

# 蚁群算法在林火扑救路径选择中的应用

刘道伟<sup>a</sup>, 关 昕<sup>b</sup>

(辽宁工程技术大学 a. 研究生学院; b. 电子与信息工程学院, 辽宁 葫芦岛 125105)

**摘 要:** 针对森林道路错综复杂的特殊情况和蚁群算法容易出现的局部收敛问题, 提出一种用于林火扑救最优路径选择的蚁群算法。在基本蚁群算法的基础上, 动态计算启发式信息值并更新信息素, 平衡收敛速度与停滞现象的出现概率, 增大算法的搜索空间。仿真结果表明, 该算法能在较短时间内找出符合实际要求的最优路径, 提高蚁群的全局搜索能力。

**关键词:** 林火扑救; 蚁群算法; 最优路径; 拓扑结构; 道路等级

## Application of Ant Colony Algorithm in Forest Fires Fighting Path Selection

LIU Dao-wei<sup>a</sup>, GUAN Xin<sup>b</sup>

(a. Graduate School; b. College of Electronic and Information Engineering, Liaoning Technical University, Huludao 125105, China)

**【Abstract】** The special circumstances of complex forest roads and the partial convergency of phenomenon which is easily to happen in ant colony algorithm, so this paper adopts to dynamic calculation method of the value of heuristic information and dynamic updating the pheromone to improve the basic ant colony algorithm. Results show that the improved algorithm can not only find to meet practical requirements of the optimal path in a relatively short time but also improve the ability of global search. It provides a new idea and method for solving the problem of optimal forest fires fighting path.

**【Key words】** forest fires fighting; ant colony algorithm; optimal path; topological structure; road grade

DOI: 10.3969/j.issn.1000-3428.2011.14.072

### 1 概述

近年来随着森林火灾的频繁发生, 基于地理信息系统(Geographic Information System, GIS)的森林防火系统越来越受到人们重视。森林防火系统中包含许多重要组成部分, 其中森林扑火就是该重要组成部分之一。在森林扑火中时间的节省对控制火势蔓延, 减少森林资源的破坏起着至关重要的作用, 因此, 选择消耗时间最短的路径到达着火点进行扑救一直是很多学者研究的重点。传统的查找最优路径的算法(如 Dijkstra 算法)是非智能非进化算法, 求出的最优路径并不一定是全局最优的, 而且算法在执行过程中存在内存资源占用大、时间效率低等缺点<sup>[1]</sup>。蚁群算法是受自然界中蚂蚁寻路行为启发产生的一种具有自适应特性的分布式算法, 它具有并行性、正反馈性、健壮性等特点, 并且蚁群算法是一种基于种群的鲁棒性较强的算法, 具有许多优良的性质, 能更好地求解复杂的组合优化问题。但蚁群算法也存在一定的缺陷, 如在算法实现的过程中容易出现停滞和收敛速度慢的现象。为提高全局搜索能力, 并解决特定情况下的最优路径选择问题, 本文提出一种改进的蚁群算法。

### 2 基本蚁群算法

蚁群算法是一种智能模拟进化算法, 由文献[2]提出, 是一种仿生学算法。它模拟了自然界中的蚂蚁觅食的行为, 采取用具有智能的蚂蚁, 通过个体间的信息交流与协作来寻找蚁穴到食物源的最优路径(最短路径)。其数学模型如下<sup>[3]</sup>:

蚂蚁  $k(k=1, 2, \dots)$  在运动过程中, 根据各条路径上的信息素浓度决定其转移方向。这里用禁忌表  $\text{tabu}_k(k=1, 2, \dots, m)$  来记录蚂蚁  $k$  当前所走过的城市, 集合随着  $\text{tabu}_k$  进化过程作动态调整。在搜索过程中, 蚂蚁根据各条路径上的信息素浓度及

路径的启发信息来计算状态转移概率。 $P_{ij}^k(t)$  表示在  $t$  时刻蚂蚁  $k$  由城市  $i$  转移到城市  $j$  的状态转移概率:

$$P_{ij}^k(t) = \begin{cases} \frac{\tau_{ij}^\alpha(t)\eta_{ij}^\beta(t)}{\sum_{s \in \text{allowed}_k} \tau_{is}^\alpha(t)\eta_{is}^\beta(t)} & j \in \text{allowed}_k \\ 0 & j \notin \text{allowed}_k \end{cases} \quad (1)$$

