

文章编号: 1001-0920(2012)05-0719-05

# 一种采用完全学习策略的量子行为粒子群优化算法

陈伟, 周颀, 孙俊, 须文波

(江南大学物联网工程学院, 江苏无锡 214000)

**摘要:** 为了进一步提高量子行为粒子群优化(QPSO)算法的全局收敛性能, 有效改善算法中存在的粒子早熟问题, 提出一种基于完全学习策略的改进 QPSO 算法(CLQPSO). 该学习策略改变了 QPSO 中局部吸引子的更新方式, 充分利用了种群的社会信息. 采用 8 个测试函数对算法性能进行比较分析. 实验结果表明, 所提出的改进算法不仅收敛速度快, 而且全局收敛能力好, 收敛精度优于 PSO 算法和 QPSO 算法.

**关键词:** 粒子群优化; 量子行为粒子群优化; 完全学习策略; 局部吸引子

中图分类号: TP301.6

文献标识码: A

## Improved quantum-behaved particle swarm optimization algorithm based on comprehensive learning strategy

CHEN Wei, ZHOU Di, SUN Jun, XU Wen-bo

(School of Internet of Things Engineering, Jiangnan University, Wuxi 214000, China. Correspondent: CHEN Wei, E-mail: chenwei\_0906@yahoo.com.cn)

**Abstract:** A quantum-behaved particle swarm optimization(CLQPSO) algorithm based on comprehensive learning strategy is presented, which helps prevent the original quantum-behaved particle swarm optimization(QPSO) algorithm's tendency to be easily trapped into local optima as a result of the rapid decline in diversity. The learning strategy changes the updating method of local attractor in QPSO, which makes fully use of the social information of the swarm. The 8 benchmark functions are used to test the performance of CLQPSO. The experiments results show that the proposed algorithm can find better solutions than the original QPSO algorithm and the PSO algorithm with higher efficiency.

**Key words:** particle swarm optimization; quantum-behaved particle swarm optimization; comprehensive learning strategy; local attractor

## 1 引言

粒子群优化(PSO)算法是由Kennedy和Eberhart<sup>[1]</sup>在 1995 年提出的一种群体智能优化技术. 其基本思想是受他们早期对鸟类的群体行为进行建模与仿真研究结果的启发. PSO 算法具有模型简单, 便于实现的优点, 是非线性连续优化问题、组合优化问题和混合整数非线性优化问题的有效求解工具. 因此, 从算法提出至今在科学研究的各个领域已有广泛的应用<sup>[2]</sup>. PSO 算法的最大缺点在于算法不能保证以概率 1 收敛于全局最优解<sup>[3]</sup>, 这就意味着在面对复杂的多峰优化问题时, PSO 算法往往只能找到问题的局部最优解, 从而降低了问题的求解精度. 面对 PSO 算法的收敛瓶颈, Sun 等人<sup>[4]</sup>从量子力学的角度出发, 提出了一种新的 PSO 模型——量子行为粒子

群优化(QPSO)算法. 该模型以 Delta 势阱为基础, 认为粒子具有量子行为. 理论证明 QPSO 算法是一种全局收敛算法, 目前在各个领域已取得优于 PSO 算法的效果<sup>[5]</sup>. 然而无论是 PSO 算法还是 QPSO 算法, 不可避免地与其他进化算法一样, 在搜索过程中容易因为粒子的早熟而陷入局部最优解. 粒子的早熟往往是由于随着算法的迭代, 种群多样性逐渐丧失, 粒子无法跳出当前搜索区域而造成的. 为了进一步提高 QPSO 算法在求解复杂优化问题时的性能, 本文对 QPSO 算法中粒子局部吸引子的更新进行了改进, 采用一种社会性更突出的学习策略, 提出一种新的基于完全学习策略的 QPSO 优化算法.

## 2 量子行为粒子群优化算法

在 PSO 算法中, 每一个粒子代表优化问题的一

收稿日期: 2010-11-26; 修回日期: 2011-02-28.

