引入基因簇求解 TSP 的遗传算法

马光志 卢炎生 宋恩民 汤海先
(华中科技大学计算机科学与技术学院 武汉 430074)

摘 要 在用遗传算法求解 TSP 时,极易破坏已经发现的较短线路片段,从而使遗传算法的收敛变慢。为了保护较短的线路片段,遗传操作以基因和基因簇为单位进行,优良基因簇可完整地遗传到下一代。在获得第一个近似最优解后,粉碎已发现的基因簇并继续寻优,以期能够获得全局最优解。使用 CHN144 及 TSPLIB 中的数据进行试验,找到了 CHN144 问题的当前最优路径。通过对 TSP225 的实验获得了最短路径 3859,优于目前已经公布的最短路径 3916。实验表明,基于基因簇的算法具备 3000 个城市左右的寻优能力。

关键词 旅行商问题,基因簇,遗传算法

中图法分类号 TP18 **文献标识码** A

Using Gene Clusters for TSP Solving Genetic Algorithm

MA Guang-zhi LU Yan-sheng SONG En-min TANG Hai-xian
(School of Computer Science & Technology, Huazhong University of Science and Technology, Wuhan 430074, China)

Abstract When a genetic algorithm is used to solve the TSP (Traveling Salesman Problem), shorter paths found before are accessible to be destroyed, and thus the convergence of the genetic algorithm slows down. To prevent the shorter paths from being destroyed, the gene cluster was introduced into genetic operations so that the superior gene piece can be wholly inherited to descendent off-springs. After a near optimum is found, all gene clusters are smashed and the optimum finding process is continued for global optimums. Experiments are done for CHN144 and instances in TSPLIB, in which the current optimal solution for CHN144 is found and a solution 3859 for TSP225 is found better than the best result 3916 up-to-date. Our experiments show that the gene cluster based genetic algorithm has the optimum searching ability for 3000-city-about TSPs.

Keywords Traveling salesman problem, Gene cluster, Genetic algorithm

TSP(Traveling Salesman Problem) 又称为哈密尔顿回路问题,是一个经典的组合优化问题,在涉及调度及优化的相关领域中应用广泛,许多问题都可以转化为 TSP 问题进行求解。

1 TSP 问题求解

TSP 问题可以简要地描述如下:旅行商从任意一个城市出发,环游 n 个城市且每个城市只经过一次,期望回到原地时旅行距离最短。城市之间的通路若无方向限制,则该 TSP 问题为对称的 TSP 问题。对称的 TSP 可用加权无向图 $G=(V,E,W)$ 描述,其中,$V=\{C_1,C_2,\cdots,C_n\}$ 为城市集或顶点集,$|V|=n$;E 为任意两个城市之间的通路或边集;W 为 E 对应的距离权值矩阵,$W(C_i,C_j)=W(C_j,C_i)$ 且 $W(C_i,C_i)=+\infty$。TSP 求解就是要找到 V 的一个排列 $(CP_1,CP_2,\cdots,CP_n)$,使得

$$\sum_{i=1}^{n-1}W(Cp_i,Cp_{i+1})+W(Cp_n,Cp_0) \tag{1}$$

最小。

TSP 是典型的 NP-Hard 问题,当城市数目 n 较大时,无法在多项式时间内用精确算法求解,故一般采用启发策略近似求解。近似求解主要有环路构造方法和环路改进方法,遗传算法是一种基于环路改进的算法。遗传算法基于个体表示环路的初始种群,通过杂交、变异等操作产生新个体,应用适者生存策略保留改良个体。遗传算法可以得到 TSP 的近似最优解,可以在不断进化中获得全局最优解。

基本遗传算法是一种完全随机的搜索算法,杂交操作可以在任意基因位上发生。如果没有采用合适的局部寻优策略,代表较短线路的基因片段极可能被破坏。郭涛[1]主张遗传算子以变异操作为主,变异时对基因采用 Inver-over 倒位操作,类似 LK 及其改进算法[2]采用的去交叉策略,大大加快了遗传算法的收敛速度。郭涛在适应度评估时采用了局部寻优的贪心策略,他认为局部寻优策略是近来遗传算法求解 TSP 的趋势[1]。

但是,采用贪心策略更易使遗传算法陷入近似最优解。为此,本文决定先采用贪心策略尽快得到近优解,然后粉碎构成近似最优解的基因簇或基因序列,改用轮盘赌策略不断寻

到稿日期:2008-08-22 返修日期:2008-11-06 本文受国家 863 目标导向项目(2006AA02Z347)资助。

马光志(1964-),男,副教授,主要研究领域为人工智能与知识工程,E-mail:maguangzhi@mail.hust.edu.cn;卢炎生(1949-),男,博士生导师,主要研究领域为数据库;宋恩民(1962-),男,博士生导师,主要研究领域为图像处理;汤海先(1985-),男,硕士研究生。