其中,  $\tau_{ij}(t)$  为  $t$  时刻路径  $(i, j)$  上的信息素浓度;  $\text{allowed}_k$  表示蚂蚁  $k$  下一步允许选择的城市;  $\alpha$  和  $\beta$  是信息素浓度和控制可见度的权衡;  $\eta_{ij}(t)$  是与路径  $(i, j)$  相关联的启发式信息值, 其表达式为:  $\eta_{ij}(t) = 1/d_{ij}$ 。  $d_{ij}$  表示相邻两城市  $i$  和  $j$  之间的距离。对于蚂蚁  $k$  而言,  $d_{ij}$  越小, 则  $\eta_{ij}(t)$  越大,  $P_{ij}^k(t)$  也就越大。显然, 该启发函数表示蚂蚁从城市  $i$  转移到城市  $j$  的期望程度。

为避免残留信息素过多引起残留信息淹没启发信息, 当每只蚂蚁完成一次循环后, 要对残留信息进行更新处理。由此,  $t+n$  时刻在路径  $(i, j)$  上的信息素可按如下公式调整:

$$\tau_{ij}(t+n) = (1-\rho) \cdot \tau_{ij}(t) + \Delta\tau_{ij}(t) \quad (2)$$

$$\Delta\tau_{ij}(t) = \sum_{k=1}^m \Delta\tau_{ij}^k(t) \quad (3)$$

其中,  $\rho$  表示信息素挥发系数, 则  $1-\rho$  表示信息素残留因子, 为防止信息的无限积累,  $\rho$  的取值范围为:  $\rho \in [0, 1]$ ;  $\Delta\tau_{ij}(t)$

**基金项目:** 2009 年度中国煤炭工业科技计划基金资助项目(MTKJ 2009-253)

**作者简介:** 刘道伟(1985—), 男, 硕士研究生, 主研方向: 网络安全, 智能网络; 关 昕, 副教授

**收稿日期:** 2011-01-17 **E-mail:** liudaoweideyou@163.com

表示本次循环中路径  $(i, j)$  上的信息素增量, 初始时刻  $\Delta\tau_{ij}(0) = 0$ ;  $\Delta\tau_{ij}^k(t)$  表示第  $k$  只蚂蚁在本次循环中留在路径  $(i, j)$  上的信息素量,  $\Delta\tau_{ij}^k(t)$  的取值采用文献[2]提出的 Ant.Cycle 模型:

$$\Delta\tau_{ij}^k = \begin{cases} \frac{Q}{L_k} & \text{第 } k \text{ 只蚂蚁在 } t \text{ 到 } t+1 \text{ 时间段之间留下的信息素} \\ 0 & \text{否则} \end{cases} \quad (4)$$

其中,  $Q$  表示信息素强度, 为常数, 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。

### 3 问题描述和改进的蚁群算法

#### 3.1 林火扑救选择最优路径问题描述

最优路径在本文指的是在基于森林路网拓扑结构情况下, 选择给定起点和终点之间所用时间最短的路径。由于森林扑火过程中扑救人员需携带大量的救火工具, 因此扑救人员往往倾向于选择通行时间短、可以行驶车辆的道路。但森林道路特殊而且复杂, 比如道路有时是宽阔平坦的, 可以行驶车辆; 有时却是崎岖狭窄的, 只能步行; 有时前一条是平坦的大道而下一条就变成崎岖的小道了, 在行驶过程中一旦选择了小道就意味着抛弃了车辆在接下来的道路就只能步行了。因此, 每条道路的通行能力是不同的。概括起来, 选择最优的路径到达扑火点主要考虑以下 2 个方面: (1)耗时短; (2)道路的通行能力好。

鉴于森林道路的特殊性, 本文使用了划分道路等级的思想<sup>[4]</sup>。根据每条道路通行能力不同将道路分为 4 个等级: 等级 1: 道路宽敞, 路面平整, 能正常行驶车辆; 等级 2: 道路宽敞, 路面起伏, 车辆能慢速行驶; 等级 3: 道路狭窄, 路面平坦, 不能行使车辆人能正常行走; 等级 4: 道路狭窄, 路面凸凹, 不能行驶车辆人只能慢行。本文用车辆和人行速度来标示道路等级, 具体如表 1 所示。

