

引用格式:赵国新,陈志炼,魏战红. 混合自适应量子粒子群优化算法[J]. 微电子学与计算机, 2019, 36(7): 76-80. [ZHAO G X, CHEN Z L, WEI Z H. Hybrid adaptive quantum particle swarm optimization algorithm [J]. Microelectronics & Computer, 2019, 36(7): 76-80.]

混合自适应量子粒子群优化算法

赵国新¹, 陈志炼^{1, 2}, 魏战红¹

(1 北京石油化工学院 信息工程学院, 北京 102617;

2 北京化工大学 信息科学与技术学院, 北京 100029)

摘 要: 为解决量子粒子群优化算法在迭代后期出现的种群多样性低导致的早熟收敛、陷入局部最优的问题, 提出三点改进: (1) 将收缩—扩张系数与适应度值联系起来, 收缩—扩张系数会随着粒子的适应度值的改变而自适应调整; (2) 使用差分策略更新粒子的随机位置, 使得粒子向种群最优位置靠近; (3) 粒子位置的更新加入 Levy 飞行策略, 利用 Levy 飞行策略的偶尔长距离的跳跃, 使得种群多样性增加, 提高了跳出局部最优的能力, 综合以上三点, 提出一种混合自适应量子粒子群优化算法 (HAQPSO). 通过对比各个算法在 6 个典型函数的仿真测试结果表明: HAQPSO 具有更好的全局收敛能力, 且收敛精度、速度和稳定性都有明显的提升.

关键词: 量子粒子群优化算法; 收缩—扩张系数; 差分策略; Levy 飞行策略

中图分类号: TP301.6

文献标识码: A

文章编号: 1000—7180(2019)07—0076—05

Hybrid adaptive quantum particle swarm optimization algorithm

ZHAO Guo-xin¹, CHEN Zhi-lian^{1, 2}, WEI Zhan-hong¹

(1 College of Information Technology, Beijing Institute of Petrochemical Technology, Beijing 102617, China;

2 College of Information Science and Technology, Beijing University of Chemical Technology, Beijing 100029, China)

Abstract: To solve the problem of quantum particle swarm optimization algorithm appears the poor diversity of population at the end of iteration, three improvements are proposed: (1) associating the contraction-expansion coefficient with the fitness value, the contraction-expansion coefficient will adjust adaptively with the change of the fitness value of the particle; (2) differential strategy is used to update the random location of particles which makes the particle approach the optimal position of the population;. (3) updating particle position with Levy flight strategy, Levy flight strategy's occasional long jumps are utilized to increase the diversity of population and the ability of jump out of the local optimum. A hybrid adaptive quantum particle swarm optimization (HAQPSO) algorithm is proposed based on the above three points. By comparing the simulation result of 6 typical functions shows that: Improved quantum particle swarm optimization algorithm has better global convergence ability than the quantum particle swarm optimization algorithm, and the convergence precision, speed and stability are improved obviously.

Key words: quantum particle swarm optimization algorithm; contraction-expansion coefficient; differential strategy; Levy flight strategy

1 引言

为了改进粒子群优化算法存在易早熟和不能保

证完全收敛的问题^[1], 孙俊等^[2]提出了一种具有量子行为的粒子群优化算法 (Quantum Particle Swarm Optimization, QPSO), 并证明了该算法的全

局收敛性. QPSO 算法与粒子群优化算法相比,该算法的收敛速度快,精度更高,全局搜索能力更强等特点,但是在算法迭代的后期,还是会出现种群多样性降低,导致易陷入局部最优,最终出现早熟收敛的问题. 为了改善 QPSO 算法的早熟收敛的问题,陈汉武^[3]等人提出了一种带交叉算子的量子粒子群优化算法,采用新的计算粒子吸引点和势阱特征长度的方法,再将交叉算子融合到算法中,让种群的多样性得以增加;关学忠^[4]等人采用正态云模型以一定概率将量子粒子群优化算法中的粒子进行变异操作,维持粒子的多样性,避免陷入局部最优.

