

# 多阶段多模型的改进微粒群优化算法

赵 嘉, 孙 辉

ZHAO Jia, SUN Hui

南昌工程学院 信息工程学院, 南昌 330099

School of Information Engineering, Nanchang Institute of Technology, Nanchang 330099, China

E-mail: zhaojia925@163.com

**ZHAO Jia, SUN Hui. Modified particle swarm optimization based on multi-stages & multi-models. Computer Engineering and Applications, 2010, 46(33): 32-35.**

**Abstract:** Modified particle swarm optimization based on multi-stages & multi-models is advanced to solve complex problems since premature convergence phenomena are common in current particle swarm optimization. This algorithm process includes three stages and various models considering development and exploration diversity in different steps. Firstly, the local extremum neighborhood is obtained through PSO. Secondly, local extremum is quickly searched out with Cognition Only to improve evolution efficiency. Lastly, the most optimal solution is sought through the improved evolution model to avoid local extremum. Results of four complex function tests show that the proposed algorithm is easier to get the optimal solution than PSO and TSE-PSO in efficiency and performance.

**Key words:** particle swarm optimization; multi-models; multi-stages; optimization

**摘 要:** 针对微粒群优化算法在解决复杂优化问题时易于出现早熟收敛现象, 提出了一种多阶段多模型的改进微粒群优化算法。考虑寻优不同阶段的开发与探测能力需求的差异, 算法将寻优过程分成3个阶段, 各阶段采用不同的模型进行进化。第一阶段利用标准微粒群优化算法发现局部极值的邻域; 第二阶段利用Cognition Only模型快速找到局部极值点, 提高寻优效率; 第三阶段, 提出了一种改进的进化模型, 利于粒子快速跳出局部极值点, 寻找到全局最优解。4种复杂测试函数的实验结果表明: 该算法比标准微粒群优化算法(PSO)和基于不同进化模型的两群优化算法(TSE-PSO)更容易找到全局最优解, 相比两群微粒群优化算法, 还能在一定程度上提高优化效率。

**关键词:** 微粒群优化算法; 多模型; 多阶段; 优化

DOI: 10.3778/j.issn.1002-8331.2010.33.009 文章编号: 1002-8331(2010)33-0032-04 文献标识码: A 中图分类号: TP301

## 1 前言

粒子群优化(Particle Swarm Optimization, PSO)算法最初是由Kennedy和Eberhart<sup>[1]</sup>于1995年提出的一种演化计算技术, 来源于对鸟群捕食行为的观察和研究。其后Shi和Eberhart<sup>[2]</sup>引入惯性权重来平衡全局性和收敛速度, 形成了当前的标准PSO。由于PSO算法概念简单、参数设置少、计算速度快、鲁棒性好等优点, 短短十多年时间, 便获得了很大发展, 并在一些领域取得很好的应用, 成为目前智能计算领域的研究热点之一<sup>[3]</sup>。

然而, Kennedy等人提出的粒子群优化算法存在易陷入局部极值点, 进化后期收敛速度慢, 计算精度较差等缺点。为了克服粒子群优化算法的不足, 研究人员提出了多种改进的粒子群优化算法: Shi和Eberhart<sup>[4]</sup>提出了基于模糊系统的惯性权

重的动态调整算法, Clerc<sup>[5]</sup>提出带收缩因子的粒子群算法, P.N.Suganthan<sup>[6]</sup>提出基于领域思想的粒子群算法等。近年来基于PSO的改进算法有F.Bergh<sup>[7]</sup>等人于2004年提出了协同粒子群优化算法(Cooperative Particle Swarm Optimizer, CPSO), J.J.Liang<sup>[8]</sup>等人于2006年提出了综合学习粒子群优化算法(Comprehensive Learning Particle Swarm Optimizer, CLPSO), 吴烈阳<sup>[9]</sup>等于2009年提出的并行文化粒子群优化算法(parallel Cultural Particle Swarm Optimization)。这些算法均是依据标准PSO模型开展的研究, 没有考虑到不同进化模型在进化过程中有不同的探测与开发能力的特点。因此提出了一种多阶段多模型的改进微粒群优化算法。该算法将进化过程分成3个阶段: 第一阶段采用标准PSO模型进化, 当粒子进化到一定的稳定状态后; 进入第二阶段, 此阶段使用开发模型跟踪

