

一种求解平面旅行商问题的新算法 ——绕中心周游法

宇德明

中南大学土木建筑学院, 长沙 410075

[摘要] 提出了一种求解平面旅行商问题的新算法——绕中心周游法, 它是一种确定型算法, 时间复杂性与最近邻算法相同, 为 $O(n^2)$, 其中 n 为城市数。利用所编写的绕中心周游法和最近邻算法程序, 对不同规模的平面旅行商问题进行了数值试验, 对两种算法的求解质量进行了对比分析。结果表明: ① 绕中心周游法和最近邻算法求解质量的相对优劣取决于具体问题中城市的数量和分布; ② 对于 4 城市问题, 绕中心周游法总能得到最优解, 而最近邻算法经常不能得到最优解; ③ 对于小规模 ($n < 20$) 问题, 绕中心周游法的求解质量一般优于最近邻算法的求解质量; ④ 对于中等规模 ($20 \leq n \leq 30$) 问题, 绕中心周游法的求解质量总体上相当于最近邻算法的求解质量; ⑤ 对于大规模 ($n > 30$) 问题, 绕中心周游法的求解质量一般次于最近邻算法的求解质量。

[关键词] 平面旅行商问题; 绕中心周游法; 数值试验; 最近邻算法; 对比分析

[中图分类号] TP301

[文献标识码] A

[文章编号] 1000-7857(2007)15-0053-05

A New Algorithm for Solving Plane Travelling Salesman Problem: Travelling-around-the-center Method

YU Deming

College of Civil Engineering and Architecture, Central South University, Changsha 410075, China

Abstract: A new algorithm, called the travelling-around-the-center method, is proposed in this paper. It is a deterministic algorithm, with its time complexity the same as the nearest neighbor algorithm, that is, $O(n^2)$, where n is the number of cities. Numerical tests for plane travelling salesman problems with various sizes are carried out by using the travelling-around-the-center method and the nearest neighbor algorithm, and the performances of both algorithms are compared. The following conclusions are drawn: (1) The relative performances of the solutions of the travelling-around-the-center method and the nearest neighbor algorithm depend on the number and distribution of the cities in a specific travelling salesman problem. (2) For $n=4$, the travelling-around-the-center method can always obtain the best solution, while the nearest neighbor algorithm often cannot do so. (3) For a small size problem with $n < 20$, the travelling-around-the-center method can generally obtain better solutions than the nearest neighbor algorithm. (4) For a middle size problem with $20 \leq n \leq 30$, the quality of the solution obtained by the travelling-around-the-center method is similar to that obtained by the nearest neighbor algorithm. (5) For a large size problem with $n > 30$, the quality of the solution obtained by the travelling-around-the-center method is generally not as good as that obtained by the nearest neighbor algorithm.

Key Words: plane travelling salesman problem; travelling-around-the-center method; numerical tests; nearest neighbor algorithm; comparison analysis

CIC Number: TP301

Document Code: A

Article ID: 1000-7857(2007)15-0053-05

0 引言

旅行商问题具有很强的科学和工程背景, 它是一个容易表达和理解, 但难以求解的典型组合优化问题。

因为对于一个 n 个城市的旅行商问题, 存在 $(n-1)!/2$ 条不同的旅行回路。当 $n=20$ 时, 若使用运算能力为 $10^6/s$ 浮点运算的计算机进行求解, 需用 1 929 a 才能找到最

收稿日期: 2007-06-15

作者简介: 宇德明, 湖南省长沙市韶山南路 22 号中南大学土木建筑学院, 教授, 研究方向为工程优化与风险评价;

E-mail: dmyu@mail.csu.edu.cn

优解^[1]。因此,对于中等规模以上的旅行商问题,用穷举搜索法求最优解几乎是不可能的。为了以较高的效率和质量求解旅行商问题,研究人员提出了许多不同的算法^[2-11],如最近邻算法^[1-3]、指引最近邻算法^[2]、模拟退火法^[1]、神经网络法^[4]、遗传算法^[5]和蚁群算法^[5-7]等。这些算法各有特点,如最近邻算法求解效率很高,其时间复杂性为 $O(n^2)$,但解的质量一般较差;指引最近邻算法是一种改进的最近邻算法,其时间复杂性与最近邻算法具有相同数量级,但解的质量要优于最近邻算法;模拟退火法、神经网络法、遗传算法和蚁群算法都属于智能优化算法,它们均能获得较高质量的解,但计算量很大,且难以确定合适的算法参数。总之,如何求解旅行商问题仍是一个未有效解决的世界性难题。

