维普资讯 http://www.cqvip.com

第19卷 第11期 Vol. 19 No. 11
控 制 与 决 策 Control and Decision
2004年11月 Nov. 2004

文章编号：1001-0920(2004)11-1286-04

求解旅行商问题的混合粒子群优化算法

高 尚[1,2]，韩 斌[1]，吴小俊[1]，杨静宇[2]
(1.江苏科技大学 电子信息学院，江苏 镇江 212003；2.南京理工大学 计算机系，江苏 南京 210094)

摘 要：结合遗传算法、蚁群算法和模拟退火算法的思想，提出用混合粒子群算法来求解著名的旅行商问题.与模拟退火算法、标准遗传算法进行比较，24种混合粒子群算法的效果都比较好，其中交叉策略D和变异策略F的混合粒子群算法的效果最好，而且简单有效.对于目前仍没有较好解法的组合优化问题，通过此算法修改很容易解决.

关键词：粒子群算法；遗传算法；模拟退火算法；蚁群算法；旅行商问题
中图分类号：TP301.6　　**文献标识码**：A

Solving traveling salesman problem by hybrid particle swarm optimization algorithm

GAO Shang[1,2], *HAN Bin*[1], *WU Xiao-jun*[1], *YANG Jing-yu*[2]
(1. School of Electronics and Information, University of Science and Technology, Zhenjiang 212003, China; 2. Department of Computer, Nanjing University of Science and Technology, Nanjing 210094, China. Correspondent: GAO Shang, E-mail: gao_shang@hotmail.com)

Abstract: The particle swarm optimization algorithm combining with the ideal of the genetic algorithm, the ant colony algorithm and the simulated annealing algorithm are presented to solve the well known traveling salesman problem. Compared with the standard GA and simulated annealing algorithm, all 24 hybrid particle swarm optimization algorithms are proved effective. Especially the hybrid particle swarm optimization algorithm with across strategy and mutation strategy is simple and effective. It can easily be modified for any combinatorial problem for which there is no good specialized algorithm up to now.

Key words: particle swarm algorithm; genetic algorithm; simulated annealing algorithm; ant colony algorithm; traveling salesman problem

1 引 言

粒子群优化算法(PSO)是一种进化计算技术，最早是由Kennedy等于1995年提出的[1]. PSO与遗传算法类似，是一种基于迭代的优化工具.系统初始化为一组随机解，通过迭代搜寻最优值.但没有遗传算法采用的交叉和变异，而是粒子在解空间追随最优粒子进行搜索.与遗传算法相比，PSO的优势在于简单、易实现，而且没有许多参数需要调整. PSO已广泛应用于函数优化.

目前已提出多种PSO改进算法[2~4]，尤其是文献[3]采用离散PSO算法解决旅行商问题(TSP)，取得了较好的结果.但文献[3]计算过程较复杂. TSP是一个NP完全问题，目前求解TSP问题的主要方法有模拟退火算法[5]、遗传算法、启发式搜索法、Hopfield神经网络算法和蚁群算法等.本文提出基于遗传算法、蚁群算法和模拟退火算法思想的一种新的混合PSO算法解决TSP问题.

收稿日期：2003-12-29；**修回日期**：2004-04-02.
作者简介：高尚(1972—)，男，江苏镇江人，副教授，博士生，从事模式识别与人工智能等方面的研究；韩斌(1968—)，男，江苏海安人，副教授，从事数字图像处理等方面的研究.

