

文章编号:1000-6788(2006)05-0093-06

一种改进的蚁群算法及其在 TSP 中的应用

屈稳太,丁伟

(浙江大学电气工程学院系统科学与工程学系,浙江 杭州 310027)

摘要: 为了提高传统蚁群优化算法求解的质量,对传统的蚁群优化算法进行了改进,引进了一种信息素适时交换方法,同时在信息素积累的过程中,自适应地改变信息素的挥发率,将算法中的正反馈作用抑制到适当的程度,扩大了可行解的范围,避免了算法过早的停滞,提高了解的质量,同时算法的收敛速度没有明显的降低.通过三种 TSP 问题的仿真实验,证明该算法具有较强的发现较好解的能力,解的稳定性也比较好.

关键词: 蚁群算法;正反馈;优化;旅行商问题

中图分类号: TP18

文献标识码: A

An Improved Ant Colony Algorithm and Application in the TSP

QU Wentai, DING Wei

(Department of System Science and Engineering, College of Electrical Engineering, Zhejiang University, Hangzhou 310027, China)

Abstract: In order to improve the earlier stagnation in the conventional ant colony optimization, which easily leads to local optimal solution, an improved algorithm was proposed. In the algorithm, a new mechanism of trail information exchange between edges was introduced; on the other hand, the trail information volatilization was modified adaptively with the algorithm operating. By those, the function of positive feedback in ACO was suppressed to a reasonably degree so that the algorithm will not stopped earlier, the area of feasible solutions was expanded, and hence, a better solution can likely be got, at the same time the convergence speed was not reduced distinctly. Experimental results on three TSPs show that the algorithm has more powerful capacity of finding global solution and stability than that of conventional ant colony optimization.

Key words: ant colony algorithm; positive feedback; optimization; traveling salesman problem

1 引言

复合式优化问题 (combinational optimization problem) 具有广泛的实际应用背景,但是这是一个 NP-Hard 问题,一直没有得到很好的解决.上世纪九十年代提出的蚁群优化 (ACO: Ant Colony Optimization) 算法^[1,2]作为一种新的启发式优化算法,对解决这类问题表现出了较好的效果.蚁群优化算法最先被应用于解决旅行商问题 (TSP: Traveling Salesman Problem)^[3],近来被扩展或改造以提升其本身的功能或解决其它选择问题,比如二次指派问题 (QAP: quadratic assignment problem)^[4],车辆行程规划问题 (VRP: vehicle routing problem)^[5,6],多目标 JIT 排序问题 (Just-In-Time sequencing problem with multiple objectives)^[7],图的着色问题 (graph coloring problem)^[8],系统建模与参数估计 (system modeling and parameter estimating)^[9,10]等等.就许多此类问题而言,蚂蚁系统显示了有效的解决问题的能力.

实验观察表明,蚂蚁在运动过程中会留下一种分泌物(在人工蚂蚁系统中,简称信息素),其后面的蚂蚁可根据前边走过的蚂蚁所留下的分泌物选择其要走的路径,一条路径上的分泌物越多,蚂蚁选择这条路径的概率就越大.蚂蚁之间通过这种信息交流寻求通向食物的最短路径.蚁群算法正是模拟了这样的正反馈机制,通过个体之间的信息交流与相互协作最终找到最优解.

收稿日期:2005-02-22

作者简介:屈稳太(1963-)博士,副教授,主要研究领域:系统分析,智能控制与模式识别, E-mail:wentaiqu@zju.edu.cn;
丁伟(1984-)博士生,主要研究领域:机器视觉与机器人控制.