找更优的基因序列，以期在多次寻优中获得全局最优解。本文引入基因簇表示已经发现的没有交叉的较短线路片段，在遗传操作中特别是在杂交操作时不破坏基因簇。具有多个城市的基因簇在整体上可看作一个城市，基因簇的引入在一定程度上减少了城市数目，从而减少了遗传算法的随机搜索空间，在一定程度上加快了遗传算法的收敛速度。

为了进一步加快算法的收敛速度，为种群建立了优良个体库，用于保存优良个体和淘汰最差个体。根据淘汰系数进化若干代后，对种群中的若干最差个体进行淘汰。本文选用CHN144[3]和TSPLIB[4]中的实例进行了测试，CHN144实验获得了文献[5,6]公布的最好结果，TSP225试验获得路径长度3904.28和3859，优于目前已经公布的最好结果[2,7]。

2 基于基因簇的遗传策略

本文以路径方式表示基因序列，内部无交叉的线路可看作基因簇。基因簇至少由3个以上的基因构成，基因簇的倒序排列被视为相同的基因簇。例如，在线路(2,3,6,4)中，若城市之间的通路均不交叉，即通路(2,3)、(2,6)、(2,4)、(3,6)、(3,4)、(6,4)互不交叉，则该线路可看作基因簇2,3,6,4，或等同的基因簇4,6,3,2。两个基因簇合成更大的基因簇时，合成后的基因簇内部也不允许路径交叉。

具有交叉路径的环路肯定不是最短线路。假定图1的路径AB和路径EF交叉于G，显然有$EG+AG>AE$且$GB+GF>FB$，即有$EG+AG+GB+GF>AE+FB$，也即$AB+EF>AE+FB$。显然，使用AE和FB代替AB和EF后，得到的实线环路会更短。注意，替换前路径序列为(A,B,C,D,E,F)，替换后路径序列为(A,E,D,C,B,F)，替换前后BE之间的路径序列顺序正好相反，这就是效率极高的LK算法[2]的去交叉策略。在替换前的城市路径序列中，若选定城市A的下一城市为E，运用郭涛的inver-over可得到替换后的城市路径序列。要说明的是，这种策略只需要近似交叉，无需AB和EF真正交叉，只要满足$AB+EF>AE+FB$即可。

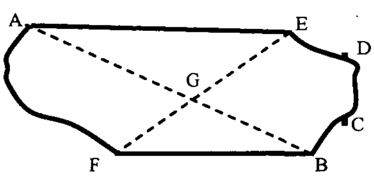


图1 TSP中交叉路径的转换

本文采用一种保基因簇的多点杂交策略，图2给出了两个个体杂交的示例。在杂交后代的第1个体中，来自父辈第1个体的基因簇保持不变，其它基因从父辈第2个体自左至右地提取，并去掉出现在父辈第1个体基因簇中的基因，剩余基因采用贪心或轮盘赌策略循环移位，以找到杂交后环路尽(极)可能短的个体。

1, 2, 3, 5, 7, 8, 6, 4, 9, 10, 11

4, 5, 3, 7, 10, 8, 1, 6, 2, 9, 11

杂交

7, 2, 3, 5, 10, 8, 6, 4, 9, 1, 11

2, 3, 5, 7, 10, 8, 1, 6, 4, 9, 11

图2 染色体保基因簇的杂交

在进行变异操作时，随机选中染色体的一个基因(簇)，同其它基因或基因簇交换位置。如图3所示，变异操作可分为3种情形：(1)一个基因簇变异为另一个基因簇；(2)一个基因簇变异为一个基因，或反之；(3)一个基因变异为另一个基因。在变异操作完成后，基因簇两端的连接路径可能交叉，只要将基因簇的基因倒序排列，就可避免基因簇两端的连接路径交叉。在图3情形1变异后的染色体中，假设基因簇2,4,5两端的连接路径交叉，即通路(8,2)和通路(5,10)交叉，将2,4,5的倒序排列后，得到的基因序列为“1,6,3,9,7,8,5,4,2,10,11”。

