

文章编号: 1004 - 4639 (2009) 02 - 0150 - 04

基于改进蚁群算法对最短路径问题的分析与仿真

杨中秋, 张延华, 郑志丽

(沈阳化工学院 信息工程学院, 辽宁 沈阳 110142)

摘 要: 使用传统蚁群算法求解图的最短路径问题时,随着节点的增加会出现搜索速度变慢且易于陷入局部最优解.针对这个问题,提出一种改进的蚁群算法,通过引入搜索方向和搜索热区机制提高算法的搜索性能.仿真实验证明:改进的蚁群算法较传统的蚁群算法具有更高搜索速度且容易得到全局最优解.

关键词: 蚁群算法; 最短路径; 信息素

中图分类号: TP391 **文献标识码:** A

20世纪 90年代初期,意大利学者 Dorigo M^[1]等人从自然界蚂蚁群体觅食行为得到启发,提出一种模拟蚂蚁行为的模拟进化算法——蚁群优化算法,简称蚁群算法.研究表明:这种算法的优点是具有分布计算、信息正反馈和启发式搜索的特征,已经成功地用于解决 TSP (旅行商)组合优化问题.为了进一步提高算法的性能,针对 TSP问题,国内外很多学者对其进行各种改进^[2-6].

虽然蚁群算法及其相应的改进算法成功地用于解决组合优化问题,但是在用它们解决图的最短路径问题时,随着节点和路径的增加,会出现搜索速度慢、易于陷入局部最优解等问题,从而限制了蚁群算法在求解最短路径问题上的应用.鉴于此,针对最短路径问题的特点,笔者在传统蚁群算法中引入搜索方向机制和搜索热区机制的方法,提高算法搜索性能.通过计算机仿真实验证明:在求解最短路径问题时,引入搜索方向机制和搜索热区机制的蚁群算法较传统的蚁群算法不但在搜索速度上有了明显的提高,且不易陷入局部最优解.

1 传统蚁群算法原理

自然界中的蚂蚁的特点是个体行为极其简

单,而群体行为却相当复杂.一群蚂蚁通过相互协作很容易找到从蚁穴到食物源的最短路径(最优路径).之所以可以做到这一点,是因为蚂蚁个体之间是通过在其所经过的路径上留下一种可称之为信息素的物质来进行信息传递,蚂蚁就是根据路径上信息素的浓度来指导自己对前进方向的选择.蚁穴到食物源之间的路径越短,则单位时间内蚂蚁往返的次数越多,相应地该路径上的信息素的浓度也越高;反过来路径上信息素的浓度越高,后来的蚂蚁选择该路径的概率就越大.这就形成了蚂蚁群体行为表现出的一种信息正反馈现象,最后,信息素浓度最大的路径就是最短路径(最优路径).蚂蚁个体之间就是通过这种信息交流达到快速找到食物源或蚁穴的.蚁群算法就是受这种行为启发,以人工蚂蚁模拟真实蚂蚁行为求解组合优化问题.

该算法模型可以通过 TSP (旅行商)问题描述^[2], TSP (旅行商)问题是指完全遍历 n 个城市一次且仅一次所走过的最短距离.其蚁群算法数学模型如下:

把 m 只蚂蚁随机放在 n 个城市,所有路径 (i, j) 上的信息素置初值 $\tau_{ij}(0) = \text{常数}(C)$,然后蚂蚁 $k(1..m)$ 在 t 时刻按照转移概率 $p_{ij}^k(t)$ 选择下一个节点.其中 $\tau_{ij}(t)$ 表示边 (i, j) 在 t 时刻

收稿日期: 2008 - 06 - 03

作者简介: 杨中秋 (1973 -),女,辽宁沈阳人,工程师,硕士,主要从事 GIS方面的研究.

的信息素浓度, $\tau_{ij}(t)$ 表示边 (i, j) 在 t 时刻的启发信息.

