

一种基于交叉和变异算子改进的遗传算法研究

谢燕丽¹, 许青林², 姜文超²

(1. 广东工业大学 信息工程学院, 广东 广州 510006;

2. 广东工业大学 计算机学院, 广东 广州 510006)

摘要:文中针对函数优化方面遗传算法(GA)存在的“早熟”与收敛速度慢的问题,设计了一种基于交叉和变异算子改进的遗传算法。通过研究分析GA,根据交叉算子和变异算子的特点,在现有的GA基础上,引入拉普拉斯算子改进交叉算子以及结合黄金分割法对变异算子做了进一步改进。通过3个测试函数对该算法与标准遗传算法,以及其他两种算法加以对比,仿真结果表明文中的算法不仅增加了个体多样性,防止了“早熟”,且比其他三种算法获得了更优解和更快的收敛速度。理论分析和实验表明,提出的算法是可行有效的。

关键词:交叉算子;变异算子;优化;遗传算法

中图分类号:TP301.6

文献标识码:A

文章编号:1673-629X(2014)04-0080-04

doi:10.3969/j.issn.1673-629X.2014.04.020

An Improved Genetic Algorithm Based on Crossover and Mutation Operators

XIE Yan-li¹, XU Qing-lin², JIANG Wen-chao²

(1. School of Information Engineering, Guangdong University of Technology, Guangzhou 510006, China;

2. Faculty of Computer, Guangdong University of Technology, Guangzhou 510006, China)

Abstract: Aiming at problems of genetic algorithm in terms of function optimization exists the "premature" and slow convergence, a kind of improved genetic algorithm based on crossover and mutation operators is designed. Through research and analysis of GA, according to the characteristics of the crossover operator and mutation operator, on the basis of existing GA, introduce the Laplacian operator to improve crossover operator and further improve mutation operator combined with golden section method. Through the three test functions to compare this algorithm with standard Genetic Algorithm (GA), and the other two algorithms, simulation results show that the proposed algorithm can not only increase the diversity of individuals, preventing the "premature", and more than the other three algorithms gain better solution and faster convergence. Theoretical analysis and experimental results show that the proposed method is feasible and effective.

Key words: crossover operator; mutation operator; optimization; Genetic Algorithm (GA)

0 引言

最优化问题是遗传算法(Genetic Algorithm, GA)的经典应用领域^[1],也是对GA进行性能评价的常用算例。采用常规方法对于一些大规模、非线性、多峰多态函数等问题的有效解决往往存在许多障碍,遗传算法却可以方便高效地得到较好的结果,并广泛应用于其他领域,取得了长足的发展。GA是模仿自然选择、物种进化而建立的随机搜索技术,其概念是由美国Michigan大学Holland教授于1975年首次提出^[2]。GA从随机产生的一组个体开始,反复经过选择、交

叉、变异三种遗传算子进行进化。其中,选择算子是根据个体的适应度评价来确定个体的生存,适应度高的被选择复制进入下一代进化,反之则被淘汰,这是“优胜劣汰,适者生存”的自然界规律体现,这个过程不会产生新的个体;交叉算子融合染色体与遗传信息探索整个搜索空间,将父代个体配对进行一个或者多个基因位重组,产生大量新个体,增加最优个体出现的机率,提高全局搜索能力;变异算子是通过改变个体内部的基因,来保持个体的多样性,克服早熟收敛^[3]。在这些遗传算子中,交叉和变异算子是产生新个体的主

收稿日期:2013-06-26

修回日期:2013-09-29

网络出版时间:2014-01-28

基金项目:广州市科技攻关项目(2012Y2-00040)