但是,蚁群算法也有一些不足:1) 由于蚁群中多个个体的运动是随机的,当群体规模较大时,要找出一条较好的路径需要较长的搜索时间,2) 由于利用了正反馈机理,正反馈强度的大小影响解的性能,如果正反馈太强,优化过程容易陷于局部最优解;正反馈强度太小,优化过程太慢,需要较长的搜索时间.因此针对算法的不足,提出了许多种改进的方法,其中 M. Dorigo 等人在基本的蚁群算法的基础上提出的改良版的蚁群系统^[1],让信息素强度最大的边以较大的概率被选中,强化了最优信息的反馈,但容易出现算法较早停滞的现象. T. Stutzle 等人提出了最大-最小蚂蚁系统(Max-Min Ant System)^[11]允许各个路径上的信息素在一个限定的范围内变化. L. M. Gambardella 等人提出了一种混合型蚁群算法 HAS^[12],在每次循环中蚂蚁建立各自的解后,再以各自的解为起点用某种局部搜索算法求局部最优解,作为相应蚂蚁的解,这样可以迅速提高解的质量,中国学者陈文生提出了基于分布均匀度的自适应蚁群算法^[13],在解决 TSP 上,得到了不同程度的成功.

本文对蚁群算法中的信息素的积累及信息的交换方式进行了改进,提出了一种新的信息交换模式和自适应的信息素挥发因子,避免了算法过早停滞的缺点,提高了解的质量,同时算法的收敛速度没有明显的降低.通过对 TSP 问题的仿真试验,验证了该方法的有效性.

2 基本原理与算法分析

2.1 TSP 原型

设有 n 个城市集 $C = (1, 2, \dots, n)$,任意两个城市 i, j 之间的距离为 d_{ij} ,求一条经过每个城市仅一次的路径 $\pi = (\pi(1), \pi(2), \dots, \pi(n))$,使得

$$\min \left(\sum_{i=1}^{n-1} d_{(\pi(i), \pi(i+1))} + d_{(\pi(n), \pi(1))} \right). \tag{1}$$

2.2 基于蚁群算法的 TSP 求解

用 $b_i(t)$ 表示 t 时刻位于城市 i 的蚂蚁的个数, $m = \sum_{i=1}^n b_i(t)$ 为蚂蚁的总数. $\tau_{ij}(t)$ 表示 t 时刻边 ij 上的信息素, $\tau_{ij}(0)$ 为 $\tau_{ij}(t)$ 的初值(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, $1 - \rho_{ij}$ 表示信息素的挥发快慢.当所有蚂蚁完成一次周游后,各条边上的信息素按下式调整:

$$\tau_{ij}(t+1) = \rho_{ij} \tau_{ij}(t) + \sum_{k=1}^m \Delta \tau_{ij}^k, \tag{2}$$

$$\Delta \tau_{ij}^k = \begin{cases} \frac{Q}{L_k} & \text{if } k \text{ th ant passed the } ij \text{ edge in this tour} \\ 0 & \text{else} \end{cases} \tag{3}$$

$\Delta \tau_{ij}^k$ 表示本次周游中路径 ij 上的信息素增量,初始时刻, $\Delta \tau_{ij}^k = 0$. τ_{ij}^k 表示第 k 只蚂蚁在周游过程中释放在边 ij 上的信息素.

$$\Delta \tau_{ij}^k = \begin{cases} \frac{Q}{L_k}, & \text{if } k \text{ th ant passed the } ij \text{ edge in this tour} \\ 0, & \text{else} \end{cases} \tag{4}$$

Q 为常数, L_k 表示本次周游后,第 k 只蚂蚁所形成的回路的长度.

蚂蚁在周游时,向那个城市转移由转移概率 p_{ij}^k 决定.

$$p_{ij}^k = \begin{cases} \frac{\tau_{ij} \eta_{ij}}{\sum_{s \in \text{allowed}_k} \tau_{is} \eta_{is}}, & j \in \text{allowed}_k \\ 0, & \text{else} \end{cases} \tag{5}$$

其中 $\text{allowed}_k = \{0, 1, \dots, n-1\} - \text{tabu}_k$ 表示蚂蚁 k 当前能选择的的城市集合, tabu_k 为禁忌表,它记录蚂蚁 k 已路过的城市,用来说明人工蚂蚁的记忆性, η_{ij} 是某种启发信息, τ_{ij} 体现了信息素和启发信息对蚂蚁决策的影响.