基金项目: 国家 863 计划项目(2008AA02Z21); 国家自然科学基金项目(60703106, 60474030).

作者简介: 陈伟(1981-), 男, 博士生, 从事人工智能、群体智能算法的研究; 须文波(1946-), 男, 教授, 博士生导师, 从事人工智能、模糊控制等研究.

个可行解, 解的优劣程度由适应度函数决定, 适应度函数取决于实际的优化问题. 假设  $M$  为种群粒子的数目, 对于第  $i$  个粒子 ( $1 \leq i \leq M$ ), 粒子在搜索空间中的当前位置记为  $X_i = (x_{i1}, x_{i2}, \dots, x_{iD})$ ,  $D$  为粒子的维数. 在算法的每一次迭代过程中, 每个粒子当前经历的个体最好位置记为  $p_{\text{best}_i} = (p_{\text{best}_{i1}}, p_{\text{best}_{i2}}, \dots, p_{\text{best}_{iD}})$ , 种群中所有粒子当前经历的全局最好位置记为  $g_{\text{best}_i} = (g_{\text{best}_1}, g_{\text{best}_2}, \dots, g_{\text{best}_D})$ .

根据 Clerc 等人<sup>[6]</sup>对 PSO 算法中粒子运行轨迹的分析研究表明, 要保证 PSO 算法的收敛性, 每一个粒子必须收敛于各自的局部吸引子, 即

$$q_{ij} = \varphi_j p_{\text{best}_{ij}} + (1 - \varphi_j) g_{\text{best}_j},$$

$$j = 1, 2, \dots, D, \quad (1)$$

其中  $\varphi_j = c_1 r_{1j} / (c_1 r_{1j} + c_2 r_{2j})$ . 这里:  $r_{1j}, r_{2j}$  为两个在 (0,1) 范围内均匀分布的随机数序列;  $c_1, c_2$  为 PSO 算法中的加速因子, 在通常情况下  $c_1 = c_2$ . 因此,  $\varphi_j$  为一个在 (0,1) 范围内均匀分布的随机数序列.

现在假设每个粒子在一个量子空间中运行, 并且具备量子行为, 那么粒子的状态便可通过其波函数  $\psi$  进行表征, 而此时  $|\psi|^2$  就是粒子位置的概率密度函数. 通过在  $q_i(t)$  的每一维上建立一个 Delta 势阱模型以防止粒子发散, 求解相应的薛定谔方程可得到粒子每一维位置在搜索空间中的概率分布函数为

$$F(x_{ij}) = e^{-2|x_{ij} - q_{ij}|/L}, \quad (2)$$

其中  $L$  为 Delta 势阱的特征长度, 它决定了粒子的搜索范围. 采用蒙特卡罗模拟的方法, 可得到粒子位置的进化公式为

$$x_{ij} = q_{ij} \pm \frac{L}{2} \ln(1/u), \quad (3)$$

其中  $u$  为 (0,1) 范围内均匀分布的随机数. 由于  $L$  的控制方法对 QPSO 算法的收敛速度和性能有关键性的影响, 文献 [7] 引入一个称为“平均最优位置”的全局变量  $m_{\text{best}}$ , 定义  $m_{\text{best}}$  为所有粒子当前最优位置的平均值, 即

$$m_{\text{best}} = \left( \frac{1}{M} \sum_{i=1}^M p_{\text{best}_{i1}}(t), \frac{1}{M} \sum_{i=1}^M p_{\text{best}_{i2}}(t), \dots, \frac{1}{M} \sum_{i=1}^M p_{\text{best}_{iD}}(t) \right). \quad (4)$$

采用  $x_{ij}$  和  $m_{\text{best}_j}$  之间的间隔对  $L$  进行评价, 即

$$L = 2\alpha |m_{\text{best}_j} - x_{ij}|. \quad (5)$$

则式 (3) 的粒子位置进化公式最后定义为

$$x_{ij}(t+1) = q_{ij}(t) \pm \alpha |m_{\text{best}_j}(t) - x_{ij}(t)| \ln(1/u), \quad (6)$$

