

基于量子粒子群优化的PID控制参数调节

余永超, 段海滨

北京航空航天大学 自动化科学与电气工程学院 飞行器控制一体化技术重点实验室, 北京 100191
E-mail: 154146928@qq.com

摘要:粒子群优化(Particle Swarm Optimization, PSO)算法是一种经典的仿生智能计算方法, 能解决多类复杂优化问题。但经典的粒子群优化算法有一定的缺陷, 例如易陷入局部极值以及算法本身参数过多等。为了克服这些缺陷, 提出了一种量子粒子群优化算法, 将粒子群优化策略与量子理论相结合, 提高了全局搜索能力, 能有效避免粒子群优化算法的不足。随后, 利用两种算法寻找函数最优解以及整定比例-微分-积分(Proportional-Integral-Derivative, PID)控制参数, 并将优化效果进行了对比分析, 实验结果验证了本文所提方法的可行性和优越性。

关键词: 粒子群优化, 量子粒子群优化, 参数整定, 收敛, PID控制

A Quantum-behaved Particle Swarm Optimization Approach to PID Control Parameters Tuning

Yongchao YU, Haibin DUAN

Science and Technology on Aircraft Control Laboratory, School of Automation Science and Electrical Engineering, Beihang University, Beijing 100191, PR China
E-mail: 154146928@qq.com

Abstract: Particle Swarm Optimization(PSO) algorithm is a bio-inspired computing algorithm which can solve many complicated problems. However, it has some defects such as it can easily fall into a local optimal situation and it has too many parameters. To overcome these defects, a Quantum-behaved Particle Swarm Optimization(QPSO) algorithm is put forward which combine standard particle swarm optimization algorithm and quantum mechanics. This new algorithm can avoid the defects which standard particle swarm optimization algorithm has. By using these two algorithms to find the best solution of a function and tuning Proportional-Integral-Derivative(PID) control parameters and comparing the results, which can demonstrate the feasibility and effectiveness of our proposed approach.

Key Words: particle swarm optimization(PSO), QPSO, parameters tuning, converge, PID control

1 引言

粒子群优化算法(Particle Swarm Optimization, PSO)是由美国心理学家Kennedy和电气工程师Eberhart于1995年提出的一种模拟鸟类群体觅食行为的仿生智能计算方法^[1], 它是一种基于群体智能的随机寻优方法, 该算法利用鸟类群体中个体对信息的共享机制, 使整个群体的运动在问题求解空间中产生从无序到有序的演化过程, 从而获取最优解。

基本粒子群优化算法具有算法简单, 收敛速度快, 易于实现等优点, 但是在处理多目标多峰值函数优化时, 容易陷入局部最优, 且多样性较差。因此, 许多研究人员先后对其进行了改进。Samal等提出了一种闭环控制的PSO动力学方法。通过Jury's test和根轨迹法分析了系统的稳定性, 并将两个加速度系数分开考虑, 选择合适的取值范围来加快算法的收敛^[2]。苗爱敏等利用算法寻优时粒子群的分部信息对算法进行改进, 提出一种动态改变粒子群寻优空间的改进粒子群优化算法, 提高算法收敛效率减少粒子搜索负担^[3]。马琳等提出一种基于免疫网络的改进多目标粒

子群优化算法。该算法通过免疫网络互通种群最优信息达到粒子群算法与人工免疫网络算法的协同搜索, 使搜索效果更优, 并将所提出的方法应用于求解电弧炉供电优化模型, 取得良好的效果^[4]。

比例-微分-积分(Proportional-Integral-Derivative, PID)控制是最早发展起来的控制技术之一, 由于其算法简单, 鲁棒性好, 可靠性高等特点被广泛应用于工业控制领域。PID控制效果的好坏取决于参数 K_p, K_i, K_d 的选取, 因此如何选取这三个参数对能否实现预想的控制效果尤为关键。目前, PID参数整定策略有很多种, 每一种整定策略在不同的环境中证实了其有效性。本文给出了一种改进粒子群智能算法-量子粒子群优化算法(Quantum-behaved Particle Swarm Optimization, QPSO)来优化PID参数, 该算法具有简单有效, 收敛速度快, 全局搜索性强的优点, 弥补了基本粒子群优化算法搜索空间有限, 易陷入局部极值的不足。

