

基于蚁群信息素的遗传操作算法

张应辉 王志伟 曾庆华

(成都东软信息技术学院 成都 611844)

摘要 在遗传操作算法中通常是随机选择交叉和变异的基因位置。基于蚁群信息素和选择基因的概率,本文提出了一种选择基因的方法以提升局部最优化的性能和加速算法的收敛。通过求解旅行商问题(TSP)的仿真实验,表明了这种方法的有效性。

关键词 遗传算法,蚁群算法,全局最优化,信息表,旅行商问题(TSP)

The Genetic Operating Based on the Pheromone of Ant Colony in the Genetic Algorithm

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

(Chengdu Neusoft Institute of Information, Chengdu 611844)

Abstract It is a normal way to randomly select crossover and mutation gene locus in genetic operating. In this paper, based on the pheromone of Ant Colony and the probability of selecting gene, a method to select gene is presented to raise the performance of local optimization and the speed of the algorithm convergence. The efficiency of the method has been shown by simulative experiments solving the Traveling Salesman Problems (TSP).

Keywords Genetic algorithms, Ant colony algorithm, Global optimization, Pheromone, Traveling salesman problems (TSP)

1 引言

近年来,启发式智能优化方法愈来愈引起人们的关注。如:模拟退火、遗传算法、蚁群算法等,它们是解决 NP 问题的有效工具。

蚁群算法(ACA, Ant Colony Algorithm)^[1,2]是意大利学者 Marco Dorigo 等提出的一种仿生寻优算法。参加寻径的蚂蚁通过留在链路上的信息素(pheromone)交互来选择新的路由,从而达到寻优的目的。其主要特点是:本身是一个增强型学习系统,具有分布式的计算特性,具有很强的顽健性,易于与其它优化算法融合。但是 ACA 在解决大型优化问题时,存在搜索空间和时间性能上的矛盾,易出现过早收敛于非全局最优解以及计算时间过长的弱点;而且决定 ACA 性能的 4 个控制参数(α, β, ρ, Q)的取值缺乏理论支持,影响算法的性能。

遗传算法(GA, Genetic Algorithm)^[3,4]是美国 Michigan 大学的 John Holland 教授等人于 1975 年提出的模拟自然界中生物遗传机制的随机搜索算法,它是基于染色体层面进行各种遗传优化操作。GA 在本质上是一个群体迭代算法,它从一组随机的初始群体出发,由目标函数(适应函数)对其进行评价,根据“优胜劣汰”的竞争原则,通过选择、交叉、变异等遗传过程,产生下一代群体,不断重复,直到获得满意的结果。GA 的特点是:具有全局搜索能力,具有潜在的并行型,具有较强的顽健性,计算过程简单,能很好地解决寻找最优解和探寻搜索空间的矛盾,具有可扩展性,易于和其它算法结合。但 GA 算法对于系统中的反馈信息利用不充分,当求解到一定范围时会做大量的无效迭代。

目前将 ACA 与 GA 相融合解决 NP 问题,特别是旅行商问题(TSP),已取得较好的效果^[6-8]。但大多数方法都是着重于参数的优化选择。

本文提出了一种将 ACA 信息素应用于 GA 求解 TSP 的新方法:利用 ACA 信息素的正反馈机制,在选择遗传操作基

因位进行遗传操作(交叉和变异)时,不同于传统的随机选择方法,而是根据结点间的信息素和距离信息,定义结点 r 和 s 构成遍历路径中边的概率 p_{rs} ,并以 p_{rs} 确定遗传操作基因位的选择概率。仿真试验表明,采用该方法的 GA 有较好寻优性能和较快收敛速度,适合求解较大规模的 TSP 问题。

2 蚁群算法与信息素

自然界中的蚂蚁,在寻找食物或遇到障碍物时,总能找到一条从食物到蚁巢或绕过障碍物的最优路径。原因在于,蚂蚁运动中会在所经过的路径上释放出信息素,后续蚂蚁可根据前面蚂蚁遗留下来的信息素选择下一条要走的路径。一条路径上的信息素越高,说明这条路径被选中的次数越多,即路径的性能更优,后续蚂蚁选择这条路径的概率就更大,由此构成一个学习信息的正反馈过程,从而逐渐逼近最优解。这就是信息素的通讯机制和正反馈机制。