其中  $\alpha$  称作收缩扩张因子, 用来控制粒子的收敛速度. 仿真实验结果表明, 当  $\alpha < 1.782$  时, 可保证算法的收

敛性<sup>[8]</sup>. 在大多数情况下, 当  $\alpha$  随着迭代步骤从  $\alpha_0$  线性减小到  $\alpha_1$  ( $\alpha_0 > \alpha_1$ ) 时, 能使算法取得比较理想的性能. 由此将满足粒子进化方程 (6) 的 PSO 模型称作量子行为粒子群优化 (QPSO) 算法.

### 3 基于完全学习策略的量子行为粒子群优化算法

#### 3.1 完全学习策略

尽管已经提出了许多 QPSO 算法的改进方法, 但在求解复杂的多峰问题时粒子的早熟仍然是 QPSO 算法的主要不足之处. 为了进一步改善 QPSO 算法的收敛性能, 本文提出一种新的学习策略来改进原来的 QPSO 算法. 在改进的算法中, 种群中所有粒子的个体最优位置  $p_{\text{best}}$  都被用来更新粒子的局部吸引子, 这种新的学习策略保证了种群的多样性, 从而防止了粒子的过早收敛.

在该学习策略中, 采用如下粒子位置进化公式:

$$\begin{cases} x_{ij}(t+1) = q_{ij}(t) \pm \alpha |q_{ij}(t) - x_{ij}(t)| \ln(1/u), \\ q_{ij}(t) = p_{\text{best}_{f_i(j)}}(t). \end{cases} \quad (7)$$

其中当前粒子局部吸引子的位置不再由式 (1) 计算得到, 而是各维的坐标随机取自任意一个粒子个体最佳位置对应维的值, 由函数  $f_i = [f_i(1), f_i(2), \dots, f_i(D)]$  决定  $q_{ij}$  取自哪一个  $p_{\text{best}}$  的对应维.  $q_{ij}$  可取自包括当前粒子自身在内的种群中所有  $p_{\text{best}}$  对应维的值, 为此引入参数学习概率  $p_c$  进行决策. 具体学习过程为: 对于粒子  $i$  的每一维, 产生一个随机数  $\text{rand}()$ , 如果  $\text{rand}() > p_c$ , 则  $q_{ij}$  取自当前粒子自身的  $p_{\text{best}}$ ; 否则就要向其他粒子的  $p_{\text{best}}$  进行学习. 当  $q_{ij}$  需要向其他粒子的  $p_{\text{best}}$  学习时, 从种群中 (除去当前粒子以外) 随机选取两个粒子的  $p_{\text{best}}$ , 比较所选取的两个粒子  $p_{\text{best}}$  的目标函数值, 选择其中具有较优目标函数值的一个, 此时  $q_{ij}$  更新为被选择的粒子的  $p_{\text{best}}$  相应维的值. 如果  $q_{ij}$  的所有维决策变量均取自其自身的  $p_{\text{best}}$ , 则在  $q_{ij}$  中随机选择某一维决策变量向其他粒子的  $p_{\text{best}}$  进行学习.

采用以上局部吸引子更新方式的 QPSO 算法, 在迭代更新时能充分利用种群中所有粒子个体最佳位置所提供的有用信息, 将其定义为基于完全学习策略<sup>[9]</sup>的量子行为粒子群优化 (CLQPSO) 算法. 具体算法流程描述如下:

Step 1: 初始化粒子的位置, 即随机产生粒子当前位置  $X_i$ , 并将个体最好位置  $p_{\text{best}_i}$  置为  $p_{\text{best}_i} = X_i$ ;