当前, 量子粒子群优化算法的相关研究及文献数量众多, 但将该算法应用于PID参数优化的研究为数不多。此外, 本文较之前相关的论文最大的特色在于

*航空科学基金(20115151019), 北航“驭远”大学生科技创新基金。

对比了基本粒子群优化算法及量子粒子群优化算法优化函数和整定PID参数的效果，直观地验证了量子理论与智能算法结合的可行性。

2 基本粒子群优化算法

粒子群优化算法的基本思想是通过群体中个体之间的协作和信息共享来寻找最优解，是一种基于群体智能的仿生计算方法。该算法采用速度-位置模型，每个粒子代表 n 维空间中的一个候选解，粒子在搜索空间中以一定的速度飞行，该飞行速度由个体的飞行经验和群体的飞行经验进行动态调整^[5]。其中设粒子 i 的当前位置为 $X_i = (x_{i1}, x_{i2}, \dots, x_{in})$ ，飞行速度为 $V_i = (v_{i1}, v_{i2}, \dots, v_{in})$ ；粒子 i 的自身最优位置为 $P_i = (p_{i1}, p_{i2}, \dots, p_{in})$ ，整个粒子群搜索到的最优解为 $P_g = (p_{g1}, p_{g2}, \dots, p_{gn})$ 。粒子群的速度和位置更新公式如下：

$$v_{ij}(t+1) = wv_{ij}(t) + c_1r_{1j}(t)(p_{ij}(t) - x_{ij}(t)) + c_2r_{2j}(t)(p_{gj}(t) - x_{ij}(t)) \quad (1)$$

$$x_{ij}(t+1) = x_{ij}(t) + v_{ij}(t+1) \quad (2)$$

式中，下标“ j ”表示粒子的第 j 维；“ i ”表示粒子 i ； t 表示第 t 代； c_1, c_2 为加速常数，通常在0~2内取值； $r_1 \sim U(0,1)$ ， $r_2 \sim U(0,1)$ 为两个相互独立的随机函数； $v_{ij} \in [-v_{\max}, v_{\max}]$ ，表示粒子搜索速度范围； w 为惯性权重系数，其更新公式通常如下：

$$w = w_{\max} - (w_{\max} - w_{\min}) \times t / t_{\max} \quad (3)$$

其中， w_{\max} 是初始惯性权重， w_{\min} 是最终惯性权重， t_{\max} 是最大迭代次数， t 是当前迭代次数^[5]。

基本粒子群优化算法流程如下：

Step 1 参数初始化。对粒子群的随机位置和速度进行初始设定。

Step 2 计算每个粒子的适应值。

Step 3 对于每个粒子，将其适应值与所经历过的最好位置 $P_i(t)$ 的适应值进行比较，若较好，则将其作为当前的最好位置。

Step 4 对每个粒子，将其适应值与全局所经历的最好位置 $P_g(t)$ 的适应值进行比较，若较好，则将其作为当前的全局最好位置。

Step 5 根据式(1)，(2)对粒子的速度和位置进行更新。

Step 6 如满足终止条件（设定为一个足够好的适应值或达到一个预设的最大迭代次数），算法结束，否则返回Step 2。

在基本粒子群优化算法中，粒子是在经典力学状态下沿着确定的轨迹飞行，因此粒子搜索的空间是一个有限的区域，因而算法容易陷入局部极值而使结果不一定是全局最优解。这也是基本粒子群优化算法的一大不足。

3 量子粒子群优化算法

量子粒子群优化算法是基于量子力学的相关理论对基本粒子群优化法进行改进得到的一种算法。两种算法的主要区别在于：量子粒子群优化算法认为粒子具有量子行为，即每一个粒子的速度和位置具有不确定性，所以每个粒子的轨迹具有不确定性，这使粒子可以在整个可行解空间中进行探索寻找全局最优解，从而可以避免基本粒子群优化算法容易陷入局部极值的缺陷，同时可以提高算法的全局收敛性^[6]。量子粒子群优化算法的进化公式如下：

$$mbest(t+1) = \frac{1}{M} \sum_{i=1}^M P_i(t) \quad (4)$$

$$= \left(\frac{1}{M} \sum_{i=1}^M P_{i1}(t), \frac{1}{M} \sum_{i=1}^M P_{i2}(t), \dots, \frac{1}{M} \sum_{i=1}^M P_{in}(t) \right)$$

$$PP_{ij}(t+1) = f_{ij}(t+1) \times P_{ij}(t) + (1 - f_{ij}(t+1)) \times P_{gj}(t) \quad (5)$$