在 TSP 问题中,ACO 基本的运行过程是这样的: m 只蚂蚁同时从某个城市出发,根据(5)选择下一次旅行的城市,已去过的城市放入 tabu_k 中,一次循环完成后,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.

2.3 改进的蚂蚁算法

在 ACO 中,选择概率 p_{ij}^k 对算法的执行效率和最终的结果都有重要的影响,在 Dorigo 提出的算法中,选取 $p_{ij} = d(c_i, c_j)^{-1}$, α 取为一个小的常数.这样的处理方式在算法的执行效率和最终的结果方面并不是最优的.

对 ACO 来说, $\tau_{ij}(t)$ ($j = i, j = 1, 2, \dots, n$) 之间的相对大小会直接影响城市 i 到城市 j 的转移概率 p_{ij}^k , 从而影响解的质量.在搜索早期,信息素的分布是分散的,随着搜索进行,信息素逐渐集中到少数边上,搜索方向也随之基本上确定下来.当某些边上的信息素强度明显高于其余边时,导致在搜索时,总是在少数几条边上进行,这样就会使解的结构过于相似,搜索过程也会停顿下来,算法陷入局部最优解.避免这种现象的方法就是要增加解的多样性.

在 ACO 中增加解的多样性,关键在于形成和保持一个合理的信息素强度的分布,在构造可行解时,让较多的边能参与进去.也就是在开发新途径和利用旧途径之间建立一个平衡点,既充分利用正反馈机制加快搜索,又尽可能地扩大算法的搜索区域,使用更多的边形成新解.陈耀提出了基于分布均匀度的自适应蚁群算法^[13],根据优化过程中的解的分布均匀度,自适应的调整路径选择概率和信息更新策略,在防止算法过早停滞和加速瘦脸上取得了较好的平衡.

本文在正反馈机制的基础上,受遗传算法中交叉与变异的启发,依概率交换部分边上的信息素,改变各边上信息素 $\tau_{ij}(t)$ 之间的相对大小和信息素的分布,让更多的边进入可行解域中,扩大解的可行域.

调整原理:如果是 n 个城市的 TSP,则在第 k 次循环结束后,将会形成表 1 所示的信息素 $\{\tau_{ij}^k | i, j = 1, 2, \dots, n\}$ 列表.

如果是对称的 TSP,则 $\tau_{ij}^k = \tau_{ji}^k$.就第 i 个城市来说,对各条边上的信息素 τ_{ij}^k ($j = 1, 2, \dots, n$) 进行排序

$$\tau_{i1}^k > \tau_{i2}^k > \dots > \tau_{in}^k. \quad (6)$$

阈值 τ_{th}^k 将随着迭代次数的增多逐渐增大.对信息素序列 (6) 进行 1/2 截断,截断后序列被分为高强度的信息素段与低强度信息素段两部分,将高强度的信息素段与低强度信息素段依照概率进行交换,信息素强度高的边交换的概率大,信息素强度低的边交换的概率小.如图 1 所示.

通过这样的调整,改变了各边上信息素 $\tau_{ij}(t)$ 之间的相对大小和信息素分布,避免了一些信息素强度低的边过早的被淘汰,从而使得更多的边进入可行解域中.

其具体步骤为:

Step1:初始化, $t = 0, NC = 1, \tau_{ij}(0) = 1, \tau_{ij}(0) = \alpha d_{ij}^{-1}, \tau_{ij} = 0$,对 n 个城市,设定城市之间的边(距离)矩阵 $D = \{d_{ij} | (i, j = 1, 2, \dots, n)\}$;设置信息素的初值 $\tau_{ij}(0) = 0$,为每个城市设定一个交换概率 $p_i, i = 1, 2, \dots, n$,设置蚂蚁总数为 M .

