

多量子粒子群协同优化算法研究

屈百达,焦竹青,徐保国

QU Bai-da, JIAO Zhu-qing, XU Bao-guo

江南大学 通信与控制工程学院,江苏 无锡 214122

School of Communication & Control Engineering, Jiangnan University, Wuxi, Jiangsu 214122, China

QU Bai-da, JIAO Zhu-qing, XU Bao-guo. Research on Quantum-behaved Particle Swarms Cooperative Optimization. Computer Engineering and Applications, 2008, 44(7): 72-74.

Abstract: Aiming at the problem that the simple Particle Swarm Optimization(PSO) algorithm and its improved algorithms are difficult to deal with local convergence and convergence velocity, this paper proposes a Quantum-behaved Particle Swarms Cooperative Optimization(QPSCO) algorithm. In its two-layer framework with particle swarms cooperative optimization, some quantum-behaved particle swarms are employed to search in the solution space independently and a mutation parameter is introduced for larger searching scale in bottom layer, and a single quantum-behaved swarm is employed to track with the globe best solution while new particles replace those particles flying out the solution space with their position in top layer. Based on the above, this algorithm is applied to PID controller design for a lower-order system with time delay. Simulation results demonstrate that the approach is more effective for parameter optimization and have better global convergence ability than PSO, QPSO et al.

Key words: particle swarm; cooperative optimization; quantum-behaved particle swarms

摘要:针对标准粒子群优化(PSO)算法及其改进算法存在的局部收敛与收敛速度问题,提出了一种多量子粒子群协同优化(QPSCO)方法。该算法采用双层的多粒子群协同优化结构:用多个量子粒子群在底层独立地搜索解空间,同时引入参数变异策略,以扩大搜索范围;上层用1个量子粒子群追逐当前全局最优解,并对飞离搜索区域粒子的位置用新位置取代,以加快算法收敛。在此基础上,将该算法应用于实际控制系统低阶时滞对象的PID控制器设计中。仿真结果表明,QPSCO是一种有效的参数优化算法,与标准PSO、QPSO等算法相比具有更好的全局收敛性能。

关键词:粒子群;协同优化;多量子粒子群

文章编号:1002-8331(2008)07-0072-03 **文献标识码:**A **中图分类号:**TP301

1 引言

粒子群优化(PSO)算法作为一种简单有效的随机搜索算法,具有较强的全局收敛能力和较强的鲁棒性,并且不需要借助问题的特征信息如导数等梯度信息,可以有效求解大量非线性、不可微和多峰值的复杂优化问题。其算法简洁,调整参数少,因而发展很快,自提出以来,得到了国内外相关领域众多学者的关注与研究,被应用于多个学科和工程技术领域^[1-4]。

标准PSO算法具有搜索空间有限,容易陷入局部最优点的缺陷,目前已有多种改进PSO算法^[5,6]。但是这些算法大多着眼于PSO的参数选择或某个参数的动态修改策略,故难以克服PSO算法易陷入局部极小的固有弱点。李爱国、J Sun等学者已尝试从改进算法模型和状态更新策略着手,解决PSO的早熟和局部收敛问题^[7,8]。

本文提出的多量子粒子群协同优化(QPSCO)方法由多个量子粒子群独立地搜索解空间,同时引入变异参数;用1个量子粒子群追逐当前全局最优解,并重置飞离搜索区域的粒子的

位置。在扩大搜索范围同时加快算法收敛,最终解决PSO算法的局部收敛与收敛速度问题。

2 PSO算法

2.1 标准PSO算法

PSO算法采用“群体”和“进化”的概念,设在 N 维搜索空间中,由 m 个粒子组成一个群体。将每个粒子视作是一个没有重量和体积的微粒,并在搜索空间中以一定的速度飞行。其飞行速度由该粒子本身的飞行经验和同伴的飞行经验进行动态调整。

