

## 改进的增强型蚁群算法

陈宏建<sup>1</sup>, 陈 峻<sup>1,2</sup>, 徐晓华<sup>1</sup>, 屠 莉<sup>1</sup>

(1. 扬州大学信息工程学院计算机科学与工程系, 扬州 225009; 2. 南京大学软件新技术国家重点实验室, 南京 210093)

**摘 要:** 针对传统增强型蚁群算法容易出现早熟和停滞现象的缺陷, 提出了一种改进的增强型蚁群算法。该方法将传统的增强型蚁群算法和遗传算法中交叉操作和变异操作相结合, 实验结果表明, 该方法比传统的增强型蚁群算法具有更好的搜索全局最优解的能力。

**关键词:** 蚁群算法; 交叉; 变异; 优化

## An Improved Augment Ant Colony Algorithm

CHEN Hongjian<sup>1</sup>, CHEN Ling<sup>1,2</sup>, XU Xiaohua<sup>1</sup>, TU Li<sup>1</sup>

(1. Dept. of Computer Sci. &amp; Eng., College of Information Engineering, Yangzhou Univ., Yangzhou 225009;

2. National Key Lab of Novel Software Tech., Nanjing Univ., Nanjing 210093)

**【Abstract】** To overcome the default of precocity and stagnation in classical augment ant colony algorithm, an modified augment ant colony algorithm is presented. The algorithm uses the operations of crossover and mutation of GA in augment ant colony optimization. Experimental results show that the algorithm has much higher capacity of global optimization than that classical augment ant colony algorithm.

**【Key words】** Ant colony algorithm; Crossover; Mutation; Optimization

蚁群算法还存在许多缺陷, 例如蚁群中多个蚂蚁的运动是随机的, 当群体规模较大时, 很难在较短时间内从复杂无章的路径中找出一条较好的路径。为此M.Dorigo等人在基本蚁群算法的基础上提出了称之为Ant-Q System的蚁群算法<sup>[7,8]</sup>, 仅让每一次循环中最短的路径上的信息素作更新。为了克服在Ant-Q中可能出现的停滞现象, T.Stutzle等人提出了MAX-MIN Ant System<sup>[9]</sup>, 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<sup>[6]</sup>, 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。我们也在文献[10,11]中提出了两种克服蚁群算法早熟和停滞现象的蚁群算法, 另外, G.Bilchev等人在使用遗传算法解决工程设计中连续空间的优化问题时, 使用了蚁群算法对遗传算法所得到的初步结果进行精确化, 取得了较好的效果<sup>[12]</sup>。为了提高蚁群算法的速度和精确度, 吴斌、史忠植等提出了基于蚁群算法的分段求解算法<sup>[13]</sup>。丁建立、陈增强等提出了先利用遗传算法再利用蚁群算法融合的方法<sup>[14]</sup>。本文针对蚁群算法容易出现停滞现象这一缺陷, 结合文献[15]提出的增强型蚁群算法(本文称为传统增强型蚁群算法), 提出了一种改进的增强型蚁群算法, 通过增强全局(或局部)最优解和全局(或局部)次优解的路径上的信息素强度以及对它们进行交叉和变异操作的遗传优化方法来改进增强型蚁群算法, 从而有效地克服了传统增强型蚁群算法中容易陷入局部最优解的问题, 实验证明, 改进的蚁群算法比传统增强型蚁群算法具有更好的搜索全局最优解的能力, 并且其收敛性较传统增强型蚁群算法有了明显的提高。

## 1 基本蚁群算法

以TSP问题为例说明基本蚁群算法的框架。设有maxcities个城市, maxants只蚂蚁, 采用 $d_{ij}$  ( $i, j = 1, 2, \dots, \text{maxcities}$ ) 表示城市i和城市j之间的距离,  $\tau_{ij}(t)$  表示在时刻t城市i和城市

j之间的路径上的残留信息素强度, 我们以此来模拟实际蚂蚁的分泌物。蚂蚁k在行进过程中, 根据各条路径上的信息素强度来决定下一步所行进的路径, 采用 $p_{ij}^k(t)$  表示在时刻t蚂蚁k由城市i转移到城市j的概率, 则有

$$p_{ij}^k(t) = \begin{cases} \frac{\tau_{ij}^k(t) \eta_{ij}^k(t)}{\sum_{r \in \text{allowed}_k} \tau_{ir}^k(t) \eta_{ir}^k(t)}, & j \in \text{allowed}_k \\ 0, & \text{otherwise} \end{cases} \quad (1)$$