在 QPSO 算法的基础上提出改进,本文提出一种混合自适应量子粒子群优化算法 (Hybrid Adaptive Quantum Particle Swarm Optimization, HAQPSO).

2 QPSO 算法

量子粒子群优化算法^[5]将每个个体用量子空间的一个粒子来表示,在一个量子空间中,同时确定不了粒子的速度和位置,所以粒子的状态必须采用波函数 ψ 来描述,粒子在 δ 势阱的基础上不断向粒子吸引点也就是粒子最优位置 $p_i(t)$ 靠近,粒子的位置概率分布由薛定谔方程确定. 为了得到粒子精确的位置,就得将量子状态塌缩到经典状态,这时,采用蒙特卡洛随机模拟的方式就可以得到粒子的运动轨迹,公式如下:

$$x_{id}(t+1) = p_i(t) \pm \frac{L_{id}(t)}{2} \ln[1/u_{id}(t)] \quad (1)$$

式中, $x_{id}(t+1)$ 为第 $t+1$ 次迭代时,粒子 i 在第 d 维的位置, $\pm$ 是由 u 的大小决定的,当 $u > 0.5$ 时,取加号,其他情况取减号; $p_i(t)$ 为第 t 次迭代时的粒子随机位置,计算公式如下:

$$p_i(t) = \varphi_d(t)p_{id}(t) + [1 - \varphi_d(t)]p_{gd}(t) \quad (2)$$

式中, p_{id} 为个体的最优位置, p_{gd} 为种群的最优位置, φ_d 为均匀分布在 $(0, 1)$ 之间的随机数.

$L_{id}(t)$ 为势阱长度,计算公式如下:

$$L_{id}(t) = 2\alpha(t) |C_d(t) - x_{id}(t)| \quad (3)$$

在 $L_{id}(t)$ 中引入平均最好位置,记为 $C(t)$,即:

$$C(t) = \frac{1}{N} \sum_{i=1}^N P_i(t) \quad (4)$$

将式(3)代入式(1)就可以得到新的粒子进化方程:

$$x_{id}(t+1) = p_i(t) \pm \alpha(t) |C_d(t) - x_{id}(t)| \times \ln[1/u_{id}(t)] \quad (5)$$

式(3)和式(5)中的 α 被称为收缩—扩张系数,它是除去群体规模以及迭代次数以外唯一的一个控制参数.

3 改进的 QPSO 算法

3.1 收缩—扩张系数控制方法的改进

上文提到的收缩—扩张系数 α 的改变直接影响粒子的行为. 对 α 的控制通常采用固定取值和线性减小的方法. 受到自适应权重 PSO 算法的启发,本文提出一种参数自适应的控制方法来调节 α , 公式如下:

$$\alpha = \begin{cases} \alpha_{\min} - \frac{(\alpha_{\max} - \alpha_{\min}) \times (f - f_{\min})}{f_{\text{avg}} - f_{\min}}; & f \leq f_{\text{avg}} \\ \alpha_{\max}; & f > f_{\text{avg}} \end{cases} \quad (6)$$

式中, $\alpha_{\max}$ 和 $\alpha_{\min}$ 分别是 α 的最大值和最小值,根据文献^[5]选取 α 的最大值和最小值分别为 1 和 0.5, f 为粒子当前的最优个体适应度值, f_{avg} 和 $f_{\min}$ 分别为当前所有粒子的平均适应度值和最小适应度值. 因参数 α 会随着粒子的适应度值的改变而自适应调整,所以称为参数的自适应.

当各粒子的适应度值趋向于一致或者局部最优时,参数 α 增大;当各粒子的适应度值比较分散时,参数 α 减小,同时对于当前的适应度值优于平均适应度值的粒子,其对应的参数 α 较小,从而该粒子得到了保留. 反之,同时对于当前的适应度值差于平均适应度值的粒子,其对应的参数 α 较大,使得该粒子向较好的搜索范围靠拢.

