

一种并行的自适应量子粒子群算法^①

熊智挺, 谭阳红, 易如方, 陈赛华

(湖南大学 电气与信息工程学院, 长沙 410082)

摘要: 针对粒子群算法存在易陷入局部最优解的问题, 提出了一种并行的自适应量子粒子群算法。通过共享粒子的两个极值, 将改进后的自适应粒子群算法和边界变异的量子粒子群算法并行搜索, 有效地克服了标准粒子群算法的缺陷。测试结果表明, 该算法在精度和全局最优解的找寻速度方面有了很大的提高。

关键词: 自适应粒子群算法; 量子粒子群算法; 并行搜索

Method of Parallel Adaptive Quantum Particle Swarm Optimization

XIONG Zhi-Ting, TAN Yang-Hong, YI Ru-Fang, CHEN Sai-Hua

(College of Electrical & Information Engineering, Hunan University, Changsha 410082, China)

Abstract: A new parallel adaptive quantum particle swarm optimization algorithm is proposed in this paper to solve the problem that standard particle swarm optimization(PSO) algorithm may easily trap into local optimal points and may obtain exact solutions at the late of the iteration with difficulty. By sharing the two extreme values of the particles, the proposed method is able to adaptively search their optimum solutions in parallel by combination of an improved adaptive PSO with a quantum Particle Swarm Optimization of boundary variation. It is proved effectively to overcome the shortcomings of standard PSO. Test results show that the accuracy and the velocity of global search for optimal solutions have been greatly improved.

Key words: adaptive PSO; quantum PSO; parallel search

1 引言

上世纪 90 年代, 粒子群算法、遗传算法、蚁群算法、免疫算法等智能算法^[1-5]都是通过模拟生物群体的行为来构造的随机算法, 形成了以群体智能 (swarm intelligence) 为核心的理论体系。尽管人们没有给出群体智能严格的数学证明, 但在函数优化、神经网络的训练、模式识别分类、模糊控制等科学和工程领域的广泛地应用, 已经证明了群体智能的有效性。作为群体智能的典型, 粒子群优化算法 (PSO, Particle Swarm Optimization) 最早于 95 年提出^[1], 是一个通过个体间协作来寻找最优解的算法。所有粒子都以随机速度在解空间移动, 在移动的同时, 粒子能记录下它所经过的最佳位置和共享群体内的最佳位置, 个体粒子通过彼此分享信息实现进化。与遗传算法通过合作和竞争来实现进化相比, 粒子群优化算法采用生物群体信息

共享策略, 使算法收敛速度特别快 (尤其是进化初期), 运算简单, 易于实现, 没有过多的参数调节, 但由于所有粒子的飞行方向是根据全体和自身经验决定, 进化后期收敛速度明显减慢, 达到一定程度后, 算法甚至陷入了停滞, 致使后期很难得到精确解。为了解决粒子群的局部最优解问题, 许多学者做了大量研究, 常用方法是改变粒子群中的惯性权重来提高粒子群的性能, 代表有: 自适应粒子群算法^[2], 或与其他群体智能算法结合, 如: 遗传粒子群算法^[4]、免疫粒子群算法^[5]、模拟退火粒子群算法^[6]、量子粒子群算法^[7]等。本文提出了一种并行的自适应量子粒子群算法 (A-QPSO, Parallel Adaptive Quantum Particle Swarm Optimization), 将改进的自适应粒子群算法 (APSO, Adaptive Particle Swarm Optimization) 和边界变异的量子粒子群算法 (QPSO, Quantum Particle Swarm

^① 基金项目: 国家自然科学基金(60876022, 50925727); 高新技术与发展基金(2006AA04A104); 湖南省自然科学基金(07JJ6132); 中央高校基本科研业务费
收稿时间: 2010-11-12; 收到修改稿时间: 2011-01-07