其中 $\text{allowed}_k$ 表示蚂蚁k下一步允许行进的城市集合, 它随蚂蚁k的行进过程而动态改变。信息素强度 $\tau_{ij}(t)$  随时间的推移会逐步消逝, 用P表示它的消逝程度。蚂蚁选择转移到哪个城市采用伪随机概率选择规则, 在这种规则下, 每当蚂蚁要选择转移的城市时, 就产生一个在[0, 1]范围内的随机数, 根据该随机数按下列公式确定蚂蚁转移的方向:

$$s = \begin{cases} \arg \max \{ [\tau(i, j)]^q [\eta(i, j)]^\beta \} & \text{若 } q \leq q_0 \\ S & \text{否则} \end{cases} \quad (2)$$

其中, q为一个在区间[0, 1]内的随机数,  $q_0$ 为[0, 1]内的一个参数, S由式(1)确定。这样, 经过maxcities时刻, 蚂蚁走完所有的城市, 即完成了一次循环。此时根据下式对各路径上的信息素进行更新:

$$\tau_{ij}(t+1) = (1-\rho) \cdot \tau_{ij}(t) + \Delta \tau_{ij} \quad (3)$$

$$\text{其中 } \Delta \tau_{ij} = \sum_{k=1}^m \Delta \tau_{ij}^k \quad (4)$$

$\Delta \tau_{ij}^k$  表示蚂蚁k在本次循环中在城市i和城市j之间的路

**基金项目:** 国家自然科学基金资助项目(60074013); 国家高性能计算基金资助项目(00210); 江苏省教育厅自然科学基金和南京大学软件新技术国家重点实验室开放基金资助项目

**作者简介:** 陈宏建(1968—), 男, 讲师, 主研方向为并行算法和软件工程; 陈 峻, 教授; 徐晓华、屠 莉, 硕士生

**定稿日期:** 2004-01-04 **E-mail:** yzchj@yzcn.net

径上留下的信息素,其计算方法可以根据计算模型而定,在最常用的Ant Circle System 模型中:

$$\Delta\tau_{ij} = \begin{cases} Q/L_k, & \text{若蚂蚁k在本次循环中经过城市i和城市j} \\ 0, & \text{otherwise} \end{cases} \quad (5)$$

其中,  $Q$ 为常数,  $L_k$ 为蚂蚁 $k$ 在本次循环中所行走路径的总长度,在式(1)中,  $\alpha$  表示蚂蚁在行进过程中所积累的信息素对它选择路径所起的作用程度,  $\eta_{ij}$ 表示由城市 $i$ 转移到城市 $j$ 的期望程度,可根据某种启发算法而定,例如可以取  $\eta_{ij} = 1/d_{ij}$ , 表示 $\eta_{ij}$ 的作用。当 $\alpha = 0$ 时,算法就是传统的贪心算法,而当 $\beta = 0$ 时,算法就成了纯粹的正反馈的启发式算法。可以用试验的方法确定参数 $\alpha$ ,  $\beta$  的最优组合。在经过若干次循环以后,可以根据适当的停止条件来结束计算。