受大城市修建环形道路以缓解城市交通拥挤现象的启迪,本文提出一种求解平面旅行商问题的新算法——绕中心周游法。它是一种十分简单的确定型启发式算法,其时间复杂性仅为 $O(n^2)$ 。数值试验结果表明:当城市数不超过 30 时,绕中心周游法和最近邻算法求解质量相当甚至更优;当城市数超过 30 时,绕中心周游法的求解质量一般次于最近邻算法。

1 绕中心周游法的算法描述

1.1 平面旅行商问题的表述

平面旅行商问题是指在平面笛卡儿坐标系 Oxy 中有 n 个位置已知的城市,旅行商从其中某个城市出发,不重复地访问其余 $n-1$ 个城市后又回到出发的城市,问题是应按什么顺序访问这些城市才能使总的旅行路线最短,最短路程是多少? 简单地说,就是如何在平面笛卡儿坐标系 Oxy 中确定一条经过各城市 1 次且仅 1 次的最短回路。

1.2 绕中心周游法的基本步骤

绕中心周游法的基本思想是:从城市群中心引出一条射线,让该射线绕城市群中心逆时针(或顺时针)旋转 1 周,则城市被射线扫过的顺序即是旅行商访问该城市的顺序。

用绕中心周游法求解平面旅行商问题包括如下 6 个基本步骤。

1) 求城市群中心坐标。

按下式计算 n 个城市构成的城市群中心的坐标

$$\left. \begin{aligned} x_o &= \frac{1}{n} \sum_{i=1}^n x_i \\ y_o &= \frac{1}{n} \sum_{i=1}^n y_i \end{aligned} \right\} \quad (1)$$

式中, (x_i, y_i) 是第 i 个城市的坐标, (x_o, y_o) 是城市群中心 O' 的坐标, n 是城市的个数。

2) 将坐标系 Oxy 平移到城市群中心 O' , 建立新的

坐标系 $O'x'y'$ 。

3) 求各个城市在新坐标系 $O'x'y'$ 中的坐标。

按下式计算各个城市在新坐标系 $O'x'y'$ 中的坐标 (x_i', y_i')

$$\left. \begin{aligned} x_i' &= x_i - x_o' & (i=1, 2, \dots, n) \\ y_i' &= y_i - y_o' & (i=1, 2, \dots, n) \end{aligned} \right\} \quad (2)$$

4) 求各个城市与 $O'x'$ 的夹角。

按下式计算各个城市与 $O'x'$ 的夹角(见图 1)。

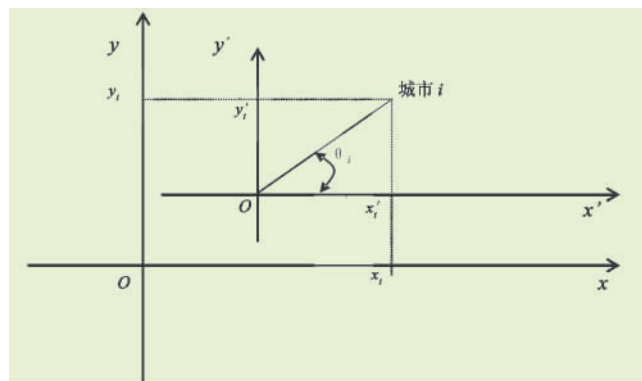


图 1 城市与 $O'x'$ 的夹角示意图

Fig. 1 Schematic diagram for determining the angle of a city with respect to coordinate axis

$$\left. \begin{aligned} \theta_i &= \arctan(y_i'/x_i') & (x_i' > 0, y_i' \geq 0) \\ \theta_i &= \pi + \arctan(y_i'/x_i') & (x_i' < 0, y_i' \geq 0) \\ \theta_i &= \pi + \arctan(y_i'/x_i') & (x_i' < 0, y_i' < 0) \\ \theta_i &= 2\pi + \arctan(y_i'/x_i') & (x_i' > 0, y_i' < 0) \\ \theta_i &= \pi/2 & (x_i' = 0, y_i' > 0) \\ \theta_i &= 3\pi/2 & (x_i' = 0, y_i' \leq 0) \end{aligned} \right\} \quad (3)$$