维普资讯 http://www.cqvip.com

2 基本粒子群优化算法

PSO 模拟鸟群的捕食行为，假设一群鸟在只有一块食物的区域内随机搜索食物，所有的鸟都不知道食物的位置，但它们知道当前位置与食物的距离. 最简单有效的方法是搜寻目前离食物最近的鸟的周围区域. PSO 从这种模型中得到启示，并将其用于解决优化问题. PSO 中每个优化问题的解都是搜索空间中的一只鸟，称之为"粒子". 所有的粒子都有一个由被优化的函数所决定的适应值，对于每个粒子，还有一个速度决定他们飞翔的方向和距离，然后粒子们就追随当前的最优粒子在解空间中搜索. PSO 初始化为一群随机粒子(随机解)，然后通过迭代找到最优解. 在每一次迭代中，粒子通过跟踪两个"极值"更新自己，一个是粒子本身所找到的最优解，称为个体极值 pbest；另一个极值是整个种群目前找到的最优解，该极值是全局极值 gbest. 另外，也可以不用整个种群而只用其中一部分为粒子的邻居，那么在所有邻居中的极值就是局部极值. 在找到这两个最优值时，每个粒子根据如下公式更新自己的速度和新的位置：

$$v_{k+1} = c_0 v_k + c_1(\mathrm{pbest}_k - x_k) + c_2(\mathrm{gbest}_k - x_k), \tag{1}$$

$$x_{k+1} = x_k + v_{k+1}. \tag{2}$$

其中：v_k 为第 k 步粒子的速度向量；x_k 为第 k 步粒子的位置；pbest_k 为第 k 步粒子本身所找到的最好解的位置；gbest_k 为第 k 步整个种群目前找到的最好解的位置；c_0, c_1, c_2 表示群体认知系数，c_0 一般取(0,1)之间的随机数，c_1 和 c_2 取(0,2)之间的随机数. 每一维粒子的速度都被限制在一个最大速度 $v_{\max}$($v_{\max} > 0$)，若 $v_k > v_{\max}$ 时，$v_k = v_{\max}$；$v_k < -v_{\max}$ 时，$v_k = -v_{\max}$.

3 混合粒子群算法

粒子群算法的本质是利用本身信息、个体极值信息和全局极值 3 个信息，指导粒子下一步迭代位置. 对于 TSP 问题，其当前位置是基本路径，若按基本粒子群算法，其速度难以表达，故采用遗传算法的思想解决. 式(1)中 $c_0 v_k$ 项可看作遗传算法的变异操作，$c_1(\mathrm{pbest}_k - x_k) + c_2(\mathrm{gbest}_k - x_k)$ 项可看作遗传算法的交叉操作，使当前解与个体极值和全局极值分别作交叉操作，产生的解为新的位置. 变异操作和交叉操作后，新的解可能比原来的解要坏，接受准则是采用模拟退火算法的思想，允许目标函数有限范围内变坏，为简化计算并不按概率取舍，直接按 $\Delta E < e$，e 为按允许目标函数变坏的范围. 下面讨论具体的操作过程.

3.1 变异操作

这里假设有 n 个城市. 由路径 C_0 变异到另一条路径 C_1，常用的有以下几种策略：

1) 变异策略 A：在第 $1 \sim n$ 个访问的城市中随机选取第 j_1 次和第 j_2 次访问的城市，在路径 C_0 中交换第 j_1 次和第 j_2 次访问的城市，其余不变，此时的路径为 C_1.

2) 变异策略 B：在第 $1 \sim n$ 个访问的城市中随机选取第 j_1 次访问的城市，在路径 C_0 中交换第 j_1 次和第 $j_1 + 1$ 次访问的城市，其余不变，此时的路径为 C_1.

3) 变异策略 C(也称逆转策略)：在第 $1 \sim n$ 个访问的城市中随机选取第 j_1 次和第 j_2 次访问的城市，在路径 C_0 中第 j_1 次到第 j_2 次访问的城市之间的子路径以反方向插入，其余不变，此时的路径为 C_1.