## 2 改进的增强型蚁群算法

演化算法就是在“探索”和“利用”之间寻找一个平衡点,也即使使得算法的搜索空间尽可能的大,以寻找那些可能存在最优解的解空间;同时又要充分利用群体内当前具有的有效信息,使得算法的侧重点放在那些可能具有较高适应值的个体所在的空间内,从而以较大概率收缩到全局最优解。在蚁群算法中,信息素是蚁群之间进行信息交互和通信的纽带和中介,是蚁群具有智能特点的关键因素,而信息素的强度随着路径距离的减少而增加,当系统搜索到较好的解时,就更更改相关路径上的信息素强度,这有利于系统搜索较好的解,因而文献[15]提出了几种增强型蚁群算法,主要是通过增强全局最优解或局部最优解的所有路径中信息素强度来提高蚁群算法的搜索能力,但这种方法在实验过程中发现很容易陷入局部最优。受遗传算法的启示,本文在增强全局最优解或局部最优解的路径中的信息素强度的基础上,再对全局(或局部)最优解和全局(或局部)次优解进行交叉和变异操作,从而得到了一种改进的蚁群算法。以求解TSP问题为例,采用路径表示TSP问题的解,首先增强全局(或局部)最优解和全局(或局部)次优解的所有路径中的信息,然后按下列方法进行交叉和变异操作:首先在最优解和次优解的路径中选取两个交叉点,然后交换两个解路径中交叉点之间的部分路径,再在原来的各自解路径中删去与现有解路径中交叉部分相同的路径点,并按顺序将剩余路径点从左至右依次填入新的解路径中。交叉完成后,再随机地在两个解路径中选取两个变异点,设为 $x_1$ ,  $x_2$ , 将该变异点 $x_1$ 和 $x_2$ 视为路径终点,再随机地产生两个值 $y_1$ 和 $y_2$ 分别替换变异点 $x_1$ ,  $x_2$ , 同时将解路径中的路径终点为 $y_1$ 的值改为 $x_1$ 值,为 $y_2$ 的值改为 $x_2$ 值。例如:设最优解的路径为:  $3 \rightarrow 4 \rightarrow 7 \rightarrow 6 \rightarrow 5 \rightarrow 2 \rightarrow 1 \rightarrow 8$ , 次优解的路径为:  $6 \rightarrow 4 \rightarrow 8 \rightarrow 7 \rightarrow 2 \rightarrow 1 \rightarrow 3 \rightarrow 5$ , 竖线之间的部分为交叉段,则交叉后两个解路径变为  $3 \rightarrow 4 \rightarrow 6 \rightarrow 7 \rightarrow 2 \rightarrow 1 \rightarrow 5 \rightarrow 8$ 和  $4 \rightarrow 8 \rightarrow 7 \rightarrow 6 \rightarrow 5 \rightarrow 2 \rightarrow 1 \rightarrow 3$ 。交叉后再变异,首先产生一个随机数作为变异点,设该数为3,再将两个解路径中的第3个路径点 $x_1=6$ 和 $x_2=7$ 视为变异点,再随机地产生两个数,设为 $y_1=5$ ,  $y_2=1$ ,则变异后的两个解路径为  $3 \rightarrow 4 \rightarrow 5 \rightarrow 7 \rightarrow 2 \rightarrow 1 \rightarrow 6 \rightarrow 8$ 和  $4 \rightarrow 8 \rightarrow 1 \rightarrow 6 \rightarrow 5 \rightarrow 2 \rightarrow 7 \rightarrow 3$ 。其算法框架描述如下:

step 1 初始化 $\alpha$ 、 $\beta$ 、 $\rho$ 等参数;

step 2 for  $i=1$  to maxcities do // maxcities为城市的数量  
for  $j=1$  to maxcities do  $\tau(i, j)=\tau_0$

step 3 for  $k=1$  to maxants do // maxants为蚂蚁的个数

{设 $i_k$ 为蚂蚁 $k$ 的开始城市;

Allowed $_k = \{1, 2, \dots, \text{maxcities}\} - i_k$ ;

$i_k = i_k$ ;

step 4 for  $p=1$  to maxcities do

{if  $p < \text{maxcities}$  then

for  $k=1$  to maxants do

{按式(1)和式(2)选择下一个城市 $j_k$ ;

Allowed $_k = \text{Allowed}_k - j_k$ ;

Tour $_k(p) = (i_k, j_k)$ ;

else

for  $k=1$  to maxants do

{ $j_k = i_k$ ;

Tour $_k(p) = (i_k, j_k)$ ;

for  $k=1$  to maxants do

{ $\tau(i_k, j_k) = (1 - \rho) \tau(i_k, j_k) + \Delta \tau_{ij}$ ;

$i_k = j_k$ ;

step 5 for  $k=1$  to maxants do 计算Length $_k$ ;

//Length $_k$ 为第 $k$ 条蚂蚁所行进的长度。

寻找最优解和次优解并更新其路径中的信息素;

对最优解和次优解实施交叉和变异操作;

评价交叉变异后解的优劣求出新的最优解并更新该最优解路径中信息素;

for 每条边edge  $(i, j)$  do 更新每条边上的信息素;

step 6 if 结束条件满足 then 输出结果 else 转向step 4。

上述算法中首先增强最优解和次优解路径中信息素强度,再使用交叉和变异操作,引进了遗传算法的优化方法,增加了解空间的多样性,提高了全局搜索能力,避免了局部收敛和早熟现象,实验证明改进后的蚁群算法比单纯地改变全局最优解或局部最优解路径中信息素强度具有更好的搜索最优解的能力。

