没有完全发挥神经网络模型的作用;而且,算法 1 中,每一步迭代结果都要加入实际过程以获得下一步迭代所需的实际过

程动态信息,造成设定点的变动次数仍然较多。

为此,对 NN-ISOPE1 进一步改进提出新的算法 NN-ISOPE2. 算法 2 中,用 NN 模型替代算法 1 中实际过程的作用,使算法形成如图 3 所示的迭代结构。

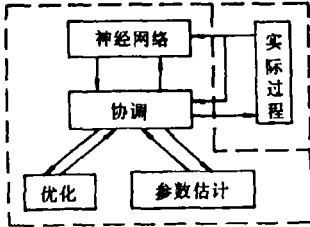


图 2 NN-ISOPE1 算法的迭代结构框图

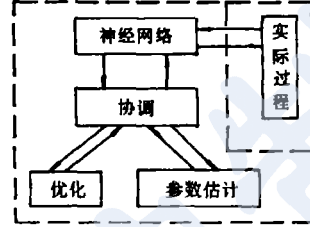


图 3 算法 2 形成的迭代结构

优化:将 NN 模型近似为实际过程进行完全离线的 ISOPE 优化运算,得到以 NN 为近似模型的最优解. 优化部分与实际过程之间没有信息交换。

修正:把优化的最终结果加到实际过程,采样实际过程的动态输入和输出,修正 NN 模型. 然后转入优化部分进行新的迭代,完成一次设定点的变动。

如此反复进行,直到最优。

很明显,算法 2 可以大大减少实际过程的设定点变动次数。

5 仿真结果

5.1 两个仿真例

本文采用两个例子对 ISOPE, NN-ISOPE1 和 NN-ISOPE2 算法进行仿真比较。

例 1 工业过程的稳态模型

$$y_1 = 1.4375c_1 - 0.1875c_2 + 1.5y_2 + \alpha_1$$

$$y_2 = 0.5c_3 - 1.5c_4 + y_1 - 2y_4 + \alpha_2$$

$$y_3 = 2.5c_4 - 0.5c_5 - y_1 + 1.5y_4 + \alpha_3$$

$$y_4 = 1.25c_6 - 3y_3 + \alpha_4$$

实际过程的稳态方程

$$y_1^* = c_1 - c_2 + 2y_2^* - 0.5c_1^2 + 0.5(c_1 + c_2 - 2)y_2^*$$

$$y_2^* = c_3 - c_4 + y_1^* - 3y_4^*$$

$$y_3^* = 2c_4 - c_5 - y_1^* + y_4^*$$

$$y_4^* = c_6 - 4y_3^* + 0.5c_6y_3^*$$

约束条件

$$R_{c,y} = \{(c, y) \in R^{10} \mid 1.006 - c_1 - y_2 \geq 0 \cap 0.375 + 2.25c_6 - 2.75y_3 - y_4 \geq 0 \cap (|c_i| \leq 0.5, i = 1, 4, 5, 6) \cap 0 \leq c_2 \leq 2.5 \cap 0 \leq c_3 \leq 2\}$$

目标函数

$$Q(c, y) = c_1^2 + (c_2 - 2)^2 + 2(c_3 - 2)^2 + c_4^2 + 3c_5^2 + (c_6 + 1)^2 + 4y_1^2 + (y_2 - 1)^2 + (y_3 - 1)^2 + y_4^2$$

例 2 工业过程的稳态模型

$$y_1 = 0.1c_1 + 0.4c_2 + 0.1y_2 + a_1$$

$$y_2 = 0.2c_1 + 0.3c_2 - 0.1y_1 + a_2$$

实际过程的动态方程

$$y_1(k) = 0.5y_1(k-1) + 0.12y_1(k-2) + 0.1y_2(k-1) + 0.28u_1(k-1) + 0.1y_1(k-1)y_2(k-1)$$

$$y_2(k) = 0.1y_1(k-1) + 0.46y_2(k-1) + 0.18y_2(k-2) + 0.16u_2(k-1) + 0.1u_1(k-1)u_2(k-1)$$