表 1 道路等级

等级	行驶速度/(km·h <sup>-1</sup> )
等级 1	60
等级 2	35
等级 3	10
等级 4	5

综上可知, 道路通行能力多样性的问题是选择最优路径首先要考虑的, 既要满足耗时短的要求, 也要满足道路通行能力好, 能尽量允许扑救人员携带大量的救火工具的通过。

#### 3.2 改进的蚁群算法

基本蚁群算法在实现的过程中容易出现停滞和收敛速度慢的现象。具体表现在蚂蚁在行进过程中常常选择信息量较大的路径, 当搜索进行到一定的程度后, 致使所有个体所选择的解相同, 不能对解空间进一步搜索, 不利于发现更好的解<sup>[5]</sup>。而且当许多蚂蚁选中同一条路径时, 路径中的信息量就会陡然增大, 从而使得多只蚂蚁集中到某一条路径上, 造成一种堵塞和停滞现象。

在具体应用过程中, 许多学者针对蚁群算法停滞和收敛速度问题提出了许多改进的方法。但是总的来说, 提高收敛速度与减少停滞现象是相互影响的。收敛速度的提高会加大出现停滞现象的概率; 反之, 减少停滞现象就会减慢收敛速度。因此, 为在提高收敛速度和减少停滞现象之间取得一个平衡, 本文采用动态的方法来更新信息素解决收敛速度与停滞现象出现概率的平衡问题。

#### 3.2.1 信息素的动态更新

根据蚂蚁所求解质量的不同, 动态地更新信息素, 只让优略路径上的信息量增强, 其余路径上的信息量被削减。每次蚂蚁完成一个循环后对搜索到的路径所用的时间进行平均化<sup>[6]</sup>, 即  $t_{AVG} = \sum_{k=1}^m t_k / m$ 。对所用时间小于  $t_{AVG}$  的蚂蚁, 在更新留在路径上的信息素时赋予一个相对较大的系数; 对所用时间大于或者等于  $t_{AVG}$  的蚂蚁, 在更新留在路径上的信息素时赋予一个稳定的系数。第  $k$  只蚂蚁在本次循环中留在路径  $(i, j)$  上的信息素  $\Delta\tau_{ij}^k(t)$  变为:

$$\Delta\tau_{ij}^k = \begin{cases} \frac{\lambda Q}{L_k} & \text{第 } k \text{ 只蚂蚁在 } t \text{ 到 } t+1 \text{ 时间段之间留下的信息素} \\ 0 & \text{否则} \end{cases} \quad (5)$$

其中,  $\lambda$  求解式如下:

$$\lambda = \begin{cases} \frac{t_{AVG} - t_k}{t_{AVG}} & t_k < t_{AVG} \\ 0 & t_k \geq t_{AVG} \end{cases} \quad (6)$$

由上式可知,  $t_k$  越小, 则  $\lambda$  越大。

#### 3.2.2 启发式信息值的动态计算

针对森林道路通行能力多样性的问题, 现已提出的蚁群算法仍达不到较优的搜索效果, 为解决该问题, 本文采用动态的方法来计算与道路相关联的启发式信息值  $\eta_{ij}(t)$ , 用变化的  $\eta_{ij}(t)$  实现对道路节点转移概率  $p_{ij}^k(t)$  的影响。具体描述如下:

在选择最优路径过程中等级越高的道路越容易被选到, 因此, 用在某道路上通行时间  $t_{ij}$  的倒数来表示该道路的启发式信息值  $\eta_{ij}(t)$ , 其中  $t_{ij} = L_{ij} / v_{ij}$ , 设 4 种等级道路的通行速度分别为  $v_1 \sim v_4$ , 从节点  $i$  到节点  $j$  的道路的通行速度为  $v_{ij}$ , 计算  $\eta_{ij}(t)$  的公式如下:

$$\eta_{ij}(t) = \begin{cases} \frac{v_{ij}}{L_{ij}} & v_{(i-1)(j-1)} \in \{v_1, v_2\} \\ \eta_{ij}^r(t) & v_{(i-1)(j-1)} \in \{v_3, v_4\} \end{cases} \quad (7)$$

$$\eta_{ij}^r(t) = \begin{cases} \frac{v_3}{L_{ij}} & v_{ij} \in \{v_1, v_2, v_3\} \\ \frac{v_4}{L_{ij}} & v_{ij} \in \{v_4\} \end{cases} \quad (8)$$

