

## 多目标量子编码遗传算法

邹 谊 魏文龙 李 斌 肖金超 庄镇泉  
(中国科学技术大学自然计算与应用实验室 合肥 230027)

**摘 要:** 如何使算法快速收敛到真正的 Pareto 前沿, 并保持解集在前沿分布的均匀性是多目标优化算法重点研究解决的问题。该文提出一种基于量子遗传算法的多目标优化算法, 利用量子遗传算法的高效全局搜索能力, 在整个解空间内快速搜索多目标函数的 Pareto 最优解, 利用量子遗传算法维持解集多样性的特点, 使搜索到的 Pareto 最优解在前沿均匀分布。通过求解带约束的多目标函数优化问题, 对该文算法的多目标优化性能进行了考察, 并与 NSGAI, PAES, MOPSO 和 Ray-Tai-Seow's 算法等知名多目标优化算法进行比较, 结果证明了该文算法的有效性和先进性。

**关键词:** 量子遗传算法; 多目标优化; Pareto 最优解

**中图分类号:** TP393

**文献标识码:** A

**文章编号:** 1009-5896(2007)11-2688-05

## A Multi-objective Q-bit Coding Genetic Algorithm

Zou Yi Wei Wen-long Li Bin Xiao Jin-chao Zhuang Zhen-quan  
(Nature Inspired Computation and Applications Laboratory, USTC, Hefei 230027, China)

**Abstract:** How causes the algorithm fast to restrain to the true Pareto optimal front, and maintains solutions distributed uniformly in the Pareto optimal front is one of the key research issue. A multi-objective optimization algorithm is proposed based on Q-bit Coding Genetic Algorithm (QCGA). By right of the capability of efficient global search and maintenance of diversity of QGA, it explores the feasible region for Pareto optimal solutions quickly and maintains the solutions distributed uniformly over the Pareto optimal front. Characteristics of the algorithm are confirmed through optimization experiments of multi-objective functions with constraints. Compared with several well-known algorithms such as NSGAI, PAES, MOPSO, experiment results prove the algorithm validity and efficiency.

**Key words:** Q-bit coding genetic algorithm; Multi-objective optimization; Pareto optimal

### 1 引言

现实世界中的很多问题, 包括很多实际的工程问题, 通常由相互冲突的多个目标组成。从理论上说, 多目标问题在整个解空间内不存在唯一的全局最优解, 而是一组相互之间无法区分优劣的均衡解, 即所谓的 Pareto 最优解。因此求解多目标优化问题的关键是求出尽可能多的、在前沿均匀分布的 Pareto 最优解。

求解多目标优化问题, 早期的方法是将各子目标聚合成一个带权系数的单目标函数<sup>[1]</sup>。这种方法存在较大的局限性, 如求解问题所需的, 与应用背景相关的启发式知识常常不能获得, 导致无法合理分配权重。80 年代中期, 进化算法(EA)开始应用于求解多目标优化问题<sup>[2,3]</sup>。90 年代前期几种多目标进化算法(MOEAs)被提出来<sup>[4-6]</sup>。MOEA 的优点在于可以不依赖问题的先验知识, 应用范围广, 可处理大规模的搜索空间, 在单轮优化期间产生多个均衡解, 可有效克服早期方

法的局限性。其中, 代表性算法有 Schaffer 的 VEGA<sup>[2]</sup>, 它是一种并列选择算法, 把全部个体均分为若干子配对库, 并分配一个子目标函数, 每个子库独立演化, 然后再混合。这种算法易产生个别子目标函数的极端最优解, 使算法过早收敛。EAs 在使用有限群体时, 易收敛于单个解<sup>[7]</sup>, 解决此问题的关键在于维持群体的多样性。1999 年 Knowles 等提出 PAES 算法<sup>[8]</sup>, 引入精英保留策略以提高收敛性, 只基于变异产生新个体, 虽然算法简单但收敛速度有待提高。针对约束处理问题, Ray 等的 Ray-Tai-Seow's 算法<sup>[9]</sup>按个体的目标函数值、约束违反值、目标函数值加约束违反值来评价个体, 取得较好的效果, 但是计算开销大, 不能有效处理复杂问题。2002 年, Deb 等提出 NSGA-II 算法<sup>[10]</sup>, 采用分层聚类策略维持解集多样性, 效果较好, 但算法较复杂, 且通用性不高。另外, Coello 等提出一种启多目标粒子群算法 MOPSO<sup>[11]</sup>, 有较快的收敛性, 但算法只能处理连续域搜索问题, 目前还不能应用到离散区域搜索问题。

