

一种基于景观特征的浮点数编码遗传算法研究^{*}

崔明义

(河南财经学院计算机科学系 郑州 450002)

摘要 遗传算法作为一种适应性搜索技术得到了普遍的应用,但其搜索效率不如启发式搜索。已有研究者将启发式知识用于二进制编码遗传算法,但浮点数编码在函数优化和约束优化领域明显有效于其它编码。本文基于算法运行时的景观特征作为启发式知识,用于浮点数编码遗传算法,力求提高其搜索效率、增强其局部搜索能力、拓展其应用领域。本文的理论研究和实验结果表明,将景观特征用于浮点数编码遗传算法,理论是可靠的,方法是可行的。

关键词 景观特征,浮点数编码,启发式知识,遗传算法

Research on Genetic Algorithm of Float Number Code Based on Landscape Character

CUI Ming-Yi

(Dept. of Computer Science, Henan University of Finance & Economics, Zhengzhou 450002)

Abstract Genetic algorithm (GA) was used widely as a adapted search technology. GA is not as good as heuristic search in search efficiency. Binary code GA was researched by scholars with heuristic knowledge at past. Float number code is superior to other codes in function optimization and restriction optimization. In this paper, landscape character in algorithm running was used to GA of float number code as heuristic knowledge. It was aimed by it that improving GA's search efficiency, enhancing local search performance, extending application fields. It was indicated by the results of it's theoretic research and experiment result that the theory is credible, the method is feasible, landscape character could be used to GA of float number code.

Keywords Landscape character, Float number code, Heuristic knowledge, Genetic algorithm

1 引言

遗传算法(Genetic Algorithm, GA)作为一种自适应搜索技术,得到了广泛的关注和发展。由于它是基于概率搜索,与启发式搜索相比,其搜索效率没有明显优势,局部搜索能力也不高。这是遗传算法理论和应用研究的重要课题之一。长期以来,许多学者都对其进行了研究和探讨,取得了一些令人瞩目的成果。将启发式知识融入到遗传算法中,以提高其运行的效率,改善其局部搜索能力,就是一种人们较常采用的方法。1996年,Back^[1]提出了进化算法(Evolutionary Algorithm, EA)中选择较高适应度个体参与交叉和变异运算,以提高算法的搜索效率;1998年,Munteanu和Lazarescu^[2]提出了适应蓄水池遗传算法,改进了算法的全局搜索性能;1999年,Corne和Dorigo^[3]等提出了混合算法(Hybrid Algorithms, HA),用不同算法的启发式知识改进遗传算法的局部搜索能力;2001年,Ribeiro和Hansen^[4]提出了元启发式算法(MetaHeuristic Algorithms, MH),利用适应性原理引导搜索活动;2004年,Munteanu和Rosa^[5]提出了适应蓄水池进化算法(Adaptive Reservoir Evolutionary Algorithm, AREA),基于算法的实时运行和寻找较优解的估计势能进行启发,以改进基本遗传算法的盲搜索特性,提高其搜索能力。

上述研究成果,都从不同角度对进化计算和算法注入启发式知识,一定程度地改善了算法的搜索能力,提高了算法的运行效率,为遗传算法的理论和应用研究做出了贡献。迄今为止,大多数研究都是对二进制编码的遗传算法进行启发,力求提高其搜索性能,对浮点数编码遗传算法的启发式搜索研

究却少之又少。相比之下,浮点数编码具有精度高、便于多维大空间搜索的优点,越来越受到学者的重视。浮点数编码在函数优化和约束优化领域明显有效于其它编码,这已被得到了广泛的验证^[6~9]。本文基于遗传算法运行过程中的景观特征进行调整,对浮点数编码予以启发,以改进浮点数编码遗传算法的搜索特性,提高其搜索性能。