如果 $f_{ij}(t+1) \geq 0.5$

$$X_{ij}(t+1) = PP_{ij}(t+1)$$

$$-a(t+1) | mbest_j(t+1) - X_{ij}(t) | \ln\left(\frac{1}{u_{ij}(t+1)}\right) \quad (6)$$

如果 $f_{ij}(t+1) < 0.5$

$$X_{ij}(t+1) = PP_{ij}(t+1)$$

$$+a(t+1) | mbest_j(t+1) - X_{ij}(t) | \ln\left(\frac{1}{u_{ij}(t+1)}\right) \quad (7)$$

式中， $f_{ij}(t+1)$ ， $u_{ij}(t+1)$ 表示[0,1]之间服从均匀分布的随机数， M 为粒子总数， n 表示粒子的维数， $P_i(t)$ 表示第 t 次迭代时第 i 个粒子的当前最佳位置， $P_g(t)$ 表示第 t 次迭代时的全局最佳位置，这里， $mbest(t+1)$ 是粒子群中所有粒子第 t 次迭代时当前最佳位置的中间位置， $PP_{ij}(t)$ 为粒子个体最优值 $P_i(t)$ 和群体全局最优值 $P_g(t)$ 之间的随机点^[5]。 $a(t)$ 为收缩扩张系数，调节它的值能控制算法的收敛速度，较大的 $a(t)$ 有利于算法跳出局部最优值，而较小的 $a(t)$ 有利于算法收敛。一般来讲，在全局搜索算法中，希望前期有较高的搜索能力以得到合适的种子，而在后期有较高的开发能力，以加快收敛速度。所以，建议算法前期时使用较大的 $a(t)$ ，以鼓励早期搜索，随着算法进行，逐渐减小 $a(t)$ ，最后把重点放在群体对最佳位置的搜索。因此， $a(t)$ 的值可由下式得出：

$$a(t) = m - (m - n) \times \frac{t}{N_c} \quad (8)$$

由上式可见， $a(t)$ 随着迭代次数的增加将从 m 线性递减到 n 。

量子粒子群优化算法的性能相对于基本粒子群优化算法的优点主要体现在以下三个方面:

- (1) 量子理论是一个“不确定”的体系。在该理论体系中可以产生更加具有随机性的不同状态的微粒,并具有更广泛的搜索空间。因此,量子粒子群优化算法可以产生多样性更强的种群。
- (2) 引进 *mbest* 是量子粒子群优化算法的另一大改进。在基本粒子群优化算法中,收敛速度很快,但有时很容易在最初的几代中就会发生收敛,从而陷入局部最优的情况。而量子粒子群优化算法中引进 *mbest*,由于每个粒子不能忽略其他粒子的影响而过快收敛,收敛的平均误差降低,这使陷入局部的可能性远低于基本粒子群优化算法。
- (3) 量子粒子群优化算法中的参数比基本粒子群优化算法少,更易于编程并运行。

因此,量子粒子群优化算法会表现出优于基本粒子群优化算法的性能^[7]。

量子粒子群优化算法流程如下:

- Step 1 参数初始化。对粒子群的随机位置和速度进行初始设定。
- Step 2 计算每个粒子的适应值。
- Step 3 对于每个粒子,将其适应值与所经历过的最好位置 $P_i(t)$ 的适应值进行比较,若较好,则将其作为当前的最好位置。
- Step 4 对每个粒子,将其适应值与全局所经历的最好位置 $P_g(t)$ 的适应值进行比较,若较好,则将其作为当前的全局最好位置。
- Step 5 根据式(4)~(7)对粒子的位置进行更新。
- Step 6 如满足终止条件(设定为一个足够好的适应度值或达到一个预设的最大迭代次数),算法结束,否则返回Step 2。

4 仿真实验

本文运用MATLAB作为开发环境,分别用基本粒子群优化算法和量子粒子群优化算法进行函数求极值的运算以及PID参数的整定,并对结果进行比较和分析。

仿真软件环境:

操作系统: Windows 7
平台: MATLAB 7.11.0.584(R2010b)

仿真硬件环境:

CPU: Intel(R) Core(TM) i7-2630QM
内存: 4.00 GB
硬盘: 640 GB

4.1 函数优化

选取一多峰值函数:

$$f(x_1, x_2) = 0.5 + \frac{\sin^2 \sqrt{x_1^2 + x_2^2} - 0.5}{(1.0 + 0.001(x_1^2 + x_2^2))^2}$$