Step 2: 评价每一个粒子的目标函数值  $f(X_i)$  (适应度值), 并与个体最好位置的目标函数值  $f(p_{\text{best}_i})$  进行比较, 如果  $f(X_i) < f(p_{\text{best}_i})$ , 则置  $p_{\text{best}_i} = X_i$ ;

Step 3: 找出群体的全局最好位置, 即先求  $g =$

$\arg \min_{1 \leq i \leq M} (f(p_{best_i}))$ , 令  $g_{best} = p_{best_g}$ ;

Step 4: 对于每一个粒子, 采用完全学习策略更新局部吸引子  $q_i$ ;

Step 5: 由式 (7) 计算更新粒子的新位置;

Step 6: 重复 Step 2~Step 5, 直到满足停止准则或达到给定的最大迭代数为止.

### 3.2 CLQPSO 算法的搜索行为分析

在 QPSO 算法中, 粒子  $i$  从个体当前位置到局部吸引子  $q_i$  点方向进行搜索. 由式 (1) 可见, 当粒子进行学习时,  $q_i$  点由自我学习和社会学习的相互作用产生,  $q_i$  点称为个体的学习倾向点 (LIP)<sup>[10]</sup>. 学习的倾向使个体必然在由已知的个体知识  $p_{best_i}$  和社会信息  $g_{best}$  决定的 LIP 附近寻找新的知识. 粒子的社会学习行为完全体现在种群粒子对  $g_{best}$  的信息共享, 如果种群的全局最优位置  $g_{best}$  在某一个远离全局最优解的局部最优区域之中, 则由于式 (1) 计算 LIP 点的局限性, 种群中的所有粒子仍然会向  $g_{best}$  学习, 导致粒子也随之陷入  $g_{best}$  所在的区域, 使得 QPSO 算法在有限的迭代步骤以内, 可能无法跳出这个区域, 从而造成算法在迭代结束时仅仅收敛到一个局部最优解. 而在所提出的新的学习策略中, LIP 的更新不仅仅局限于  $p_{best_i}$  和  $g_{best}$  界定的超矩形中, 每一维都有可能向任何粒子的  $p_{best}$  进行学习. 因此本文充分利用了种群中所有粒子提供的社会信息, 并且通过粒子间的社会合作行为来提高种群多样性, 以此改善算法在求解多峰优化问题时的收敛性能.

另外, QPSO 算法是在局部吸引子  $q_i$  点建立一个 Delta 势阱模型以模拟粒子的学习倾向性. 在执行下一步知识寻找前, 参数  $L$  必须先被评价.  $L$  可看作是个体的创造力或想象力,  $L$  值越大, 粒子找到新知识的可能性就越大, 创造性越强, 它决定了粒子知识搜寻的范围. 为了比较 QPSO 算法和 CLQPSO 算法的潜在知识搜寻空间, 假设式 (5) 中  $\alpha = 0.5$  固定不变, 则 QPSO 算法和 CLQPSO 算法中单个粒子在第  $j$  维上的潜在搜索空间可分别表示为:  $r_{ij}^1 = |m_{best_j} - x_{ij}|$ ,  $r_{ij}^2 = |p_{best_{f_i(j)}} - x_{ij}|$ ; 单个粒子在整个解空间的搜索范围分别表示为:  $R_i^1 = \prod_{j=1}^D r_{ij}^1$ ,  $R_i^2 = \prod_{j=1}^D r_{ij}^2$ ;  $\bar{R}^1, \bar{R}^2$  分别表示 QPSO 算法和 CLQPSO 算法种群所有粒子搜索范围的平均值. 对 4.1 节中定义的 Sphere 函数 (单峰) 和 Griewank 函数 (多峰) 进行优化求解, 两种算法分别独立运行 20 次, 并记录每次迭代时  $\bar{R}^1$  和  $\bar{R}^2$  的值.  $\bar{R}^1, \bar{R}^2$  随算法迭代次数变化的曲线如图 1 所示. 从图 1 可以看出, 无论是单峰还是多峰函数, CLQPSO 算法对参数  $L$  的评价能在算法的进化过程中产生更

大的知识搜索空间, 使算法更具有创造性. 这主要是因为 CLQPSO 算法中新的学习策略使 LIP 点的选取更多元化, 而每一个粒子的  $p_{best}$  都有可能提供有用的社会信息. 因此, 相比 QPSO 算法, CLQPSO 算法可在更广泛的潜在区域中搜索优化问题的最优解, 以此来改善算法的收敛性能.