**基金项目:** 国家自然科学基金(the National Natural Science Foundation of China under Grant No.50539020); 江西省自然科学基金(the Natural Science Foundation of Jiangxi Province of China under Grant No.2007GZS1056, No.2009GZW0020); 江西省教育厅科技项目(No. GJJ09365, No.GJJ10630); 南昌工程学院青年基金-科技项目(No.2010KJ015, No.2010KJ018)。

**作者简介:** 赵嘉(1981-), 男, 讲师, 研究方向: 智能优化算法; 孙辉(1959-), 男, 博士, 教授, 硕士生导师, 研究方向: 智能优化算法, 变分不等式等。

**收稿日期:** 2010-03-10 **修回日期:** 2010-06-29

粒子的局部吸引空间, 细化开发新的最优解; 在第三阶段使用探测模型扩大探测空间, 进一步提高种群的全局寻优能力。实验结果表明: 算法的全局收敛性能得到了一定的提高, 能有效避免微粒群优化算法中的早熟收敛问题。

## 2 基本微粒群算法

在科学研究中, 许多问题最终都归结为带有约束的函数优化问题, 而带有的约束通过微粒群优化算法的位置和速度限制可以转换成不带约束条件的优化算法。即主要讨论如下的优化问题:

$$\min f(x), x \in S \subseteq R^n \quad (1)$$

PSO算法是基于群体与适应度的优化算法。系统初始化微粒(代表潜在的解)为一组随机解, 它具有位置、速度两个特征, 粒子的适应值由粒子的位置决定。微粒在解空间中运动, 由速度矢量决定运动方向和距离<sup>[10]</sup>, 由个体最优位置  $pbest$ 、全局最优位置  $gbest$  与粒子当前速度更新得到新的速度和位置, 粒子依据适应值的大小在解空间追寻最优的微粒进行搜索, 逐步收敛于最优解。

假设在一个  $d$  维的目标搜索空间中, 有  $n$  个微粒组成一个群体, 其中第  $t$  代第  $i(i=1, 2, \dots, n)$  个微粒的位置坐标  $x_i^t = (x_{i1}, x_{i2}, \dots, x_{id})$ , 速度  $v_i^t = (v_{i1}, v_{i2}, \dots, v_{id})$ , 个体最优位置  $p_i^t = (p_{i1}, p_{i2}, \dots, p_{id})$ , 全局最优位置  $p_g^t = (p_{g1}, p_{g2}, \dots, p_{gd})$ 。对第  $i$  个第  $d$  维第  $t$  代微粒其迭代公式可表示为:

$$v_{id}^{t+1} = \omega v_{id}^t + c_1 r_1 (p_{id}^t - x_{id}^t) + c_2 r_2 (p_{gd}^t - x_{id}^t) \quad (2)$$

$$x_{id}^{t+1} = x_{id}^t + v_{id}^{t+1} \quad (3)$$

其中,  $v_{id}^t$  为当前的速度;  $v_{id}^{t+1}$  为粒子  $i$  在第  $t$  次迭代后的新速度;  $\omega$  为惯性权重;  $c_1, c_2$  为加速(学习)因子;  $r_1, r_2$  为 0 到 1 之间均匀分布的随机数;  $x_{id}^t$  为粒子  $i$  当前的位置;  $x_{id}^{t+1}$  为粒子  $i$  第  $t$  次迭代后的新位置。

加速因子  $c_1, c_2$  是调整微粒自身经验与社会(群体)经验在其运动中所起作用的权重。如果  $c_2 = 0$ , 则微粒没有群体共享信息, 只有“自身经验”, 因为个体间没有交互, 每个微粒仅受自己的飞行经验影响, 不受其他微粒的影响, 它的收敛速度较快, 容易陷入局部最优点。Kennedy 称之为 Cognition Only 模型<sup>[11]</sup>, 其迭代公式为:

$$v_{id}^{t+1} = \omega v_{id}^t + c_1 r_1 (p_{id}^t - x_{id}^t) \quad (4)$$

$$x_{id}^{t+1} = x_{id}^t + v_{id}^{t+1} \quad (5)$$

其中, 式(4)、式(5)中的参数说明同式(2)、式(3)。

## 3 多阶段多模型的改进微粒群优化算法

### 3.1 算法原理

探测与开发一直是微粒群算法搜索过程中的一对矛盾, 探测强调在全局范围内寻找新的搜索区域, 而开发则侧重于在所得区域内实现精细搜索。虽然标准 PSO 算法简单且优化性能良好, 但在整个迭代过程中, 存在迭代前期探测能力强, 开发能力弱, 此时如果微粒落在最优微粒所在邻域, 可能由于探测能力太强, 而逃离该最优微粒所在邻域; 后期探测能力弱, 开发能力强, 此时如果微粒遇到局部极值, 所有微粒的速度可能很快降为零而不再飞行, 容易导致微粒群收敛于局部极值点<sup>[12]</sup>。

