

# 遗传算法的混合算子策略

张应辉 曾庆华 王志伟

(成都东软信息技术学院 成都 611844)

**摘要** 在一般遗传算法中,求最优解时既可避免早熟收敛,又能提高收敛速度是困难的,因为算法中使用了单独一组交叉算子/变异算子。本文提出一种新的基于混合算子的遗传算法执行策略。在求解旅行商问题(TSP)中,为了提高局部搜索能力和收敛速度,给出了一种基于边重组的启发式交叉算子。仿真实验表明了这种算法的有效性。

**关键词** 遗传算法,遗传算子,全局优化,早熟收敛,旅行商问题(TSP)

## The Genetic Algorithm Based on Mixed Genetic Operators

ZHANG Ying-Hui WANG Zhi-Wei ZENG Qing-Hua

(Chengdu Neusoft Institute of Information, Chengdu 611844)

**Abstract** In a general Genetic Algorithms (GA), it is difficulty to avoid prematurely convergence and raise the speed of the algorithm convergence for complex finding the optimal solution, in which the algorithm is run with a single set of crossover/mutation operators. In this paper, a new run-strategy of Genetic Algorithms based on mixed genetic operators is presented, a heuristic crossover operator based on the edge recombination is also given to raise the ability of the local searching and the speed of convergence in solving the Traveling Salesman Problems (TSP). The efficiency of the algorithm has been shown by simulative experiments.

**Keywords** Genetic algorithms, Genetic operators, Global optimization, Prematurely convergence, Traveling Salesman Problems (TSP)

## 1 引言

遗传算法(GA)是一种模仿生命与智能的产生与进化过程的仿生算法,也是一种随机优化与搜索方法,适合于大规模、高度非线性的不连续多峰函数的优化以及无解析表达式的目标函数的优化,特别是能非常有效地解决组合优化问题。

在遗传算法中,交叉算子和变异算子是决定遗传算法寻优能力的最主要因素。目前的理论和实验证明,没有一种交叉算子和变异算子能够适合所有的优化问题,正如自然界各种生物各有各的遗传机制一样。特别是在组合优化中,收敛速度和全局搜索能力往往是矛盾的,有些遗传算子收敛速度很快,但很容易陷入局部优化而产生“早熟收敛”现象<sup>[1]</sup>,即在找到全局最优解或近优解之前,遗传算法的寻优过程可能导致其主导基因固定在一种染色体基因上。而有些算子的局部搜索能力较差,解空间太大,适应值分布结构不合理,导致收敛速度慢。

观察和研究自然界生物进化过程,可以发现以下现象:

- 1)相同的原始基因在不同的生存环境进化出不同的物种;
- 2)往往由多种进化机制产生的物种具有更高级的生命形式或智能;
- 3)环境对生物进化有着相当重要的影响。

为提高遗传算法性能,避免出现“早熟收敛”现象,提高其局部搜索能力,人们进行了许多研究和探索,包括遗传算子的研究<sup>[1,3]</sup>、交叉概率和变异概率的研究<sup>[2,3]</sup>等。

本文以 TSP 问题为例,通过对常用遗传算子的分析,提出了一种基于混合遗传进化机制的混合算子进化策略,即充分利用不同遗传算子各自的遗传进化特性,在遗传进化的不同阶段采取不同的遗传算子进行遗传进化,以提高算法收敛速度,增强算法的全局搜索能力,避免“早熟收敛”现象。本文也提出了一种具有较强局部搜索能力和快速收敛能力的遗传

算子 HEX,但该算子较易陷入早熟收敛,它与传统算子混合使用,能较好地提高寻优性能。仿真试验也证明了利用混合遗传算子的遗传算法的有效性。

## 2 TSP 遗传算法常用遗传算子分析

旅行商问题(TSP)是近代组合优化的一个典型难题。它可简单描述为,寻找一条遍历  $n$  个结点的最短路径,或者说搜索整数子集  $X = \{1, 2, \dots, n\}$  ( $X$  的元素表示对  $n$  个结点的编号)的一个排列  $R = (v_1 v_2 \dots v_n)$  (称为一条遍历路径),使

$$\sum_{i=1}^{n-1} d(v_i, v_{i+1}) + d(v_n, v_1)$$

取最小值。其中  $d(v_i, v_j)$  为结点  $v_i$  到结点  $v_j$  的距离。

用遗传算法求解 TSP 问题时,对最直观的路径表示编码方法,主要的遗传交叉算子有<sup>[4,5]</sup>:

1)部分映射交叉(Partially Mapped Crossover, PMX)。交叉过程由两步来完成,首先对父代进行常规的双点交叉,再根据交叉区域内各个基因值之间的映射关系来修改交叉区域之外的各个基因座的基因值。这种交叉算子保留了部分结点的绝对访问顺序,但更多地产生了父代遍历路径所没有的部分新路径,其收敛速度较低,但全局搜索能力较强。

2)顺序交叉(Order Crossover, OX)。其交叉过程是:先进行常规的双点交叉,再对个体遍历路径进行有效顺序修改,修改时尽量维持各结点原有的相对遍历顺序。这种交叉算子保留了部分结点的相对访问顺序,但同 PMX 一样,也更多地产生了父代遍历路径所没有的部分新路径,其收敛速度也较低。

3)循环交叉(Cycle Crossover, CX)。其交叉的基本思想是:任意两条遍历路径都可能组成一些互不相交的循环路径,相互交换其中一个循环路径的基因就有可能产生出新的遍历路径。

4)边重组交叉(Edge Recombination Crossover, EX)。其

主要思想是强调对父代遍历路径上边之间的邻接关系的遗传。

主要的遗传变异算子有<sup>[4,5]</sup>:

1) 双点逆转变异——2Rev。随机选择个体中结点  $a_i$  和  $a_j (i < j < 1)$ , 将  $a_{i+1} \dots a_j$  逆转后代替原来的路径段, 即:

$$(a_1 \dots a_i \underline{a_{i+1} a_{i+2} \dots a_j a_{j+1}} \dots a_n) \\ \downarrow 2Rev$$

$$(a_1 \dots a_i \underline{a_j a_{j-1} \dots a_{i+1} a_{i+2}} \dots a_n)$$

2) 对换变异——Swap。随机选择个体中结点  $a_i$  和  $a_j (i < j)$ , 交换  $a_i$  和  $a_j$ , 即

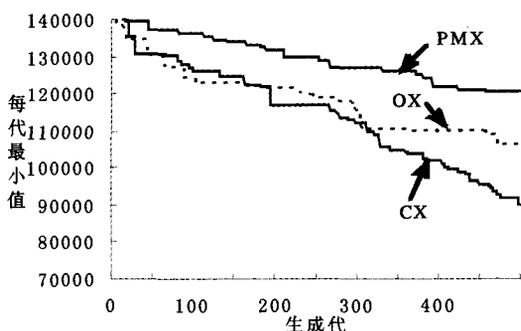
$$(a_1 \dots a_{i-1} \underline{a_i a_{i+1}} \dots a_{j-1} \underline{a_j a_{j+1}} \dots a_n) \\ \downarrow Swap$$

$$(a_1 \dots a_{i-1} \underline{a_j a_{j+1}} \dots a_{i-1} \underline{a_i a_{i+1}} \dots a_n)$$

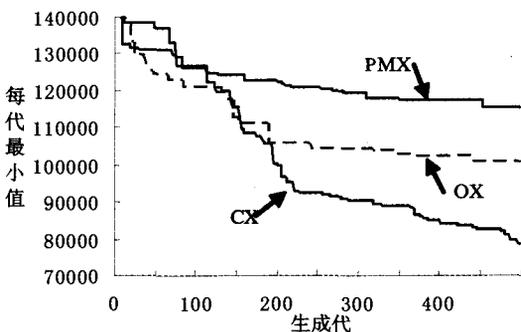
3) 插入变异——Ins。随机选择个体中结点  $a_i$  和  $a_j (a_j$  不是  $a_i$  的直接后续结点), 将  $a_j$  插入  $a_i$  和  $a_{i+1}$  之间, 即:

$$(a_1 \dots \underline{a_i a_{i+1}} \dots a_{j-1} \underline{a_j a_{j+1}} \dots a_n) \\ \downarrow Ins$$