其中, $-100 \leq x_1, x_2 \leq 100$ 。该函数的全局极小点是(0,0),而在距全局极小点大约3.14范围内凹陷部有无数多的局部极小点,一般算法很难搜索到全局极小点。

用两种算法分别对该函数求极值,为避免偶然误差,两算法各运行三次,对每一代的最优适应值绘制成曲线。

设置初始试验参数:

粒子总数为20,迭代次数为1000,基本粒子群算法中 $c_1=2, c_2=2, w_{\max}=0.9, w_{\min}=0.1, v_{\max}=2, v_{\min}=-v_{\max}$;量子粒子群算法中 $m=1, n=0.5$ 。

仿真结果如下:

(1) 基本粒子群优化算法

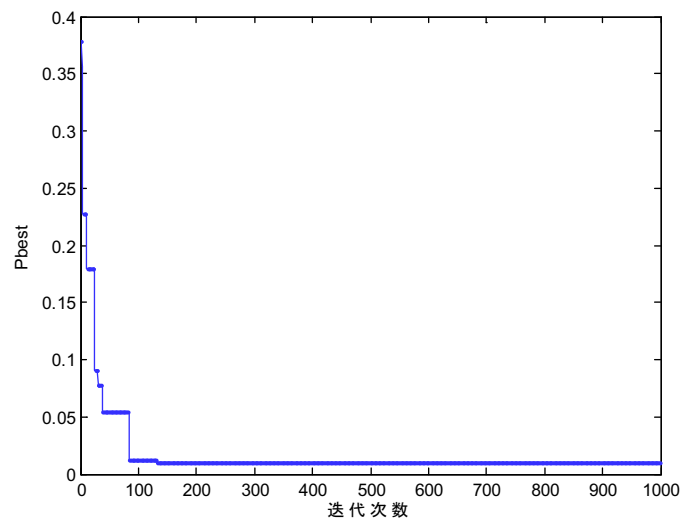


图 1: PSO第一次运行结果

稳定时, $f=0.0097, x_1=-1.7563, x_2=-2.6011$ 。

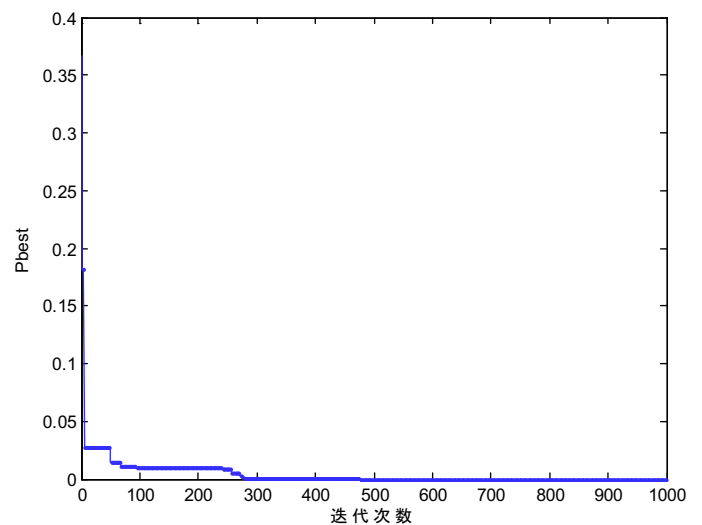


图 2: PSO第二次运行结果

稳定时, $f=0, x_1=0, x_2=0$ 。