2000 年, Han 等提出遗传量子算法<sup>[12]</sup>(Genetic Quantum Algorithm), 算法利用了量子的态矢量表述和量子旋转门操作, 但算法设计的通用性不强, 效率有待提高。李斌等提出

2006-04-10 收到, 2006-11-07 改回

国家自然科学基金(60401015, 60572012)和安徽省自然科学基金(050420201)资助课题

了量子编码遗传算法, 采用通用的量子门旋转操作方法和量子交叉算子<sup>[13]</sup>, 对单目标优化问题求解有很好的效果, 但算法的设计没有扩展到多目标。量子遗传算法在求解单目标问题上的成功<sup>[14]</sup>, 使我们有信心将其应用到多目标优化问题。本文提出了一个多目标量子编码遗传算法, 在以上量子编码遗传算法的基础上, 增加了pareto评优、解集过滤器、和约束处理机制, 改进了量子交叉操作。通过求解带约束的多目标函数优化问题, 对本文算法的多目标优化性能进行了考察, 并与NSGAI, PAES, MOPSO和Ray-Tai-Seow's算法等知名多目标优化算法进行比较, 结果证明了本文算法的有效性和先进性。

本文后续部分安排如下: 第2节介绍多目标量子遗传算法, 第3节给出实验结果和讨论, 最后总结全文。

## 2 多目标量子编码遗传算法

### 2.1 量子比特编码和初始化

在量子计算机中, 充当信息存储单元的物理介质是一个双态量子系统, 称为量子比特(qubit)。与经典比特不同, 量子比特可同时存储和表达“0”和“1”两种状态, 可表示为  $\varphi = \alpha|0\rangle + \beta|1\rangle$ , 其中  $(\alpha, \beta)$  是两个复常数, 称为量子比特的概率幅, 满足  $|\alpha|^2 + |\beta|^2 = 1$ ;  $|0\rangle$  和  $|1\rangle$  分别表示“0”和“1”两种状态。

采用量子比特幅编码的染色体结构如下:

$$q_j^t = \begin{pmatrix} \alpha_{11}^t | \alpha_{12}^t | \dots | \alpha_{1k}^t | \alpha_{21}^t | \alpha_{22}^t | \dots | \alpha_{2k}^t | \dots | \alpha_{m1}^t | \alpha_{m2}^t | \dots | \alpha_{mk}^t \\ \beta_{11}^t | \beta_{12}^t | \dots | \beta_{1k}^t | \beta_{21}^t | \beta_{22}^t | \dots | \beta_{2k}^t | \dots | \beta_{m1}^t | \beta_{m2}^t | \dots | \beta_{mk}^t \end{pmatrix} \quad (1)$$

$q_j^t$  代表第  $t$  代第  $j$  个个体的染色体,  $m$  为染色体基因个数, 对应于函数的变量个数,  $k$  为编码每个基因的量子比特数。初始化时, 令所有基因位的概率幅  $(\alpha_i, \beta_i)$  都为  $(1/\sqrt{2}, 1/\sqrt{2})$ , 这意味着在整个解空间中所以可能解的取值概率相同。对一个用概率幅编码的染色体进行“测量”(采样), 可获得一个确定解。某一位在某一次“测量”中取“0”还是取“1”的概率由其相应概率幅大小决定。个体进化过程中, 概率幅的改变由量子旋转、交叉操作实现。

### 2.2 染色体更新

对染色体的更新通过量子旋转门实现, 其工作原理如下:

$$\begin{pmatrix} \alpha'_i \\ \beta'_i \end{pmatrix} = \begin{pmatrix} \cos(\theta_i) & -\sin(\theta_i) \\ \sin(\theta_i) & \cos(\theta_i) \end{pmatrix} \begin{pmatrix} \alpha_i \\ \beta_i \end{pmatrix} \quad (2)$$

其中  $(\alpha_i, \beta_i)$  为染色体中第  $i$  个量子比特,  $\theta_i = s(\alpha_i, \beta_i)\Delta\theta_i$ ,  $s(\alpha_i, \beta_i)$  和  $\Delta\theta_i$  的值根据某一策略事先确定, 以表格的形式提供查询。本文采用一个通用的调整策略, 如下式所示<sup>[13]</sup>。

$$\theta_i = \text{sign}((f(x) - f(b)) \times (x_i - b_i) \times \alpha_i \times \beta_i) \times \Delta\theta \quad (3)$$