正反馈机制能加快算法的搜索,也会导致算法陷于局部最优解,出现停滞现象,而通讯机制能使个体相互协作,有利于算法搜索到更优解。

TSP 是一个典型的 NP 问题,它可简单描述为,求一条遍历 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

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

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

设 m 为蚂蚁的总数, $\tau_n(t)$ 表示 t 时刻边 e_n 上的信息素量, $\tau_n(0) = \tau_0$ (τ_0 为常数)。随着时间的推移,新的信息素加进来,旧的信息素要挥发。当所有蚂蚁完成一次遍历后,各边的信息素按下式调整:

$$\tau_n(t+1) = \rho \cdot \tau_n(t) + \Delta\tau_n \quad (2)$$

$$\Delta\tau_n = \sum_{k=1}^m \Delta\tau_n^k \quad (3)$$

$\Delta\tau_n$ 表示本次遍历边 e_n 上的信息素增量, $1-\rho$ 表示信息素的

挥发速度, $\Delta\tau_n^k$ 表示第 k 只蚂蚁本次遍历释放在边 e_n 上的信息素。

$$\Delta\tau_n^k = \begin{cases} Q/L_k, & e_n \in E_k \\ 0, & \text{else} \end{cases} \quad (4)$$

Q 为常数, E_k 为本次遍历第 k 只蚂蚁的遍历路径中的边集合, L_k 表示 E_k 中边长之和, 即遍历路径的长度。蚂蚁在结点 r 向结点 s 转移的概率 p_{rs}^k 为:

$$p_{rs}^k = \begin{cases} \frac{\tau_{rs}^\alpha \eta_{rs}^\beta}{\sum_{q \in A_k} \tau_{rq}^\alpha \eta_{rq}^\beta}, & s \in A_k \\ 0, & \text{else} \end{cases} \quad (5)$$

其中 $A_k = \{1, 2, \dots, n\} - \text{tabu}_k$ 表示蚂蚁 k 当前能选择的结点集合, tabu_k 为禁忌表, 记录蚂蚁 k 已路过的结点。 $\eta_{rs} = 1/d(r, s)$ 为边 e_{rs} 的能见度。 α, β 体现了信息素和能见度对蚂蚁决策的重要性。

ACA 基本的运行过程是: m 只蚂蚁随机从若干个结点出发, 根据转移概率 p_{rs}^k 选择下一个遍历结点, 已遍历的结点放入 tabu_k 中。一次遍历完成后, 由公式 (2)~(4) 更新每条边上的信息素, 反复重复上述过程, 直到终止条件成立。

对 ACA 来说, $\tau_{ij}(t)$ 之间的相对大小会直接影响转移概率 p_{ij}^k , 从而影响可行解的质量。在搜索早期, 信息素的分布是分散的, 随着搜索进行, 信息素慢慢地集中到部分边上, 搜索方向也随之基本确定。当某些边上信息素强度明显高于其它边, 导致在构造可行解时, 总是使用少数几条边, 就会使解结构过于相似, 搜索停顿下来, 算法陷入停滞。

3 基于信息素的遗传算法

用 GA 求解 TSP, 交叉/变异机制将直接影响算法性能, 也是研究的重点。通常是根据个体的适应函数确定个体的选择概率选择父代个体, 再根据交叉概率 p_c 和变异概率 p_m 对父代个体进行遗传操作 (交叉和变异), 生成子代个体。在遗传操作时, 一般是随机地选择参与操作的基因位, 这虽然增强了算法的全局搜索能力, 但也增加了算法的无效操作次数, 降低了算法性能。

如利用 ACA 结点转移概率的概念, 对父代个体遍历路径的每一边赋予一个选择概率, 根据该概率选择遗传操作的基因位, 则可减少算法的无效操作次数, 提高算法的有效性。

由于 TSP 中结点分布的多样性和解空间的复杂性, 如全部按选择概率选择遗传操作的基因位, 也易出现过早收敛于非全局最优解, 因此定义一个阈值 p_t ($0 \leq p_t \leq 1$), 即每次遗传操作有 p_t 的概率采用按选择概率选择基因位。这样既能减少算法的无效操作次数, 提高算法的有效性, 又能保持算法的全局搜索能力。