基于上述分析, 在粒子迭代过程中, 不同阶段采用具有不同探测与开发能力的进化模型进行进化:

第一阶段, 采用标准 PSO 优化算法, 如若前后两次迭代的最优适应值之差小于预先设定的阈值(此时表明粒子群已经找到一个极值点的搜索邻域), 进入下一阶段;

第二阶段, 使用具有极强开发能力的 Cognition Only 模型找寻该区域的局部极小值;

第三阶段, 利用提出的模型进化, 使之能跳出局部极值点, 扩大搜索区域。

重复上述第二与第三两阶段直到找到最优解。

在整个搜索过程中, 搜索邻域的极值点被一一找出, 且粒子的多样性得到了保证。

### 3.2 算法方程

在基于多模型的改进微粒群优化算法中, 首先, 采用式(2)、式(3)的标准 PSO 模型进行进化, 在后两阶段中, 前一阶段采用式(4)、式(5)的 Cognition Only 模型; 后一阶段为扩大搜索区域, 提出了一种改进的微粒群进化模型。

在该模型的设计中, 基于两点考虑:

(1) 扩大搜索区域并使模型的数据来源简单(只用已知的数据和变量, 不新定义其他数据和变量)。由于经过第二阶段进化后, 大部分粒子的个体最优位置  $p_i$  与全局最优位置  $p_g$  之间已经很靠近, 为扩大搜索区域, 在分析现有数据后, 发现群体中个别粒子个体最优位置与全局最优位置和其他个体最优位置之间的距离相差较远, 故在模型中提出速度不仅与粒子本身的飞行经验和全局历史最优微粒有关, 还与全局各个体最优位置中的最差微粒有关。

(2) 模型的结构形式标准。为保证模型和标准 PSO 算法的结构形式一致并考虑微粒群算法的统计规律性, 定义一个减速因子(与加速因子  $c_1, c_2$  对应)和一个 0 到 1 之间均匀分布的随机数(与  $r_1, r_2$  对应)。

模型迭代公式为:

$$v_{id}^{t+1} = \omega v_{id}^t + c_1 r_1 (p_{id}^t - x_{id}^t) + c_2 r_2 (p_{gd}^t - x_{id}^t) + c_3 r_3 (p_{bd}^t - x_{id}^t) \quad (6)$$

$$x_{id}^{t+1} = x_{id}^t + v_{id}^{t+1} \quad (7)$$

$$p_{bd}^t = \max(p_1^t, p_2^t, \dots, p_n^t) \quad (8)$$

其中,  $\omega, c_1, c_2, r_1, r_2$  与式(2)、式(3)定义相同,  $c_3$  为减速因子,  $r_3$  为 0~1 之间均匀分布的随机数。


### 3.3 算法流程

(1) 初始化设置粒子群粒子个数、最大迭代次数、惯性权重、加速因子、减速因子、标准 PSO 迭代阈值  $\sigma$ 、模型迭代次数  $\epsilon$ 、各粒子初始位置和初始速度等。

(2) 评价各微粒的初始适应值、保存初始最好位置及初始最优适应值。

(3) 根据式(2)计算各微粒新的速度, 根据式(3)计算各微粒新的位置、并对各微粒新的速度和位置进行限幅处理。

(4) 更新各微粒的个体历史最好适应值和个体历史最好位置; 更新全局历史最好适应值和全局历史最好位置。

(5) 比较前后  全局历史最好适应值的差, 如果该差值大于事先设定标准 PSO 迭代阈值  $\sigma$ , 返回步骤(3), 否则进入步骤(6)。

(6) 采用 Cognition Only 模型对粒子的速度和位置进行

进化,并更新各微粒的个体历史最好适应值和个体历史最好位置;更新全局历史最好适应值和全局历史最好位置,按照设定的参数重复步骤(6)  $\varepsilon$  次后,转入步骤(7)。