式中, π 为圆周率, θ_i 是城市 i ($i=1, 2, \dots, n$) 与 $O'x'$ 的夹角, 取值范围为 $[0, 2\pi)$ 。

5) 确定旅行回路。

按照下列原则排列各个城市的顺序: 将各个城市与 $O'x'$ 的夹角 θ_i 按照由小到大原则进行排列; 如果两个城市与 $O'x'$ 的夹角相等, 则按照它们在新坐标系 $O'x'y'$ 中的横坐标 x_i' 由小到大原则进行排列; 如果两个城市在新坐标系 $O'x'y'$ 中的横坐标 x_i' 也相等, 则按照它们在新坐标系 $O'x'y'$ 中的纵坐标 y_i' 由小到大原则进行排列。设最终得到的城市排列顺序为 $\pi_1, \pi_2, \dots, \pi_n$, 则回路 $\pi_1 \rightarrow \pi_2 \rightarrow \dots \rightarrow \pi_n \rightarrow \pi_1$ 即为所要求的旅行回路。

6) 确定旅行回路的长度。

将回路上相邻两城市间的距离相加, 可得到旅行回路的长度 L , 即

$$L = \sum_{i=1}^n d_{\pi_i, \pi_{i+1}} = \sum_{i=1}^n \sqrt{(x_{\pi_i} - x_{\pi_{i+1}})^2 + (y_{\pi_i} - y_{\pi_{i+1}})^2} \quad (4)$$

式中, 假设 $\pi_{n+1} = \pi_1$ 。

1.3 绕中心周游法的复杂性分析

分析前面提出的绕中心周游法的基本步骤可以发现,该算法的主要运算是求坐标运算、求角度运算、求距离运算和比较运算,其中求坐标运算 $2(n+1)$ 次、求角度运算 n 次、求距离运算 n 次、比较运算 $n(n-1)/2$ 次。因此,绕中心周游法的算法复杂性为 $0.5n^2+3.5n+2$ 。忽略低阶项,则绕中心周游法的算法复杂性仅为 $O(n^2)$,与最近邻算法相同。

2 平面旅行商问题数值试验

2.1 随机平面旅行商问题数值试验

在边长为 1 的正方形内随机产生 n 个城市,构成城市数为 n 的随机平面旅行商问题实例,分别用绕中心周游法和最小近邻法(从城市 1 出发)求解,每组运行 10 次。数值试验结果见表 1~3。表 2、表 3 和文中其他地方提到的相对误差,等于绕中心周游法得到回路的平均长度与最小近邻法得到回路的平均长度之差除以最小近邻法得到回路的平均长度。

分析表 1 中数据可以发现,当 $n=4$ 时,在 10 次数值试验中,绕中心周游法得到的回路长度总是小于或等于最近邻算法得到的回路长度。事实上,绕中心周游法每次都得到了最优长度,而最近邻算法只有 6 次得到了最优长度。例如,在第 5 组试验中,绕中心周游法和最近邻算法得到的回路长度分别为 2.58 和 2.89,前者比后者缩短了 11%。

分析表 2 中数据可以发现,城市分布情况对两种算法求解质量的相对优劣具有重要影响。例如,当 $n=30$ 时,在第 7 组试验中,绕中心周游法得到的回路平均长度比最近邻算法得到的回路平均长度缩短 11%,而在第 8 组试验中,绕中心周游法得到的回路平均长度比最近邻算法得到的回路平均长度长 15%。

表 1 随机 4 城市平面旅行商问题数值试验结果
Table 1 Results of numerical tests for plane travelling salesman problems with 4 cities

仿真序号	绕中心周游法		最近邻算法	
	回路	长度	回路	长度
1	1-2-3-4-1	1.98	1-2-3-4-1	1.98
2	1-2-4-3-1	2.49	1-2-4-3-1	2.49
3	1-3-2-4-1	2.78	1-3-2-4-1	2.78
4	1-3-2-4-1	1.56	1-2-4-3-1	1.58
5	1-3-2-4-1	2.58	1-3-4-2-1	2.89
6	1-3-2-4-1	1.92	1-2-4-3-1	1.95
7	1-2-4-3-1	1.33	1-2-4-3-1	1.33
8	1-3-4-2-1	1.10	1-3-2-4-1	1.17
9	1-4-2-3-1	1.69	1-4-2-3-1	1.69
10	1-4-3-2-1	2.38	1-4-3-2-1	2.38
平均		1.98		2.02