2 算法原理和描述

为阐述方便,在后面的叙述中,除特别说明以外,均以“算法”表示本文描述的算法。

2.1 基本原理

定义1 当算法在搜索空间的某区域内搜索到一个新的最优解(相对于搜索开始后的搜索过程),而更好的最优解尚未搜索时,称这种景观为景观A。

定义2 当算法已搜索到了搜索空间的某区域的最优解,若再在该区域搜索时,已不可能有更优解出现,该区域的搜索复杂度较低,需转移到搜索空间的另一区域搜索,称这种景观为景观B。

定义3 当景观A出现后,算法需在该区域继续搜索,直到搜索到更好的最优解或者景观B出现,称这种景观为景观C。

为了便于实现算法,需要设置两个指针:一个是最优解指针 BestSolution,记录搜索过程的最优解值;一个是搜索复杂度指针 Complexity,作为复杂度的度量记录搜索指数。还要构造一个虚拟小生境,聚集景观A出现后景观C时的子群体。当 BestSolution 有新值时,景观A出现,虚拟小生境开始

^{*} 本课题受到河南省高校杰出科研人才创新工程项目(2004KYCX014)和河南省自然科学基金(0411014500)的资助。崔明义 教授,博士,研究方向为计算智能、软件和网络安全。

生成,在该小生境进行有限的连续变异,直到景观 A 再次出现或虚拟小生境规模大于群体规模。当 Complexity 值变小,搜索复杂度降低,景观 B 出现。在景观 A,只进行选择和交叉操作;在景观 B,只在虚拟小生境中进行变异操作。

浮点数编码遗传算法的运行过程,是上述三种景观不断出现和消失的过程。在整个运行过程中,算法应时刻保持搜索空间的三种景观的适度平衡,将搜索的实时景观状态特征作为启发式知识,应用到搜索的动态过程,建立算法的动态适应机制,使之能够应对复杂的、高维的多峰搜索状态,是本文研究的基本原理。

定理 1 如果景观 A、B、C 存在,那么在有限次迭代后,算法将以概率 1 收敛于全局最优解(不计初始化)。

证明:设 $X^* = \{x \in X : f(x) = f^*\}$ 是全局最优解的集合。由于景观 B 存在有通过连续变异可遍历的从任意 $x \notin X^*$ 到某一 $x^* \in X^*$ 的有限路径,设 k_x 是 $x \notin X^*$ 到集合 X^* 的最短路径长度, $k^* = \max\{k_x : x \notin X^*\}$ 。

考虑某群体的一个任意双亲 x ,景观 A 保证这个双亲参与交叉运算且至少以概率 δ_c ($\delta_c > 0$) 不使该双亲发生变化。景观 B 保证新子代个体转换到 x^* 最短路径下一个点的概率至少为 δ_m ($\delta_m > 0$)。由于景观 C,新子代个体将至少以概率 δ_s ($\delta_s > 0$) 进行选择并继续生存,因而,双亲 x 进化到表示成 x^* 最短路径下一个点的双亲的概率至少为 $\delta_c \cdot \delta_m \cdot \delta_s > 0$ 。经过 k_x 次转换,在第 k_x 次迭代,从 $x \notin X^*$ 到集合 X^* 的转换概率至少为 $(\delta_c \cdot \delta_m \cdot \delta_s)^{k_x-1} \cdot \delta_c \cdot \delta_m > 0$ 。因此,可以推断在 k^* 次迭代之后,算法搜索到全局最优解的概率至少为 $\delta = (\delta_c \cdot \delta_m \cdot \delta_s)^{k^*-1} \cdot \delta_c \cdot \delta_m > 0$ 。从而,在 k 次迭代之后不能搜索到一个全局最优解的概率至多为 $(1-\delta)^{\lfloor k/k^* \rfloor}$,且随着 $k \rightarrow \infty$ 指数级地快速收敛于零。这就意味着 $P\{T < \infty\}$,即在有限次迭代之后,算法将以概率 1 第一次收敛于全局最优解。