Optimization) 并行搜索。A-QPSO 用主线程来记录粒子的两个极值, 子线程与主线程相互通信来更新粒子的极值, 以实现子线程的极值共享。其中一个子线程改进自适应粒子群算法, 采用了惯性权值和学习因子均非线性可变, 既加强了前期的全局搜索能力, 也保证了后期局部搜索能力, 确保得到精确解。另一个子线程边界变异量子粒子群算法, 使用的是能在全部的解空间内寻解的非线性量子模型, 使算法摆脱有限的区域, 跳出局部最优解的能力得到增强, 并对边界粒子的处理方式改进, 保持了种群的多样性, 增加了全局搜索能力, 大大提高了算法的稳定性和收敛性。通过并行策略, 解决了 APSO 局部最优解和 QPSO 运算速度慢的问题, 结合了两个算法的优点。实验测试表明, APSO 比标准粒子群算法的性能有了很大的提高, 但跳出局部解的能力不是很强; QPSO 不易陷入局部最优解, 在后期却很难提高解的精度; 而并行的 A-QPSO 结合两种算法的优点, 高精度、快收敛、不易陷入局部解。

2 并行的自适应量子粒子群算法

为了克服标准 PSO 的不足, 这里采用异步并行计算的思想, 将改进自适应粒子群算法和边界变异量子粒子群算法并行运行, 用一个主线程来记载全局最优值和个体最优值, 每个子线程负责各自的计算, 并将计算中得到的两个极值传给主线程, 主线程只保存最优的两个极值, 各个子线程从主线程中调取这两个值, 并更新各个子线程粒子的速度和位置信息, 以延迟信息交换的功能和减少重叠信息交换, 从而减少运行时间, 算法流程如图 1:

Step1: 启动主线程, 初始化 pbest (个体最优值) 和 gbest (全局最优值), 开启 pbest 和 gbest;

Step2: 启动子线程, 运行 APSO 和 QPSO。初始化解的维数、种群数、最大进化代数以及 APSO 和 QPSO 中的参数。初始化种群的速度和位置;

Step3: 计算 pbest 和 gbest 值, 放入主线程中, 所有子线程共享主线程中的 pbest 和 gbest 值。子线程根据主线程中的信息更新位置和速度;

Step4: 所有子程序判断是否完成所达到周期值且其他子程序是否完成该次循环, 未达到周期且其他子程序已经完成循环则转 Step3, 若达到最大周期值, 输出结果, 终止程序。

APSO 和 QPSO 是子线程, 两者互不干扰、相互独立, 只对主线程中的 pbest 和 gbest 值调用和更新。APSO 和 QPSO 子程序可以同步运行, 也可以异步运行。本文采用异步运行机制, 一周期内主线程中的值更新次数由子程序数目决定, 更新次数为两次。算法如图 1 所示。其中两子线程分别为改进自适应粒子群 (APSO) 和改进量子粒子群 (QPSO)。

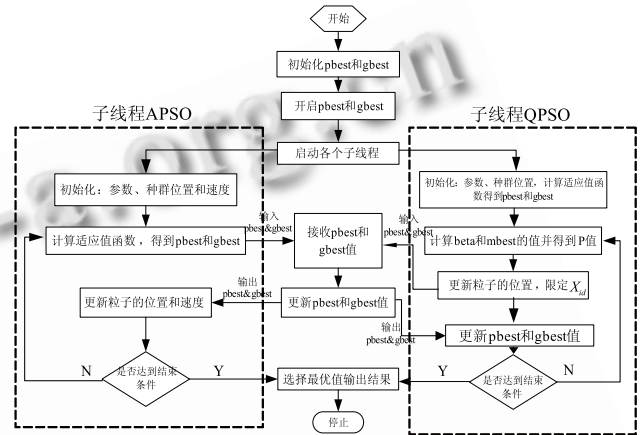


图 1 并行自适应量子粒子群算法流程

2.1 子线程: 改进自适应粒子群算法 (APSO, Adaptive Particle Swarm Optimization)