据上述思路, 可设计如下基于信息素的遗传算法 GAOp (Genetic Algorithm Based on Pheromone)。

GAOp 算法:

① 输入阈值 p_t ($0 \leq p_t \leq 1$) 和其他参数; 初始化种群 G ; 初始化信息素 $\tau_{ij}(0)$ ($i, j = 1, 2, \dots, n$)。

② 评估种群 G ; 更新 τ_{ij} 。

③ 选择父代个体; 根据 p_t 确定基因位选择方法, 选择交叉操作基因位, 进行交叉操作, 生成子代个体。

④ 对每个子代个体, 根据 p_t 确定基因位选择方法, 选择变异操作基因位, 进行变异操作, 生成新种群 G' 。

⑤ 评估种群 G' ; 更新 τ_{ij} 。

⑥ 如未满足终止条件, 用 G' 更新 G , 转向 ③; 否则结束算法。

上述算法只是提供了一种新的优化算法思路, 针对不同的优化问题, GA 和 ACA 的研究成果均可用于该算法的实现。

4 GAOp 的实现和仿真试验

为了探讨 p_t 在 GAOp 中的作用, 简化算法的实现, 本文实现的仿真程序对其他参数和操作进行了改进和简化, 主要包括以下内容。

1) 参数: 种群规模 $m = 6n$ (n 为城市数目), $\alpha = 1, \beta = 2$ 。

2) 个体选择机制:

a) 适应函数 f_i 为路径长度的倒数;

b) 选择概率 $p_i = f_i / \sum_{k=1}^m f_k$;

c) 采用杰出选择基本遗传算法。

3) 信息素更新机制:

a) $\tau_{ij}(0) = 1$ ($i, j = 1, 2, \dots, n$);

b) 若本代遗传产生历史最优个体 b (记其边的集合为 E_b), 则更新信息素:

$$\tau_{ij}(t+1) = \rho \cdot \tau_{ij}(t) + \Delta\tau_{ij}, i, j = 1, \dots, n$$

$$\text{其中, } \rho = \frac{n-3}{n-1}; \Delta\tau_{ij} = \begin{cases} 1, & e_{ij} \in E_b; \\ 0, & \text{else} \end{cases}$$

4) 交叉变异机制:

a) $p_c = p_m = 1$;

b) 经交叉操作产生的子代个体其路径长度如大于种群 G 中最短路径长度的 k 倍 (k 为解空间系数。一般, $1.2 < k < 2$), 则放弃;

c) 经双点交叉变异、对换变异和插入变异^[3~5]生成的 3 个个体与原个体比较, 选择最短路径长的个体作为变异操作的结果;

d) 基因位选择概率 p_{ij} : 设 E_c 为个体 c 的边集合, 对每一边 $e_{ij} \in E_c$, 由其能见度和信息数量定义其对个体的贡献度函数 h_{ij} 和选择概率 p_{ij} :

$$h_{ij} = \frac{\tau_{ij}^\alpha}{d(i, j)^\beta}; p_{ij} = \frac{h_{ij}^{-1}}{\sum_{e_{rs} \in E_c} h_{rs}^{-1}}$$

贡献度越高, 保留它的概率越大, 或者说选择其端点作为遗传操作基因位的概率 p_{ij} 越小; 反之, h_{ij} 越小, p_{ij} 越大。

e) 根据阈值 p_t 确定本次遗传操作作用 p_t 选择基因位的概率。

利用该仿真程序对文[9]中 KroA 100 个城市 (遗传代 2000)、KroA 150 个城市 (遗传代 5000)、KroA 200 个城市 (遗传代 10000) 等 3 个 TSP 进行求解;

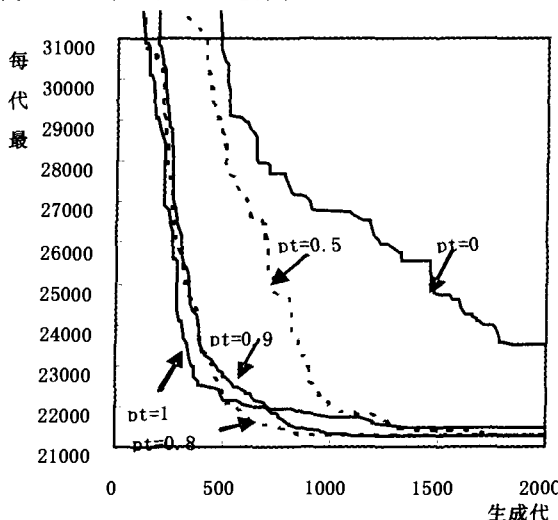


图1 KroA 100 个城市 TSP 求解收敛曲线 (OX)