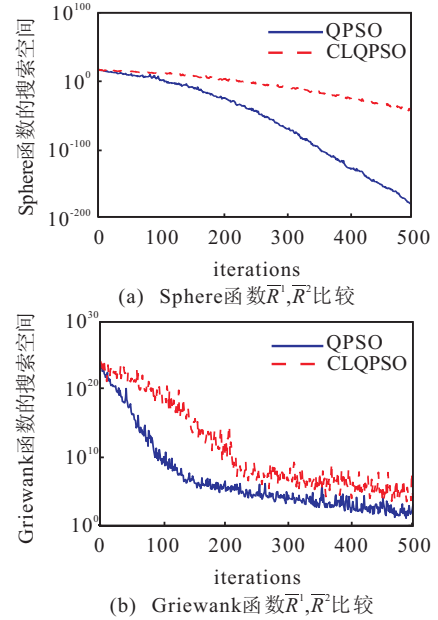


图 1 QPSO 和 CLQPSO 潜在搜索空间的比较

### 3.3 学习概率 $p_c$

在 CLQPSO 算法的设计和数值实验过程中发现, 如果对于种群中的所有粒子都设置相同的学习概率  $p_c$ , 则当  $p_c$  取不同的值时, 会在同一个优化函数上产生不同的结果. 尤其在多峰函数的优化问题中, 由于  $p_c$  值的大小决定了粒子的全局勘探能力和局部开发能力,  $p_c$  的选取会对算法的结果产生关键性的影响. 为了使种群粒子的勘探能力和开发能力能达到一个理想的平衡状态, 使 CLQPSO 算法能适应不同类型目标函数的优化问题, 对种群中的第  $i$  ( $1 \leq i \leq M$ ) 个粒子采用如下学习概率设置方法:

$$p_{c_i} = 0.05 + 0.45 \frac{\exp\left(\frac{10(i-1)}{M-1}\right) - 1}{\exp(10) - 1}. \quad (8)$$

即各粒子的  $p_c$  按照式 (8) 的非线性规律由 0.05 增加到 0.5. 本文实验中种群的粒子数为 30, 各粒子学习概率  $p_c$  的分配如图 2 所示.

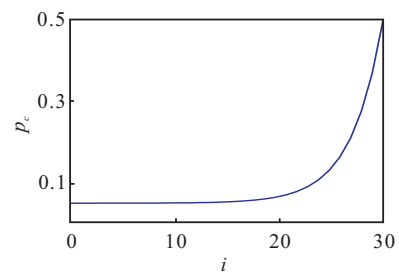


图 2 种群粒子数为 30 时学习概率  $p_c$  的分配

## 4 数值仿真实验

### 4.1 测试函数

为了测试 CLQPSO 算法的性能, 本文选取 8 个标准测试函数对各算法进行数值仿真实验. 各个测试函数的表达式、搜索范围以及初始化范围如表 1 所示. 在这 8 个测试函数中,  $f_1$  为 Sphere 函数,  $f_2$  为 Rosenbrock 函数, 二者为单峰函数, 前者比较简单, 后者在极小值附近有陡峭的峡谷, 很容易陷入局部最优值; 其他 6 个测试函数为多峰函数.  $f_3$  为 Ackley 函数, 有一个很狭窄的全局最优凹陷区域, 在这个区域的周围分布着许多深度较浅的局部最优点;  $f_4$  为 Griewank 函数, 其中  $\prod_{i=1}^D \cos(x_i/\sqrt{i})$  部分使得各变量之间相互关联, 这种关联使得优化算法很难搜索到其全局最优解;  $f_5$  为 Weierstrass 函数, 虽然是连续