4) 变异策略 D：在第 $1 \sim n$ 个访问的城市中随机选取第 j_1 次和第 j_2 次访问的城市，假设 $j_1 < j_2$，在路径 C_0 中将第 j_1 次访问的城市安排到第 j_2 次访问的城市之后，其余不变，此时的路径为 C_1；

5) 变异策略 E：上述策略未利用城市间距离大小的信息，变异策略 E 将利用点的邻接关系，依据蚁群算法的思想，距离近的邻接点以较大的概率被选为下一个访问点，所以在局部调整时依据此思想. 设 $d(i,j)$ 表示城市 i 与城市 j 的距离，在 $1 \sim n$ 的城市中随机选取城市 i_1，离城市 i_1 最远城市的距离为

$$d_{\max} = \max_j d(i_1, j),$$

为了排除下一个访问点为其本身，令 $d(i_1, i_1) = d_{\max}$，则下一个访问点为城市 j 的概率为

$$p_j = \frac{d_{\max} - d(i_1, j)}{\sum_{k=1}^{n}(d_{\max} - d(i_1, k))}. \tag{3}$$

假设以式(3)的概率选取城市 j_1，在路径 C_0 中将城市 j_1 安排到城市 i_1 之后，其余不变，此时的路径为 C_1.

6) 变异策略 F：在 $1 \sim n$ 的城市中随机选取城市 i_1，为了使路径总长度之和达到最小，优先解决薄弱环节，这里采用路径中相邻城市间距离大的两个城市以较大的概率被选取，在它们之间插入其他城市. 采用 $l(n)$ 数组记录路径 C_0 相邻城市之间的距离，具体数据如下：

$$l(k) = d[c(k), c(k+1)],$$
$$k = 1, 2, \cdots, n-1;$$
$$l(n) = d[c(n), c(1)]. \tag{4}$$

维普资讯 http://www.cqvip.com

选取城市 i 的概率为

$$p_i = l(i) / \sum_{k=1}^{n} l(k). \tag{5}$$

按式(5)选取城市 i_1，后面方法同策略变异 E，按式(3)选取城市 j_1，在路径 C_0 中将城市 j_1 安排到城市 i_1 之后，其余不变，此时的路径为 C_1.

3.2 交叉操作

交叉的方法很多，下面几种方法最常用：

1) 交叉策略 A：在第 2 个串中随机选择一个交叉区域；将 old2 的交叉区域加到 old1 前面(或后面)，删除 old1 中已在 old2 交叉区中出现过的城市.

2) 交叉策略 B：在第 2 个串中随机选择一个交叉区域；将 old2 的交叉区域加到 old1 对应的位置，删除 old1 中已在 old2 的交叉区中出现过的城市.

3) 交叉策略 C：在第 2 个串中随机选择一个交叉区域；将 old2 的交叉区域加到 old1 的随机位置，删除 old1 中在 old2 的交叉区中已出现的城.

4) 交叉策略 D：在第 2 个串中随机选择一个交叉区域，如交叉区域为：6,5,4,3；找非交叉区域城市离交叉区域两端城市 6 和城市 3 最近的城市，若城市 8,3 最近，则将 old2 的交叉区域加到 old1 的城市 8 处的位置(有可能逆转)，删除 old1 中已在 old2 的交叉区中出现过的城市.

3.3 混合粒子群算法

解 TSP 的混合粒子群算法如下：

设定粒子数 n_p，规定迭代次数 $n_{\max}$，随机产生 n_p 个初始解(初始路径)C_0；

根据当前位置计算适应值(各路径的长度)ltsp0，设置当前适应值为个体极值 plbest，当前位置为个体极值位置 pcbest，根据各个粒子的个体极值 plbest，找出全局极值 glbest 和全局极值位置 gcbest；

While(迭代次数 < 规定迭代次数 $n_{\max}$) do

　For$j = 1 : n_p$

　　第 j 个粒子路径 $C_0(j)$ 与 gcbest 交叉得到 $C_1'(j)$；

　　$C_1'(j)$ 与 pcbest 交叉得到 $C_1''(j)$；

　　对 $C_1''(j)$ 产生变异得到 $C_1(j)$；

　　根据当前位置计算适应值 ltsp1；

　　计算两个位置所引起的适应值的变化量 ΔE；若 $\Delta E < e$(e 为按允许目标函数变坏的范围)，$\Delta E \leqslant 0$，接受新值；否则，拒绝，第 j 个粒子路径 $C_1(j)$ 仍然为 $C_0(j)$；