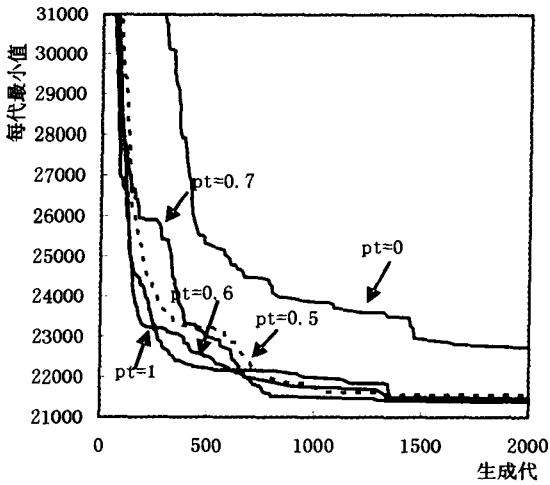


图2 KroA 100个城市 TSP 求解收敛曲线(PMX)

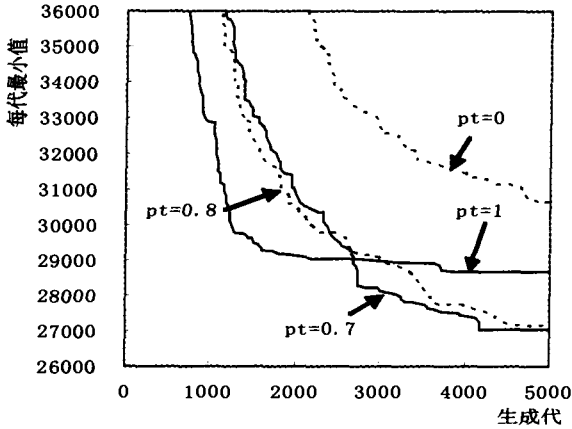


图3 KroA 150个城市 TSP 求解收敛曲线(OX)

1)交叉操作分别采用顺序交叉(OX)和部分匹配交叉(PMX)^[3,4]操作;

2) $p_t \in \{0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1\}$, $p_t=0$ 表示所有基因位均采用随机选择, $p_t=1$ 表示所有基因位均采用概率选择。

3)解空间系数 $k=1.3$ 。

其寻优过程见图1~图6,最优解结果分别见图7~图9,试验收敛情况见表1~表3。

表1 仿真试验结果——KroA100

最大遗传代:2000

p_t	OX		PMX	
	世代数	寻优解	世代数	寻优解
0	1931	23501	1954	22733
0.1	1901	22753	1177	22333
0.2	1936	22376	1845	22015
0.3	1873	22911	1975	22350
0.4	1949	22058	1560	22330
0.5	1822	21285 *	1338	21554
0.6	1941	21614	1334	21409
0.7	1912	21821	1696	21386
0.8	1021	21285 *	821	21615
0.9	1109	21285 *	1800	22254
1	1364	21501	1356	21490

* 文[9]的最优解 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后取整,其结果即为 21285

表2 仿真试验结果——KroA150

最大遗传代:5000

p_t	OX		PMX	
	世代数	寻优解	世代数	寻优解
0	4841	30628	4835	29452
0.1	4873	29447	4688	28160
0.2	4998	29889	4694	28702
0.3	4946	29474	3291	27894
0.4	4618	28634	3608	27854
0.5	4987	27721	4817	28631
0.6	4958	27726	4036	27499
0.7	4608	27036	3261	27042
0.8	4545	27116	4518	27326
0.9	4481	27825	3977	26956
1	4028	28693	4827	26682