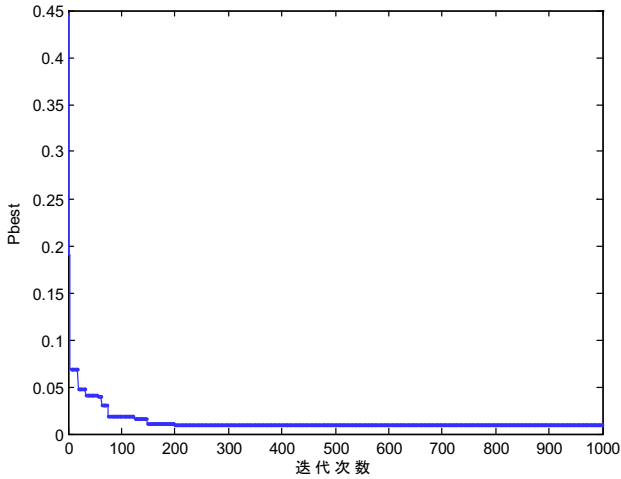


图 3: PSO第三次运行结果

稳定时, $f=0.0097$, $x_1=-2.4929$, $x_2=1.9067$ 。

(2) 量子粒子群优化算法

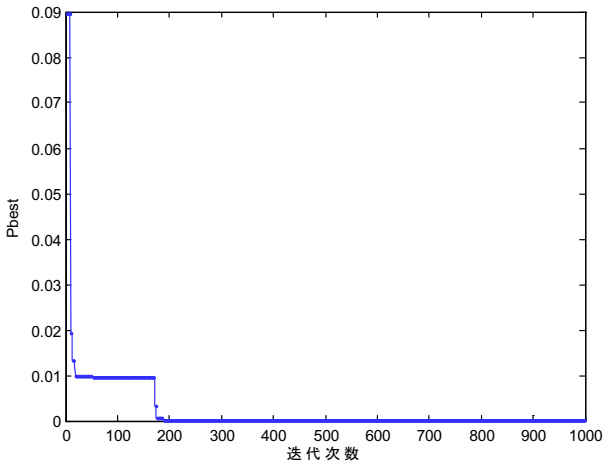


图 4: QPSO第一次运行结果

稳定时, $f=0$, $x_1=0$, $x_2=0$ 。

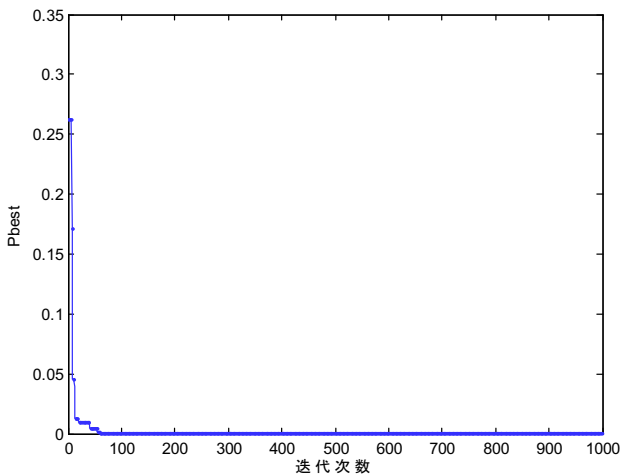


图 5: QPSO第二次运行结果

稳定时, $f=0$, $x_1=0$, $x_2=0$ 。

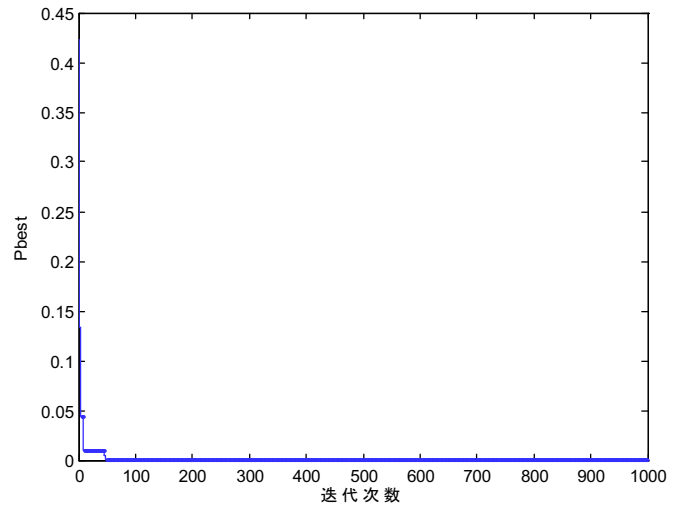


图 6: QPSO第三次运行结果

稳定时, $f=0$, $x_1=0$, $x_2=0$ 。

从仿真结果中可以清晰地看出,基本粒子群优化算法在处理此问题时,容易陷入局部最优且收敛速度较慢,而量子粒子群优化算法可以轻松找到全局最优解并且收敛速度很快。在此问题中,量子粒子群优化算法表现出了更为良好的性能。