APSO 初始化后, 得到一群随机粒子 (初始解)。然后通过迭代和追踪两个极值 (pbest, gbest) 来更新自己。找到两个极值后, 粒子按下式 1、2 更新自己的速度和位置:

$$V_{i+1} = w * V_i + c_1 * rand() * (pbest - x_i) + c_2 * rand() * (gbest - x_i) \tag{1}$$

$$x_{i+1} = x_i + V_i \tag{2}$$

其中, V_i 是粒子的速度向量, x_i 是粒子的位置向量; pbest 是粒子本身发现的最好位置, gbest 是群体中所有粒子发现的最好位置; rand() 是介于 (0~1) 之间的随机数; 学习因子, 为惯性权重因子。式 (1) 第一部分称为记忆项, 表示上次速度大小和方向的影响; 第二部分为自身认知项, 表示粒子动作依赖于经验的程度; 第三部分称为群体认知项, 反映了粒子间的协同合作和知识共享。公式 (1) 中学习因子 c_1, c_2 分别调节 pbest 和 gbest 方向的飞行距离, 可以调节自身认知和群体认知的比例大小, 从而优化全局最优解的精度。然而 PSO 算法是一个非线性的复杂过程, 让惯性

权重因子采用线性的过渡方法不能反映实际的搜索过程,描述的不够精确。本文采用 Sigmoid 函数(S 函数)更新惯性权重,它比余弦函数更平滑,有很好的平衡性。设 G 为当前迭代次数, G_{\max} 是最大迭代次数,其更新式为:

$$w = (w_{\max} - w_{\min}) / \left(1 + e^{(20G/G_{\max} - 10)} \right) + w_{\min} \quad (3)$$

在标准 PSO 中, c_1 和 c_2 是一个固定的常数,而合理的调节 c_1 , c_2 参数的大小,能有效的抑制局部解,加强搜索精度。本文采用三次函数 $y = ax^3 + b$ 调节学习因子项:

$$c_1 = (c_{1f} - c_{1s}) * (G/G_{\max})^3 + c_{1s},$$

$$c_2 = (c_{2f} - c_{2s}) * (G/G_{\max})^3 + c_{2s} \quad (4)$$

c_{1s} 和 c_{1f} 为 c_1 参数的初始值和最终值, c_{2s} 、 c_{2f} 为 c_2 参数的初始值和最终值。搜索前期增大群体认知比例,防止陷入局部最优解,后期加强自身认知比例,能搜索到更精确解。

子线程改进的 APSO 算法流程:

Step1: 初始化每个微粒(群体规模为 SwarmSize)及相应的速度,创建初始解矩阵;

While(当前迭代次数<最大迭代次数&&其他子程序已完成)

Step2: 评价每个粒子的适应度(代入适应值函数 fit); 对每个微粒,将其适应值与当前个体粒子最好位置 pbest 比较,若较好,则将其当前适应值作为 pbest;

Step3: 从所有微粒的个体最好位置 pbest 找出全局最好位置 gbest,并与与主线程中的 pbest 和 gbest 比较,若好于主线程,则更新,反之用主线程中的值替代;

Step4: 按(1)(2)式更新粒子的位置速度和位置,按(3)(4)式更新 w 、 c_1 和 c_2 ;

End

2.2 子线程: 边界变异量子粒子群算法 (Quantum Particle Swarm Optimization)