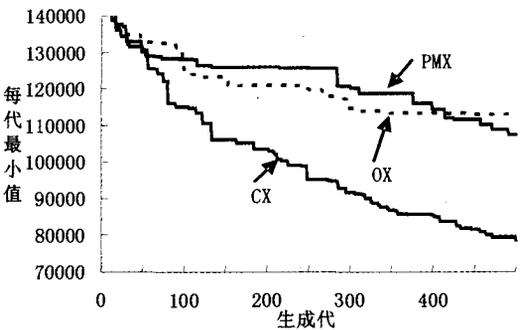
$$(a_1 \dots \underline{a_i a_j a_{i+1}} \dots a_{j-1} \underline{a_j a_{j+1}} \dots a_n)$$



(a) 2Rev 变异



(b) Ins 变异



(c) Swap 变异

图 1

图 1(a)~图 1(c) 分别为利用 PMX, OX, CX 交叉算子和 3 个变异算子, 采用精英选择基本遗传算法对 kroA 100 个城市的 TSP 求解的收敛过程。其中种群大小 1000,  $p_c = 0.8$ ,

$p_m = 0.1$ , 适应值函数采用遍历路径长度的倒数。从图中可看出:

1) 影响收敛速度的主要因素是交叉算子, 变异算子的作用相对较小。

2) 局部搜索能力相对较强的算子 (如 CX) 其收敛速度高于全局搜索能力较强的算子 (如 PMX 和 OX)。

3) PMX, OX, CX 算子主要考虑的是结点的位置和顺序, 未考虑边的信息 (而这是 TSP 问题的另一个重要信息), 不能将遍历路径上的重要信息很好地遗传到下一代种群。

4) 边重组交叉算子能使子个体从父体中继承 95%~99% 的边的信息, 但收敛速度也不理想。

5) 这几个交叉算子的局部搜索能力均较弱, 当结点较多时, 其收敛速度均过于缓慢。这也是基本遗传算法所固有的弱点。

### 3 启发式边重组交叉算子

求解 TSP 即是求使边长总和最小的结点的排列, 所以边长的信息是 TSP 中相当重要的信息。为了提高算法收敛速度, 增强算法的局部收缩能力, 这里介绍一种基于边重组交叉算子的、具有一定学习能力的启发式边重组交叉算子 HEX, 它不但能很好地继承父代中边信息, 还能很好地继承父代中精英基因段信息。

HEX 交叉的基本思想是: 从父代遍历路径中随机选取一个结点作为子代的开始结点, 比较父代中与该结点邻接的、不在子代中的结点, 选取其邻接边长最小的结点作为下一结点。如果该结点的所有邻接点均已选入子代, 则从未选结点中选择与该结点邻接边最短的结点。如此反复, 直至形成一条完整的遍历路径。其算法描述如下:

启发式边重组交叉——HEX

设  $L_a = (a_1 a_2 \dots a_n)$ ,  $L_b = (b_1 b_2 \dots b_n)$  为两条父代遍历路径,  $L_x = (x_1 x_2 \dots x_n)$ ,  $L_y = (y_1 y_2 \dots y_n)$  为将要产生的两条子代遍历路径。  $D = \{1, 2, \dots, n\}$  为结点的集合。

① 对每个  $a_i \in D$ , 建立其在遍历路径  $L_a$  和  $L_b$  中的邻结点集合  $A_i (i=1, 2, \dots, n)$ 。

② 随机选择结点  $a_i$ , 令  $k=1, x_k = a_i, D = D - \{x_k\}$ , 对所有  $A_i (i=1, 2, \dots, n), A_i = A_i - \{x_k\}$ 。

③ 记  $x_k$  对应的邻结点集为  $A$ 。

若  $A = \emptyset$ , 则选择  $x_{k+1} \in A$  且

$$d(x_k, x_{k+1}) = \min\{d(x_k, q) | q \in A\};$$

若  $A = \emptyset$ , 则选择  $x_{k+1} \in D$  且

$$d(x_k, x_{k+1}) = \min\{d(x_k, q) | q \in D\}.$$

④ 令  $k = k + 1, D = D - \{x_k\}$ , 对所有  $A_i (i=1, 2, \dots, n), A_i = A_i - \{x_k\}$ 。