的, 但只在一部分点上可微的;  $f_6$  为 Rastrigin 函数, 是一个包含大量局部最优解的复杂多峰问题, 在求解这个函数时, 算法往往很容易陷入一个局部最优位置, 因此能有效保持种群多样性的群体智能优化算法可求得更好的解;  $f_7$  为 Noncontinuous Rastrigin 函数, 是基于 Rastrigin 函数构建的, 因此它和 Rastrigin 函数拥有同样数目的局部最优解;  $f_8$  为 Schwefel 函数, 其复杂性取决于远离全局最优解的深度局部最优解, 如果许多粒子陷入某个深度的局部最优解, 会使整个优化算法很难搜索到其全局最优解. 在这 8 个测试函数中, 除了  $f_2$  的全局最优位置为  $[1, 1, \dots, 1]$ ,  $f_8$  的全局最优位置为  $[420.96, 420.96, \dots, 420.96]$ , 其余函数的全局最优位置均为  $[0, 0, \dots, 0]$ , 在其各自的全局最优位置上对应的全局最优值为  $f(X^*) = 0$ .

表 1 实验采用的测试函数及其搜索范围和初始化范围

名称	表达式	搜索范围	初始化范围
Sphere	$f_1(X) = \sum_{i=1}^D x_i^2$	$[-100, 100]^D$	$[-100, 50]^D$
Rosenbrock	$f_2(X) = \sum_{i=1}^{D-1} (100(x_{i+1} - x_i^2)^2 + (x_i - 1)^2)$	$[-2.048, 2.048]^D$	$[-2.048, 2.048]^D$
Ackley	$f_3(X) = e + 20 - 20 \exp\left(-0.2 \sqrt{\frac{1}{D} \sum_{i=1}^D x_i^2}\right) - \exp\left(\frac{1}{D} \sum_{i=1}^D \cos(2\pi x_i)\right)$	$[-32.768, 32.768]^D$	$[-32.768, 16]^D$
Griewank	$f_4(X) = \frac{1}{4000} \sum_{i=1}^D x_i^2 - \prod_{i=1}^D \cos\left(\frac{x_i}{\sqrt{i}}\right) + 1$	$[-600, 600]^D$	$[-600, 200]^D$
Weierstrass	$f_5(X) = \sum_{i=1}^D \left( \sum_{k=0}^{k_{\max}} [a^k \cos(2\pi b^k (x_i + 0.5))] \right) - D \sum_{k=0}^{k_{\max}} [a^k \cos(2\pi b^k \cdot 0.5)]$ $a = 0.5, b = 3, k_{\max} = 20$	$[-0.5, 0.5]^D$	$[-0.5, 0.2]^D$
Rastrigin	$f_6(X) = \sum_{i=1}^D (x_i^2 - 10 \cos(2\pi x_i) + 10)$	$[-5.12, 5.12]^D$	$[-5.12, 2]^D$
Noncontinuous Rastrigin	$f_7(X) = \sum_{i=1}^D (y_i^2 - 10 \cos(2\pi y_i) + 10)$ $y_i = \begin{cases} x_i, & x_i < 0.5 \\ \frac{\text{round}(2x_i)}{2}, & x_i \geq 0.5 \end{cases}$	$[-5.12, 5.12]^D$	$[-5.12, 2]^D$
Schwefel	$f_8(X) = 418.9829D - \sum_{i=1}^D x_i \sin(x_i^{\frac{1}{2}})$	$[-500, 500]^D$	$[-500, 500]^D$