群体中第 i 个粒子的当前位置为 $x_i=(x_{i1}, x_{i2}, \dots, x_{iN})$, $i=1, 2, \dots, m$;粒子 i 的当前飞行速度 $v_i=(v_{i1}, v_{i2}, \dots, v_{iN})$;搜索到的最优位置,也就是粒子 i 所经历过的具有最好适应度值的位置, $p_i=(p_{i1}, p_{i2}, \dots, p_{iN})$;整个粒子群搜索到的最优位置,也就是群体中所有粒子所经过的最好的位置为 $p_g=(p_{g1}, p_{g2}, \dots, p_{gN})$ 。将 x_i 带入目标函数计算其适应度值。

基金项目:国家科技部“十五”攻关项目(No.2001BA204B01-03)。

作者简介:屈百达(1956-),男,教授,博士,主要研究领域为现代控制理论与技术;焦竹青(1983-),男,硕士研究生,主要研究领域为计算机控制。

收稿日期:2007-06-20

修回日期:2007-08-16

PSO 中粒子的状态更新策略为:

$$v_{in}(k+1)=wv_{in}(k)+c_1r_1(p_{in}-x_{in}(k))+c_2r_2(p_{gn}-x_{in}(k)) \quad (1)$$

$$x_{in}(k+1)=x_{in}(k)+v_{in}(k) \quad (2)$$

式中: $i=1, 2, \dots, m; n=1, 2, \dots, N$ 表示解空间的维数; $v_{in} \in [-v_{\max}, v_{\max}]$, v_{\max} 表示速度搜索范围; c_1, c_2 为加速常数; w 为惯性权重; r_1, r_2 为 [0, 1] 之间的随机数; k 为当前迭代次数。

目前, 有关 PSO 的研究大多以这种带惯性权重的 PSO 算法为基础进行扩展和修改, 为此将该算法称为标准 PSO 算法。但是在标准 PSO 的粒子群中, 粒子的收敛是以轨道的形式实现的, 且粒子速度总是有限的, 因此在搜索过程中粒子的搜索空间是一个有限的区域, 不能覆盖整个可行空间, 即标准 PSO 算法不能保证以概率 1 收敛到全局最优解。

2.2 QPSO 算法

由于 Van Den Bergh F 已经证明了 PSO 算法不能收敛于全局最优解甚至局部最优解^[9], 许多学者采用多种方法以改进算法的收敛性能。2004 年 J Sun 在研究了 Clerc 等人关于粒子收敛行为的研究成果后, 从量子力学的角度提出了一种新的 PSO 算法模型, 这种模型以 DELTA 势阱为基础, 认为粒子具有量子行为, 因此称之为 QPSO 算法, 并通过实验结果证明了 QPSO 收敛性能有了很大的改进。该算法的描述如下:

在有 m 个粒子进行寻优的 N 维搜索空间中, 第 i 个粒子的位置为 $x_i=(x_{i1}, x_{i2}, \dots, x_{iN})$; 第 i 个粒子自身的最优位置为 $p_i=(p_{i1}, p_{i2}, \dots, p_{iN})$; 整个粒子群所搜索到的最优解 $p_g=(p_{g1}, p_{g2}, \dots, p_{gN})$ 。

粒子的更新操作如下:

$$L = \frac{1}{g|x_{in}(k)-p|} \quad (3)$$

if $u > 0.5$

$$x_{in}(k+1) = P_{in} - L * \ln\left(\frac{1}{u}\right) \quad (4)$$

else

$$x_{in}(k+1) = P_{in} + L * \ln\left(\frac{1}{u}\right) \quad (5)$$

$$P_{in} = \frac{r_1 p_{in} + r_2 p_{gn}}{r_1 + r_2} \quad (6)$$

式中: $i=1, 2, \dots, m; n=1, 2, \dots, N$ 表示解空间的维数; $x_{in} \in [-x_{\max}, x_{\max}]$, x_{\max} 表示位置搜索范围; g 为大于 $\ln\sqrt{2}$ 的常数; u, r_1, r_2 是 [0, 1] 之间的随机数; k 为当前迭代次数。