其中,  $\Delta\theta$  为每次调整的角步长, 对正常调整和交叉调整可以取不同的步长, 以达到控制演化速度和交叉作用影响力的目的。该调整策略的思路是: 将个体  $q_j^t$  当前测量所得解  $X$  与该

个体当前的目标  $B$  进行比较, 若  $X$  支配  $B$ , 则调整  $q_j^t$  中相应位量子比特  $(x_i \neq b_j)$ , 使得几率幅  $(\alpha_i, \beta_i)$  向着有利于  $x_i$  出现的方向演化; 若  $B$  支配  $X$ , 使几率幅  $(\alpha_i, \beta_i)$  向着有利于  $b_i$  出现的方向演化; 否则, 随机选择一个, 使几率幅  $(\alpha_i, \beta_i)$  向着有利于那个随机选择的解出现的方向演化。

### 2.3 设置 Pareto 解集过滤器

在种群外部设置Pareto解集过滤器(Archive<sup>[8]</sup>), 用来存放进化时产生的Pareto最优解。对于不违反约束的解:

(1) 如果该解支配过滤器中某个解, 则把该解加入到过滤器中, 同时剔除那个被支配解;

(2) 如果过滤器中某个解支配该解, 则不对过滤器进行操作;

(3) 否则, 则把该解加入到过滤器中。

过滤器的大小可以事先设定。在向其中加入解时, 同时进行“划分”操作。本文采用线性划分法, 即对每个子目标, 在过滤器的所有解中找出最大和最小值, 作为子空间一个维度的边界, 在上界和下界之间均匀地划分  $n$  等份。假设优化目标个数为  $m$  个, 其中某个解在每个维度上的坐标为:  $(x_1, x_2, \dots, x_m)$ , 则它所属子空间的编号为  $x_1 \cdot n^0 + x_2 \cdot n^1 + \dots + x_m \cdot n^{m-1}$ 。

当过滤器中的解的数目超出预定大小时, 选择一个包含解数目最多的子空间, 从中随机剔除一个解, 目的是维持解的多样性, 使解尽可能均匀分布。

### 2.4 量子交叉

通过引入量子交叉操作, 在个体间实现信息交换, 以进一步提高搜索效率。实现步骤:

(1) 按照事先设定的交叉概率  $p_c$ , 从种群中随机选取若干对个体  $q_{Ai}$  和  $q_{Bi}$ ;

(2) 对它们分别进行一次测量, 得到一组确定解  $q_{Ai}$  和  $q_{Bi}$ ;

(3) 以个体  $q_{Bi}$  演化目标作为个体  $q_{Ai}$  当前演化目标, 对个体  $q_{Ai}$  实施量子旋转; 以个体  $q_{Ai}$  演化目标作为个体  $q_{Bi}$  当前演化目标, 对个体  $q_{Bi}$  实施量子旋转。

本文算法中, 交叉包括种群中个体与 Pareto 解集过滤器中解的交叉。若过滤器中有解, 以过滤器每个子空间中的解的数目多少为适应度, 按赌轮选择选出若干个解, 与种群中的个体配对进行量子交叉。实施这种交叉操作, 目的是为了向这些解进化, 加快求解速率。

### 2.5 约束处理

本文把个体对约束的违反看作该个体的一个“属性”, 用约束违反度  $R^{[10]}$  来衡量该“属性”值。设有约束  $g_1(x) \leq 0, g_2(x) \leq 0, \dots, g_n(x) \leq 0$ , 对个体  $A$ , 若  $g_i(A) \leq 0$ , 则  $R_i = 0$ ; 若  $g_i(A) > 0$ , 则  $R_i = g_i(A)$ 。个体  $A$  的约束违反度  $R = \sum R_i$  ( $i=1, 2, \dots, n$ )。显然, 不违反约束的个体  $R$  值最小, 为零。

结合个体约束违反度  $R$ , 个体间的支配关系为: 不违反约束的个体支配违反约束的个体, 约束违反度小的个体支配

约束违反度大的个体。用约束违反度  $R$  来处理约束, 通过比较  $R$  值, 可以比较两个违反约束个体的支配关系, 在进化中向合理解收敛。

2.6 算法流程

为了充分利用量子计算的多态叠加和并行搜索特性, 本文对量子遗传算法中各个个体各自独立演化, 每个个体都拥有自己独立的演化目标。同时个体间通过量子交叉实现演化信息的交换。算法的主要流程如下:

- (1)初始化种群  $Q(t_0)$ , 产生  $n$  个以量子比特编码的染色体;
- (2)对初始种群中的每个个体  $q_j^0 (j = 1, 2, \dots, n)$  实施  $K$  次测量, 产生  $K$  个确定解。个体测量产生的  $K$  个确定解中, 在不违反约束的解中, 随机选取一个 Pareto 最优解作为该个体的当前目标, 如果产生的  $K$  个解都违反约束, 则选取约束违反度最小的解作为该个体的当前目标。
- (3)While 非结束状态 do:
- (4)Begin
- (a)  $t = t + 1$ ;
- (b)对种群  $Q(t-1)$  中每一个个体  $q_j^{t-1} (j = 1, 2, \dots, n)$  实施一次测量, 得到一个确定解;
- (c)对每个个体的确定解, 与该个体的当前目标进行比较。利用旋转门实施对个体  $q_j^{t-1} (j = 1, 2, \dots, n)$  的染色体更新, 得到新个体  $q_j^t$ ;
- (d)实施量子交叉操作。
- (e)如达到一定演化代数, 实施量子移民策略;
- (5)END

对每次测量产生的确定解, 若测量产生的确定解支配该个体当前目标, 则选择该解为作为下一步演化的目标; 若该个体当前目标支配测量产生的确定解, 保持当前目标不变;

否则, 随机选择一个作为下一步演化的目标。

在第 1 步中, 我们认为量子编码策略已暗含了个体的多样性, 故种群规模  $n$  不需太大。第 2 步对初始个体实施  $K$  次测量的目的是为各个个体提供一个尽可能好的初始演化目标,  $K$  的取值视具体情况而定, 但不宜过大, 以免增加计算量, 可取 1~10 以内的自然数。

3 实验结果及分析

为了定量地比较, 引入衡量算法所得解与真实 Pareto 前沿间差距的参数  $M_1^*$  [10]:

$$M_1^* = \frac{1}{|Y'|} \sum_{d' \in Y'} \min\{\|d' - \bar{d}\|^* ; \bar{d} \in \bar{Y}\}$$

其中  $Y'$  为算法得到的解集,  $d'$  为  $Y'$  中的解;  $\bar{Y}$  为真实的 Pareto 解集, 可以用枚举法得到,  $\bar{d}$  为  $\bar{Y}$  中的解。

首先采用文献[11]中引用的函数实验(表 1)。3 个函数优化都是使目标函数最小化。实验采用标准 C 语言在 Pentium 2.0G, 256M 内存, windows2000 环境下实现。3 个函数的实验参数取一样: 个体数  $n$  取 60, 每个变量编码位数取 12, 初始测量次数  $K$  取 8, 旋转调整角步长取  $0.08\pi$ , 交叉调整角步长取  $0.04\pi$ , 移民代数取 200, 交叉概率取 0.2, 迭代终止次数取 600; 只是移民概率不一样, 分别取: 0.2, 0.5, 0.5。运行 20 次, 与文献[11]中的实验对比, 结果见表 2。

从实验结果可以看出, 本文的量子遗传算法求解质量上优于其他 3 种算法, 并且具有更好的鲁棒性。

再引用文献[10]中的 3 个函数实验(表 3)。

3 个函数优化都是使目标函数最小化。量子遗传算法的实验参数和前面的实验一样。由于文献[8]中没有给出定量比较, 与文献[10]中实验结果对比, 定性比较如图 1-图 3:

表 1 测试函数

函数序号	变量范围	目标函数
1	$x_i \in [0, 1]$	$f_1(x) = x_1$
		$f_2(x) = (1 + 10x_2) \times \left[ 1 - \left( \frac{x_1}{1 + 10x_2} \right)^2 - \frac{x_1}{1 + 10x_2} \sin(8\pi x_1) \right]$
2	$x \in [-5, 10]$	$f_1(x) = \begin{cases} -x, & x \leq 1 \\ -2 + x, & 1 < x \leq 3 \\ 4 - x, & 3 < x \leq 4 \\ -4 + x, & x > 4 \end{cases}$
		$f_2(x) = (x - 5)^2$
3	$x_1 \in [0, 1]$ $x_2 \in [-30, 30]$	$f_1(x) = x_1$
		$f_2(x) = g(x)h(x)$
		$g(x) = 11 + x_2^2 - 10 \cos(2\pi x_2)$
		$h(x) = \begin{cases} 1 - \sqrt{\frac{f_1(x)}{g(x)}}, & f_1(x) \leq g(x) \\ 0, & \text{其他} \end{cases}$