* 文[9]的最优解为 26524,没提供最优路径

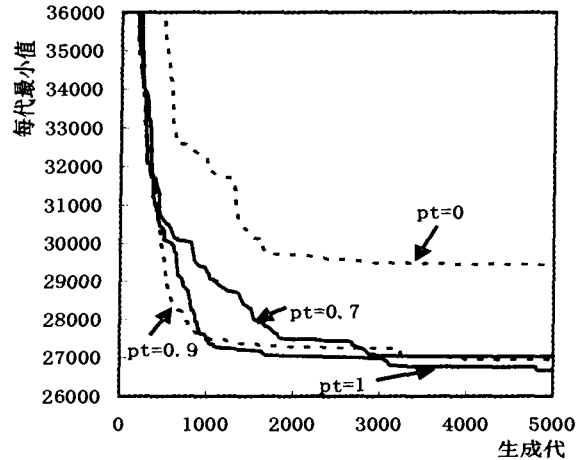


图4 KroA 150个城市 TSP 求解收敛曲线(PMX)

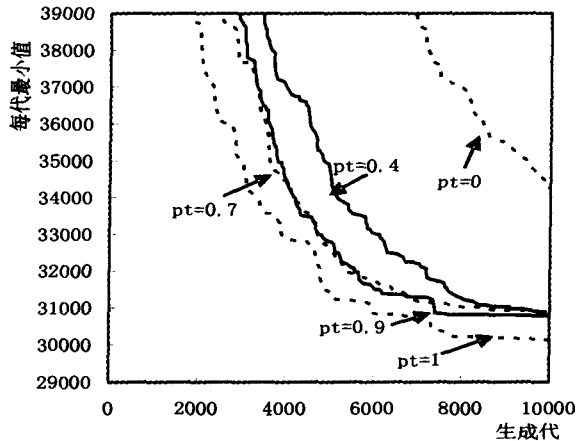


图5 KroA 200个城市 TSP 求解收敛曲线(OX)

结束语 本文提出的 GAoP 算法,在遗传算法的遗传操作中,利用蚁群信息素的通讯机制和正反馈机制选取基因位进行遗传操作,提高了遗传算法的性能。该方法同时融合了两种算法的优点,即有较快的收敛速度,也有较强的全局寻优能力,GA 和 ACA 方面的大多数研究成果也可融于该算法中,有较强的适应性和可扩展性。

仿真试验实现的 GAoP 算法主要目的是探讨信息素在 CA 中的作用和算法的有效性,因此对信息素的更新和 CA 的实现均做了一些改进和简化。仿真试验结果表明,阈值 P_t 是 GAoP 算法的一个重要参数,它直接影响到算法的有效性和

寻优性能。

GAoP算法中其它的、基于GA和ACA的参数(如 $m, \alpha, \beta, \rho, Q, p_c, p_m$ 等)设定和操作方法(如信息素更新机制,交叉变异机制,个体选择机制等),以及它们与 p_t 之间的相互影响,还可进行进一步的研究和探讨。

和GA一样,GAoP算法不但可解TSP,也可用于求解其他的优化问题。

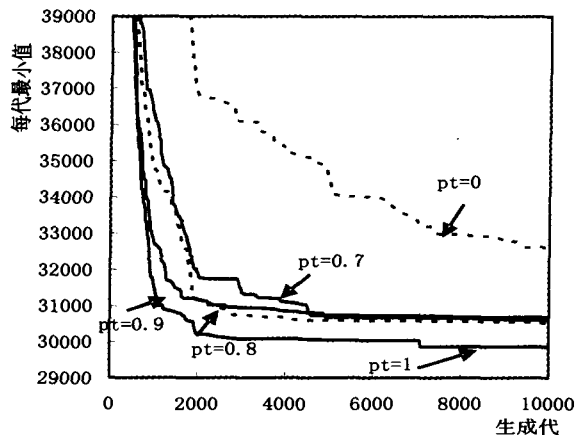


图6 KroA 200个城市TSP求解收敛曲线(PMX)

表3 仿真试验结果——KroA200

最大遗传代:10000

p_t	OX		PMX	
	世代数	寻优解	世代数	寻优解
0	9988	34243	9437	32623
0.1	9914	33184	9517	31599
0.2	9998	32667	9959	31111
0.3	9715	31568	8421	31100
0.4	9909	30883	9943	31697
0.5	9835	31719	7261	30981
0.6	9873	31315	8694	31251
0.7	9609	30926	8292	30686
0.8	9618	31270	9786	30524
0.9	9276	30791	9548	30609
1	9691	30125	7080	29864