表 2 随机 30 城市平面旅行商问题数值试验结果
Table 2 Results of numerical tests for plane travelling salesman problems with 30 cities

仿真序号	回路长度		相对误差/%
	绕中心周游法	最近邻算法	
1	5.61	5.26	7
2	5.76	5.33	8
3	5.24	5.23	0
4	4.75	5.03	-6
5	5.32	6.10	-13
6	5.26	6.08	-13
7	5.41	6.05	-11
8	5.77	5.00	15
9	5.36	5.43	-1
10	6.55	6.30	4
平均	5.50	5.58	-1

分析表 3 中数据可以发现:① 对于小规模($n<20$)问题,绕中心周游法的求解质量一般优于最近邻算法的求解质量。例如,当 $n=15$ 时,绕中心周游法得到的回路平均长度比最近邻算法得到的回路平均长度缩短 10%。② 对于中等规模($20\leq n\leq 30$)问题,绕中心周游法的求解质量总体上相当于最近邻算法的求解质量,例如,当 $n=25$ 时,绕中心周游法得到的回路平均长度与最近邻算法得到的回路平均长度相差不到 1%。③ 对

表 3 随机平面旅行商问题数值试验汇总结果
Table 3 Summary results of numerical tests for plane travelling salesman problems

城市数 n	平均回路长度		相对误差/%
	绕中心周游法	最近邻算法	
4	1.98	2.02	-2
10	2.96	3.13	-5
15	3.59	4.00	-10
16	3.86	4.19	-8
17	3.85	4.14	-7
18	4.27	4.36	-2
19	4.67	4.83	-3
20	4.43	4.45	0
25	5.00	4.98	0
30	5.50	5.58	-1
31	5.72	5.35	7
32	6.01	5.81	3
33	6.31	5.94	6
34	6.41	6.04	6
35	6.43	5.87	10
40	7.90	6.41	23
50	7.88	7.04	12

于大规模($n>30$)问题,绕中心周游法的求解质量一般次于最近邻算法的求解质量。例如,当 $n=35$ 时,绕中心周游法得到的回路平均长度比最近邻算法得到的回路平均长度长 10%。

2.2 标准平面旅行商问题数值试验

下面分别用绕中心周游法和最近邻算法求解文献[5]、[12]和[13]中提供的 9 个标准平面旅行商问题,即:10 个城市的 Hopfield10,30 个城市的 Oliver30,48 个城市的 att48,51 个城市的 Eil51,70 个城市的 st70,76 个城市的 Eil76,100 个城市的 KroA100,130 个城市的 Ch130 和 225 个城市的 tsp225 问题。表 4 列出了求解 Hopfield10 问题的详细结果,表 5 为求解其余 8 个问题的汇总结果。图 2~4 分别是用绕中心周游法求解 Hopfield10, Oliver30 和 Eil51 问题得到的旅行回路。

分析表 4 和表 5 中的数据可以发现: ① 利用最近邻算法求解 Hopfield10 问题时,得到的解的质量与出发城市的选取有很大关系,其最优回路和最差回路长度分别为 2.69 和 3.41,二者相差 27%; ② 用绕中心周游法求解 Hopfield10 问题,得到的回路与用最近邻算法得到的最优回路相同,长度为 2.69,比用最近邻算法得到回路的平均长度缩短 7%; ③ 用绕中心周游法求解 Oliver30 问题,得到的回路长度比用最近邻算法得到回路的平均长度缩短 8%; ④ 用绕中心周游法求解城市数大于 40 的其他标准平面旅行商问题,得到解的

表 4 利用绕中心周游法和最近邻算法求解 Hopfield10 问题所得结果
Table 4 Results of Hopfield10 problem by the travelling-around-the-center method and the nearest neighbor algorithm

算法名称	旅行回路	回路长度	备注
最近邻算法	1-10-9-8-7-6-5-4-3-2-1*	2.78	① 标同一符号的为同一回路
	2-3-1-10-9-8-7-6-5-4-2**	2.84	
	3-2-1-10-9-8-7-6-5-4-3*	2.78	② 最近邻算法的回路长度与出发城市的选取有关
	4-5-7-6-8-9-10-1-3-2-4***	2.90	
	5-4-1-10-9-8-7-6-2-3-5	3.41	
	6-7-8-9-10-1-3-2-4-5-6**	2.84	③ 最近邻算法的平均回路长度为 2.89
	7-6-8-9-10-1-3-2-4-5-7***	2.90	
	8-9-10-7-6-5-4-1-3-2-8	2.98	
	9-10-8-7-6-5-4-1-3-2-9	2.77	
	10-9-8-7-6-5-4-1-3-2-10****	2.69	
绕中心周游	1-3-2-10-9-8-7-6-5-4-1****	2.69	