(7)采用式(6)与式(7)对粒子的速度和位置进行进化,并更新各微粒的个体历史最好适应值和个体历史最好位置;更新全局历史最好适应值和全局历史最好位置,按照设定的参数重复步骤(7)  $\varepsilon$  次后,转入步骤(8)。

(8)若满足停止条件(迭代次数超过最大允许迭代次数),搜索停止,输出全局历史最好位置和全局历史最好适应值;否则,返回步骤(6)继续搜索。

## 4 仿真实验

### 4.1 测试函数

为了测试改进算法的效果,选择了4个经典的复杂非线性多峰函数进行测试。

Griewank 函数:

$$\min f(x_i)=\sum_{i=1}^N\frac{x_i^2}{4\,000}-\prod_{i=1}^N\cos(\frac{x_i}{\sqrt{i}})+1, x_i\in[-600, 600]$$

Rastrigrin 函数:

$$\min f(x_i)=\sum_{i=1}^N[x_i^2-10\cos(2\pi x_i)+10], x_i\in[-5.12, 5.12]$$

Ackley 函数:

$$\min f(x_i)=20+e-20\exp(-0.2\sqrt{\frac{1}{N}\sum_{i=1}^Nx_i^2})-\exp(-\frac{1}{N}\sum_{i=1}^N\cos(2\pi x_i))$$
$$x_i\in[-15, 30]$$

Rosenbrock 函数:

$$\min f(x)=\sum_{i=1}^{N-1}[100(x_i^2-x_{i+1})^2+(x_i-1)^2], x_i\in[-5, 10]$$

以上测试函数都是多峰值复杂函数,其参数设置及函数的全局最优值  $x^*$  与函数的全局最优值  $f(x^*)$  信息如表1。

表1 高维函数参数设置及函数的全局最优信息

函数	Initial rang	Dimension	$x^*$	$f(x^*)$
Griewank 函数	[-600, 600]	10	[420.96, 420.96, ..., 420.96]	0
Rastrigrin 函数	[-5.12, 5.12]	10	[0, 0, ..., 0]	0
Ackley 函数	[-15, 30]	10	[0, 0, ..., 0]	0
Rosenbrock 函数	[-5, 10]	10	[1, 1, ..., 1]	0

### 4.2 算法参数讨论

模型迭代次数  $\varepsilon$  的取值直接影响到算法的优劣。 $\varepsilon$  太小,当使用 Cognition Only 搜索时,尚未找到该邻域的最优值,就已经结束 Cognition Only 模型的搜索,跳出该邻域,当使用式(6)进化时,粒子的多样性尚未得到足够的保证; $\varepsilon$  太大,由于 Cognition Only 模型迭代次数太多,  $p_{bd}^i$  已经越来越靠近  $p_{gd}^i$ ,式(6)已经很难跳出该搜索区域。为了讨论迭代次数  $\varepsilon$  对算法优化性能的影响,作了不同迭代次数  $\varepsilon$  与达优率的关系实验。实验时,惯性权值从0.9衰变到0.4,加速因子  $c_1$ 、 $c_2$  为2,减速因子  $c_3$  为2,最大迭代次数为2 000,群体微粒个数为60。由于微粒群优化算法是随机搜索算法,每次搜索的结果有可能不同,所以对每一种情况进行500次独立实验,然后进行统计汇总。**达优率是500次实验中找到最优解的次数在总次数中所占的比率。**

从图1可以看出,4种测试函数的达优率开始时随着迭代

次数  $\varepsilon$  的增加而增大,当迭代次数约为20~30时,达优率较高,之后随着迭代次数的增大达优率下降。迭代次数过少,粒子很难搜索到搜索邻域的极小值;迭代次数过多,粒子的多样性得不到保证。为了较好地利用基于多模型的改进微粒群优化算法的优势,迭代次数约为20~30次较好。