3.2 粒子随机位置更新的改进

DE 算法^[6]主要特性就是差分策略,其中以 DE/rand/1/bin 和 DE/best/2/bin 应用最为广泛,本文结合第一种差分策略对粒子最优位置的更新进行改进,改进公式如下:

$$P_i(t+1) = P_{(r0)gd}(t) + F(P_{(r1)gd}(t) - P_{(r2)gd}(t)) \quad (7)$$

式中, F 为缩放因子,采取与参数 α 控制方法相同的自适应改变的方法,公式如下:

$$F = \begin{cases} F_{\min} - \frac{(F_{\max} - F_{\min}) \times (f - f_{\min})}{f_{\text{avg}} - f_{\min}}; & f \leq f_{\text{avg}} \\ F_{\max}; & f > f_{\text{avg}} \end{cases} \quad (8)$$

根据文献^[6]选取 F 的最大值和最小值分别为 0.9 和 0.4.

3.3 粒子位置更新的改进

对于粒子位置的更新,加入 Levy 飞行策略.

Levy 飞行策略是模拟动物在陌生环境中搜寻食物的理想策略^[7],鉴于 Levy 飞行策略的特点,许多学者在仿生学智能算法的进化方程中加入 Levy 飞行策略^[8-9],改进了算法的性能并取得良好的优化效果.

Levy 飞行的步长与时间 t 服从 Levy 分布,在进行简化和傅里叶变换后可以得到幂指数形式的概率密度函数:

$$\text{Levy} \sim \mu = t^{-\lambda}; 1 < \lambda < 3 \tag{9}$$

式中, λ 为幂指数. 式(9) 是带有宽尾的概率分布. 在实际使用过程中,通过简单的编程语言实现起来比较困难,因此,通常使用 Mantegna 提出的 Levy 飞行路径表达式:

$$\text{Levy}(\lambda) = \mu / |\nu|^{1/\beta} \tag{10}$$

式中,参数 β 的取值范围为 $(0, 2)$,通常取 $\beta = 1.5$;参数 μ 和 ν 是服从正太分布的随机数,即 $\mu \sim N(0, \sigma_\mu^2)$, $\nu \sim N(0, \sigma_\nu^2)$;参数 μ 和 ν 所对应的正太分布的标准差 σ_μ 和 σ_ν 取值满足式(11):

$$\begin{cases} \sigma_\mu = \{ \frac{\Gamma(1+\beta) \times \sin(\pi\beta/2)}{\Gamma[(1+\beta)/2] \times \beta \times 2^{(\beta-1)/2}} \} \\ \sigma_\nu = 1 \end{cases} \tag{11}$$

将式(11)代入式(10)就可以得到 Levy 飞行路径 $\text{Levy}(\lambda)$. 基于 Levy 飞行策略的 QPSO 算法需要将原算法中的粒子进化方程更新为:

$$\begin{cases} x_{id}(t+1) = p_i(t) + \text{Levy}(\lambda)[x_{id}(t) - p_{igd}] + \\ \alpha(t) | C_d(t) - x_{id}(t) | \times \ln[1/u_{id}(t)]; u_{id}(t) > 0.5 \\ x_{id}(t+1) = p_i(t) + \text{Levy}(\lambda)[x_{id}(t) - p_{igd}] - \\ \alpha(t) | C_d(t) - x_{id}(t) | \times \ln[1/u_{id}(t)]; u_{id}(t) \leq 0.5 \end{cases} \tag{12}$$

以下是 HAQPSO 算法的执行步骤:

(1)对粒子群数目、最大迭代次数、粒子群中的每个粒子的位置进行初始化,并将粒子个体最好位置初始化为当前位置;

(2)计算粒子群的平均最好位置;

(3)计算粒子当前的适应度值,根据适应度最小原则,与前一次迭代的适应度值进行比较,如果当前的适应度值小于前一次迭代的适应度值则把粒子的最好位置更新为粒子当前的位置,否则粒子的最好位置不变;

(4)计算出种群当前的全局最好位置. 如果粒子的最好位置对应的适应度值小于前一次迭代的全局最好位置的适应度值则把全局最优位置更新为粒子的最好位置,否则群体的全局最好位置不变;

(5)分别使用式(2)、式(12)来计算出随机点的位置和粒子新的位置;

(6)重复执行(2)~(5),直到满足设定的结束条件或者达到最大迭代次数.