作者简介:谢燕丽(1989-),女,广西北海人,硕士,研究方向为软件工程、企业信息化;许青林,硕士,副教授,研究方向为软件工程、企业信息化。

网络出版地址: <http://www.cnki.net/kcms/detail/61.1450.TP.20140128.1140.017.html>

要操作,对 GA 是否可以得到全局最优解十分重要,是影响算法收敛性能的关键。实践表明,种群经过一定的进化代数之后,失去了种群多样性,GA 会产生早熟、搜索效率低和容易陷入局部搜索的问题。为了克服这些问题,国内外学者已经做了大量的研究和探索,并提出了许多有效的方法^[4-6]。文中着重从交叉和变异算子改进的方向出发,以增加种群多样性、提高算法全局与局部搜索能力、加快收敛速度为目的,设计一种新的遗传算法。

1 一种新的遗传算法

1.1 交叉算子的改进

交叉算子的设计一般要与个体编码设计一致^[7],在采用实数编码机制的 GA 中,算术交叉算子^[8]广泛被人们所青睐。文中引入算术交叉算子,改进其生成系数。其算子表示如下:

$$X_A^k = \beta X_B^{k-1} + (1 - \beta) X_A^{k-1} \quad (1)$$

$$X_B^k = \beta X_A^{k-1} + (1 - \beta) X_B^{k-1} \quad (2)$$

式中, X_A^k , X_B^k 表示第 k 代种群中的两个个体 X_A 、 X_B , k 是进化代数。 β 为参数,若 β 为常数,此时称为均匀算术交叉;若与进化代数相关,则称为非均匀算术交叉。

为了使 k 代中 A 、 B 两个体中优良个体在后代中占有更大的比例,并始终保持种群的多样性,在这里用拉普拉斯系数^[9]改进 β 的系数生成。新一代的个体主要由前一代的个体和系数 β 决定,且后代个体多样性与父代个体多样性是密切相关的,而个体之间的距离是保持多样性的关键, λ 用来调节与父代个体之间的距离, λ 越小越靠近父代个体,种群的多样性越高; λ 越大越远离父代个体,则种群多样性越低。子代是对父代成比例的延伸,在 u 、 λ 共同作用下,父代的两个个体远离或靠近对方,子代的两个个体随之远离或靠近对方。基于以上特点,文中设计的算术交叉算子生成系数如下:

$$\beta = \begin{cases} u - \lambda \log_e(r), & r \leq \frac{1}{2} \\ u + \lambda \log_e(r), & r > \frac{1}{2} \end{cases} \quad (3)$$

式中, r 是在 $[0,1]$ 内符合均匀分布的随机数; u 、 λ 是拉普拉斯系数, $u \in R$ 是位置参数, $\lambda > 0$ 是尺度参数。二者可从父代群体个体分布反推得来。

1.2 变异算子的改进

2008 年,李小宁利用黄金分割法与 GA 结合^[10],采用两个黄金分割变异点,构造出了一种新的混合搜索方法。文中在此基础上,改进该方法中的黄金分割变异算子,在使种群在进化过程中尽可能地获得优良

新个体的基础上,为了避免算法陷入局部搜索,由两个点增加到 4 个点,尽最大可能寻得全局最优解。第 k 代种群个体 X 中第 i 维个体表示如下:

$$X_i^k = X_i^{k-1} + 0.618 \cdot (X_i^{k-1} - a_i) \quad (4)$$

$$X_i^k = X_i^{k-1} + 0.618 \cdot (b_i - X_i^{k-1}) \quad (5)$$

$$X_i^k = X_i^{k-1} + 0.382 \cdot (X_i^{k-1} - a_i) \quad (6)$$

$$X_i^k = X_i^{k-1} + 0.382 \cdot (b_i - X_i^{k-1}) \quad (7)$$

其中, a_i , b_i 对应 X_i 的定义域,即 $X_i \in [a_i, b_i]$ 。

在黄金分割变异操作中,第 k 代个体生成原则可以有以上四种表示方法。其选择策略主要由对应的适应度大小决定,比较四个黄金分割点的适应度大小,选择适应度值最大的个体对应的个体,如式(8)表示:

$$\max \begin{cases} f(X_i^{k-1} + 0.618 \cdot (X_i^{k-1} - a_i)), \\ f(X_i^{k-1} + 0.618 \cdot (b_i - X_i^{k-1})), \\ f(X_i^{k-1} + 0.382 \cdot (X_i^{k-1} - a_i)), \\ f(X_i^{k-1} + 0.382 \cdot (b_i - X_i^{k-1})) \end{cases} \quad (8)$$