⑤ 若  $k < n$ , 则转向③求子代的下一结点; 否则已求出  $L_x$ 。

⑥ 用  $L_y$  代替  $L_x, L_a$  和  $L_b$  交换, 重复①~⑤, 求  $L_y$ 。

图 2 为用 HEX 交叉求解 kroA 100 个城市 TSP<sup>[6]</sup> 的收敛曲线。明显地, HEX 算子收敛速度相当迅速, 但很快就陷入局部最优 (用对换变异, 178 代时收敛于近优解 22362.68)。

通过对以上遗传算子的分析和大量的实验统计, 可得出如下结论:

1) 全局搜索能力较强的交叉算子 (如 PMX, OX), 由于每一代的解空间跨度较大, 往往收敛速度较慢;

2) 收敛速度较快的交叉算子 (如 HEX), 由于较早产生较

优个体而容易产生局部最优,导致“早熟收敛”。

3)变异算子中,Swap 的全局搜索能力强于 Ins,而 Ins 又强于 2Rev。

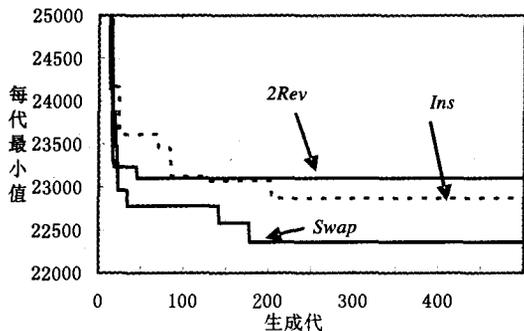


图2 HEX交叉算子

因此,如果在遗传算法中交替地使用收敛速度快的算子和全局搜索能力强的算子,一方面提高收敛速度,当一定时间内不再收敛时(这时往往陷入了局部优化或出现了“早熟收敛”),再扩大搜索空间,多样化种群中的个体类型,如此反复,便可以较快求出全局最优解或近优解。

对变异算子,由于它对算法收敛速度影响较小,但能提高全局搜索能力,在算法执行时可随机选取。

#### 4 混合算子仿真试验

在设计混合算子遗传算法时,需要考虑以下几个问题:1)算子的组合与选择;2)算子交替的周期或频率;3)算子交替的顺序。针对这些问题,以 TSP kroA 100 个城市问题的求解为例进行了大量的仿真试验。为了便于比较和筛选,在仿真试验中,采用精英选择基本遗传算法,种群大小设定为 1000,  $p_c = 0.8$ ,  $p_m = 0.1$ , 适应值函数为遍历路径长度的倒数。对变异算子的随机选择,根据它们的全局搜索能力,对各变异算子赋予如下选择概率:  $p_{2Rev} = 0.25$ ,  $p_{Ins} = 0.3$ ,  $p_{Swap} = 0.45$ 。

交叉算子采用了如下一些组合: HEX + PMX (先执行 HEX,再执行 PMX,下同), HEX + OX, HEX + CX, PMX + HEX, OX + HEX 以及 CX + HEX。

算子交替周期从 10 代起,至 100 代,每隔 10 代试验一次。

表 1 给出了一些较有价值的试验数据。在 HEX + PMX 的算子组合中,交替周期为 40 代时,在 800 代之前已求得已知最优解 21285,如图 3 所示。文[6]给出的已知最优解 21282 是将路径中各边长四舍五入取整求和所得,如将其提供的最优路径各边长按实数求和取整,其结果即为 21285。

表 1 仿真试验统计结果

首先执行的算子		HEX		CX	PMX	平均值	
其次执行的算子		CX	PMX	HEX			
500 代时 结果	交替	20	21608	21495	21591	21384	21519
	周期	30	21729	21441	21646	21469	21571
	(代)	40	21609	21458	21756	21618	21610
	平均值		21648	21464	21664	21490	21567
800 代时 结果	交替	20	21386	21307	21481	21382	21389
	周期	30	21406	21369	21418	21382	21394
	(代)	40	21520	<b>21285</b>	21443	21448	21424
	平均值		21438	21321	21447	21404	21402