4 实验仿真和结果分析

为了验证 HAQPSO 算法的性能,本文对 6 个标准测试函数进行测试,在 MATLAB R2016a 平台编写了相关程序,并对 HAQPSO 算法与未改进的 QPSO 算法仿真实验结果进行了比较验证. 实验需要设置的参数有:最大迭代次数为 2 000,群体规模为 20,收缩—扩张系数 α 的取值范围为 $[0.5, 1]$,缩放因子 F 的取值范围为 $[0.4, 0.9]$,本文所用到的测试函数如表 1 所示.

4.1 HAQPSO 与 QPSO 方法的性能对比

HAQPSO 算法与未改进的 QPSO 算法分别对 6 个测试函数独立运行 30 次,相应的测试结果如表 2 中的所示(结果保留 4 位小数),包括 30 次实验结果中的最差值(Max),最优值(Min),平均值(Mean)以及标准差(Std). 为了更直观的比较算法的性能,从 30 次的测试结果中挑选出适应度值表现最好的一次画成迭代曲线,如图 1~图 6 所示.

表 1 典型的测试函数

测试函数	表达式	搜索边界	最优值
Sphere	$\sum_{i=1}^{20} x_i^2$	$[-100, 100]$	0
Rosenbrock	$\sum_{i=1}^{19} [100(x_{i+1} - x_i^2)^2 + (x_i - 1)^2]$	$[-10, 10]$	0
Rastrigrin	$\sum_{i=1}^{20} (x_i^2 - 10\cos(2\pi x) + 10)$	$[-5.12, 5.12]$	0
Griewank	$\frac{1}{4\,000} \sum_{i=1}^{20} x_i^2 - \prod_{i=1}^{20} \cos(\frac{x_i}{\sqrt{i}}) + 1$	$[-600, 600]$	0
Ackley	$-20\exp(-0.2\sqrt{\frac{1}{20} \sum_{i=1}^{20} x_i^2}) - \exp[\frac{1}{20} \sum_{i=1}^{20} \cos(2\pi x_i^2)] + 20 + e$	$[-32, 32]$	0
Schaffer	$0.5 + \frac{(\sin \sqrt{x_i^2 + x_{i+1}^2})^2 - 0.5}{(1 + 0.001(x_i^2 + x_{i+1}^2))^2}$	$[-10, 10]$	0

表 2 测试结果			
测试函数	指标	HAQPSO	QPSO
Sphere	Max	0	1.704 6e-7
	Min	0	2.976 8e-11
	Mean	0	1.440 8e-8
	Std	0	3.282 1e-8
Rosenbrock	Max	0.000 7	4.835 7
	Min	4.827 9e-6	0.065 9
	Mean	0.000 2	1.341 6
	Std	0.000 2	0.904 7
Rastrigrin	Max	0	0.004 6
	Min	0	3.094 3e-6
	Mean	0	0.000 5
	Std	0	0.000 9
Griewank	Max	0.109 9	3.062 0
	Min	0.007 0	0.215 2
	Mean	0.038 9	1.148 7
	Std	0.025 7	0.715 5
Ackley	Max	8.881 8e-16	0.000 4
	Min	8.881 8e-16	1.225 7e-6
	Mean	8.881 8e-16	6.939 2e-5
	Std	0	7.623 7e-5
Schaffer	Max	0	0.001 2
	Min	0	1.621 3e-6
	Mean	0	0.000 2
	Std	0	0.000 3