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

启发信息因子设为:

$$\eta_{ij}(t) = 1/d_{ij} \quad (2)$$

d_{ij} 表示边 (i, j) 的长度. 参数 α 、 β 分别表示信息素浓度和启发信息的重要性, $allowed_k = \{0, 1, \dots, n-1\}$ 表示蚂蚁 k 下一步允许选择的节点 (访问过的节点不允许选择). 经过 n 个时刻, 所有蚂蚁完成一次循环, 各路径上的信息素根据式 (3) 调整.

$$\tau_{ij}(t+n) = (1-\rho) \cdot \tau_{ij}(t) + \rho \sum_{k=1}^m \tau_{ij}^k(t) \quad (3)$$

其中 $1-\rho$ 表示信息素残留因子, $\rho \in (0, 1)$;

$$\tau_{ij}^k(t) = \begin{cases} Q/L_k & \text{如果蚂蚁 } k \text{ 在本次循环中经过 } (i, j) \\ 0 & \text{否则} \end{cases} \quad (4)$$

$\tau_{ij}(t)$ 表示本次循环中路径 (i, j) 上的信息素增量, 初始时刻 $\tau_{ij}(0) = 0$, $\tau_{ij}^k(t)$ 表示第 k 只蚂蚁在本次循环中留在路径 (i, j) 上的信息量.

根据信息素更新策略的不同, Dorigo M 提出了 3 种不同的基本蚁群算法模型, 其中 Ant-Cycle 模型在求解最短路径时性能最好, 具体模型如下:

$$\tau_{ij}^k(t) = \begin{cases} \frac{Q}{L_k} & \text{如果蚂蚁 } k \text{ 在本次循环中经过 } (i, j) \\ 0 & \text{否则} \end{cases} \quad (5)$$

Q : 表示信息素强度, 它在一定程度上影响算法的收敛速度, L_k 表示第 k 只蚂蚁在本次循环中所走路径的总长度.

当所有蚂蚁完成一次循环后即结束一次迭代过程, 记录当前所得到的最优路径. 重复以上过程直到预定的迭代次数为止, 即得出最终的最优路径.

2 改进的蚁群算法

AS 算法在求解诸如 TSP、二次分配、车辆调度、网络路由等组合优化问题时取得了非常好的效果. 但是使用该算法求解图的最短路径问题时, 却存在搜索速度缓慢、并且极易陷入局部最

优解等缺点. 鉴于此, 本文针对最短路径问题的特点, 对 AS 算法进行改进, 即在 AS 算法中引入搜索方向机制和搜索热区机制提高对最短路径的搜索效率和搜索性能.

2.1 引入搜索方向机制

在 AS 算法的启发信息 (2) 中, d_{ij} 表示为可选边的长度, 即 d_{ij} 越短则该边被选中的概率就越大. 但在求解最短路径问题时, 若路径启发信息仍然用式 (2) 表示, 则没有考虑到由源节点到目的节点的方向性问题, 从而造成算法的搜索速度下降和易于陷入局部最优解. 由此对启发信息因子修改如下:

$$\eta_{ij}(t) = 1/d_{jo} \quad (6)$$

其中 d_{jo} 表示节点 j 到节点 o 的距离. 这样距离目标节点近的节点被选中的概率就高些, 从而提高了蚁群算法的搜索性能.

2.2 引入搜索热区机制

搜索热区是指由源节点和目标节点为对角节点所组成的矩形区域. 在实际求最短路径问题时, 求得的最优轨迹节点的位置落在由源节点和目标节点为对角节点所形成的搜索热区的概率往往要大大高于落在非搜索热区内的概率, 因此, 在蚁群算法中引入搜索热区的概念后, 不但使得搜索的速度大大提高, 而且搜索不易陷入局部最优解. 引入搜索热区机制后转移概率 p_{ij}^k 如式 (7) 所示.

$$p_{ij}^k(t) = \begin{cases} \frac{\tau_{ij}(t) \eta_{ij}(t) \delta_{ij}(t)}{\sum_{s \in allowed_k} \tau_{is}(t) \eta_{is}(t) \delta_{is}(t)}, & j \in allowed_k \\ 0 & \text{otherwise} \end{cases} \quad (7)$$

其中 $\delta_{ij}(t)$ 表示在 t 时刻根据边 (i, j) 是否在搜索热区中而取不同的值, 其它参数意义不变.

$$\delta_{ij}(t) = \begin{cases} A, & \text{边 } (i, j) \text{ 在搜索热区内} \\ B, & \text{边 } (i, j) \text{ 不在搜索热区内} \end{cases} \quad (8)$$

参数 A 、 B 取值随具体问题而定, 通常情况 A 应取大于 1 的数, B 取小于 1 的数.