实际过程动态控制器方程

$$u_1(k) = 0.6u_1(k-1) + 0.22u_1(k-2) + 0.1u_2(k-1) + 1.2(c_1 - y_1(k))$$

$$u_2(k) = 0.2u_1(k-1) + 0.64u_2(k-1) + 0.13u_2(k-2) + 1.0(c_2 - y_2(k))$$

约束条件

$$R_{c,y} = \{(c, y) \in R^4 | 0 \leq c_1 \leq 1 \cap 0 \leq c_2 \leq 1\}$$

目标函数

$$Q(c, y) = 2(c_1 - 0.5)^2 + 1.6(c_2 - 0.3)^2 + 1.3(y_1 - 0.4)^2 + 2.3(y_2 - 0.2)^2 + 2.5(c_1 + c_2 - 1)^2$$

5.2 仿真结果

从表 1 可以看出, NN-ISOPE1 方法有较好的效果, 但收敛速度仍然比较慢, 设定点的变动次数比较多, 不能满足实际控制的要求; NN-ISOPE2 能明显减少设定点变动次数, 具有优越的性能. 原因很简单: 内部优化迭代的最终结果是以神经网络为近似模型的最优解(设定点), 此解已经接近系统的真实最优解, 算法将此解作为设定点加到实际系统, 当然可以显著减少设定点的变动次数.

表 1 仿真结果

	设定点变动次数			目标最优值			实际目标值
	ISOPE	新算法 1	新算法 2	ISOPE	新算法 1	新算法 2	
例 1	32	42	4	2.1405	2.1498	2.1436	2.1405
例 2	34*	28	4	0.1069*	0.1071	0.1070	0.1068

* 因为 ISOPE 方法本身不能直接对动态模型进行优化, 所以在对例 2 进行仿真时, 先从动态模型推导出稳态模型, 然后开始优化

6 结论

本文基于神经网络对稳态优化 ISOPE 算法进行了改进. 利用神经网络模型的优越性改善 ISOPE 算法的性能, 避免 ISOPE 算法为了求导对实际过程所作的摄动; 进一步改进后, 大大减少了 ISOPE 算法设定点的变动次数, 提高了算法的效率和实用性, 使算法更适用于实际应用.

在改进算法中, 用动态 NN 模型直接进行稳态优化, 这样充分利用了动态信息进行建模和在线修正, 从而避免了为得到稳态 NN 模型对实际系统造成的影响, 大大减少了系统建模的工作量, 同时提高了算法的适用性和效率.

仿真结果表明, 对非线性较强的系统, 改进算法具有同样优越的性能.

同时,改进算法具有较好的迭代结构,可以很容易地与关联平衡法或关联预测法相结合,扩展应用到大系统的稳态优化。

参 考 文 献

- 1 Brdys M, Chen S, Roberts P D. An Extension to the Modified Two Step Algorithm for Steady State System Optimisation and Parameter Estimation. *Int J System Sci*, 1986, 17, 1229~1243
- 2 Narendra K S, Parthasarathy K. Identification and Control of Dynamical Systems Using Neural Networks. *IEEE Trans on Neural Networks*, 1990, 1(1): 4~27
- 3 Wang Wuyi, Wan Baiwu. Neural Networks Identification of Steady State Process Model Using Dynamic Data. *Proceedings of Symposium on Control and Automation Applications CAI'93, Hong Kong*, 1993
- 4 李玉桥,万百五,梁天培. 基于神经网络的工业大系统辨识及稳态递阶优化方法研究. 中国自动化学会第六届过程控制科学报告会论文集,苏州,1993,524~529
- 5 陈庆新,万百五. 利用工业过程动态信息建立稳态模型及其强一致性分析, SISO 情形. *控制与决策*, 1991, 6, 90~96

ALGORITHMS OF INTEGRATED SYSTEM OPTIMIZATION AND PARAMETER ESTIMATION (ISOPE) BASED ON NEURAL NETWORK

ZHENG Lei WAN Baiwu

(*Institute of Systems Engineering of Xi'an Jiaotong University, Xi'an 710049*)

Abstract This paper proposes two modified algorithms of ISOPE based on neural network (NN). In order to avoid perturbation on the real process and decrease the number of setpoint changes, dynamic information is fully used to identify the dynamic NN model, which is applied in steady-state optimization of industrial processes. The results of simulation show that the two modified algorithms are effective.

Key words steady-state optimization, neural network, ISOPE

作者简介

郑磊,男,24岁,硕士.研究领域为智能控制和大工业过程稳态优化。

万百五,男,教授,博士生导师.研究领域为大系统稳态优化、分解-协调及大工业过程的递阶控制等。