量子粒子群算法 (Quantum Particle Swarm Optimization) 于 2004 年由 Sun 等人提出^[9],以 DELTA 势阱为基础,粒子可以在全部的量子空间内进行搜索,全局搜索能力很强。QPSO 算法利用波函数 $\varphi(x,t)$ 描述粒子状态,通过求解薛定谔方程得到粒子在空间某点出现的概率密度函数,用 Monte Carlo 随机模拟得到粒子位置方程:

$$X(t) = P \pm \ln(1/u) * L / 2 \quad (5)$$

u 为在 $[0,1]$ 上均匀分布的随机数, L 值由 $L(t+1) = 2\beta |mbest - X(t)|$ 确定, QPSO 算法的进化方程为:

$$mbest = 1/M * \sum_{i=1}^M pbest_i$$

$$= 1/M * \left(\sum_{i=1}^M pbest_{i1}, \sum_{i=1}^M pbest_{i2}, \dots, \sum_{i=1}^M pbest_{iDim} \right) \quad (6)$$

$$P = \varphi * pbest + (1 - \varphi) * gbest \quad (7)$$

$$X_{id}(t+1) = P \pm \beta * |mbest_{id} - X_{id}(t)| * \ln(1/u) \quad (8)$$

M 为种群中粒子的数目; Dim 为粒子的位数; φ 是在 $[0,1]$ 上均匀分布的随机数; $mbest$ 是种群的平均最好位置点,(6)式中的为每一维粒子所经历的最好位置;

(7)式中 $pbest_{iDim}$ 的 pbest 和 gbest 与 PSO 一样,表示粒子所经历的最好位置和种群的最好位置。称为搜索扩张系数,是 QPSO 中的唯一参数,本文中 $\beta = (1 - 0.5) * (G_{\max} - G) / G_{\max} + 0.5$ 。

在 QPSO 算法中,当粒子 X_{id} 飞越出搜索边界 X_{\max} 后,通常将边界位置赋予该粒子:

$$\text{If } X_{id} > X_{\max} \quad X_{id} = X_{\max} \quad \text{或者}$$

$$\text{If } X_{id} < -X_{\max} \quad X_{id} = -X_{\max} \quad (9)$$

采用这样的边界设置后,所有超过边界的粒子都强制限制在边界处,一旦在边界有局部最优解,粒子将很容易陷入该局部最优点,随着边界处粒子的增多,种群的多样性也会进一步降低,势必影响算法的全局搜索能力。因此本文做如下改进:

$$\text{If } X_{id} > X_{\max} \quad X_{id} = X_{\max} - c * (X_{id} - X_{\max}) * \text{rand}() \quad \text{或者}$$

$$\text{If } X_{id} < -X_{\max} \quad X_{id} = -X_{\max} - c * (X_{id} + X_{\max}) * \text{rand}() \quad (10)$$

c 值用于调节范围大小,当粒子越界后,不再聚集在边界处,而分布在边界附近的可行域内,其大小主要取决于 $X_{id} - X_{\max}$,即越界的大小。既保证粒子位于可行性空间内,又克服了原算法易在边界聚集或陷入局部最优解的缺点,增加种群的多样性,有利于提高全局搜索能力。

子线程边界变异的 QPSO 算法流程:

Step1: 初始化每个粒子的位置向量,计算适应值函数,得到 pbest 和 gbest;

While(当前迭代次数<最大迭代次数&&其他子程序已完成)

Step2: 计算 beta 值和 mbest 的值,由式(7)得

到 P 值;

Step3: 按式 (8) 更新粒子的位置;

Step4: 按 (10) 式判定是否超过最大搜索区域, 限定 X_{id} 范围;

Step5: 得到 pbest 和 gbest 的值, 与主线程中的 pbest 和 gbest 比较, 若较好, 则更新, 反之用主线程中的值替代;

End

3 测试结果和分析

3.1 测试函数

为了测试并行的自适应量子粒子群算法的效果, 本文选用了下面 4 组函数对其实验, 并将实验结果与标准 PSO、APSO 和 QPSO 所获的实验结果进行了比较。

DeJong 函数: $F_1 = \sum_{i=1}^n x_i^2$

由 DeJong 提出也称为 Sphere 函数在 $(x_1, x_2, \dots, x_n) = (0, 0, \dots, 0)$ 达到极小值, 只有一个极值点, 全局最优值为 0, 易于求解。