PSO 算法中, 粒子 i 的状态由位置 x_i 和速度 v_i 共同来描述, 两者共同决定了粒子的飞行轨迹, 粒子沿着确定的轨迹运行。但在量子力学中, 根据测不准原理, 粒子的位置 x_i 和速度 v_i 是不能同时被决定的。因此在 QPSO 算法中, 粒子的状态只需用位置向量来描述, 并且算法中只有一个控制参数 g 。该参数的选择和控制非常重要, 它关系到整个算法的收敛速度。

在量子空间中, 粒子满足聚集态的性质是完全不同的, 它可以在整个可行解空间中进行搜索, 因而 QPSO 算法的全局搜索性能远远优于标准 PSO 算法。但是, QPSO 本身是一种中心随机搜索算法, 其算法模型中粒子的分布比标准 PSO 模型更加离散化, 因此当粒子个数比较少时 QPSO 更容易陷入局部最优。另外, QPSO 算法对参数 g 非常敏感, 若 g 值选取过大, 则算法收敛过慢, 寻优过程需要花费较长的时间, 若 g 值过小则相对容易陷入局部最优。

3 QPSCO 算法设计原理

标准 PSO 算法及 QPSO 算法都是着眼于如何更有效地用一个粒子群在解空间中搜索最优解。分析式(1)~式(5)不难发现, 粒子在搜索时, 总是追逐当前全局最优点和自己迄今搜索到的最优值, 因此粒子们的速度很快降到接近于 0, 导致粒子们陷入局部极小而无法摆脱, 本文称这种现象为粒子群的“趋同性”。这种“趋同性”限制了粒子的搜索范围, 要想扩大搜索范围, 就要增加粒子群的粒子数, 或者减弱粒子对整个粒子群当前搜索到的全局最优点的追逐。增加粒子数将导致算法计算复杂度增高, 而减弱粒子对全局最优点的追逐又存在算法不易收敛的缺点。

QPSCO 算法的基本思想是利用 $S(S>1)$ 个独立的量子粒子群进行协同优化。其中, 前 $S-1$ 个量子粒子群根据本粒子群迄今搜索到的最优值来修正群中粒子的位置。将各粒子群中的粒子赋予变异参数 g , 在进化初期 g 选取较大的值, 以拓展搜索空间, 防止过早的陷入局部最优; 在进化过程后期, g 选取一个相对较小的值, 使粒子能够在已经搜索到的最优解区域里进行局部搜索, 以便寻找到最优解:

$$g = \begin{cases} g_1 & k < K \\ g_2 & k \geq K \end{cases} \quad (7)$$

式中, K 为界定进化初期与后期的迭代次数。

第 S 个粒子群则是根据全部粒子群迄今搜索到的最优值修正群中粒子的位置。并对本群中飞离搜索区域粒子的位置用新的位置取代, 这样不仅可以扩大搜索范围, 而且加快了算法收敛。取代策略表示为:

$$\text{if } x_{in} > x_{\max} \text{ then reset } x_{in};$$

为避免 QPSCO 和普通 QPSO 的计算复杂度相同, 该算法并不要求每个粒子群的粒子数相等, 也不要求所有粒子群的粒子状态更新策略相同^[8]。

在适应度函数方面, 本文引进的时域标准综合了超调 σ 、上升时间 t_r 、稳定时间 t_s 和稳态误差 e_s , 并引入权值 β , 克服了传统评价标准如 IAE、ISE 标准中减小超调与缩短稳定时间的矛盾, 弥补了 ISTE 标准解析复杂和计算耗时缺陷。选取适应度函数为:

$$J(X) = (1 - e^{-\beta})(\sigma + e_s) + e^{-\beta}(t_s - t_r) \quad (8)$$

式中: X 为 PID 参数; β 为权值, 可以设定 $0.7 < \beta < 1.5$ 以减小超调和稳态误差, 另一方面可以设定 $0.15 < \beta < 0.7$ 来减小上升时间和稳定时间。