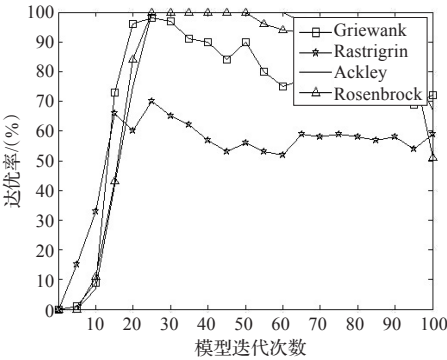


图1 不同函数的达优率与模型迭代次数  $\varepsilon$  的关系图

### 4.3 实验结果及分析

为了更好地测试算法的性能,这里选择标准的PSO算法、文献[10]提出的基于不同进化模型的双群交换微粒群优化算法(TSE-PSO)与本文算法对上述4个函数进行测试实验。寻优时,相关参数设置相同。惯性权值从0.9衰变到0.4,加速因子  $c_1$ 、 $c_2$  为2,减速因子  $c_3$  为2,最大迭代次数为2 000,本文算法、标准PSO算法群体微粒个数为60,本文算法的模型迭代次数  $\varepsilon$  依据上述讨论结果选择25。为了比较的公平性,TSE-PSO的每分群微粒个数为30。由于微粒群优化算法是随机搜索算法,从概率的角度来对比算法的优化性能,优化时用每一算法对每一函数进行500次独立测试,然后进行统计汇总,在求运行时间时,采用同一硬件配置的计算机系统进行计时。4种优化算法实验得到的具体结果如表2所示。

表2 4种优化算法实验结果表

函数	算法	全局平均最优值	运行时间/s	达优率/(%)
Griewank 函数	PSO	2.647 868E-01	2.27	1
	TSE-PSO	6.817 678E-02	2.88	17
	本文算法	2.271 5E-04	2.27	54
Rastrigrin 函数	PSO	5.154 4	2.30	0
	TSE-PSO	2.633 647E-01	2.83	5
	本文算法	2.614 4E-03	2.31	41
Ackley 函数	PSO	9.666 4E-05	2.35	78
	TSE-PSO	0	2.95	100
	本文算法	0	2.37	100
Rosenbrock 函数	PSO	3.412 2	2.36	0
	TSE-PSO	1.428 3E-01	3.10	8
	本文算法	6.317 4E-03	2.37	43

由表2可知,在算法参数相同的情况下,本文算法的时间复杂度与标准PSO算法的时间复杂度在同一个数量级,其运行时间相差无几,而其全局平均最优值和达优率却有明显的提高,特别是对Rastrigrin和Rosenbrock测试函数,标准PSO算法达优率为0,本文算法的达优率却有40%之多。本文算法和TSE-PSO算法进行比较,本文算法不仅在平均运行时间上优于TSE-PSO算法,且全局平均最优值和达优率也都有明显的改善。

图2显示了Griewank函数在3种算法下的进化过程,从图2



可看出,本文算法比标准微粒群和TSE-PSO算法能更好地跳出局部极值点,保证粒子的多样性。

综上可知,提出的算法具有更佳的整体寻优性能。

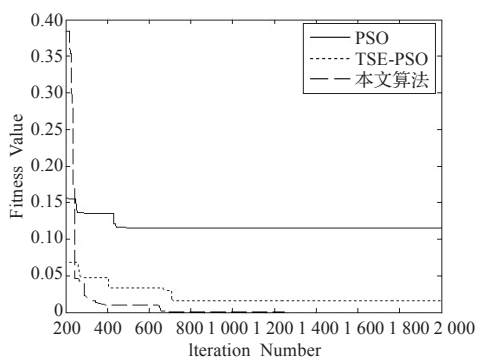


图2 Griewank函数下各算法的进化过程图

## 5 结束语

通过对标准粒子群优化算法在对算杂问题优化时很容易陷入局部极值点的分析,提出了多阶段多模型的改进微粒群优化算法。该算法利用各模型具有不同的探测与开发能力的特点进行粒子群的进化。这种处理,兼顾了探测与开发的平衡,有效地防止了群体多样性的降低。仿真实验表明,本文算法比标准微粒群算法有更好的全局寻优能力和达优率,比TSE-PSO算法有更好的全局平均最优值、运行效率和达优率。该方法概念明确,程序实现简单,全局寻优能力强,是一种有效的改进算法。虽然算法有更好的全局的寻优性能、达优率和运行效率,如何进一步提高算法的全局寻优性能是今后研究的重点。

## 参考文献:

[1] Kennedy J, Eberhart R C. Particle swarm optimization[C]//Proceedings of IEEE International Conference on Neural Networks. Piscataway, NJ: IEEE Service Center, 1995: 1942-1948.