1.3 算法实现

改进的 GA 实现步骤如下:

步骤①:初始化种群规模 popsize、种群代数 maxgens、变异概率 P_M 、交叉概率 P_X 、最优个体 best_fit=0 等,根据参数的上下限采用实数编码初始化种群,种群个体为: $\{x_1, x_2, \dots, x_{\text{popsize}}\}$ 。

步骤②:计算初始种群中每个个体的适应度值,给出适应度函数 $\text{fit}(x)$ ^[11]。目标函数求最大值的,则 $\text{fit}(x) = f(x)$,目标函数求最小值,则 $\text{fit}(x) = -f(x)$ 。并对个体适应度排序,将 num 个最好个体放在最优个体存储池^[12],记为 Best(num)。存储适应度最大或最小值,记为 fit_max 或 fit_min,以及对应的个体。

步骤③:选择操作。计算种群个体的选择概率 P_s ^[13], $P_s(i) = \frac{\text{fit}(i)}{\sum_{i=0}^{\text{popsize}} \text{fit}(i)}$,选择出要进化的个体。

步骤④:利用交叉概率 P_X 选出 $P_X \times \text{popsize}$ 组个体进行改进的算术交叉运算。

步骤⑤:种群的变异运算。文中采用黄金分割变异算子,计算四个黄金分割点的适应度值,并找出最大值的点。以 P_M 的概率执行 $X^{k+1} = X^k + \alpha \cdot (b - X^k)$ 或者 $X^{k+1} = X^k + \alpha \cdot (X^k - a)$, $\alpha = 0.618$ 或者 0.382 。

步骤⑥:种群的适应度评价。 $\text{fit}(x) = f(x)$ 或者 $\text{fit}(x) = -f(x)$ 。并对个体适应度排序,若此代的最优值大于或小于前代的最优值,则替换原来的 best_fit 以及对应的个体。并将 Best(num) 替代种群中最差的 num 个个体,并将种群中前 num 个较好的个体放到 Best(num)。

步骤⑦:判断种群代数是否迭代结束,否则回到步骤③。

2 算法的测试及结果分析

实验选用 GA 常用的其中三个测试函数^[14], 如下所示:

$$\max f_1(x) = 0.5 - \frac{\sin^2 \sqrt{(x_1^2 + x_2^2)} - 0.5}{[1 + 0.001(x_1^2 + x_2^2)]^2},$$
$$[-10, 10]$$
$$\max f_2(x) = 100(x_1^2 - x_2)^2 + (1 - x_1)^2,$$
$$[-2.048, 2.048]$$
$$\min f_3(x) = 2x_1^2 + 3x_2^2 - 0.8\sin(2\pi x_1) - 1.2\cos(3\pi x_2), [-1.024, 1.024]$$

上述三个函数除第三个求全局最小值外, 其他两个都是求全局最大值。3 个函数都是多峰值, 都有可能出现陷入局部极值点, 出现早熟现象, 收敛速度慢等问题。采用传统的优化方法来求解, 有些函数很难找到全局最优解, 但采用文中提出的改进的 GA 却几乎都能找到全局最优解。三个函数主要用于测试算法克服早熟现象的能力以及算法的收敛速度。三个测试函数的理论最优解为: $\max f_1(0, 0) = 1$; $\max f_2(-2.048\ 0, -2.048\ 0) = 3\ 905.926\ 2$; $\min f_3(0.221\ 8, -0.000\ 5) = -1.889\ 1$ 。