证毕。

定理 2 如果景观 A、B、C 存在,在有限次迭代后,算法将无限地搜索到全局最优解。

证明:定理 1 的证明已经表明,景观 A、B、C 足以在有限次搜索后,以概率 1 收敛于全局最优解。假设在第 k_1 步第一次搜索到全局最优解,在第 $k_2 > k_1$ 步群体释放了所有最优个体,那么,同样的假设可以保证经过有限次迭代后,在第 $k_3 > k_2$ 步算法将以概率 1 搜索到全局最优解,如此反复。证毕。

上述两个定理,从理论上保证了在景观 A、B、C 下,算法对全局最优解的收敛性。

2.2 虚拟小生境机理

算法的虚拟小生境反映了景观的状态。在算法搜索过程中,它始终保持着子群体的多样性。与其他进化算法在整个群体中进行变异运算不同,算法的变异运算只在虚拟小生境中进行,随着虚拟小生境的不断调整,景观 A、B、C 也在不断转换。当景观 B 出现时,虚拟小生境的规模增大,群体的多样性增加,在一个更好小生境内重新开始了搜索;当景观 C 出现时,虚拟小生境的规模保持常量;当景观 A 出现时,虚拟小生境重新初始化,一个新的小生境诞生并开始发育。

虚拟小生境规模的调整依据两个指针 BestSolution 和 Complexity 的值。若指针 BestSolution 有新值,说明算法搜索到新的最优解,找到了更好的峰值,虚拟小生境初始化,景观 A 出现;若指针 Complexity 的值变小,搜索复杂度降低,虚拟小生境的规模增大,子群体的多样性增加,景观 A 出现。

当出现三种情况之一时,虚拟小生境初始化。这三种情况是:(1)算法开始运行;(2)搜索到了新的最优解;(3)虚拟小

生境规模大于等于群体规模。在这三种情况之外,如果没有搜索到新的最优解,虚拟小生境规模进行调整。设中间变量 δ 表示虚拟小生境规模调整量,初始化为 $\delta(0) = \Delta(0)$,则

$$\Delta(t) = \begin{cases} \delta(t), & \text{有 } \delta(t) = \lceil \delta(t-1) + c \rceil & \text{如果 } [\delta(t-1) + c] = \\ & \lceil \delta(t-1) + c \rceil \\ \Delta(t-1), & \text{有 } \delta(t) = \delta(t-1) + c & \text{其它} \end{cases} \quad (1)$$

式中,参数 $c > 0$,它决定了规模扩大的速度。

如果从上一代到当前没有搜索到更好的适应度,虚拟小生境规模依据下式保持常量。

$$\Delta(t) = \begin{cases} \delta(t), & \text{有 } \delta(t) = \lceil \delta(t-1) \rceil & \text{如果 } [\delta(t-1)] \\ & = \lceil \delta(t-1) \rceil \\ \Delta(t-1), & \text{有 } \delta(t) = \delta(t-1) & \text{其它} \end{cases} \quad (2)$$

2.3 虚拟小生境的选择

为了增加虚拟小生境群体的多样性和提高运行效率,算法采用两种选择机制组成小生境:

(1)由于虚拟小生境只进行变异运算,为了不使因变异运算破坏掉最优个体,需从群体中选择一些最坏个体,以增加群体的多样性。

(2)将 k 精英计划和锦标赛法结合起来:在变异后,从当前群体中选择 k 个最好个体,然后每次从 2 个个体中选择一个最好的个体,这种操作重复 M 次,得到 M 个个体,其操作的时间复杂度为 $O(M)$ 。因为这是一个常量,使算法的选择压力很低。其优点是在算法运行时依靠这种选择特性提高运行效率。

2.4 算法描述

根据上述思想,算法的描述如下:

Step 1: $t \leftarrow 0$;

Step 2: 在特定范围内随机初始化浮点数编码群体 $P(t)$;

Step 3: 初始化虚拟小生境规模 M ;

Step 4: 计算群体 $P(t)$ 中个体适应度;

Step 5: 从群体 $P(t)$ 中选择双亲 $P'(t)$;

Step 6: 使用 $P(t)$ 估计搜索是否在期望区域;

Step 7: 用估计值和指针 BestSolution 值调整虚拟小生境,决定即时景观;