表 2 比较实验结果

函数序号	算法	$M_1^*$ 均值	$M_1^*$ 的标准差
1	NSGA-II	0.002536	0.000138
	PAES	0.002881	0.00213
	MOPSO	0.002057	0.000286
	本文	0.001851	0.000080
2	NSGA-II	0.001594	0.000122
	PAES	0.070003	0.158081
	MOPSO	0.00147396	0.00020178
	本文	0.000348	0.000009
3	NSGA-II	0.094644	0.117608
	PAES	0.259664	0.573286
	MOPSO	0.0011611	0.0007205
	本文	0.000583	0.000030

表 3 测试函数

测试函数	变量范围	目标函数	约束条件
CONSTR	$x_1 \in [0.1, 1.0]$	$f_1(x) = x_1$	$g_1(x) = x_2 + 9x_1 \geq 6$
	$x_2 \in [0, 5]$	$f_2(x) = (1 + x_2) / x_1$	$g_2(x) = -x_2 + 9x_1 \geq 1$
SRN	$x_i \in [-20, 20]$	$f_1(x) = (x_1 - 2)^2 + (x_2 - 1)^2 + 2$	$g_1(x) = x_1^2 + x_2^2 \leq 225$
	$i = 1, 2$	$f_2(x) = 9x_1 - (x_2 - 1)^2$	$g_2(x) = x_1 - 3x_2 \leq -10$
TNK	$x_i \in [0, \pi]$	$f_1(x) = x_1$	$g_1(x) = -x_1^2 - x_2^2 + 1 + 0.1 \cos(16 \arctan(x_1 / x_2)) \leq 0$
	$i = 1, 2$	$f_2(x) = x_2$	$g_2(x) = (x_1 - 0.5)^2 + (x_2 - 0.5)^2 \leq 0.5$

图中, 阴影部分为采用穷举法得到的满足约束的所有解, 圆圈表示算法得到的非支配解。由图可见, 对连续 Pareto 前沿(图 1 函数 CONSTR, 图 2 函数 SNR)和非连续的 Pareto 前沿(图 3 函数 TNK), 本文的算法和 NSGA-II 算法一样, 都能使解有效地收敛到整个真正的 Pareto 前沿, 明显优于 Ray-Tai-Seow's 算法。

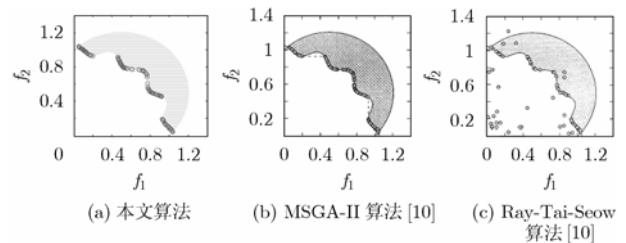


图 3 TNK 函数实验结果

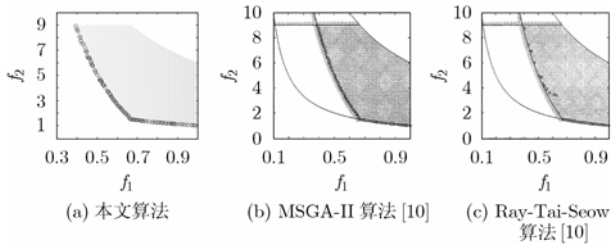


图 1 本文算法测试 CONSTR 实验结果

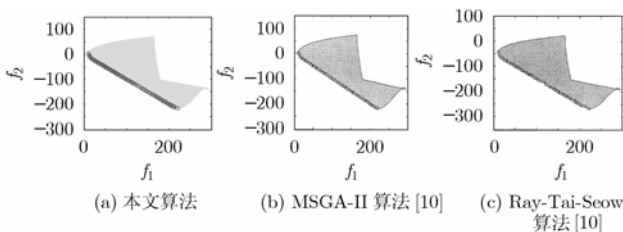


图 2 SNR 函数实验结果

### 4 结束语

量子概率编码的表达多样性, 使得量子遗传算法很适合于求解多目标问题。本文将量子遗传算法应用于带约束的多目标函数优化, 通过在外部设立解集过滤器, 同时通过对解集过滤器的操作, 以及引入移民机制, 来维护群体的多样性。本文还提出一个对约束(Constraint)的处理方法, 使其能有效处理带约束的函数优化问题。实验结果表明该算法在求解效率和维持群体多样性方面的有效性。今后的工作还有: 分析算法运行的参数对实验结果的影响, 以及如何提高算法的收敛速度, 以提高算法的效率。