算法流程描述如下:

(1) 初始化前 $S-1$ 个量子粒子群。确定群体规模 m ; 随机产生每个粒子的位置; 给定算法的惯性权重取值范围 $[w_{\min}, w_{\max}]$; 设定最大迭代次数 k_{\max} 。

(2) 将各粒子群中每个粒子的当前位置 x_i 设置为个体极值 p_i , 计算所有粒子的适应度, 该群体适应度最好粒子的 p_i 设置为全局极值 p_g 。

(3) 若某粒子的 x_i 适应度优于 p_i 的适应度, p_i 被 x_i 替换; 若粒子群中当前最优适应度优于 p_g 的适应度, p_g 被当前最优位置替换。

(4) 根据式(4)、(5)更新前 $S-1$ 个粒子群中每个粒子的位置, 并按式(7)对参数 g 进行变异操作, 计算群体适应度。

(5) 检查终止条件, 判断前 $S-1$ 个粒子群是否达到全局收

敛或最大迭代次数。如果达到,继续执行步骤(6);否则返回步骤(3)。

(6)输出上述各粒子群的全局极值 $p_{g1}, p_{g2}, \dots, p_{gS-1}$ 。

(7)初始化第 S 个粒子群。群体规模设为 $S-1 (S-1 \neq m)$, 将前 $S-1$ 个粒子群的 p_g 设置为该粒子群中各粒子的当前位置。

(8)按步骤(3)的方法更新粒子的位置,若粒子飞离搜索区域,则重置该粒子位置。

(9)检查终止条件,判断第 S 个粒子群是否达到全局收敛或最大迭代次数,如果达到,继续执行步骤(10);否则跳转返回步骤(8)。

(10)输出全局极值 p_{gs} ,算法运行结束。

在传统 PSO 算法中,有限的搜索范围将粒子限制在一个固定的区域。而 QPSO 算法中,粒子能够以某一确定的概率出现在整个可行的搜索空间中任意一个位置,而这样的位置可能比当前群体中的 p_g 具有更好的适应度值。QPSCO 的算法结构既利用前 $S-1$ 个量子粒子群的独立搜索来保证寻优搜索过程可以在搜索空间中的较大范围内进行,又利用第 S 个量子粒子群追逐当前全局最优点来保证算法的收敛性,从而兼顾优化过程的精度和效率,克服一般 PSO 算法及 QPSO 算法在收敛性能上的缺陷。

4 仿真实例

为了验证 QPSCO 算法的寻优效果和优越性,将 QPSCO、标准 PSO 和 QPSO 算法分别应用于仿真实例,结合 Simulink 完善的系统仿真功能,交互地进行 PID 参数寻优。

PID 参数优化实质上是对三维函数优化,QPSCO 算法采用实数编码,对 PID 参数寻优的粒子直接进行编码为 $X=(K_p, K_i, K_d)$ 。通过一系列寻优迭代,获得最佳 PID 参数,以此来获得较快的响应速度和较小的超调量。在实际控制系统中,被控对象模型一般为带时滞环节的低阶系统,选取下列典型传递函数进行仿真测试:

$$G(s) = \frac{e^{-0.5s}}{s+1} \quad (9)$$

标准 PSO 算法中,粒子数为 40, $c_1=c_2=2$,迭代次数 50 次, w 随着迭代次数增加由 0.9 变到 0.4。QPSO 算法中,粒子数为 40, $g=1.0, \beta=1.0$;迭代次数 50 次。QPSCO 算法中,粒子群数 $S=10; \beta=1.0$;前 9 个粒子群中,每群粒子数设为 30~40 之间随机数; g 随着迭代次数增加由 1.5 变为 0.8;第 10 个粒子群中, $g=0.8$;前后迭代次数分别取 50 次和 30 次。 K_p, K_i 和 K_d 取值范围(粒子搜索区域)均取 $[0, 5]$ 。仿真结果如图 1、表 1 所示。