## 3 仿真实验

为了说明改进的增强型蚁群算法的搜索全局最优解的能力,本文同样以TSPLIB和文献[15]中提供的TSP问题GR21、GR24、Swiss42、Hk48、Gr120及Eil76为例。实验中所采取的各项参数如下:初始化中蚂蚁的数目等于TSP中城市的个数,每个城市分布的蚂蚁个数为0~2个,  $\alpha = 3, \beta = 5, \rho = 0.2$ , 系统信息素强度的初始值为1, 常数为1, 统计次数为15, 最大迭代次数Gr21、Gr24为300次, Eil76、Swiss42、Hk48为500次, Gr120为1500次。实验中采用的方法有两种:方法1是增强全局最优解和全局次优解路径中的信息素强度,其实验结果如表1所示(带\*的数据参照文献[15]);方法2是增强局部最优解和局部次优解路径中的信息素强度,其实验结果如表2所示;表中的最优解是统计中的最小值,平均解表示每次最优解的平均值。

表1 方法1中几种蚁群算法的最优解和平均解的进化情况

算法及解的情况		Gr21	Gr24	Eil76	Swiss42	Hk48	Gr120
标准蚁群算法	最优解	2 707	1 272	604.88	1 281	11 461	7 482
	平均解	2 817.13*	1 274*	621.07	1 286.3*	11 465.3*	7 573.3*
增强型蚁群算法	最优解	2 707*	1 272*	572.88	1 273*	11 461*	7 435.0*
	平均解	2 713.45*	1 272*	575.50	1 279*	11 461*	7 546.7*
本文改进蚁群算法	最优解	2 407.40	1 244.55	563.88	757.35	9 838.28	7 423.82
	平均解	2 407.40	1 244.55	564.28	763.39	9 979.16	7 461.04

表2 方法2中几种蚁群算法的最优解和平均解的进化情况

算法及解的情况		Gr21	Gr24	Eil76	Swiss42	Hk48	Gr120
标准蚁群算法	最优解	2 707	1 272	604.88	1281	11 461	7 482
	平均解	2 817.13*	1 274*	621.07	1 286.3*	11 465.3*	7 573.3*
增强型蚁群算法	最优解	2 707*	1 272*	572.88	1 300*	11 744*	14 303.0*
	平均解	2 710.4*	1 275.6*	575.50	1308.6*	11 897.6*	14 544.7*
本文改进蚁群算法	最优解	2 407.40	1 244.55	563.88	770.71	10 283.50	7 457.11
	平均解	2 407.40	1 251.81	576.57	781.63	10 302.76	7 797.12

改进的蚁群算法实验中上述诸问题所找到的最优解的路径如图1至图6所示。

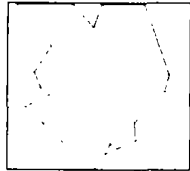


图1 Gr21找到的最优路径



图2 Gr24找到的最优路径

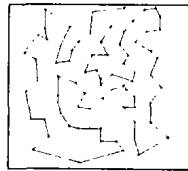


图3 Eil76找到的最优路径

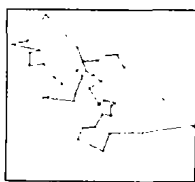


图4 Swiss42找到的最优路径

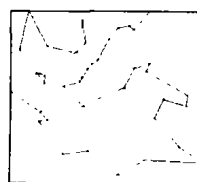


图5 Hk48所找到的最优路径



图6 Gr120找到的最优路径

由上述实验可知,改进后的蚁群算法具有比传统增强型蚁群算法更强的搜索全局最优解的能力。另外,笔者还采用传统的蚁群算法和改进的蚁群算法对问题Hk48和Gr120的优化过程进行了比较,其使用的参数同上,其优化过程如图7至图10所示。由该实验结果不难发现改进后的蚁群算法较一般的增强型蚁群算法具有更好的稳定性和收敛性。