(1) 1, $\boxed{2, 4, 5}$, 7, 8, $\boxed{6, 3, 9}$, 10, 11

1, $\boxed{6, 3, 9}$, 7, 8, $\boxed{2, 4, 5}$, 10, 11

(2) 2, $\boxed{3}$, 5, 7, 10, $\boxed{8, 6, 1, 4}$, 9, 11

2, $\boxed{8, 6, 1, 4}$, 5, 7, 10, $\boxed{3}$, 9, 11

(3) 4, 5, $\boxed{3}$, 7, 8, 10, 1, $\boxed{6}$, 2, 9, 11

4, 5, $\boxed{6}$, 1, 10, 8, 7, $\boxed{3}$, 2, 9, 11

图3 染色体保基因簇的变异

在情形3中，对于贪心或轮盘赌策略选中的基因3,6，考虑城市3前的城市5以及城市6后的城市2，如果通路(5,3)和通路(6,2)近似交叉，则可采用前面介绍的去交叉策略；或者选定第一个城市为5，同时选下一城市为6，采用郭涛提出的inver-over操作[1]。

在交叉和变异的过程中，包含多个城市的基因簇保持不变，可被视为一个城市加以保护。在进化过程中，随着基因簇长度的不断增长，可杂交和变异的城市不断减少，从而使算法的收敛速度逐步加快。但是，基因簇更可能使算法陷入局部最优，为此，在获得第一个近似最优解后，可根据需要粉碎基因簇，而后采用轮盘赌策略进一步寻优。

3 基于基因簇的遗传算法

除了前面描述的杂交变异操作外，本文还引入了淘汰操作，根据预先设定的淘汰概率，在种群进化一定的代数之后，选用优良个体库中的个体，对种群中的最差个体进行淘汰。在优良个体库存放进化过程中得到最优个体。

在没有得到近似最优解以前，即当前最优个体还包含交叉路径以前，基于基因簇的遗传算法采用贪心策略进行交叉和变异。算法如下：

(1)读入和设置种群大小P、优良个体库个体数D、杂交概率C、变异概率M，淘汰概率W；读入城市坐标文件，随机产生初始种群；

(2)初始化进化代数$G=0$；

(3)贪心策略标志$F=$true；

(4)下一代种群数$N=0$；

(5)若$F=$true则采用贪心策略，否则采用轮盘赌策略，从上代种群中随机选择两个个体$S1$和$S2$，根据杂交概率多点杂交。为下一代种群产生两个新个体$S1'$，$S2'$；

(6)若$F=$ true则采用贪心策略，否则采用轮盘赌策略，对$S1'$，$S2'$根据变异概率变异，得到两个新个体$S1''$，$S2''$；

(7)若$S1''$或者$S2''$的适应度高于优良个体库中的最优个体，则将当前个体为最优个体，以当前个体基因簇为最优基因簇；

(8)若当前个体为最优个体，若优良个体库个体数$<D$，则

将当前个体加入优良个体库，否则替换优良个体库的最差个体；

(9)若当前个体为最优个体，且其环路不包含交叉路径，即该个体为近似最优解，则令贪心策略标志 F=*false*；

(10)若当前个体为近似最优解，且贪心策略标志 F=*true*，则粉碎基因簇；

(11)N=N+2，若 N<P 则转步骤(5)；

(12)当 N=P 时，根据淘汰概率，对新一代种群中的最差个体，选择优良个体库中的优良个体进行替代淘汰；

(13)G=G+1，若 G 等于指定进化代数则停止，否则转步骤(4)。

4 实验结果

本文的算法使用 Visual 2005 C#.NET 开发，实验所用的计算机为 Genuine Intel(R) CPU 2140@1.60GHz。除 CHN144 的坐标数据从文献[3]得到外，所有其它实验数据均来自 TSPLIB。

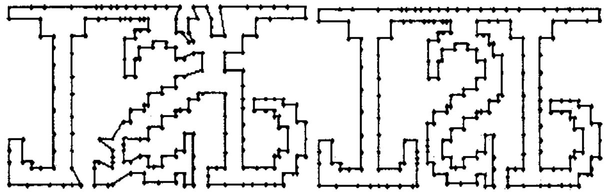
实验 1(不同规模 TSP 实验)　为方便起见，记种群大小为 P、优良个体库大小为 D、杂交概率为 C、变异概率为 M、淘汰概率为 W、进化时间为 T，环路长度为 L、当前最优路径长度为 O。T 以秒为单位，为便于比较，以下参数值固定：$C=0.05, M=0.08, W=0.04$。表 1 给出了 T, L, O 的实验结果，它们经过 10 次运行获得的第一个近优解时的平均值。