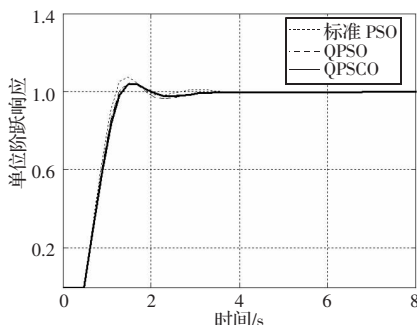


图 1 控制系统单位阶跃响应

表 1 仿真数据

| 参数寻优 | K_p | K_i | K_d | $\sigma/\%$ | e_s | t_s/s |
|--------|-------|-------|-------|-------------|-------|---------|
| 标准 PSO | 1.77 | 1.310 | 0.25 | 5.03 | 0 | 3.5 |
| QPSO | 1.68 | 1.176 | 0.20 | 4.57 | 0 | 3.4 |
| QPSCO | 1.60 | 0.172 | 0.18 | 3.86 | 0 | 2.6 |

由仿真结果可以看出,在同等条件下对 PID 参数寻优,标准 PSO 算法能寻优到较好的 PID 参数,并在足够时间范围内优化,将超调量控制在 5% 左右。QPSO 算法的优化结果要好于前面优化算法,寻优速度也要明显快于标准 PSO 算法,但是系统稳定时间(设 $\Delta=0.2\%$)并不令人满意。QPSCO 合理地照顾了超调与稳定时间,使 PID 参数整定后的系统达到了一个比较理想的状态。从响应曲线总体比较来看,本文所提的 QPSCO 算法优化效果要好于其它 PSO 算法。

5 结论

本文提出了一种基于多量子粒子群的协同优化算法。该算法采用双层结构:在底层由多个量子粒子群独立地搜索解空间,同时引入了参数变异策略,扩大了搜索范围;上层用 1 个量子粒子群追逐当前全局最优解,并重置飞离搜索区域的粒子的位置,从而加快了算法的收敛;各粒子群的粒子数及寻优过程中粒子状态更新策略均不要求相同。将该算法应用于实际控制系统低阶时滞对象的 PID 控制器设计中,仿真结果表明,所提出算法与标准 PSO、QPSO 等算法相比具备了更好的全局收敛性能。

参考文献:

- [1] Zhang Hong, Li Xiao-dong, Li Heng, et al. Particle swarm optimization-based schemes for resource-constrained project scheduling[J]. Automation in Construction, 2005, 14(3): 393-404.
- [2] 王介生, 王金城, 王伟. 基于粒子群算法的 PID 参数自整定[J]. 控制与决策, 2005, 20(1): 73-76.
- [3] Kannan S, Slochanal S M R, Subbaraj P, et al. Application of particle swarm optimization technique and its variants to generation expansion planning problem[J]. Electric Power Systems Research, 2004, 70(3): 203-210.
- [4] Zwe-Lee G.A particle swarm optimization approach for optimum design of PID controller in AVR system[J]. IEEE Transactions on Energy Conversion, 2004, 19(2): 384-391.
- [5] Kennedy J. Probability and dynamics in the particle swarm[C]//Proc of 2004 Congress on Evolutionary Computation, 2004: 340-347.
- [6] Clerc M, Kennedy J. The particle swarm: explosion, stability, and convergence in a multidimensional complex space[J]. IEEE Trans on Control Parameters Affecting Online Performance of Evolutionary Computation, 2002, 6(1): 58-73.
- [7] 李爱国. 多粒子群协同优化算法[J]. 复旦学报: 自然科学版, 2004, 43(5): 923-925.
- [8] Sun Jun, Feng Bin, Xu Wen-bo. Particle swarm optimization with particles having quantum behavior[C]//Proc of 2004 Congress on Evolutionary Computation, 2004: 325-331.
- [9] van Den Bergh F. An analysis of particle swarm optimizers[D]. South Africa: Department of Computer Science, University of Pretoria, 2002.