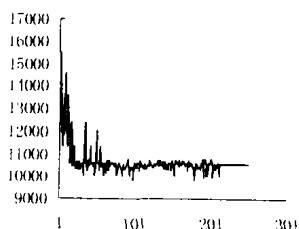


图7 传统蚁群算法中Hk48的最优解进化过程

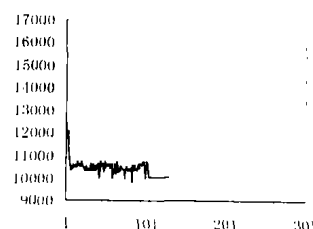


图8 改进蚁群算法中Hk48的最优解进化过程

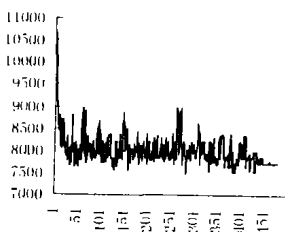


图9 传统蚁群算法中Gr120的最优解进化过程

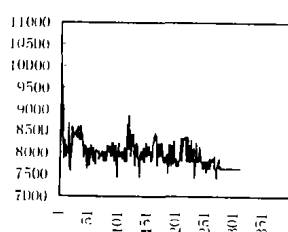


图10 改进蚁群算法中Gr120的最优解进化过程

## 4 结束语

本文结合文献[15]给出的增强型蚁群算法,提出了一种改进的增强型蚁群算法,通过增强全局(或局部)最优解和全局(或局部)次优解路径上的信息素强度以及使用遗传算法中的交叉和变异操作的优化手段相结合的方法改进传统的蚁群算法,从而有效地克服了传统蚁群算法中容易陷入局部最优解的缺陷,使得传统的蚁群算法的性能有了显著的提高。这有利于蚁群算法的应用与推广。相信随着蚁群算法研究的不断深入,蚁群算法将会获得更广泛的应用。

## 参考文献

- 1 Dorigo M, Maniezzo V, Colomi A. Ant System: Optimization by a Colony of Cooperating Agents. IEEE Transactions on SMC, 1996, 26(1): 8-41
- 2 Dorigo M, Gambardella L M. Ant Colony System: a Cooperative Learning Approach to the Traveling Salesman Problem. IEEE Transactions on Evolutionary Computing, 1997, 1(1): 53-56
- 3 Colomi A, Dorigo M, Maniezzo V. Ant Colony System for Job-shop Scheduling. Belgian Journal of Operations Research Statistics and Computer Science, 1994, 34(1): 39-53
- 4 Maniezzo V. Exact and Approximate Nondeterministic Tree Search Procedures for the Quadratic Assignment Problem. Inform Journal of Computer, 1999, 11(4): 358-369
- 5 Maniezzo V, Carbonaro A. An ANTS Heuristic for the Frequency Assignment Problem. Future Generation Computer Systems, 2000, 16(8): 927-935
- 6 Gambardella L M, Dorigo M. HAS-SOP: A Hybrid Ant System for the Sequential Ordering Problem. Technique Report, No. IDSIA 97-11, IDSIA, Lugano, Switzerland, 1997
- 7 Gambardella L M, Dorigo M. Ant-Q: A Reinforcement Learning Approach to the Traveling Salesman Problem. In: Prieditis A, Russell S (eds.) Proceedings of the 12th International Conference on Machine Learning, Tahoe City, CA: Morgan Kaufmann, 1995: 252-260
- 8 Dorigo M, Luca M. A Study of Some Properties of Ant-Q. Technical Report TR/IRIDIA/1996-4, IRIDIA, University Libre de Bruxelles, 1996
- 9 Stutzle T, Hoos H H. Improvements on the Ant System: Introducing the MAX-MIN Ant System. In: Artificial Neural Networks and Genetic Algorithms, New York: Springer-Verlag, 1988: 245-249
- 10 陈 峻, 沈 洁, 秦 玲等. 基于分布均匀度的自适应蚁群算法. 软件学报, 2003, 14(8): 1379-1387
- 11 陈 峻, 秦 玲, 陈宏建. 具有感觉和知觉特征的蚁群算法. 系统仿真学报, 2003, 15(10): 1418-1425
- 12 Bilchev G, Parmee I C. Adaptive Search Strategies for Heavily Constrained Design Spaces. Proceedings of 22nd International Conference on Computer Aided Design 95 Yelta[C]. Ukraine, 1995: 230-235
- 13 吴 斌, 史忠植. 一种基于蚁群算法的TSP问题分段求解算法. 计算机学报, 2001, 24(12): 1328-1333
- 14 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
- 15 燕 忠, 袁春伟. 增强型的蚁群优化算法. 计算机工程与应用, 2003, 39(23): 62-64