[2] Shi Y, Eberhart R C. A modified particle swarm optimizer[C]//Proceedings of the IEEE International Conference on Evolutionary Computation. Piscataway, NJ: IEEE Press, 1998: 69-73.

[3] 沈林成, 霍霄华, 牛铁峰. 离散粒子群优化算法研究现状综述[J]. 系统工程与电子技术, 2008, 30(10): 1986-1994.

[4] Shi Y, Eberhart R C. Fuzzy adaptive particle swarm optimization[C]//Proceedings of the IEEE Congress on Evolutionary Computation, Seoul, Korea, 2001.

[5] Clerc M. The swarm and the queen: Toward a deterministic and adaptive particle swarm optimization[C]//Proceedings of the Congress on Evolutionary Computation, 1999: 1951-1957.

[6] Suganthan P N. Particle swarm optimizer with neighborhood topology on particle swarm performance[C]//Proceedings of the 1999 Congress on Evolutionary Computation. Piscataway, NJ: IEEE Press, 1999: 1958-1962.

[7] Bergh F, Engelbrecht A P. A cooperative approach to particle swarm optimization[J]. IEEE Transactions on Evolutionary Computation, 2004, 8(3): 225-239.

[8] Liang J J, Qin A K, Suganthan P N, et al. Comprehensive learning particle swarm optimizer for global optimization of multimodal functions[J]. IEEE Transactions on Evolutionary Computation, 2006, 10(3): 281-295.

[9] 吴烈阳, 孙辉, 白明明, 等. 一种新的并行文化微粒群优化算法[J]. 计算机工程与应用, 2009, 45(35): 44-46.

[10] 吴烈阳, 孙辉, 白明明. 基于不同进化模型的双群交换微粒群优化算法[J]. 南昌工程学院学报, 2008(4): 1-4.

[11] Kennedy J. Small worlds and mega-minds: Effects of neighborhood topology on particle swarm performance[C]//Proceedings of the IEEE Congress on Evolutionary Computation, 1999, 7: 1931-1938.

[12] 陈国初, 俞金寿, 郭伟. 两群替代微粒群优化算法及其应用[J]. 华东理工大学学报: 自然科学版, 2005, 31(6): 787-791.

(上接24页)

的问题,提出了动态选择雅可比矩阵奇异值比率确定参数搜索方向的方法,并给出了下融合参数局部逼近性能信息的辨识算法。分析了辨识算法的收敛性,给出了算法收敛速度的解析表达式。最后进行了数值仿真实验,结果说明了所提出辨识方法的有效性。

## 参考文献:

[1] 邱晓华, 陈偕雄. 随机自治状态空间模型的正交梯度辨识[J]. 浙江大学学报: 理学版, 2009, 36(2): 170-174.

[2] Van Overschee P, De Moor B. Subspace algorithms for the stochastic identification problem[J]. Automatica, 1993, 29(3): 649-660.

[3] Goethals I, Van Gestel T, Suykens J, et al. Identification of positive real model models in subspace identification by using regularization[J]. IEEE Transactions on Automatic Control, 2003, 48(10): 1843-1847.

[4] Dahlén A, Lindquist A, Mari J. Experiment evidence showing that

stochastic subspace identification methods may fail[J]. Systems & Control Letters, 1998, 34(5): 303-312.

[5] 衷路生, 宋执环. 基于正交梯度搜索的动态系统递阶优化辨识[J]. 自动化学报, 2008, 34(6): 711-715.

[6] Ober R J. Balanced parametrization of classes of linear systems[J]. SIAM Journal on Control and Optimization, 1991, 29(6): 1251-1287.

[7] McKelvey T, Helmersson A, Ribiari T. Data driven local coordinates for multivariable linear systems and their application to system identification[J]. Automatica, 2004, 40(9): 1629-1635.

[8] Ribiari T, Deistler M, Hanzon B. An analysis of separable least squares data driven local coordinates for maximum likelihood estimation of linear systems[J]. Automatica, 2005, 41(3): 531-544.

[9] Dennis J E, Schnabel R B. Numerical methods for unconstrained optimization and nonlinear equations[M]. Englewood Cliffs, NJ: Prentice-Hall, 1983.

[10] Wills A, Ninness B. On gradient-based search for multivariable system estimates[J]. IEEE Transactions on Automatic Control, 2008, 53(1): 298-306.