为了对比实验结果, 文中采用四种算法: 简单的 GA、文献[9]中的算法(定义为 LGA)、文献[10]中的算法(定义为 GGA)、文中提出的算法(定义为 LGGA)。对以上三个测试函数分别从搜索到的函数的最优解、收敛代数、收敛率, 以及种群平均适应度、标准差几个方面进行比较。初始参数设置。种群大小: $\text{pop-size} = 1\ 000$, 最大进化代数: $\text{maxgens} = 300$, 变异概率: $P_M = 0.1$, 交叉概率: $P_X = 0.8$ 。实验结果如表 1 ~ 表 3, 以及图 1 ~ 图 3 所示。

实验结果分析: 从表 1、表 2、表 3 综合分析, 文中提出的算法无论从全局最优解、收敛速度, 以及收敛率

表 1 f_1 测试实验结果

函数	算法	收敛代数	所得最优解	误差	收敛率/%
f_1	GA	95	$\max f_1(1.68, -2.7) = 0.997\ 5$	0.002 5	99.75
	LGA	45	$\max f_1(2.920, -1.160) = 0.999\ 0$	0.001 0	99.9
	GGA	46	$\max f_1(-0.010\ 6, 0.003\ 8) = 0.999\ 9$	0.000 1	99.99
	LGGA	51	$\max f_1(0.000\ 0, 0.002\ 6) = 1.000\ 0$	0.000 0	100

表 2 f_2 测试实验结果

函数	算法	收敛代数	所得最优解	误差	收敛率/%
f_2	GA	160	$\max f_2(-2.035\ 7, -1.986\ 6) = 3\ 767.743\ 4$	138.182 8	96.46
	LGA	51	$\max f_2(-2.039\ 8, -2.039\ 8) = 3\ 854.015\ 1$	51.911 1	98.67
	GGA	53	$\max f_2(-2.035\ 7, -2.025\ 7) = 3\ 818.133\ 7$	87.792 5	97.75
	LGGA	49	$\max f_2(-2.048\ 0, -2.048\ 0) = 3\ 905.926\ 2$	0.000 0	100

表 3 f_3 测试实验结果

函数	算法	收敛代数	所得最优解	误差	收敛率/%
f_3	GA	103	$\min f_3(0.221\ 2, -0.014\ 3) = -1.877\ 5$	0.011 6	99.39
	LGA	49	$\min f_3(0.210\ 9, -0.004\ 1) = -1.886\ 1$	0.003	99.84
	GGA	54	$\min f_3(0.221\ 2, -0.004\ 1) = -1.888\ 1$	0.001 0	99.94
	LGGA	48	$\min f_3(0.221\ 8, -0.002\ 1) = -1.888\ 8$	0.000 3	99.98