　　如果ltsp1(j) <plbest(j)，则 pcbest(j) = $C_1(j)$，plbest(j) = ltsp1(j)；

　End For

　根据各个粒子的个体极值 plbest，找出全局极值 glbest 和全局极值位置 gcbest；

　$C_0 \leftarrow C_1$；

End While

最后输出全局极值 glbest 和全局极值位置 gcbest.

混合粒子群算法的时间复杂性可估算如下：以交叉时间花费最多，内循环需要作 $O(2n_p)$ 交叉操作，外循环执行 $n_{\max}$ 次，所以时间复杂性大约为 $O(2n_p n_{\max})$.

4 算法测试

分别采用模拟退火算法、遗传算法和混合粒子群算法解决 31 个城市的 CTSP 问题，数据来源于文献[5]. 模拟退火算法采用文献[5]的算法，起始温度 $T = 100\,000$，终止温度 $T_0 = 1$，退火速度 $\alpha = 0.99$；遗传算法程序采用 Matlab 的遗传算法工具箱[6]，参数如下：染色体个数 $N = 30$，交叉概率 $P_c = 0.2$，变异概率 $P_m = 0.5$，迭代次数 100；混合粒子群算法参数为：粒子数 $n_p = 30$，最大迭代次数 $n_{\max} = 2\,000$，$e = 100$，混合粒子群算法分别有 4 种交叉策略和 6 种变异策略，组合起来共 24 种方法进行比较. 对各种算法测试 100 次，结果如表 1 所示.

从表 1 可以看出，混合粒子群算法的 24 种组合大都比较好，其中交叉策略 D 和变异策略 F 的混合粒子群算法最好. 对于31个城市的CTSP，目前用模拟退火算法得到的最好解[5]如图 1 所示，总路程为 15 398 km；遗传算法最好解的总路程为 15 387 km(见图 2)；混合粒子群最好解的总路程为 15 383 km(见图 3). 这说明采用混合粒子群算法较好. 采用模拟退火算法解旅行商问题时，在邻域内找下一解实质上是进行变异操作，变异操作采用随机策略，未充分利用已知信息；而本文除了采用变异操作，还进

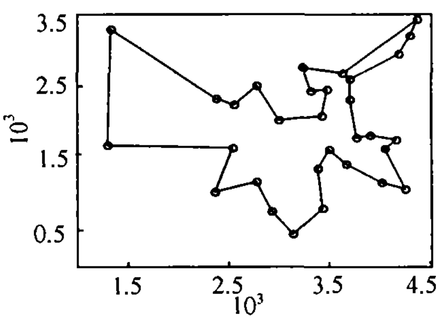

图 1 采用模拟退火算法解 CTSP 的最好解

维普资讯 http://www.cqvip.com

表 1　几种算法测试结果

算法		平均值/km	最好解/km	最差解/km
模拟退火算法		16 902	15 398	18 247
遗传算法		16 920	15 387	17 298
交叉策略A	变异策略A	16 368	15 390	17 682
	变异策略B	15 738	15 391	16 649
	变异策略C	15 965	15 383	17 456
	变异策略D	16 854	15 392	17 346
	变异策略E	16 885	15 383	18 023
	变异策略F	15 849	15 383	17 367
交叉策略B	变异策略A	16 844	15 393	17 673
	变异策略B	16 963	15 389	17 743
	变异策略C	15 893	15 383	17 896
	变异策略D	16 874	15 390	18 023
	变异策略E	16 894	15 393	17 931
	变异策略F	15 835	15 383	16 697
交叉策略C	变异策略A	16 848	15 394	16 935
	变异策略B	16 383	15 390	17 849
	变异策略C	16 427	15 383	17 384
	变异策略D	16 853	15 389	17 673
	变异策略E	16 389	15 383	17 843
	变异策略F	15 733	15 383	16 682
交叉策略D	变异策略A	16 383	15 393	18 024
	变异策略B	16 795	15 390	17 384
	变异策略C	16 383	15 383	17 032
	变异策略D	15 963	15 389	18 024
	变异策略E	15 837	15 383	17 473
	变异策略F	15 737	15 383	16 740