Rosenbrock 函数: $F_2 = \sum_{i=1}^n \left[100(x_{i+1} - x_i^2)^2 + (x_i - 1)^2 \right]$

该函数是一个非凸、病态函数, 在 $(x_1, x_2, \dots, x_n) = (1, 1, \dots, 1)$ 达到全局极小值 0。

Rastrigrin 函数: $F_3 = \sum_{i=1}^n \left[x_i^2 - 10 \cos(2\pi x) + 10 \right]$

该函数是一个多峰值函数, 在 $(x_1, x_2, \dots, x_n) = (0, 0, \dots, 0)$ 达到全局极小点 0, 在 $\{x_i \in (-5.12, 5.12), i=1, 2, \dots, n\}$ 范围内大约有各局部极小点。

Griewank 函数: $F_4 = \frac{1}{4000} \sum_{i=1}^n x_i^2 - \prod_{i=1}^n \cos\left(\frac{x_i}{\sqrt{i}}\right) + 1$

该函数拥有很多个局部极小点, 局部极小点的个数与问题的维数有关, 极小值在 $x_i = \pm k\pi\sqrt{i} (i=1, 2, \dots, n)$, 而全局极小点为 0。

为了对比研究并行量子粒子群算法 (A-QPSO) 的性能, 将其与标准的粒子群优化算法 (PSO)、改进自适应粒子群算法 (APSO)、边界变异的量子粒子群算法 (QPSO) 进行比较。

表 1 各函数参数设置

	DeJong	Rosenbrock	Rastrigrin	Griewank
维数	30	30	30	30
初始范围	[-100,100]	[-30,30]	[-5.12,5.12]	[-600,600]
精度	0.01	100	100	0.1

3.2 实验结果与分析

算法中所有的初始参数如下: SwarmSize 为 80, w_{max} 为 0.9, w_{min} 为 0.4, c_{1s} 、 c_{1f} 分别取 1.5 和 2.5, c_{2s} 、 c_{2f} 分别取 2.5 和 1.5, Xmax 等于区间边界, 维数、区间、目标精度见表 1。算法共运行 50 次, 成功率是达到目标精度解的百分比, 运行结果如表 2、3。

表 2 四种算法在四种函数对比

	最大迭代次数	PSO			APSO		
		成功率	最优值	平均值	成功率	最优值	平均值
DeJong	500	0%	0.250513	2.294273	26%	0.002177	0.03527
	1000	96%	0.0001199	0.0018919	100%	1.13E-07	3.92E-06
	2000	100%	8.01E-11	1.55E-09	100%	7.16E-14	9.27E-14
Rosenbrock	500	0%	144.1424	593.8643	40%	30.03333	270.0934
	1000	60%	21.14301	139.45996	82%	10.743105	66.769981
	2000	86%	2.114676	53.97822	96%	4.334606	52.00391
Rastrigrin	500	100%	28.41722	48.38213	100%	27.89762	51.47116
	1000	100%	19.029863	34.975025	100%	27.858866	42.564784
	2000	100%	14.92438	25.85927	100%	17.90927	37.21146
Griewank	500	2%	0.091383	0.413209	90%	0.000518	0.035312
	1000	100%	2.80E-05	0.0111527	100%	2.18E-08	0.0093023
	2000	100%	9.62E-11	0.013924	100%	7.76E-14	0.015306

表 3 四种算法在四种函数对比 (续表)