4.2 PID参数的整定

假设被控对象为二阶系统

$$G(s) = \frac{400}{s^2 + 50s}$$

采样时间为1ms,输入指令为一阶跃信号。为获取满意的过度过程动态特性,采用误差绝对值时间积分性能指标作为参数选择的最小目标函数。为了防止控制能量过大,在目标函数中加入控制输入的平方项。选用下式作为参数选取的最优指标:

$$J = \int_0^{\infty} (w_1 |e(t)| + w_2 u^2(t)) dt + w_3 \cdot t_u \quad (9)$$

式中, $e(t)$ 为系统误差, $u(t)$ 为控制器输出, t_u 为上升时间, w_1 , w_2 , w_3 为权值。

为了避免超调,采用了惩罚功能,即一旦产生超调,将超调量作为最优指标的一项,此时最优指标为 $\text{If } e(t) < 0$

$$J = \int_0^{\infty} (w_1 |e(t)| + w_2 u^2(t) + w_4 |e(t)|) dt + w_3 \cdot t_u \quad (10)$$

令参数 K_p 的取值范围为[0,20], K_i, K_d 的取值范围为[0,1], 取 $w_1=0.999$, $w_2=0.001$, $w_3=2.0$, $w_4=100$ 。分别用两算法进行参数整定。绘制代价函数 J 的优化过程和采用整定后的PID控制阶跃响应图像。

设置初始试验参数:

粒子总数为20;

迭代次数为100;

基本粒子群算法中 $c_1=2$, $c_2=2$, $w_{\max}=0.9$,

$w_{\min}=0.1$, $v_{\max}=0.001$, $v_{\min}=-v_{\max}$; 量子粒子群算法中 $m=1$, $n=0.5$ 。

仿真结果如下:

(1) 基本粒子群优化算法

经100次迭代, 获得优化参数如下: PID整定结果为 $K_p=20.0000$, $K_d=0.2538$, $K_i=0.0035$; 性能指标 $J=24.0122$ 。

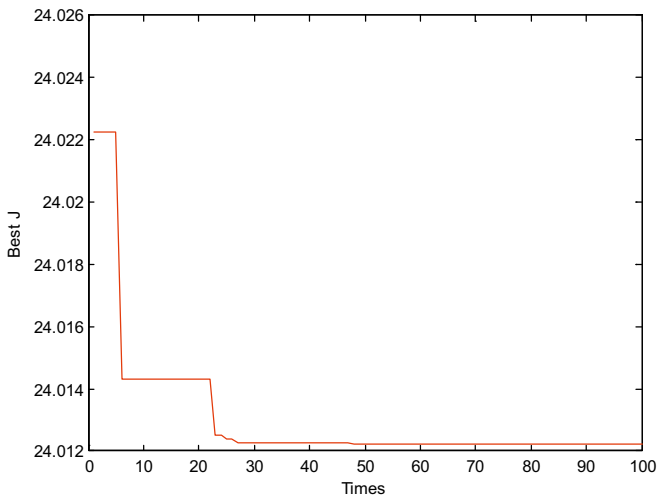


图 7: 代价函数 J 的优化过程

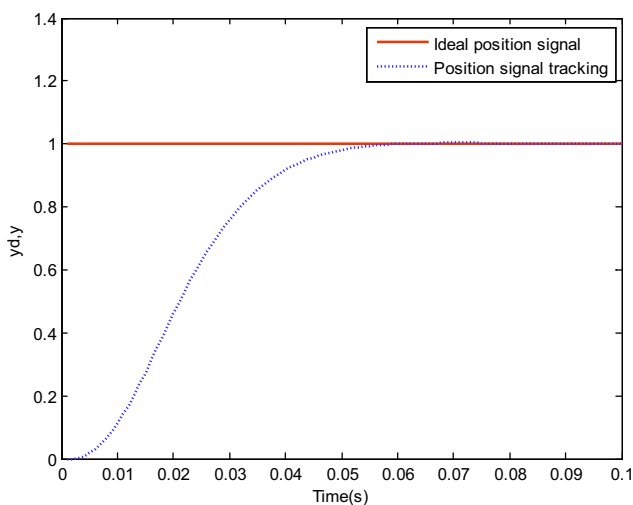


图 8: 整定后的PID阶跃响应

(2) 量子粒子群优化算法

经100次迭代, 获得优化参数如下: PID整定结果为 $K_p=20.0000$, $K_d=0.2537$, $K_i=0.0039$; 性能指标 $J=24.0123$ 。