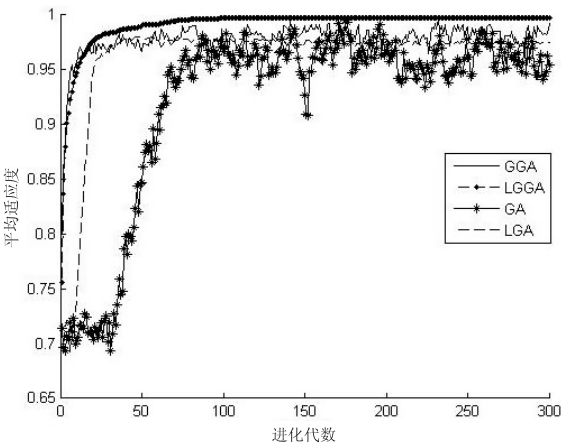


图 1 f_1 的平均适应度图

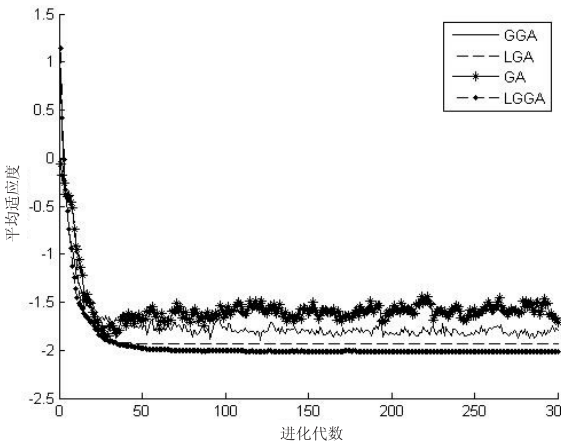


图 2 f_3 的平均适应度图

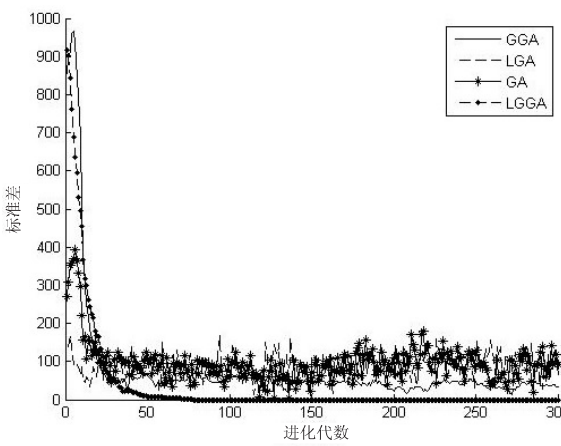


图 3 f_2 的标准差变化图

都明显优于其他三种算法。但由于篇幅有限, 在此只展示 f_1 、 f_3 的平均适应度图, 对应图 1、图 2, 从两个测试函数的平均适应度变化的四种算法对比图而知, 四

种算法都会达到一个最优解,只是大小不同,求最大值的,在可行域范围内,越大越好。相反,求最小值的,越小越好。对于收敛速度,代数越少,说明收敛速度越快。文中的算法收敛速度较其他三种快,且最值较其他三种更优且更接近理论值。其中,LGA、GGA 又比 GA 的收敛速度快,且搜索到的最值更优。标准差能反映一个数据集的离散程度,离散度是评价方法的好坏的最重要也是最基本的指标。在此,标准差代表的是此代的种群个体间的离散程度,从图 3 中 f_2 函数的标准差变化图来看,LGGA 比 GGA 变化平稳,而 GGA 比 LGA、GA 较平稳得多。总体来说,文中提出的算法是可行有效的。

3 结束语

虽然 GA 能寻找到问题的最优解,但它仍然有一些不尽如人意的地方,最主要的是早熟现象以及收敛速度慢的问题。为了避免这种缺陷,文中针对交叉算子与变异算子在整个 GA 中的特点,对其加以改进,然后应用于 3 个多峰值函数优化。结果显示,LGGA 的寻优能力优于其他三种算法。LGGA 的全局搜索能力有所提高,减轻了早熟现象,收敛速度也加快了,表现出了较强的寻优性能。

参考文献:

[1] Dejong K A. An analysis of the behavior of a class of genetic adaptive systems [D]. Michigan: University of Michigan, 1975.

[2] Holland J H. Adaptation in natural and artificial systems[M].

(上接第 79 页)

参考文献:

[1] Kenned J,Eberhart R. Particle swarm optimization[C]//Proc of IEEE international conference on neural networks. Perth, Australia;[s. n.],1995:1942-1948.

[2] Eberhart R,Kennedy J. A new optimizer using particle swarm theory[C]//Proc of the 16th international symposium on micromachine and human science. Nagoya, Japan; IEEE Press, 1995:39-43.

[3] Shi Yuhui,Eberhart R C. A modified particle swarm optimizer [C]//Proc of IEEE international conference on evolutionary computation. Piscataway, USA; IEEE Press,1998:69-73.

[4] Higashi N,Iba H. Particle swarm optimization with Gaussian mutation[C]//Proc of the IEEE swarm intelligence symposium. Indianapolis, Indiana, USA; IEEE Press,2003:72-79.

[5] 胡建秀,曾建潮. 具有随机惯性权重的 PSO 算法[J]. 计算机仿真,2006,23(8):164-167.