表 2 10 维函数测试结果

算法	$f_1$	$f_2$	$f_3$	$f_4$	$f_5$	$f_6$	$f_7$	$f_8$
SPSO	3.694e-124	1.223e+000	2.594e-015	6.823e-002	0.000e+000	1.811e+000	5.600e-001	8.314e+002
QPSO	<b>4.626e-295</b>	7.055e-001	<b>2.380e-015</b>	3.674e-002	0.000e+000	1.217e+000	2.522e+000	6.410e+002
CLQPSO	1.397e-078	<b>2.867e-001</b>	2.665e-015	<b>0.000e+000</b>	<b>0.000e+000</b>	<b>0.000e+000</b>	<b>0.000e+000</b>	<b>7.106e+000</b>

表 3 20 维函数测试结果

算法	$f_1$	$f_2$	$f_3$	$f_4$	$f_5$	$f_6$	$f_7$	$f_8$
SPSO	5.543e-052	1.180e+001	6.075e-015	3.196e-002	2.453e-005	1.180e+001	4.040e+000	2.229e+003
QPSO	<b>4.913e-109</b>	9.912e+000	<b>4.228e-015</b>	2.241e-002	0.000e+000	7.796e+000	1.332e+001	1.752e+003
CLQPSO	1.849e-034	<b>7.776e+000</b>	6.217e-015	<b>0.000e+000</b>	<b>0.000e+000</b>	<b>7.960e-002</b>	<b>0.000e+000</b>	<b>9.475e+000</b>

表 4 30 维函数测试结果

算法	$f_1$	$f_2$	$f_3$	$f_4$	$f_5$	$f_6$	$f_7$	$f_8$
SPSO	1.513e-028	2.652e+001	9.557e-015	1.732e-002	1.124e-002	3.002e+001	1.913e+001	3.671e+003
QPSO	<b>6.321e-060</b>	1.710e+001	<b>6.146e-015</b>	9.901e-003	4.441e-018	1.982e+001	2.872e+001	3.004e+003
CLQPSO	1.625e-021	<b>1.587e+001</b>	6.217e-015	<b>0.000e+000</b>	<b>0.000e+000</b>	<b>2.189e-001</b>	<b>0.000e+000</b>	<b>5.457e+001</b>

## 4.2 实验环境及算法参数设置

实验将 CLQPSO 算法与 PSO 算法和 QPSO 算法进行比较来验证 CLQPSO 算法的优越性. 各算法的参数设置描述如下: 标准粒子群优化算法 (SPSO)<sup>[11]</sup>中, 设置加速因子  $c_1 = c_2 = 2$ , 惯性权重  $\omega$  随着迭代次数由 0.9 线性减小到 0.4; 量子行为粒子群优化算法 (QPSO) 中, 收缩扩张因子  $\alpha$  参考文献 [4], 设置为随着迭代次数, 由 1.0 线性减小到 0.5; 基于完全学习策略的 QPSO 优化 (CLQPSO) 算法中, 种群各粒子的学习概率  $p_c$  由式 (8) 决定,  $\alpha$  的设置与 QPSO 相同. 此外, 3 种算法的种群粒子数均设为 30, 最大迭代次数设置为 5000. 实验环境为 Matlab 7.0.4, Intel Pentium(R) 4, 2.93 GHz, 1 GB RAM. 每种算法对每个函数在维数分别取 10, 20 和 30 的情况下独立运行 50 次.

## 4.3 实验结果及讨论

表 2~表 4 显示了 3 种算法对各测试函数在维数分别为 10, 20 和 30 的情况下进行优化求解所得到的平均最好函数值. 从表 2~表 4 中的数据可以看出, 对于单峰函数  $f_1$ , 由于函数形式比较简单, 函数最优值所在的区域比较平坦, 不需要优化算法对解空间进行更大范围的探索. 因此对于粒子搜索空间较大的 CLQPSO 算法而言, 反而增加了寻优过程的复杂性, 降低了算法收敛精度, 使得在算法迭代结束时得到的结果不如 SPSO 和 QPSO 算法. 对于函数  $f_2$  而言, 虽然也是单峰问题, 但由于函数最优值附近有陡峭的区域, 使得算法极易陷入局部最优, 增加了算法的搜索难度, 此时 CLQPSO 算法的进化策略便能有效增加