其中,  $v_{(i-1)(j-1)}$  表示蚂蚁从节点  $i$  向节点  $j$  转移时通过前一条路的速度。显然,  $\eta_{ij}(t)$  与  $p_{ij}^k(t)$  和  $v_{ij}$  均呈正比,  $v_{ij}$  越大, 则  $\eta_{ij}(t)$  越大;  $\eta_{ij}(t)$  越大, 则  $p_{ij}^k(t)$  越大。

在求解最优路径时, 每个蚂蚁根据通过前面一条道路的速度和候选道路的通行速度计算启发信息值, 再根据状态转移规则式转移状态, 完成道路规划。

#### 3.3 改进算法描述

改进算法的具体步骤如下:

**步骤 1** 参数初始化。令迭代次数为  $nc$ , 且初始  $nc=0$ , 最大  $nc=NC$ ; 设定蚂蚁个数为  $m$ , 将  $m$  个蚂蚁置于初始顶点上; 令道路拓扑图上每条边  $(i, j)$  的初始化信息素  $\tau_{ij}(t) = C$ , 且初始时刻  $\Delta\tau_{ij}(0) = 0$ 。

**步骤 2** 将各蚂蚁的出发点置于当前解中。

**步骤 3** 对每个蚂蚁  $k(k=1, 2, \dots)$  按改进后的状态转移规则  $p_{ij}^k(t)$  移至下一顶点  $j$ , 将顶点  $j$  置于当前解中。

**步骤 4** 若所有蚂蚁的当前解集包含了终点, 转到步骤 5, 否则转步骤 3。

**步骤 5** 计算各蚂蚁的行驶时间，记录耗时最短的行驶路径。

**步骤 6** 将当前耗时最短的行驶路径各边按动态更新信息素的方法进行信息素更新。

**步骤 7** 对各边弧 $(i,j)$ ，置 $\Delta\tau_{ij}=0$ ， $nc=nc+1$ 。

**步骤 8** 如果 $nc < NC$ ，转到步骤 2，否则终止迭代并输出最优路径。

## 4 仿真实验

### 4.1 实验条件

要进行最短路径的仿真实验，根据仿真结果选择最优的最短路径，首先必须将道路中节点和边的关系抽象为图的结构。

图是由节点的集合和边的集合组成的，其中节点是道路的交叉点、端点，边是两节点间的一段道路<sup>[7]</sup>。为减少存储信息所占用的内存空间，在此只存储有用信息。

(1)节点的存储结构表如表 2 所示。其中，节点 ID 号为唯一标示该节点的信息；X、Y 分别为该节点在地图中的横坐标和纵坐标。

**表 2 节点的存储结构**

ID	X	Y
节点 ID 号	节点横坐标	节点纵坐标

(2)边的存储结构表如表 3 所示。其中，Node\_FID、Node\_LID 为该道路的起始节点和终止节点；Road\_Long 为标示该道路长度的信息；Road\_Class 为标示道路等级的信息。

**表 3 边的存储结构**

Node_FID	Node_LID	Road_Long	Road_Class
起始节点 ID 号	终止节点 ID 号	道路长度	道路等级

建立道路拓扑结构图的步骤如下：

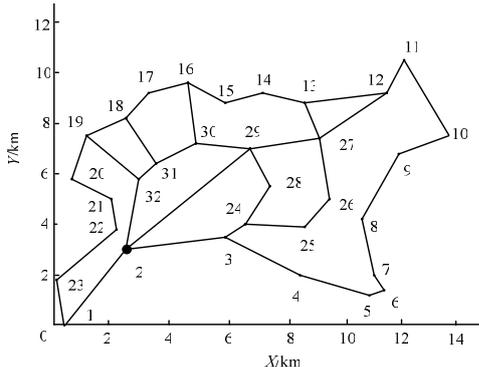
**步骤 1** 从 GIS 数据源读取线图层信息，并从中提取节点信息和边信息；