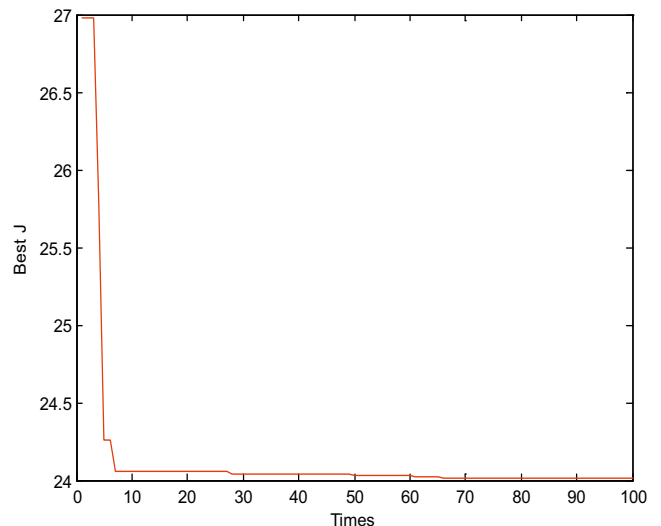


图 9: 代价函数 J 的优化过程

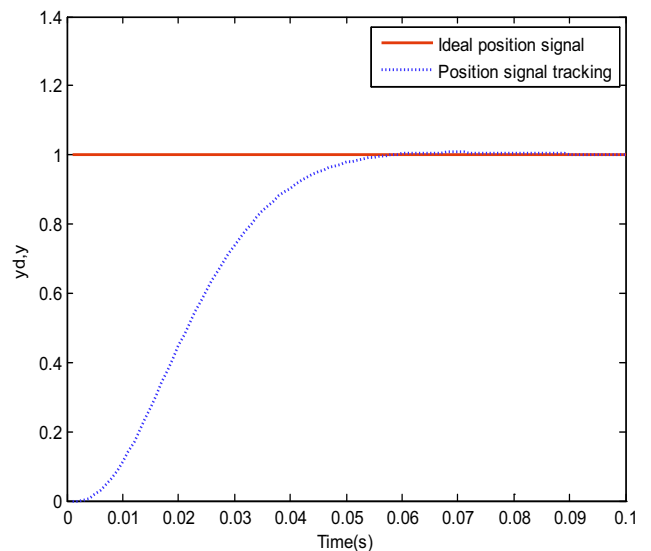


图 10: 整定后的PID阶跃响应

由仿真结果可见, 基本粒子群优化算法与量子粒子群优化算法整定PID后所得的最优参数相差无几, 整定后的PID控制阶跃响应也基本一致, 调节品质良好。然而在优化过程中量子粒子优化算法的收敛速度明显快于基本粒子群优化算法, 由此可见在PID参数整定中, 量子粒子群优化算法较之基本粒子群优化算法表现出更好的性能。

5 结束语

本文将一种量子粒子群优化算法应用于函数优化以及PID参数的整定, 并与基本粒子群算法进行了实验对比分析。量子粒子群优化算法相比基本粒子群优化算法, 具有更好的全局搜索性以及更快的收敛速度, 同时能有效避免陷入局部极值。

参考文献

[1] J. Kennedy, R.C. Eberhart. Particle swarm optimization,

- Proceedings of IEEE International Conference on Neural Networks*, Perth, WA, 1995, vol. 4, 1942-1948.
- [2] N. R. Samal, A. Konar, S. Das, A. Abraham. A closed loop stability analysis and parameter selection of the particle swarm optimization dynamics for faster convergence. *Proceedings of IEEE Congress on Evolutionary Computation*, Singapore, 2007, 1769-1776.
- [3] 苗爱敏, 施心陵, 宋执环. 基于动态特性的改进粒子群优化算法. *中国科技论文在线*, 2011, 6(10), 754-760.
- [4] 马琳, 毛志忠, 袁平. 一种改进的多目标粒子群优化算法及应用. *控制与决策*, 2012, 27(9): 1313-1319.
- [5] 段海滨, 张祥银, 徐春芳. 仿生智能计算. 北京: 科学出版社, 2011.
- [6] 冯斌, 王璋, 孙俊. 基于混沌变异算子的小生境量子粒子群算法. *计算机应用与软件*, 2009, 26(1), 50-52.
- [7] F. Liu, H. Duan, Y. Deng. A chaotic quantum-behaved particle swarm optimization based on lateral inhibition for image matching. *Optik*, 2012, 123(21), 1955-1960.