*文[9]的最优解为29368,没提供最优路径

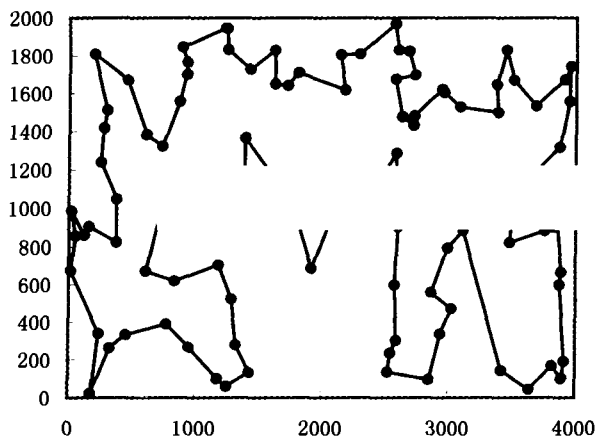


图7 KroA 100个城市TSP已知最优解

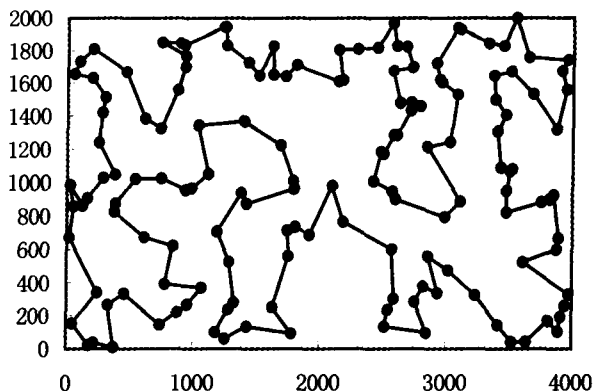


图8 KroA 150个城市TSP近似最优解

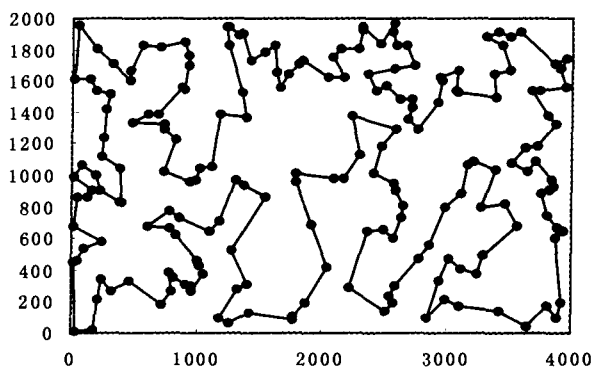


图9 KroA 200个城市TSP近似最优解

参考文献

- 1 Dorigo M, Maniezzo C A. The ant system: optimization by a colony of cooperating agents. IEEE Trans. on System, Man and Cybernetics, 1996, 26(1): 1~13
- 2 Dorigo M, Gambardella L M. Ant colony system: a cooperative learning approach to the traveling salesman problem. IEEE Trans. on Evolutionary Computation, 1997, 1(1): 53~66
- 3 王小平, 曹立明. 遗传算法——理论、应用与软件实现. 西安: 西安交通大学出版社, 2002
- 4 周明, 孙树栋. 遗传算法原理及应用. 国防工业出版社, 1999
- 5 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
- 6 丁建立, 陈增强, 袁著祉. 遗传算法与蚂蚁算法的融合. 计算机研究与发展, 2003, 40(9): 1351~1356
- 7 Pilat M L, White T. Using genetic algorithms to optimize ACS TSP. In: Proceedings of Ant Algorithms: 3th International Workshop, ANTS 2002, Brussels, Belgium, 2002. 282~287
- 8 孙力娟, 王良俊, 王汝传. 改进的蚁群算法及其在TSP中的应用研究. 通信学报, 2004, 25(10): 111~116
- 9 TSPLIB[EB/OL]. <http://www.iwr.uni-heidelberg.de/groups/comopt/software/TSPLIB95/index.html>, 2003