3 使用改进的蚁群算法求最短路径问题的实现

求图中任意源点到目标点最短路径问题的

改进蚁群算法具体实现步骤如下：

- (1) 将 m 只蚂蚁放在源点.
- (2) 所有路径 (i, j) 上的信息素置初值 $_{ij}(0) = \text{常数}(C)$, 当前最短距离初始化为一个极大值.
- (3) 然后蚂蚁 $k(1..m)$ 在 t 时刻按照转移概率公式 (7) 选择下一个节点, 其中 $_{ij}(t)$ 、 $_{ij}(t)$ 如公式 (6)、(8) 所示. 当所选节点是终点时, 则该只蚂蚁结束搜索. 此时, 如果该蚂蚁走过的路径长度小于当前最短距离, 则当前最短距离用该蚂蚁走过的路径长度取代.
- (4) 经过 n 个时刻, 所有蚂蚁完成一次循环, 即结束一次迭代过程, 各路径上的信息素根据式 (3)、(4)、(5) 调整.
- (5) 重复步骤 ~ , 直到预期的迭代次数为止, 即得到由源点到目标点的最短路径.

4 实验仿真

实验的硬件平台是赛扬 1.8 G 的 PC 机, 256 M 内存, 操作系统是 Window s2000 server 开发工具是 VC 6.0 实验数据采用 1 000 个节点的带权有向图, 各个节点的横坐标范围为 1 ~ 1 000, 纵坐标用随机函数 rand ()除以 1 000 的余数生成. V_i 表示最优路径经过节点 (或城市) 排序的序号: $V_1, V_2 \dots V_1\,000$, 图中每个节点的入度和出度均为 10. 蚁群算法中各参数设置如下: 蚂蚁数为 30, 信息素初始值 $_{ij}(0) = 20$, 路径信息释放率 $= 0.1$, 参数 $= 0.3$, $= 5, A = 3, B = 0.3$, 蚂蚁携带信息素总量 $Q = 1\,000$, 最大迭代次数为 100. 随机选取 10 对节点, V_8 与 V_{901} , ..., V_{420} 与 V_{461} , 分别采用传统蚁群算法和改进的蚁群算法进行验证, 结果如表 1、表 2 所示, 2 种算法对应的进化曲线如图 1 和图 2 所示.

表 1 传统蚁群算法和改进蚁群算法
所求最短路径长度对照

Table 1 Contrast length with shortest path by using
traditional ant colony algorithm and
modified ant colony algorithm

序号	源点	目标点	实际最短 路径长度	传统蚁群算法 搜索最短长度	改进蚁群算法 搜索最短长度
1	V8	V901	1 459.45	1 548.94	1 548.94
2	V48	V896	1 098.67	3 070.56	1 098.67
3	V108	V756	1 025.83	1 034.16	1 025.83

续 表

Continued table

序号	源点	目标点	实际最短 路径长度	传统蚁群算法 搜索最短长度	改进蚁群算法 搜索最短长度
4	V149	V951	992.07	1 403.82	992.07
5	V208	V646	1 031.25	1 031.25	1 031.25
6	V359	V588	1 192.51	1 203.44	1 192.51
7	V405	V611	880.99	1 197.22	880.99
8	V512	V581	987.91	1 108.11	1 108.11
9	V580	V591	1 044.97	1 044.97	1 044.97
10	V420	V461	1 097.03	1 097.03	1 097.03

表 2 传统蚁群算法和改进蚁群算法所用时间比较

Table 2 Comparision of using time between traditional ant
colony algorithm and modified ant colony algorithm

序号	源点	目标点	传统蚁群算法 所用时间 /s	改进蚁群算 法所用时间 /s
1	V8	V901	20.049	4.617
2	V48	V896	58.184	5.067
3	V108	V756	17.195	1.522
4	V149	V951	13.770	1.562
5	V208	V646	10.82	0.301
6	V359	V588	27.269	3.305
7	V405	V611	25.076	1.082
8	V512	V581	9.374	3.095
9	V580	V591	10.485	0.401
10	V420	V461	0.331	0.120

由此可知, 在求图的最短路径问题时, 改进的蚁群算法较传统的蚁群算法无论在搜索速度还是搜索精度上都有了显著的提高.