### 参考文献

[1] Koski J. Multi-criterion optimization in structural design. In Attek E, Gallagher R H, and Ragsdell K M, *et al.* Ed. *New Directions in Optimum Structural Design*. New York, Wiley,

- 1984: 483-503.
- [2] Schaffer J D. Multiple Objective Optimization with Vector Evaluated Genetic Algorithms. In: Proceedings of the 1st International Conference on Genetic Algorithms, Lawrence Erlbaum Associates, Hillsdale, 1985: 93-100.
- [3] Fourman M P. Compaction of symbolic layout using genetic algorithms. In Grefenstette J J(Ed.), Proceedings of an International Conference on Genetic Algorithms and Their Applications. Pittsburgh, PA, 1985: 141-153.
- [4] Kursawe F. A variant of evolution strategies for vector optimization. In: Schwefel H P and Männer R(Ed.). Parallel Problem Solving from Nature Proceedings of the first Workshop PPSN, Berlin, Springer, 1991: 193-197.
- [5] Fonseca C M and Fleming P J. Genetic algorithms for multi-objective optimization: formulation, discussion and generalization. Proceedings of the 5th international conference on genetic algorithms, Forrest Ed, San Mateo, CA: Morgan Kaufmann Publishers, 1993: 416-423.
- [6] Srinivas N and Deb Kalyanmoy. Multiobjective optimization using non-dominated sorting in Genetic algorithms. *Evolutionary Computation*, 1994, 2(3): 221-248.
- [7] Deb K and Goldberg D E. An investigation of niche and species formation in genetic function optimization. In: Schaffer J D(Ed.), Proceedings of the 3rd International Conference on Genetic Algorithms, George Mason University, Fairfax, VA, USA. 1989: 42-50.
- [8] Joshua D K and David W C. Approximating the nondominated front using the Pareto archived evolution strategy. *Evolutionary Computation*, 2000, 8(2): 149-172.
- [9] Ray T, TRai K, and Seow K C. An evolutionary algorithm for multiobjective optimization. *Eng. Optim*, 2001, 33(3): 399-424.
- [10] Deb K, Pratap A, Agarwal S, and Meyarivan T. A fast and elitist multi- objective genetic algorithm: NSGA-II. *IEEE Trans. on Evolutionary Computation*, 2002, 6(2): 182-196.
- [11] Carlos A Coello Coello and Maximino Salazar Lechuga. MOPSO: A proposal for multiobjective particle swarm optimization. Proceedings of the 2002 Congress on Evolutionary Computation, Hawaii, USA, 12-17 May 2002. vol.2: 1051-1056.
- [12] Han Kuk-Hyun and Kim Jong-Hwan. Genetic quantum algorithm and its application to combinatorial optimization problem[A]. Proceeding of the 2000 IEEE Congress on Evolutionary Computation [C]. San, Diego, 2000, 2: 1354-1360.
- [13] Li Bin, *et al.* Genetic algorithm based on the quantum probability representation[R]. Yin H, *et al.* (Ed.). Lecture Notes in Computer Science (LNCS2412), 2002: 500-505.
- [14] 李斌, 庄镇泉等. 量子概率编码遗传算法及其应用. 电子与信息学报, 2005, 27(5): 808-810.
- Li Bin and Zhuang Zhen-quan, *et al.* Quantum probability coding genetic algorithm and its applications. *Journal of Electronics. & Information Technology*, 2005, 27(5): 805-810.
- 邹 谊: 男, 1971年生, 博士, 专业方向为软硬件协同设计、量子计算.
- 魏文龙: 男, 1982年生, 硕士, 专业方向为软硬件协同设计、遗传算法.
- 李 斌: 男, 1970年生, 博士, 副教授, 专业方向为遗传算法、量子计算.
- 肖金超: 男, 1982年生, 本科生, 专业方向为遗传算法、进化计算.
- 庄镇泉: 男, 1938年生, 教授, 博士生导师, 专业方向为电子设计自动化、智能信息处理.