Step 8: 虚拟小生境内变异

Step 8.1 选择虚拟小生境 $\rho(t)$: 从 $P'(t)$ 中选择较小适应度的个体 $\Delta(t)$;

Step 8.2 对 $\Delta(t)$ 的个体随机变异;

Step 8.3 将变异后的小生境替换变异前的 $\rho(t)$, 并加入到 $P'(t)$ 中;

Step 9: 在 $P'(t)$ 中以概率 p_c 交叉运算;

Step 10: 形成新的群体 $P(t+1)$: 将 k 精英计划用于 $P'(t)$;

Step 11: 计算群体 $P(t+1)$ 的个体适应度;

Step 12: 判断是否满足终止条件: 若满足, 终止; 否则, $t \leftarrow t+1$, 转 Step 5。

3 应用实例

为了验证算法的实用性和可行性,本文取三个著名的遗传算法测试函数 F6、F7、F8 进行了实验。这三个函数如下:

(1) F6 函数 (Schaffer1 function)

$$\min f(x) = (x_1^2 + x_2^2)^{0.25} [\sin^2(50 \times x_1^2 + x_2^2)^{0.1} + 1.0] \quad (3)$$

s. t. $x_i \in [-10, 10]$

(2)F7 函数(Schaffer2 function)

$$\min f(x)=0.5+\frac{\sin \sqrt{x_1^2+x_2^2}-0.5}{\left[1+\alpha \times\left(x_1^2+x_2^2\right)\right]^2} \quad (4)$$

$$\text { s. t. } \alpha \in\{0.001,0.01,0.1,1,1.0\}$$

$$x_i \in[-10,10]$$

(3)F8 函数(Rastrigin's function)

$$\min f(x)=10 \times n+\sum_{i=1}^n\left[x_i^2-10 \times \cos \left(2 \pi x_i\right)\right] \quad (5)$$

$$\text { s. t. } x_i \in[-5.12,5.12]$$

在实验中,进化代数均为 100 代,根据测试函数的复杂程度,函数 F6、F7、F8 的群体规模分别为 60、60、80。三个函数的理论最优值、最优值和算法的实验结果如表 1 所示。函数 F6、F7、F8 的算法最优解、平均解和最佳个体分别如图 1、图 2、图 3 所示。

表 1 三个函数的理论最优值、最优值和算法的实验结果比较

测试函数	F6	F7($\alpha=0.01$)	F8($n=10$)
函数理论最优值	0	0	0
函数算法仿真值	0.034837	0.020754	0.020536
平均适应度	0.11251	0.026125	0.46594
理论最优值	(0,0)	(0,0)	(0,0,0,0,0,0,0,0,0,0)
算法最优解	(0.00087, -0.00082)	(0.0001, -0.00018)	(0.00303, -0.00262, -0.0007, 0.00234, -0.00017, -0.00415, -0.00309, 0.0008 -0.00735, -0.00013)