表 5 其他标准平面旅行商问题求解结果
Table 5 Results for other standard plane travelling salesman problems

标准平面旅行商问题名称	绕中心周游法	最近邻算法 *	相对误差 / % **
Hopfield10	2.69	2.78(2.89)	-3(-7)
Oliver30	492	573(536)	-14(-8)
att48	54 431	40 526(41 718)	34(30)
Eil51	660	514(563)	28(17)
st70	1 131	806(826)	40(37)
Eil76	935.6	712(665)	31(41)
KroA100	50 542	26 856(27 045)	88(87)
Ch130	13 319	7 575(7 720)	76(73)
tsp225	12 525.23	4 829(4 919)	159(155)

注: * 括号外的数字对应从城市 1 出发得到回路的长度,括号内的数字对应从不同城市出发得到回路的平均长度。
** 括号外的数字对应从城市 1 出发得到回路长度的相对误差,括号内的数字对应从不同城市出发得到回路平均长度的相对误差。

质量不如用最近邻算法得到解的质量。例如,用绕中心周游法求解 Eil51 问题,得到的回路长度为 660,而用

最近邻算法求解 Eil51 问题,得到的回路平均长度为 563,前者比后者长 17%。

回路为
1-3-2-10-9-
8-7-6-5-4-1



图 2 用绕中心周游法求解 Hopfield10 问题得到的回路
Fig. 2 Loop of Hopfield10 problem obtained by the travelling-around-the-center method

回路为
1-2-6-4-5-13-12-30-
23-22-17-16-28-29-
27-26-25-24-8-15-14-
7-11-10-21-19-20-9-
18-3-1

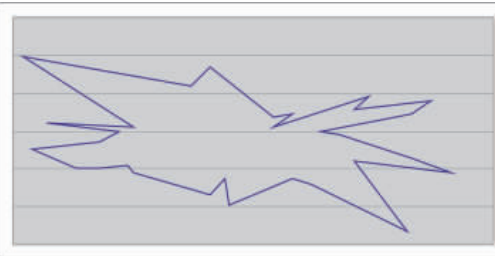


图 3 用绕中心周游法求解 Oliver30 问题得到的回路
Fig. 3 Loop of Oliver30 problem obtained by the travelling-around-the-center method

回路为

1-31-8-26-27-48-7-23-43-6-
24-51-14-46-25-18-13-47-4-
41-40-19-12-17-42-44-37-
15-45-33-5-10-39-49-30-38-
9-34-50-21-16-11-29-2-20-
35-36-3-32-22-28-1

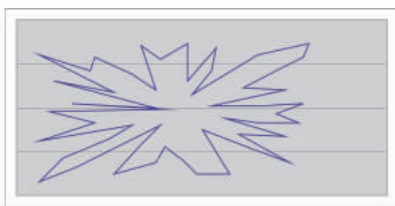


图 4 用绕中心周游法求解 Eil51 问题得到的回路

Fig. 4 Loop of Eil51 problem obtained by the travelling-around-the-center method

这些发现进一步验证了前面从随机数值试验中得出的结论。

3 结论

1) 提出了一种求解平面旅行商问题的新确定型算法——绕中心周游法, 该算法时间复杂性与最近邻算法相当, 为 $O(n^2)$, 求解质量稳定。

2) 对绕中心周游法和最近邻算法的求解质量进行了比较研究, 发现两种算法求解质量的相对优劣取决于具体问题中城市的数量和分布: ① 对于 4 城市问题, 绕中心周游法总能得到最优解, 而最近邻算法经常不能得到最优解; ② 对于小规模 ($n < 20$) 问题, 绕中心周游法的求解质量一般优于最近邻算法的求解质量; ③ 对于中等规模 ($20 \leq n \leq 30$) 问题, 绕中心周游法的求解质量总体上相当于最近邻算法的求解质量; ④ 对于大规模 ($n > 30$) 问题, 绕中心周游法的求解质量一般次于最近邻算法的求解质量。

参考文献 (References)