	最大迭代次数	QPSOB			A-QPSOB		
		成功率	最优值	平均值	成功率	最优值	平均值
DeJong	500	100%	8.24E-10	5.80E-08	100%	9.72E-11	1.01E-09
	1000	100%	2.96E-20	3.99E-16	100%	0.00E+00	5.48E-22
	2000	100%	5.80E-21	8.94E-21	100%	8.64E-50	1.11E-47
Rosenbrock	500	88%	25.11884	56.11258	94%	20.8955	38.92937
	1000	98%	22.407117	37.867165	98%	22.718772	37.727766
	2000	100%	20.45611	33.84106	100%	21.09844	30.62408
Rastrigrin	500	66%	11.2739	83.90442	100%	17.15963	31.01745
	1000	84%	11.581667	45.893655	100%	12.058462	25.073191
	2000	98%	4.995355	23.0341	100%	10.94455	21.98869
Griewank	500	100%	1.59E-06	0.01167	100%	9.57E-09	0.0088479
	1000	100%	1.17E-08	0.0088628	100%	0.00E+00	0.0078753
	2000	100%	0.00E+00	0.005156	100%	0.00E+00	0.0046403

表 2、3 随着迭代次数的增加, 四种算法的成功率和精度方面都有提高。当迭代次数相同时, APSO、QPSO 和 A-QPSO 都比标准 PSO 都有了很大提高, 其中 QPSO 和 A-QPSO 最为明显。在 DeJong 函数 2000 代中, QPSO 和 A-QPSO 至少超过标准 PSO 有 $1e12$ 数量级, A-QPSO 算法的精度最高, 提高精度有 $1e38$ 数量级, 性能提升相当明显。对于病态 Rosenbrock 函数 500 代结果, A-QPSO 相比标准 PSO 的成功率有了 94% 的提升, 超过 QPSO 有 6%。而 Rastrigrin 函数得测试结果显示, QPSO 算法成功率是四个算法中最差的, 而标准 PSO、APSO 和 A-QPSO 效果较好。Griewank 函数有许多局部最小值, 500 代结果显示 QPSO 和 A-QPSO 摆脱局部最优值效果较好, 成功率比标准 PSO 提高了 98%, 并且 A-QPSO 可以得到精确解。综合以上可知, A-QPSO 实际是 APSO 和 QPSO 的一个结合体, 在寻找最优解方面和解的精度都是取二者之长, 在参数变量方面, 它所占用的变量空间方面是二者之和。但随着计算机技术的发展, 在变量空间方面对算法的制约已经越来越小, 对运行时间的要求则是越来越短。例如运行 DeJong 函数共 50 次并达到 $1e-2$ 精度内所用时间, 标准 PSO 为 14.400569s, APSO 为 13.613697s, QPSO 为 24.589759s, A-QPSO 为 15.691526s, 由此看出, 在达到相同的精度的条件下, A-QPSO 算法的运行时间比较短。

4 结语

本文提出了一种并行的自适应量子粒子群算法, 利用并行方式将改进的自适应粒子群算法和边界变异的量子粒子群算法结合起来, 通过共享粒子自身历史最优解和全局最优值, 使新的算法拥有了两种算法的优点。其中一个子线程自适应粒子群算法, 改进了标准粒子群算法中惯性权值和学习因子, 让其成为非线性可变, 加强了后期精确解的搜寻能力; 另一个子线程边界变异的量子粒子群算法, 对量子粒子群算法在边界处不易寻解的不足之处进行了改进, 进一步提高跳出局部最优解的能力。通过实验测试表明并行的自适应量子粒子群算法相比标准粒子群算法有求解精度高、不易陷入局部最优解、快速收敛的优点, 但这种算法所需要设的参量比较多, 运行时间还可以进一步的缩短, 这也是下一步改进方向。

参考文献

- 1 Kennedy J, Eberhart R. Particle swarm optimization. Proc. of the IEEE International Conference on Neural Networks. 1995. 1942-1948.
- 2 Xiao RY, Yu JH. A Newly Self-Adaptive Strategy for the PSO. Natural Computation. 2008. ICNC 2008. Fourth International Conference on. 2008. 396-400.
- 3 Dorigo M, Maniezzo V, Colomni A. Ant System: Optimization by a colony of cooperating agents. IEEE Trans. on SMC, 1996, 26(1):29-41.