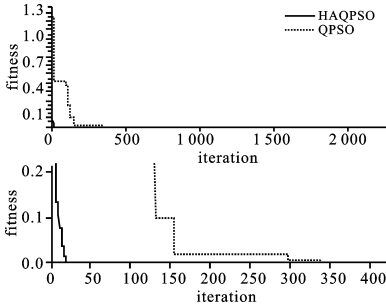


图 1 Sphere 函数迭代曲线

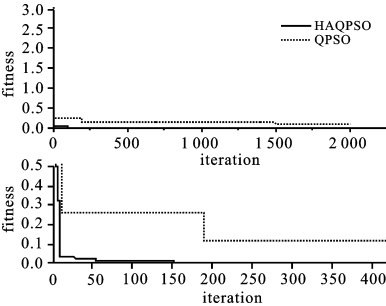


图 2 Rosenbrock 函数迭代曲线

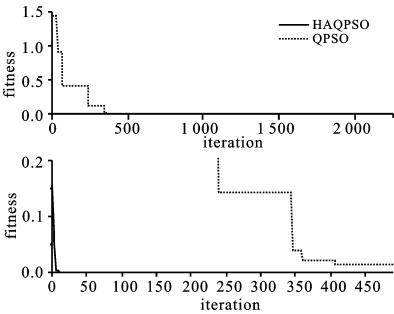


图 3 Rastrigrin 函数迭代曲线

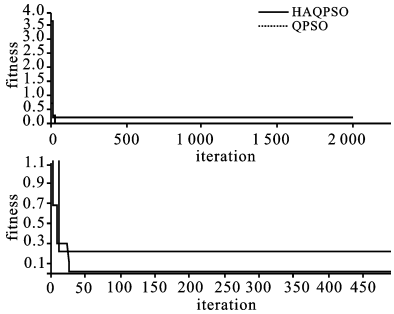


图 4 Griewank 函数迭代曲线

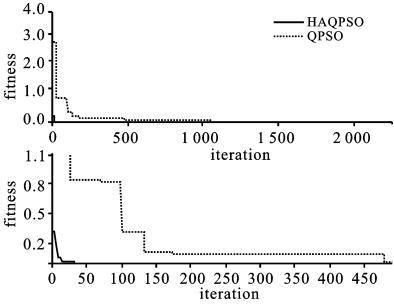


图 5 Ackley 函数迭代曲线

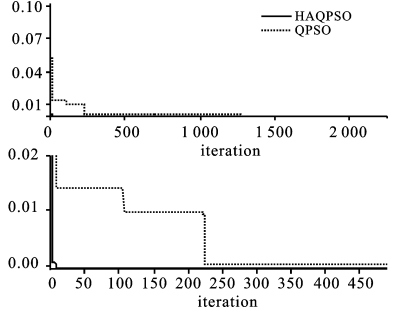


图 6 Schaffer 函数迭代曲线

从表 2 可以看到, HAQPSO 算法在所有的测试函数的性能指标(Max, Min, Mean)表现都比未改进的 QPSO 算法好, 特别是对于 Sphere、Rastrigrin、Ackley、Schaffer 函数的寻优时, 提升效果十分明显, Sphere、Rastrigrin 和 Schaffer 函数都搜寻到了理论极值, Ackley 函数最少也提高了 11 个数

量级,表明了 HAQPSO 算法收敛精度更高. 对于标准差,HAQPSO 算法的标准差也比 QPSO 算法小很多,有的甚至为 0,表明了 HAQPSO 算法有着比 QPSO 算法更好的稳定性.

在图 1~图 6 中,上图为算法的整体迭代过程,下图为局部放大的迭代过程. 可以看出,QPSO 算法在陷入局部最优的时候,无法跳出,导致收敛的精度不高. 从整体迭代过程来看,HAQPSO 算法向理论极值下降的速度比 QPSO 算法快很多,例如 Sphere 函数,HAQPSO 算法在 25 次迭代之前就找到最优值,而 QPSO 算法要到 340 次迭代才找到最优值,其他测试函数相同,HAQPSO 算法搜寻到最优值的速度明显优于 QPSO. 综合以上的分析,说明了算法改进的有效性.

4.2 HAQPSO 与参考文献中方法的性能对比

将 HAQPSO 算法与参考文献[3, 10-13]中的 5 种算法进行对比,评价指标为算法对测试函数进行 30 次实验的所得的最优适应度值的平均值(表中“—”代表参考文献中的算法没有对这个函数做测试,“CQPSO”为带交叉算子的量子粒子群优化算法,“LFOA”为具有 Levy 飞行特征的双子群果蝇优化算法,“DAAFSA”为双自适应人工鱼群优化算法,“IGWO”为基于动态搜索的改进灰狼优化算法,“HJCSO”为基于模式搜索的鸡群算法),如表 3 所示.