表 1　不同规模 TSP 实验结果

样例	P	D	T	L	O
BERLIN52	60	5	0.586	8281.25	7542.0
CHN144	100	10	24	32388.90	30391.6
TSP225	220	20	173	4292.78	3919.0
LIN318	220	20	490	45136.17	42029.0
D493	220	20	2029	37791.70	35002.0
D657	300	30	7309	54674.50	48912.0
PR1002	350	30	5301	287008.20	259045.0

由表 1 可知，城市数目在 300 以下时，算法求解速度较快。随着问题规模的扩大，求解时间成指数级增长。种群规模对问题求解也有很大影响，当种群规模不变时，求解时间随问题规模成指数级增长，如表 1 中 TSP225，LIN318，D493 所示。对不同规模的问题应选择合适的种群规模，适当增加种群规模有助于降低求解时间，如表 1 中 D657，PR1002 所示。因此，在多种群并行计算的情况下，有可能显著提高 TSP 求解的速度。

实验 2(持续寻优实验)　CHN144 实验得到的最优路径图同文献[5]的图 2 以及文献[6]的图 1，但得到的最优路径长度为 30367.28，不同于文献[5]的 30355.119 以及文献[6]的 30353.86；得到的次优路径图同文献[5]的图 3，但次优路径长度为 30368.20。考虑到 CHN144 的结果有些混乱，本文采用 TSPLIB 中的多组数据进行了试验，除了 TSP225 外结果均和文献[7]一致。TSP225 实验的两个结果如图 4 所示，这两个结果均优于 TSPLIB 和文献[7]公布的 3916。



(a) 路径长度 3904.28　　(b) 路径长度 3859

图 4　TSP225 实验得到的环路

实验 3(大规模 TSP 求解实验)　对 3000 个城市以内的 TSP 进行了实验，包括 PR1003、NWL1379 以及 PR2392 等。PR2392 的实验结果如图 5 所示(因路径太密没有显示城市坐标点)。PR2392 实验的种群大小 $P=400$，优良个体库大小 $D=30$，交叉系数 $C=0.05$，变异系数 $M=0.08$，淘汰系数 $=0.04$，反复寻优 4 小时 42 分钟得到长度为 378062.83 的路径，接近于文献[7]公布的当前最优路径 378032。

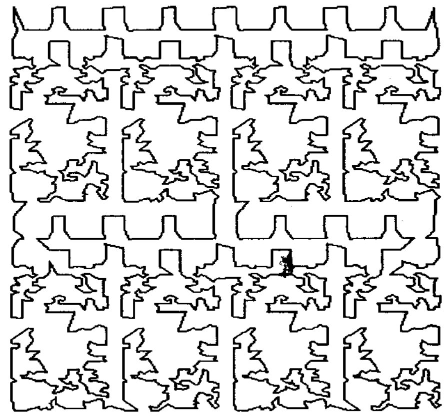


图 5　PR2392 的路径图(378062.83)

结束语　本文基于基因簇设计了 TSP 求解遗传算法。在进化过程中，该算法能保护已经发现的较短路径不受破坏、间接降低城市数量获得较快的收敛速度；通过粉碎基因簇继续寻优，使得算法具有获得全局最优解的能力，对 TSP225 的试验获得了迄今为止最好的试验结果。

下一步研究拟从采用 P2P 技术着手，通过并行计算提高求解大规模 TSP 的能力。

参考文献

[1] Guo Tao, Michalewicz Z. Inver-over Operator for the TSP[C]// Eiben A E, et al., eds. Proceedings of the 5th Parallel Problem Soving from Nature Conference. Lecture Notes in Computer Science 1998. Berlin: Springer, 1998: 803-812

[2] Helsgaun K. An effective implementation of the Lin-kernighan traveling salesman heuristic [J]. European Journal of Operational Research, 2000, 126: 106-130

[3] 康立山，谢云，尤矢勇，等. 非数值并行算法(第一册)：模拟退火算法[M]. 北京：科学出版社，1997

[4] Reinelt G. TSPLIB-A Traveling Salesman Problem Library [J]. ORSA J. Comput., 1991, 3(4): 376-385

[5] 宋勇，陈贤富，戴蓓倩. 弹性 TSP 及其并行遗传优化[J]. 小型微型计算机系统，2006, 27(5): 842-845

[6] 蔡之华，彭锦国，高伟，等. 一种改进的求解 TSP 问题的演化算法[J]. 计算机学报，2005, 28(5): 823-828

[7] Lee S H. Greedy Randomized Adaptive Search Procedure for Traveling Salesman Problem[D]. Texas A&M University, 2006