(下转第 71 页)

4.4 改进方案与标准方案的性能对比分析

首先,硬件方面,改进方案对原方案的硬件设备改动很小,只需要调整实现KDF函数的电路,若KDF函数采用软件实现,则不需要对硬件做改动,为升级带来了便利。可见,硬件方面,改进方案的性能与原方案保持一致。

其次,在消息、信令的网络传输方面,改进方案通过X2接口传递的网络消息数量比标准方案少2次,通过Uu和S1接口传递的网络消息数量与标准方案一致。由于在终端和实体上的处理时间相对于网络传输时延可以忽略不计,可见,改进方案在网络传输方面比标准方案开销略小。

最后,在设计思想方面,改进方案最大限度的继承了标准方案的思想,没有调整标准方案中的密钥参数个数以及种类,也不需要修改相应的S1切换密钥更新协议,只需调整网络传输的参数和部分密钥推导函数等软、硬件,使得标准方案可以顺利过渡到本方案,在性能上比标准方案略好。

5 结语

本文主要针对LTE标准技术规范中的X2切换密钥更新协议进行了介绍,并分析了其缺陷——仅提供两跳前向安全性。同时,我们提出了一个改进方案,不仅最大限度的继承了原有方案的思想,还弥补了原

方案的缺陷,提供了一跳前向安全性,并使得原方案可以顺利过渡到本方案。

参考文献

- 1 Astély D, Dahlman E, Furuskär A, et al. LTE: the evolution of mobile broadband. IEEE Communications Magazine, 2009, 47: 44-51.
- 2 Forsberg D, Huang L, Tsuyoshi K, Alanara S. Enhancing security and privacy in 3GPP E-UTRAN radio interface, in proc. 18th IEEE International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC07). 2007. 1-5.
- 3 Forsberg D. LTE key management analysis with session keys context. Computer Communications, 2010,33:1907-1915.
- 4 3GPP, 3GPP System Architecture Evolution (SAE), Security architecture, 3GPP TS 33.401 v9.4.0, 2010.
- 5 3GPP, Evolved Universal Terrestrial Radio Access (E-UTR A) and Evolved Universal Terrestrial Radio Access Network (E-UTRAN), Overall description, 3GPP TS 36.300 v10.0.0, 2010.
- 6 3GPP, Evolved Universal Terrestrial Radio Access Network (E-UTRAN), architecture description, 3GPP TS 36.401 v9.2.0, 2010.

(上接第51页)

- 4 雷秀娟,史忠科,孙瑰琪.基于遗传算子的粒子群优化算法的比较分析.计算机工程与应用,2008,44(14):65-67.
- 5 高鹰,谢胜利.免疫粒子群优化算法.计算机工程与应用,2004,6:4-6.
- 6 高鹰,谢胜利.基于模拟退火的粒子群优化算法.计算机工程与应用,2004,1:47-49.
- 7 Thangaraj R, Pant M, Nagar AK. Maximization of Expected Target Damage Value Using Quantum Particle Swarm Optimization. Developments in eSystems Engineering (DESE), 2009 Second International Conference on. 2009. 329-334.
- 8 Shi YH, Eberhart RC. A modified particle swarm optimizer. Proc. of the IEEE Congress on Evolutionary Computation. Piscataway, USA: IEEE Service Center, 1998. 69-73.
- 9 Sun J, Feng B, Xu WB. Particle swarm optimization with particles having quantum behavior. Proc. of 2004 Congress on Evolution Computation. Piscataway. NJ: IEEE Press, 2004. 325-331.
- 10 刘昊,李大卫,王莉.遗传并行粒子群优化算法及其性能分析.辽宁科技大学学报,2008,31(5):495-499.
- 11 林星,冯斌,孙俊.基于边界变异的量子粒子群优化算法.计算机工程,2008,34(12):187-188.
- 12 李剑,王乘.一种改进的自适应微粒群优化算法.华中科技大学学报(自然科学学报),2008,38(3):118-121.