Step2:让 M 只蚂蚁从某一城市出发,蚂蚁依据概率(5)选择下一个城市(初次周游时,因为没有信息素的积累, p_{ij} 是不定的,选择是随机的),周游完一个循环后,依据(2~4)式计算信息素的积累.

Step3:当所有蚂蚁完成一次周游后,先根据 $\tau_{ij}(t)$ 对各边的信息素强度进行调整,调整原理如上所述.对每个城市 i ,通过转轮法产生多个 $(0, 1)$ 之间均匀分布随机数 r_i ,对高强度信息素段和低强度信息素段同时进行,判断 r_i 落在哪一个区间,就选择那一个边进行交换.随机数 r_i 的个数决定了交换的个数,也就是决定了可行解的范围,个数越多,可行解范围越大,但是算法搜索时间越长,因此,随机数 r_i 的个数可根据计算精度的要求和问题的规模折衷选择.

表 1 信息素列表

	1	2	3	...	n
1	0	τ_{12}^k	τ_{13}^k	...	τ_{1n}^k
2	τ_{21}^k	0	τ_{23}^k	...	τ_{2n}^k
3	τ_{31}^k	τ_{32}^k	0	...	τ_{3n}^k
...
n	τ_{n1}^k	τ_{n2}^k	τ_{n3}^k	...	τ_{nn}^k

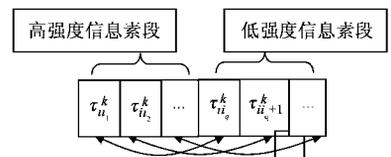


图 1 信息素的交换

Step4: $NC = NC + 1, t = t + 1$, 检验解的精度是否达到要求, 或者循环次数是否达到, 如果没有达到, 则转 Step2, 否则结束.

信息素的挥发也影响着信息素的分布. 如何确定 (2) 式中 τ_{ij} 的大小, 是应用 ACO 算法的另一个关键. 通常 τ_{ij} 被设置成常数, 由经验或实验而定, 且每条边的挥发率相同. 实际上, 在以城市 i 为起点与其它城市形成的 $n - 1$ 条边中, 不一定每次每条边上的信息素都要挥发, 各个边之间也应有所差别. 例如: 在搜索刚开始的一段时间内, 让所有边上的信息素都不减少, 这种处理能够加快搜索过程. 经过一段时间后, 各条边上信息素的挥发将会出现不同. 一般情况下, $\tau_{ij}(t)$ 较大的边用于构造可行解的概率较高, 为了防止部分边上的信息素浓度过大, 相应地, 这些边上的信息素应挥发快一些, 相应的 τ_{ij} 值应该小一些; 而为了避免最优路径上的某些边由于信息素强度过低而失去选择机会, 这些边的 τ_{ij} 值应该较大.

$\tau_{ij}(t)$ 按下式确定

$$\tau_{ij}(t) = \begin{cases} \tau_{ij}(t_h) & t = t_h; \\ k_1 \tau_{ij}(t) & C, t > t_h. \\ k_2 & \text{else} \end{cases} \quad (7)$$

$\tau_{ij}(t)$ 表示 t 时刻边 ij 上信息素挥发率, t_h 是早期搜索时间, $k_1, k_2 \in (0, 1), k_1 > k_2$, 但两者的差别不能太大. $C \in [\tau_{i,\max}, \tau_{i,\text{mean}}]$, 为一常数

$$\tau_{i,\max} = \max_{j=1, \dots, n-j} \{ \tau_{ij}(t) \}, \quad (9)$$

$$\tau_{i,\text{mean}} = \frac{1}{n-1} \sum_{j=1, j \neq i}^n \tau_{ij}(t). \quad (10)$$

3 仿真试验

通过对随机产生的 50 个城市、75 个城市以及从通用 TSPLIB^[14] 中选用的 ch150 (150 个城市) TSP 问题进行仿真研究, 分别采用基本的蚁群算法 Basic-ACO^[3]、基于分布均匀度的自适应蚁群算法 Chen-ACO^[13] 和本文提出的改进型蚁群算法 Improved-ACO 进行仿真, 比较几种算法的计算结果. 仿真采用 Matlab6.5 软件编程, 每个程序运行 20 次, 仿真结果如表 2 所示. 针对 50 个城市和 75 个城市, 分别用三种计算方法所获得的最优路径如图 2、图 3 所示.