表 1 中的试验结果表明:

1)算子组合: HEX 和 PMX 组合的求解能力最强,其次

为 HEX 和 CX 组合,HEX 和 OX 组合能力最差。

2)交替周期在 20~40 代之间时能取得较好结果,过于小或过于大,均会影响全局收敛速度。

3)算子的交替顺序对算法的执行结果有一定影响,但有限。HEX + PMX 组合的全局收敛速度最快,其次为 PMX + HEX,HEX + CX。CX + HEX 效果较差。

**结束语** 利用遗传算法求解 TSP 问题是遗传算法在组合优化中的重要应用。当城市点数较多时,用传统的单一遗传算子求解,收敛速度就成为主要问题,而往往收敛速度较快的算子很容易陷入局部优化,产生早熟现象。

本文提出的混合算子遗传算法执行策略,融合多种性能特征的遗传算子于同一算法中,既提高了算法收敛速度,也避免出现“早熟收敛”现象,增加了遗传进化的多样性。

本文给出的启发式边重组交叉(HEX)算子,能对求解 TSP 提供迅速收敛能力,将它和传统的 PMX、OX、CX 交叉算子进行组合操作,可提高算法的收敛速度,也能提高算法的全局搜索能力。图 4 是用混合算子遗传算法执行策略(HEX + PMX)执行 18,000 代后,求出的 rat 575 个城市的 TSP 近似最优解 6874(文[6]给出的已知最优解为 6773)。

混合算子遗传算法执行策略也可用于求解其他组合优化问题和大规模、高度非线性的不连续多峰函数的优化问题,关键是要针对具体问题,找出具有快速收敛能力的交叉算子和全局搜索能力强的交叉算子。

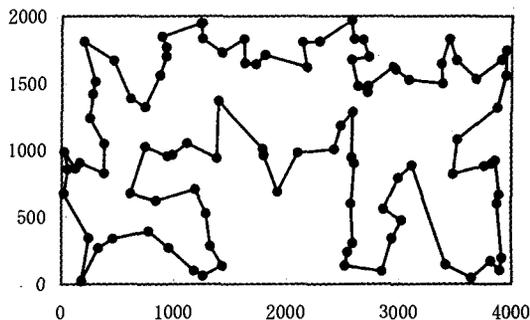


图3 kroA 100个城市的 TSP 最优解

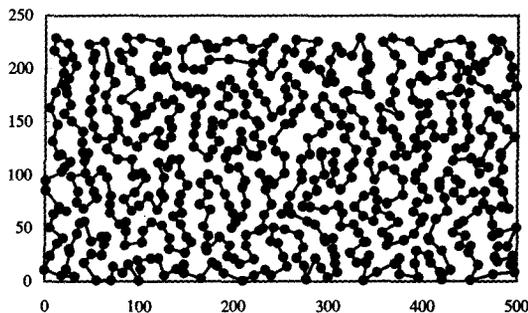


图4 Rat 575个城市的 TSP 最优解

#### 参考文献

- 1 Runwei C, Mitsuo G. Crossover on Intensive Search and Traveling Salesman Problem. In: Processing of 16<sup>th</sup> International Conference on computers & Industrial Engineering, 1994. 568~570
- 2 Srinivas M, Panik L M. Adaptive Probabilities of Crossover and Mutation in Genetic Algorithms. IEEE trans. on System, Man and Cybernetics, 1994, 24(4): 656667
- 3 张应辉, 王兴伟, 刘积仁, 李华天. 遗传算法中一种有效的自适应概率参数模型. 清华大学学报(自然科学版)1998, 38(S2): 110~113
- 4 王小平, 曹立明. 遗传算法——理论、应用与软件实现. 西安交通大学出版社, 2002
- 5 周明, 孙树栋. 遗传算法原理及应用. 国防工业出版社, 1999
- 6 TSPLIB [EB/OL]. <http://www.iwr.uni-heidelberg.de/groups/comopt/software/TSPLIB95/index.html>, 2003