- [1] 康立山, 谢云, 等. 非数值并行算法 (第 1 册)——模拟退火法[M]. 北京: 科学出版社, 2003: 6-21.
KANG Lishan, XIE Yun, *et al.* Non-numerical parallel algorithm (I) Simulation annealing method [M]. Beijing: Science Press, 2003: 6-21.
- [2] 苏丽杰, 聂义勇, 李长军. 求解旅行商问题的两个启发算法的改进[J]. 计算机应用, 2004, 24: 258-260.
SU Lijie, NIE Yiyong, LI Changjun. The improvement of two heuristic algorithm for solving the TSP [J]. Computer Applications, 2004, 24: 258-260.
- [3] 王凌, 童行行, 郑大钟. TSP 基于参考点的相邻插入法和两阶段方法[J]. 控制与决策, 2004, 19(7): 831-837.
WANG Ling, TONG Hanghang, ZHENG Dazhong. Reference point based near insertion approach and two-stage approach for TSP [J]. Control and Decision, 2004, 19(7): 831-837.
- [4] 黎明, 杨小芹, 周林霞. 基于局部进化的 Hopfield 神经网络的优化计算方法[J]. 中国图象图形学报, 2004, 9(2): 207-213.
LI Ming, YANG Xiaoqin, ZHOU Linxia. Optimization with partially evolved Hopfield neural networks [J]. China Journal of Image and Graphics, 2004, 9(2): 207-213.
- [5] 李士勇. 蚁群算法及其实现 [M]. 哈尔滨: 哈尔滨工业大学出版社, 2004: 42-233.
LI Shiyong. Ant group algorithm and its realization [M]. Harbin: Harbin Industry University Press, 2004: 42-233.
- [6] 萧蕴诗, 李炳宇, 吴启迪. 求解 TSP 问题的模式学习并行蚁群算法[J]. 控制与决策, 2004, 19(8): 885-888.
XIAO Yunshi, LI Bingyu, WU Qidi. Parallel model-learning ant colony optimization algorithm for TSP [J]. Control and Decision, 2004, 19(8): 885-888.
- [7] 叶志伟, 郑肇葆. 蚁群算法中参数 α, β, ρ 设置的研究——以 TSP 问题为例[J]. 武汉大学学报(信息科学版), 2004, 29(7): 597-601.
YE Zhiwei, ZHENG Zhaobao. Configuration of parameters α, β, ρ in Ant Algorithm [J]. Geomatics and Information Science of Wuhan University, 2004, 29(7): 597-601.
- [8] 万颖瑜, 周智, 陈国良, 等. SizeScale 求解旅行商问题(TSP)的新算法[J]. 计算机研究与发展, 2002, 39(10): 1294-1302.
WAN Yingyu, ZHOU Zhi, CHEN Guoliang, *et al.* Size-Scale: new algorithm for the traveling salesman problem [J]. Journal of Computer Research and Development, 2002, 39(10): 1294-1302.
- [9] 周培德. 几何算法求解货郎担问题 [J]. 计算机研究与发展, 1995, 32(10): 63-65.
ZHOU Peide. Using a geometrical algorithm to solve TSP [J]. Journal of Computer Research and Development, 1995, 32(10): 63-65.
- [10] 周培德, 周忠平, 张欢. 寻求中国货郎担问题最短回路的多项式时间算法[J]. 北京理工大学学报, 2000, 20(2): 201-204.
ZHOU Peide, ZHOU Zhongping, ZHANG Huan. A polynomial time algorithm for solving the shortest circuit of China traveling salesman problem [J]. Journal of Beijing Institute of Technology, 2000, 20(2): 201-204.
- [11] 潘立登, 黄晓峰. 用启发式贪心法求解旅行商问题 [J]. 北京化工大学学报, 1998, 25(2): 46-51.
PAN Lideng, HUANG Xiaofeng. A heuristic greedy method for the traveling salesman problem [J]. Journal of Beijing University of Chemical Technology, 1998, 25(2): 46-51.
- [12] REINELT G. A traveling salesman problem library [JO]. ORSA Journal on Computing, 1991 (3): 376-384. <http://elib.zib.de/pub/Packages/mp-testdata/tsp/tsplib>.
- [13] APPLEGATE D L, BIXBY R E, CHVATAL V, *et al.* The traveling salesman problem—A computational study [JO]. Princeton Series in Applied Mathematics, 2006. <http://www.iwr.uni-heidelberg.de/groups/comopt/software/TSPLIB95/index.html>.

(责任编辑 代丽)