**步骤 2** 建立节点集合和边的集合；

**步骤 3** 创建节点和边的邻接关系；

**步骤 4** 建立拓扑结构图。

本文利用 Matlab 7.0 作为仿真软件，从节点集合中选出包括救火站在内的 32 个邻接的节点，并对节点和边的信息进行相应比例的缩小，通过仿真软件创建所选中的 32 个节点的拓扑结构，如图 1 所示。图中节点 2 为救火站所在地，道路的通行能力分为 4 个等级： $v_1 \sim v_4$ ，分别为表 1 中的 4 个等级。其中，11-12 道路通行速度为  $v_4$ ；19-32、18-31、16-30、13-27 道路通行速度为  $v_3$ ；30-31-32、2-29-27-12、3-24-25-26-27、5-6-7、9-10-11 道路的通行速度为  $v_2$ ；剩余道路的通行速度均为  $v_1$ 。

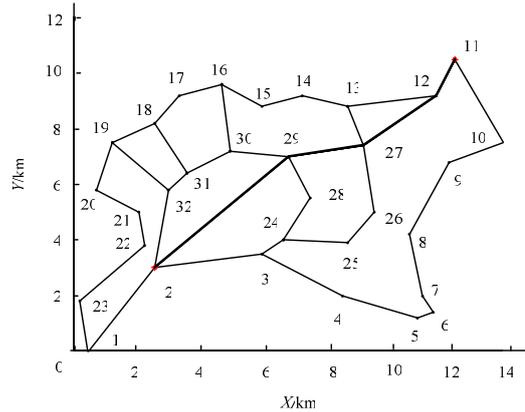


**图 1 道路拓扑结构**

### 4.2 实验结果

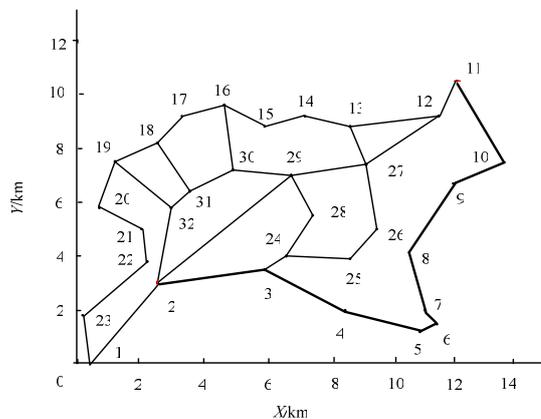
蚁群算法和改进的蚁群算法在 Matlab 7.0 仿真软件实现过程中选用的参数  $\alpha$ 、 $\beta$ 、 $\rho$ 、蚂蚁数  $m$ 、最迭代次数  $NC$  等都是相同的。假如节点 11 的位置发生火灾，则要求选择一条从节点 2 到节点 11 耗时最短的路径，同时考虑尽可能的可以通过汽车。

在上述情况下将基本蚁群算法加载到 Matlab 7.0 中，选出的最优路径为 2-29-27-12-11，如图 2 所示(加粗部分)。其中各路段的通行时间为在考虑前一阶段通行能力的情况下计算该路段长度与该路段速度的比值。因此基本蚁群算法选出的最优路径中各路段通行时间的和为 0.665 714 h。



**图 2 基本蚁群算法结果**

同样的情况下改进的蚁群算法选出的最优路径为 2-3-4-5-6-7-8-9-10-11，如图 3 所示(加粗部分)。改进的蚁群算法选出的最优路径中各路段通行时间的和为 0.467 381 h。



**图 3 改进蚁群算法结果**

2 种算法的实验结果对比如表 4 所示。

**表 4 2 种算法的实验结果对比**

算法	所走路径	路径长度/km	行驶时间/h
基本蚁群算法	2-29-27-12-11	14.3	0.665 714
改进蚁群算法	2-3-4-5-6-7-8-9-10-11	22.9	0.467 381

仿真实验表明：改进的蚁群算法选出了一条耗时最短的而且汽车可以通过的路径；基本蚁群算法选出的是路径长度是最短的路径，但是行驶时间比较长而且到达着火点时扑救人员是步行的，这样大型的扑救工具不方便运输到火灾现场。改进的算法虽然距离比基本算法长，但是所需时间明显减少，路况较好，找出的路径较优。

(下转第 219 页)