表 2 三种算法的计算结果

参数设定, $\alpha = 0.4, \beta = 0.8, p_i = 0.1, k_1 = 0.99, M = 125$

算法	50-city 问题		75-city 问题		ch150 问题 (标准解: 6528)	
	平均值	最好结果	平均值	最好结果	平均值	最好结果
Basic-ACO	434.2	430.4	552.3	549.9	7512.6	7472.4
Chen-ACO	428.9	428.5	548.6	545.8	6871.3	6786.0
Improved-ACO	427.5	427.3	544.6	543.0	6729.6	6617.8

50 个城市的 TSP 问题在 Basic-ACO 求解下的最优路径长度为 430.4, 路径选择如图 2(a) 所示; 在 Chen-ACO 求解下的最优路径长度为 428.5, 路径选择如图 2(b) 所示; 在 Improved-ACO 求解下的最优路径长度为 427.3, 路径选择如图 2(c) 所示. 图 3 所示为 75 个城市的 TSP 问题的路径图. 在 Basic-ACO 求解下的最优路径长度为 549.9, 路径选择如图 3(a) 所示, 在 Chen-ACO 求解下的最优路径长度为 545.8, 路径选择如图 3(b) 所示; 在 Improved-ACO 求解下的最优路径长度为 543.0, 路径选择如图 3(c). ch150 问题的标准解为 6528^[14], 在 Basic-ACO 求解下, 求得最优路径长度为 7472.4, 在 Chen-ACO 求解下, 求得最优路径长度为 6786.0, 在 Improved-ACO 求解下, 求得最优路径长度为 6617.8.

从表 1 可以发现, 改进后 ACO 算法明显优于 Basic-ACO 算法, 较之 Chen-ACO 算法也要好一些. 分析原因, 主要是 Basic-ACO 算法过分加强了信息的正反馈作用, 虽然加快了算法的收敛速度, 但易于使算法过