表 3 各算法最优适应度均值比较

测试函数	HAQPSO	CQPSO	LFOA	DAAFSA	IGWO	HJCSO
Sphere	0	2.60e-109	2.290 0e-16	2.05e-15	6.439 4e-60	1.775 3e-9
Rosenbrock	0.000 2	2.50e-3	27.129 2	—	3.975 2	4.912 2e-8
Rastrigrin	0	0	3.286 3e-15	4.79e-13	0	4.555 5e-16
Griewank	0.038 9	0	0	1.11e-17	0	6.119 8e-18
Ackley	8.881 8e-16	4.00e-15	9.710 0e-9	2.77e-4	1.509 9e-14	2.078 9e-6
Schaffer	0	—	0	3.15e-15	—	9.238 9e-17

从整体来看,除了 Griewank 和 Rosenbrock 函数以外,HAQPSO 算法在另外的 4 种测试函数的表现相较于其他 5 种方法都要好. 从单个测试函数来看,对于 Sphere 函数,只有 HAQPSO 能搜寻到了理论极值;而 HAQPSO 对于 Rosenbrock 函数的表现也比 HJCSO 差,但是比 LFOA 和 IGWO 好很多,与 CQPSO 的结果相当;对于 Rastrigrin 函数,HAQPSO、CQPSO 和 IGWO 都能搜寻到理论极值 0;对于 Griewank 函数的求解寻优,HAQPSO 的最优适应度值的平均值只有 0.038 9,在 6 种算法种表现最差;而 Ackley 函数,HAQPSO 的表现最好,与

CQPSO 和 IGWO 结果表现比较接近,相差最多的 DAAFSA 达到了 12 个数量级;最后的 Schaffer 函数,HAQPSO 和 LFOA 都找到了其理论极值.

5 结束语

本文针对 QPSO 算法在迭代的后期会出现种群多样性不足导致易陷入局部最优的问题,提出 3 点改进,首先采用自适应改变的方法来控制收缩—扩张系数;然后利用差分策略对粒子的随机位置进行更新;最后在粒子的位置更新中加入 Levy 飞行策略,增加了种群的多样性,扩大了搜索空间,提高了跳出局部最优的能力. 通过 6 个测试函数的实验结果和与参考文献中的方法进行比较可得,HAQPSO 算法的表现较好,改进的方法提升了算法的全局搜索能力,提高了稳定性以及收敛的精度、速度.

参考文献:

[1] VAN DEN BERGH F. An analysis of particle swarm optimizers[D]. Hatfield, South Africa: University of Pretoria, 2002.

[2] SUN J, FENG B, XU W B. Particle swarm optimization with particles having quantum behavior[C]//Proceedings of 2004 Congress on Evolutionary Computation. Piscataway, NJ, USA: IEEE, 2004: 325-331.

[3] 陈汉武, 朱建锋, 阮越. 带交叉算子的量子粒子群优化算法[J]. 东南大学学报(自然科学版), 2016, 46(1): 23-29.

[4] 关学忠, 皇甫旭, 李欣, 等. 基于正态云模型的自适应变量子粒子群优化算法[J]. 电子设计工程, 2016, 24(8): 64-67.

[5] 方伟, 孙俊, 谢振平. 量子粒子群优化算法的收敛性分析及控制参数研究[J]. 物理学报, 2010, 59(6): 3686-3694.

[6] 丁青锋, 尹晓宇. 差分进化算法综述[J]. 智能系统学报, 2017, 12(4): 431-432.

[7] CHECHKINA V, METZLER R, KLAFTER J, et al. Introduction to the theory of Levy flights[C]//Anomalous Transport: Foundations and Applications. Weinheim: John Wiley Sons, 2008: 129-162.

[8] 刘长平, 叶春明. 具有 Levy 飞行特征的蝙蝠算法[J]. 智能系统学报, 2013, 8(3): 240-246.