粒子的多样性, 使得算法得到更好的优化结果. 对于多峰函数而言, 除了  $f_3$ , 3 种算法所得的优化结果相当, QPSO 算法略好之外, 其他函数 CLQPSO 算法都获得了比 SPSO 算法和 QPSO 算法明显更优的结果, 并且随着函数维数的增加, 复杂性的提高, CLQPSO 的全局收敛能力更加突出, 这充分说明 CLQPSO 算法更适用于求解复杂的多峰问题. 图 3 为各算法优化求解  $f_3 \sim f_8$  函数时最优函数值随算法迭代步骤变化的收敛曲线. 将图中 3 种算法的收敛曲线进行比较, 很明显可以看出, 对于多峰函数的优化问题, CLQPSO 算法相比其他两种算法, 具有较快的收敛速度, 能在较少的迭代步骤内, 收敛到和其他两种算法相当或更好的最优函数值. 随着算法进入迭代的后期, 更能凭借其突出的全局搜索能力, 使算法在有限的迭代步骤内达到较高的收敛精度.

## 5 结 论

为了改善 QPSO 算法存在的粒子早熟问题, 从算法中粒子局部吸引子的更新方式入手, 提出了一种基于完全学习策略的 QPSO 算法. 在该算法中, 种群中所有粒子的  $p_{best}$  都有可能以一定的概率参与局部吸引子的更新过程, 充分利用了种群中的社会信息. 由新的学习策略产生的局部吸引子对 QPSO 算法的参数  $L$  进行评价也能有效提高算法的创造能力, 增大了算法潜在的搜索空间, 使算法在迭代后期更容易跳出局部最优解并进行更深入的搜索. 数值实验及分析表明, CLQPSO 算法对各测试函数具有较好的寻优结果, 尤其适用于对多峰函数的优化求解, 在保证收敛精度的同时, 也具有较快的收敛速度, 是一种很有潜力和应用前景的全局优化算法.

## 参考文献(References)

- [1] Kennedy J, Eberhart R. Particle swarm optimization[C]. Proc of IEEE Int Conf on Neural Networks. Perth: IEEE, 1995: 1942-1948.
- [2] 倪庆剑, 刑汉承, 张志政, 等. 粒子群优化算法研究进展[J]. 模式识别与人工智能, 2007, 20(3): 349-357.  
(Ni Q J, Xing H C, Zhang Z Z, et al. Survey of particle swarm optimization algorithm[J]. Pattern Recognition and Artificial Intelligence, 2007, 20(3): 349-357.)
- [3] Bergh FVD. An analysis of particle swarm optimizers[D]. Pretoria: University of Pretoria, 2001.
- [4] Sun J, Feng B, Xu W B. Particle swarm optimization with particles having quantum behavior[C]. Proc of 2004 Congress on Evolutionary Computation. Piscataway, 2004: 325-331.
- [5] Fang W, Sun J, Ding Y R, et al. A review of quantum-behaved particle swarm optimization[J]. IETE Technical Review, 2010, 27(4): 336-348.

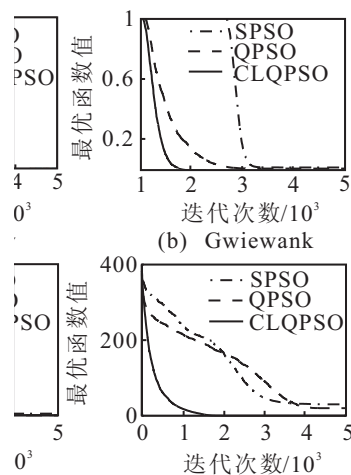


图 3  $D = 30$  时, 各算法在  $f_3 \sim f_8$  上的收敛曲线