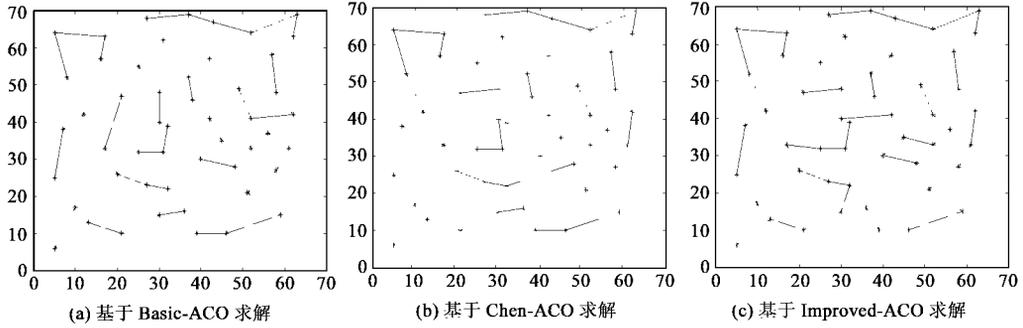


图 2 50 个城市的 TSP 路径

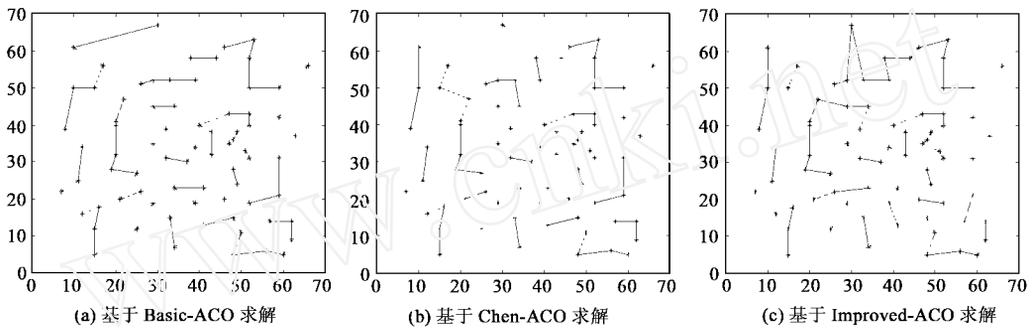


图 3 75 个城市的 TSP 路径

早停滞,陷入局部解;而 Chen-ACO 算法在加速算法收敛与防止算法停止之间取得了较好的平衡,但是在信息素重新调整的过程中,使用了一种基于信息素均匀度的方法,该方法虽然能动态地调整各路径上信息素的分布,使其不至于过分集中或分散,但这种调整策略是一种固定的模式,当信息分布过分集中在某些边上时,采取一种大幅度消减其信息素的策略,这样做其解的多样性和算法收敛速度都受到一定的影响.本算法通过边的信息素的随机交换,交换的概率依赖于各边的信息素强度,从而使得可行解的范围扩大,而又保证了一定的正反馈作用,改善了算法的求解质量,解的稳定性也比较好,其平均值与最优值之间的误差保持在 2% 之内.

4 结论

本文对传统蚁群优化算法容易陷于局部最优解的不足进行了改进.传统的蚁群优化算法是基于一种启发式的正反馈的搜索机制,当算法进行到一定程度时,由于正反馈的作用使得算法较早停止,陷于局部最优解.本文主要采用了一种信息素交换的思想,当信息素积累到一定程度时,对相关边上的信息素进行交换,扩大可行解的范围.在算法中信息素的挥发快慢对算法的过程和结果有着重要的影响,文章中对信息素挥发率进行了改进,从原来的常量变成实时可变的变量,随着算法的进行,对一些信息素浓度大的边设置较大的挥发率,而对信息素浓度小的边设置较小的挥发率,这样在一定程度上虽然抑制了正反馈的作用,但扩大了可行解的范围,提高了解的质量.通过对三个 TSP 问题的仿真实验,可以看出该算法具有较强的发现较好解的能力,不容易陷入局部最优.其原因是该算法不仅利用了正反馈原理,在一定程度上加快了优化过程,而且,考虑到蚁群算法本质上是一种并行的算法,个体之间不断的进行信息交流和传递,有利于发现较好解.所以充分利用了蚂蚁的这种信息交换机制,通过信息素的交换和信息素挥发因子的自适应改变,扩大了可行解的范围.

但这种算法在得到较好解的同时,加长了搜索时间,降低了算法的效率.因为,蚁群中各个体的运动是随机的,虽然通过信息交换能够向着最优路径进化,但是当群体规模较大时,很难在较短的时间内从大量杂乱无章的路径中找出一条较好的路径.尤其在进化的初级阶段,各个路径上的信息素相差不明显,需经过较长一段时间,通过信息正反馈,使得较好路径上的信息素逐渐增大,才能使得较好路径上的信息素明

显高于其它路径上的信息素,最终收敛于较好的路径.