[9] ZHANG Y, WANG L, WU Q. Modified adaptive Cuckoo search(MACS) algorithm and formal description for global optimization[J]. Int J of Computer Applications in Technology, 2012, 44(2): 73-79.

根据图 6(b)所示:从宏观角度分析,50 人的调查结果中权值最低与最高都出现在 GA 所重构的图形方案中,其最高权值值为 0.312 5,最低权值值为 0.062 5;从调查结果微观角度分析,绝大多数人选择的 GA 与 BP 神经网络各权值总和都在 0.187 5 至 0.25 之间;调查的 50 人中 GA 与 BP 神经网络所获得的权值总和分别为 9.875 和 10.437 5.

6 结束语

本文以非物质文化遗产中蜡染图形为载体,提出并运用 GA 和 BP 神经网络两种算法对其框架进行重构;同时,利用拓扑构型对重新生成的图形框架进行纹样填充,生成 16 幅图案进行调查分析.针对分析结果,分别得出在 4 种因素影响下,何种算法更适合于本案例研究;并且根据人工打分结果显示,BP 神经网络生成方案的权值总和大于 GA 生成方案的权值总和,其大于的值为 0.5625;即在美观性、组合合理性方面,BP 神经网络更适合于本案例图形重构设计.

参考文献:

[1] 林晓凡,林琳. 基于网络的辅助设计系统构建[J]. 微电子学与计算机,2007,24(7):163-165.

[2] 王晨琛,王业琳,葛中芹,等. 基于卷积神经网络的中国水墨画风格提取[J]. 图学学报,2017,38(5):754-759.

[3] SIMONYAN K, ZISSERMAN A. Very deep convolu-

[4] 张前图,房立清,赵玉龙. 具有 Levy 飞行特征的双子群果蝇优化算法[J]. 计算机应用,2015,35(5):1348-1352.

[5] 徐建波,戴月明,严大虎. 双自适应人工鱼群优化算法[J]. 微电子学与计算机,2018,35(4):54-57.

[6] 罗佳,唐斌. 新型灰狼优化算法在函数优化中的应用[J]. 兰州理工大学学报,2016,43(3):96-101.

[7] 张慕雪,张达敏,何锐亮. 基于模式搜索法的鸡群优化

ational networks for large-scale image recognition [EB/OL]. [2017-01-02]. <https://core.ac.uk>.

[4] HE K, ZHANG X, REN S, et al. Deep residual learning for image recognition [EB/OL]. [2017-01-02]. <https://arxiv.org>.

[5] 周爱武,翟增辉,刘慧婷. 基于模拟退火算法改进的 BP 神经网络算法[J]. 微电子学与计算机,2016,33(4):144-147.

[6] 赵婷婷,魏小鹏,易鹏飞. 基于改进遗传算法的形状进化设计及其三维图形生成[J]. 机械工程学报,2010,46(19):147-154.

[7] 黄贤英,李沁东,熊李媛. 结合拓扑势用户聚类的协同过滤推荐算法[J]. 计算机工程与设计,2018,39(1):90-95.

[8] 董青. 高级时装定制中传统吉祥纹样的图形重构及应用性研究[D]. 大连:大连工业大学,2015.

作者简介:

丁 宁 女,(1993-),硕士研究生. 研究方向为计算机辅助图形设计.

吕 健(通讯作者) 男,(1983-),博士,副高. 研究方向为设计学、信息与交互设计. E-mail:305515940@qq.com.

胡 涑 男,(1993-),硕士研究生. 研究方向为计算机辅助设计与制造.

赵慧亮 男,(1988-),博士,讲师. 研究方向为工业设计、现代设计方法.

王伟祎 女,(1992-),硕士研究生. 研究方向为工业设计.

(上接第 80 页)

算法[J]. 微电子学与计算机,2018,35(4):46-52.

作者简介:

赵国新 男,(1962-),硕士,教授,硕士生导师. 研究方向为工业控制系统、工业控制系统信息安全.

陈志炼(通讯作者) 男,(1994-),硕士研究生. 研究方向为工业控制系统信息安全. E-mail:247058468@qq.com.

魏战红 女,(1985-),博士,讲师. 研究方向为保密通信.