行与交叉操作，在采用变异操作时，特别采用了蚁群算法的思想"距离近的邻接点以较大的概率被选"，因而效果较好．采用遗传算法解旅行商问题时，使用随机策略，虽然保证解分布比较均匀，但个体的质量不能保证，解群中的个体很大一部分远离最优解，这样在经过交叉操作后的个体性能不能保证比原来的好；而本文采用的交叉操作，是与个体极优和群体极优进行交叉，因此会改变下一代群体的质量，有助于提高搜索效率．

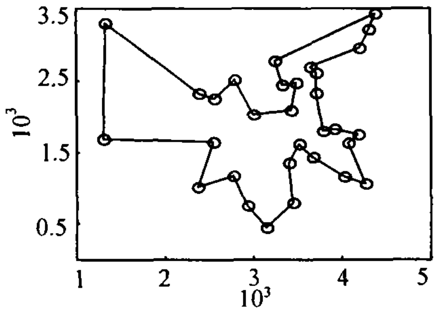

图 2　采用遗传算法解 CTSP 的最好解

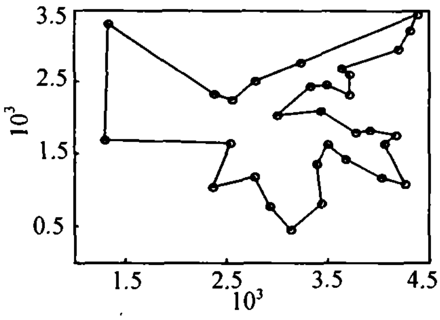

图 3　采用混合粒子群算法解 CTSP 的最好解

5　结　语

本文尝试采用结合遗传算法、蚁群算法和模拟退火算法的思想的混合粒子群算法解决典型的离散优化问题TSP问题．对于目前还没有较好解法的组合优化问题，很容易通过修改此算法来加以解决．PSO研究处于初期，还有许多问题值得研究，如算法的收敛性、理论依据等．但从当前的应用效果看，这种模仿自然生物的新型系统寻优思想具有光明的前景，更多深入细致的工作还有待于进一步展开．

参考文献(References)：

[1] Eberhart R C, Kennedy J. A new optimizer using particles swarm theory[A]. *Proc Sixth Int Symposium on Micro Machine and Human Science*[C]. Nagoya, 1995. 39-43.

[2] Shi Y H, Eberhart R C. A modified particle swarm optimizer[A]. *IEEE Int Conf on Evolutionary Computation*[C]. Anchorage, 1998. 69-73.

[3] Maurice Clerc. Discrete particle swarm optimization illustrated by the traveling salesman problem[DB]. http://www.mauriceclerc.net, 2000.

[4] 李爱国，覃征，鲍复民，等．粒子群优化算法[J]．计算机工程与应用，2002，38(21)：1-3.
(Li A G, Qin Z, Bao F M, et al. Particle swarm optimization algorithms[J]. *Computer Engineering and Applications*, 2002, 38(21): 1-3.)

[5] 康立山，谢云，尤矢勇，等．模拟退火算法[M]．北京：科学出版社，1994. 50-97.

[6] 高尚．基于Matlab遗传算法优化工具箱的优化计算[J]．微型电脑应用，2002，18(8)：52-54.
(Gao S. Optimization computation babed on the genetic algorithm optimization toolbox in matlab [J]. *Microcomputer Applications*, 2002, 18(8): 52-54.)