[6] Eusuff M M,Lansey K,Pasha F. Shuffled frog leaping algorithm; A memetic meta- heuristic for combinatorial optimiza-

Ann Arbor:The University of Michigan Press,1975.

[3] 刘红,韦穗. 遗传算子的分析[J]. 计算机技术与发展, 2006,16(10):80-82.

[4] Ichikawa Y,Ishii Y. Retaining diversity of genetic algorithms for multivariable optimization and neural network learning [C]//Proc of IEEE international conference on neural network. [s. l.]:[s. n.],1993:1110-1114.

[5] Srinivas M,Patnaik L M. Adaptive probabilities of crossover and mutation in genetic algorithm[J]. IEEE trans on systems, man and cybernetics,1994,24(4):656-667.

[6] 杨启文,蒋静坪,张国宏. 遗传算法优化速度的改进[J]. 软件学报,2001,12(2):270-275.

[7] 张晋,李冬黎,李平. 遗传算法编码机制的比较研究 [J]. 中国矿业大学学报,2002,31(6):637-640.

[8] 陈国龙,陈火旺,郭文忠,等. 基于随机错位算术交叉的遗传算法及其应用[J]. 模式识别与人工智能,2004,17(2): 250-256.

[9] Deep K,Thakur M. A new crossover operator for real coded genetic algorithms[J]. Applied mathematics and computation, 2007,188(1):895-911.

[10] 李小宁. 关于混合遗传算法改进的研究[D]. 西安:西北大学,2008.

[11] 金芬,孙春华,钟鸣. 遗传算法中适应度函数的改进术 [J]. 机械设计与制造,2010(3):218-219.

[12] 杨平,郑金华. 遗传选择算子的比较与研究[J]. 计算机工程与应用,2007,43(15):59-62.

[13] 何琳,王科俊,李国斌,等. 最优保留遗传算法及其收敛性分析[J]. 控制与决策,2000,15(1):63-66.

[14] 王小平,曹立明. 遗传算法理论、应用与软件实现[M]. 西安:西安交通大学出版社,2002.

tion[M]. London;J. Heuristics,2000.

[7] Zhen Ziyang,Wang Daobo,Liu Yuanyuan. Improved shuffled frog leaping algorithm for continuous optimization problem [C]//Proc of 2009 IEEE congress on evolutionary computation. Trondheim,Norway;[s. n.],2009.

[8] Zhang Xuncai,Hu Xuemei,Cui Guangzhao,et al. An improved shuffled frog leaping algorithm with cognitive behavior[C]// Proceedings of the 7th world congress on intelligent control and automation. Chongqing, China;[s. n.],2008.

[9] 赵鹏军,刘三阳. 求解复杂函数优化问题的混合蛙跳算法 [J]. 计算机应用研究,2009,26(7):2435-2437.

[10] 赵守法. 蛙跳算法的研究与应用[D]. 上海:华东师范大学,2008.

[11] Emad E,Tarek H,Donald G. A modified shuffled frog-leaping optimization algorithm: Applications to project management [J]. Structure and Infrastructure Engineering,2007,3(1):53-60.

[12] 罗雪晖,杨焱,李霞. 改进混合蛙跳算法求解旅行商问题[J]. 通信学报,2009,30(7):130-135.

一种基于交叉和变异算子改进的遗传算法研究

作者：谢燕丽, 许青林, 姜文超, XIE Yan-li, XU Qing-lin, JIANG Wen-chao

作者单位：谢燕丽, XIE Yan-li (广东工业大学 信息工程学院, 广东 广州, 510006), 许青林, 姜文超, XU Qing-lin, JIANG Wen-chao (广东工业大学 计算机学院, 广东 广州, 510006)

刊名：计算机技术与发展

ISTIC

英文刊名：Computer Technology and Development

年, 卷(期): 2014(4)

本文链接: http://d.wanfangdata.com.cn/Periodical_wjfz201404020.aspx