参考文献:

- [1] Dorigo M, Stutzle T. Ant Colony Optimization[M]. MIT Press, Cambridge, MA, 2004.
- [2] Dorigo M, Blum C. Ant colony optimization theory: A survey[J]. Theoretical Computer Science, 2005,344:243 - 278.
- [3] Dorigo M, Gambardella L M. Ant colony system: A cooperative learning approach to the traveling salesman problem[J]. IEEE Trans Evolutionary Computation, 1997,1(1): 53 - 66.
- [4] Talbi E G, Roux O, Fonlupt C, et al. Parallel ant colonies for the quadratic assignment problem[J]. Future Generation Computer Systems(FGCS). 2001,17(4): 441 - 449.
- [5] Bullnheimer B, Hartl R F, Strauss C. An improved ant system algorithm for the vehicle routing problem[R]. Ann Oper Res, 1999, 89: 319 - 28.
- [6] Bell J E, McMullen P R. Ant colony optimization techniques for the vehicle routing problem[J]. Advanced Engineering Informatics, 2004, 18(1): 41 - 48.
- [7] McMullen P R. An ant colony optimization approach to addressing a JIT sequencing problem with multiple objectives[J]. Artificial Intelligence in Engineering, 2001,15(3): 309 - 317.
- [8] Ahn S H, Lee S G, Chung T C. Modified ant colony system for coloring graphs[A]. Proceedings of the 2003 Joint Conference of the Fourth International Conference on Information Communications and Signal Processing and the Fourth Pacific Rim Conference on Multimedia[C], 2003, 3(15 - 18): 1849 - 1853.
- [9] Shelokar P S, Jayaraman V K, Kulkarni B D. An ant colony classifier system: Application to some process engineering problems [J]. Computers and Chemical Engineering, 2004, 28(9): 1577 - 1584.
- [10] 汪镭, 吴启迪. 蚁群算法在系统辨识中的应用[J]. 自动化学报, 2003,29(1):102 - 109.
Wang L, Wu Q D. Ant system algorithm based system identification[J]. Acta Automatica Sinica, 2003,29(1):102 - 109.
- [11] Thomas S, Holger H H. Max-Min ant system [J]. Future Generation Computer Systems(FGCS), 2000, 16(8): 889 - 914.
- [12] Gambardella L M, Dorigo M. An ant colony system hybridized with a new local search for the sequential ordering problem[J]. Inform Journal on Computing, 2000,12(3):237 - 255.
- [13] 陈耀耀, 沈洁, 秦玲, 等. 基于分布均匀度的自适应蚁群算法[J]. 软件学报, 2003,14(8):1379 - 1387.
Chen L, Shen J, Qin L, et al. An adaptive ant colony algorithm based on equilibrium of distribution[J]. Journal of Software, 2003,14(8):1379 - 1387.
- [14] www.iwr.uni-heidelberg.de/group/comopt/software/TSPLIB.

(上接第 82 页)

参考文献:

- [1] 廖祖仁, 傅崇伦. 产品寿命周期费用评价法[M]. 北京:国防工业出版社, 1993,12:16 - 89.
Liao Zuren, Fu Chonglun. The Methods of Products Evaluate in Life Cycle[M]. Peking: National Defence Industry Press, 1993,16 - 89.
- [2] 徐扬光. 设备综合工程学概论[M]. 北京:国防工业出版社, 1989.
Xu Yangguang. The Generality of Facility Integration Engineering[M]. Peking: National Defence Industry Press, 55 - 102.
- [3] (美)Booz, Allen&Hamilton Inc. 美国系统工程管理[M]. 北京:航空工业出版社, 1991,63 - 88.
Booz, Allen&Hamilton Inc. American System Engineering Management[M]. Peking: Aviation Industry Press, 1991,63 - 88.
- [4] 李晓龙, 李亮, 武振业. 工程设备索赔的系统分析方法[J]. 世界科技研究与发展, 2002(2):81 - 84.
Li Xiaolong, Li liang, Wu Zhenye, The systematic analyzing methods of project equipment claims [J]. World Science Technology R&D, 2002(2):81 - 84.
- [5] 全林, 赵俊和, 成盛超. 效用评估的模糊效用方法[J]. 上海交通大学学报, 1997(4):121 - 123.
Quan Lin, Zhao Junhe, Cheng Shengchao. Fuzzy utility method of the utility evaluation [J]. Journal of Shanghai Jiaotong University, 1997(4):121 - 123.