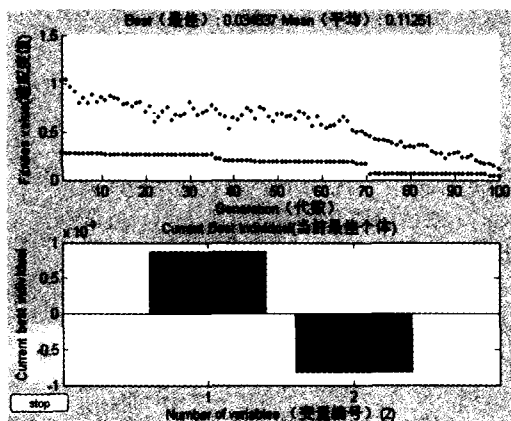


图 1 函数 F6 的最优解、平均解和最佳个体

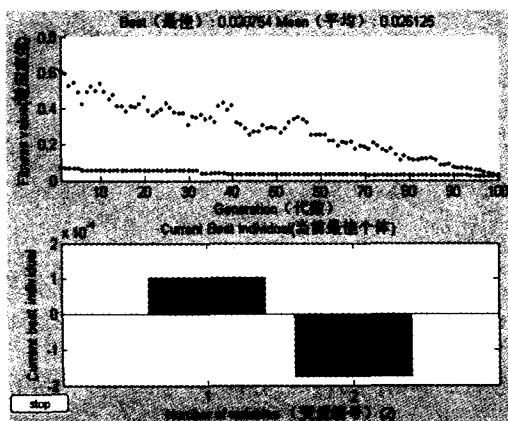


图 2 函数 F7 的最优解、平均解和最佳个体

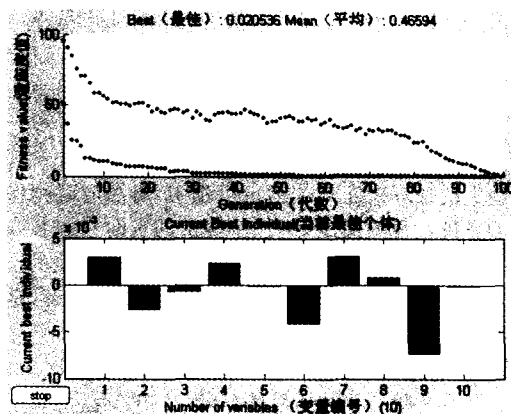


图 3 函数 F8 的最优解、平均解和最佳个体

实验结果表明,三个函数的算法仿真值均接近于其理论最优值,算法最优解也近似于理论最优值,其误差是由于浮点数本身的误差和算法的近似搜索性能所引起的。结果显示,算法具有较广泛的实用性、较强的可行性和较高的可靠性,完全可以满足经常的工程优化需要。

结束语 遗传算法以其较强的全局搜索能力得到了普遍的应用。但其搜索效率尤其是复杂优化的搜索效率不高、局部搜索能力不强、二进制个体编码的局限性、容易产生欺骗性“早熟”等是制约其推广应用的主要因素,也是遗传算法理论和应用研究的热点问题。而采用浮点数编码、增加启发式知识是解决上述问题的重要途径。本文将算法运行时的景观特征作为启发式知识,用于浮点数编码的遗传算法中,从理论和实践上进行了探索,取得了一些经验,力图为遗传算法的理论和应用研究增加一些积累,使遗传算法能够更有效地得到推广和使用。有理由相信,经过众多研究者的努力,遗传算法的理论和应用研究必将会取得更大的进展。

参考文献

- 1 Bäck T. Evolutionary Algorithms in Theory and Practice. New York: Oxford University Press, 1996
- 2 Munteanu C, Lazarescu V. A New Approach to Global Search Using a Genetic Framework: The Adaptive Reservoir Genetic Algorithm. In: Proceedings of MENDEL'98, 1998, 74~79
- 3 Corne D, Dorigo M, Glover F. New Ideas in Optimization. London: McGrawHill, 1999
- 4 Ribeiro C, Hansen P. Essays and Surveys in Metaheuristics. Kluwer Academic Publishers, 2001
- 5 Munteanu C, Rosa A C. Adaptive Reservoir Evolutionary Algorithm: An Evolutionary On-Line Adaptation Scheme for Global Function Optimization. Journal of Heuristics, 2004, 10: 555~586
- 6 Eshelman L, Schaffer J. Real-Coded Genetic Algorithms and Interval Schemata. In: Whitley L, ed. Foundations of Genetic Algorithms, Morgan Kaufmann Publishers. San Francisco, 1993, 2: 187~202
- 7 McCormick W T, Schweitzer P J, White T W. Problem Decomposition and Data Reorganization by a Cluster Technique. Operations Research, 1972, 20(5): 993~1009
- 8 Michalewicz Z. Genetic Algorithm + Data Structure = Evolution Programs, 3rd edition. Springer-Verlag, New York, 1996
- 9 Walters G A, Smith D K. Evolutionary Design Algorithm for Optimal Layout of Tree Networks. Engineering Optimization, 1995, 24: 261~281