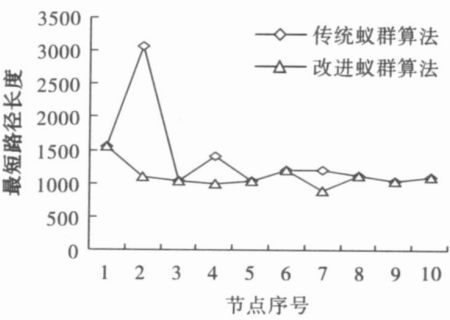


图 1 传统蚁群算法和改进的蚁群算
法搜索最短路径长度比较

Fig 1 Contrast length with shortest path by using
traditional ant colony algorithm and
modified ant colony algorithm

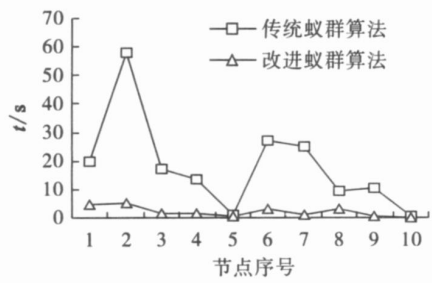


图 2 传统蚁群算法和改进的蚁群算法所用时间比较
Fig 2 Comparison of using time between traditional ant colony algorithm and modified ant colony algorithm

5 结束语

使用传统的蚁群算法解决最短路径问题时,存在搜索速度慢、易于陷入局部最优解等缺陷。本文针对最短路径问题的特点,在传统蚁群算法中引入了搜索方向和搜索热区机制,提高了算法的搜索效率和搜索性能。实验证明:在求解最短路径问题时,使用改进的蚁群算法较传统的蚁群

算法无论在搜索效率还是在搜索性能都有了明显的改善。

参考文献:

[1] Coloni A, Dorigo M, Maniezzo V. Ant System: Optimization By a Colony of Cooperating Agents [J]. IEEE Transactions On Systems, Man and Cybernetics - Part B, 1996, 26 (1): 1 - 13.

[2] 申春,彭秀增,罗凡,等. 基于方向启发因子的蚁群算法 [J]. 计算机科学, 2006, 33 (8): 175 - 176

[3] 张志民,张小娟,李明华,等. 一种引入奖励与惩罚机制的蚁群算法 [J]. 计算机仿真, 2006, 23 (7): 161 - 163.

[4] 张曦煌,李彦中,李岩. 基于加速寻径收敛的改进型蚁群算法 [J]. 计算机工程与应用, 2007, 43 (24): 75 - 77.

[5] 张毅,梁艳春. 基于选路优化的改进蚁群算法 [J]. 计算机工程与应用, 2007, 43 (2): 60 - 63.

[6] 宁爱兵,马良,熊小华. 基于复杂适应系统的蚂蚁群体智能研究 [J]. 微计算机信息, 2008 (1): 265 - 267.

Analysis and Simulation of Shortest Path Problem Based on Modified Ant Colony Algorithm

YANG Zhong-qiu, ZHANG Yan-hua, ZHENG Zhi-li

(Shenyang University of Chemical Technology, Shenyang 110142, China)

Abstract: When resolving shortest path problem with traditional ant colony algorithms, the denser the node is, the slower the speed is and the easier it is to get local optimization. To deal with this problem, a kind of modified ant colony algorithm is proposed, in which the mechanism of search direction and search hot section are introduced. Simulated experiments show that the modified ant colony algorithms have a higher searching speed and are easier to use to obtain global optimization than the traditional ones.

Key words: ant colony algorithm; shortest path; pheromone