

# 嵌入式 GIS 最短路径分析中 Dijkstra 算法的改进

王玉琨, 吴 锋

WANG Yu-kun, WU Feng

河南理工大学 计算机学院, 河南 焦作 464200

Department of Computer, Henan Polytechnic University, Jiaozuo, Henan 464200, China

E-mail: w\_wufeng@163.com

WANG Yu-kun, WU Feng. Improvement of Dijkstra in Embedded-GIS route analysis. Computer Engineering and Applications, 2008, 44(28): 128-129.

**Abstract:** Dijkstra is a classic arithmetic for net route analysis. By banding it together with the embedded GIS characteristics, the efficiency of the best route finding arithmetic is greatly improved, the graphic storage structure and finding technique are changed, the needed memory is decreased, and the time for finding the best route is greatly shortened. Applying this algorithm into "Geographic Information System for Communal Query of Jiaozuo City" system, gains satisfying result.

**Key words:** the best route finding analysis; Dijkstra arithmetic; embedded

**摘 要:** Dijkstra 算法是求解网络中最短路径的经典算法, 文中通过改变图的存储结构及搜索方法, 减少了内存存储空间, 缩短了查询时间, 以提高该算法在嵌入式 GIS (Geographic Information System) 系统中路径优化的效率。并将该算法应用在嵌入式焦作市地理信息公众查询系统中, 取得满意的效果。

**关键词:** 最短路径分析; Dijkstra 算法; 嵌入式

**DOI:** 10.3778/j.issn.1002-8331.2008.28.043 **文章编号:** 1002-8331(2008)28-0128-02 **文献标识码:** A **中图分类号:** TP301

地理网络分析是 GIS 中一个主要功能, 它包括最短路径分析、资源配置、等时性问题等等, 其中最短路径分析是最基本的, 也是最关键的。现有的最短路径算法有许多, 其中 Dijkstra 算法由于能适应网络拓扑的变化, 性能稳定, 因而在计算机网络拓扑路径选择以及 GIS 中得到广泛的应用。

在以时间换空间的算法下, 虽然节省了存储资源, 但是却耗费了大量的时间, 系统响应速度慢, 使用户无法忍受长时间的运行等待; 在以空间换时间的算法中, 搜索时间虽然缩短了, 但却占用了大量的存储资源, 这在嵌入式 GIS 中可能无法正常运行。本文中将对 Dijkstra 算法进行优化处理, 将它应用于 GIS 路径分析, 同时考虑提高时间与空间效率。

## 1 Dijkstra 算法及其存在的问题

Dijkstra 算法用于计算一个源结点到所有其他结点的最短代价路径, 它是按路径长度递增的次序来产生最短路径的算法。

下面以邻接矩阵描述 Dijkstra 算法的实现过程。假设用带权的邻接矩阵  $Cost$  来表示具有  $N$  个结点的带权有向图 (这里讨论的最短路径算法主要是针对平面有向图),  $Cost[i, j]$  表示弧  $\langle V_i, V_j \rangle$  的权值, 如果从  $V_i$  到  $V_j$  不通, 则  $Cost[i, j] = \infty$ 。引进一个辅助向量  $Dist$ , 并设  $V_s$  为起始点, 每个分量  $Dist[i]$  表示已找

到的从起始点  $V_s$  到每个终点  $V_i$  的最小权值。则该向量的初始值为:  $Dist[i] = Cost[s, i], V_i \in V$ 。其中,  $V$  是结点的集合。令  $S$  为已经找到的从起点出发的最短路径的终点集合, 初始值为  $S = \{V_s\}$ , 则从  $V_s$  出发到图  $G$  上其它所有结点  $V_i$  可能达到的最短路径长度为  $Dist[i] = Cost[s, i], V_i \in V$ 。

**步骤 1** 选择  $V_j$ , 使得  $Dist[j] = \min\{Dist[i] | V_i \in V - S\}$ ,  $V_j$  就是当前求得的一条从  $V_s$  出发的最短路径的终点, 令  $S = S \cup \{V_j\}$ 。

**步骤 2** 修改从  $V_s$  出发到集合  $V - S$  中任意一顶点  $V_k$  的最短路径长度。如果  $Dist[j] + Cost[j, k] < Dist[k]$ 。

**步骤 3** 修改  $Dist[k]$  为  $Dist[k] = Dist[j] + Cost[j, k]$ 。

重复步骤 2、3 操作共  $N-1$  次, 由此求得从  $V_s$  出发到图上各个顶点的最优路径是依路径权值递增的序列。

从上述算法描述过程可知, 步骤 1 是影响该算法效率的关键。由于每次搜索最小权值的结点  $V_j$ , 都需要对  $V - S$  集合上的所有结点进行扫描, 而这些元素是无序的, 会花费大量的时间; 另外, 搜索最小权值的过程中, 还会出现某些结点被多次访问的情况, 增加了额外的运算量。因此, 在大数据量的情况下, 这些必将成为算法的瓶颈。由于传统 Dijkstra 算法的时间复杂度为平方阶的, 可以考虑通过尽量减少最短路径分析过程中搜索的临时结点数量来对效率和存储空间进行优化。

基金项目: 河南省科技公关资助项目 (the Key Technologies R&D Program of Henan Province, China under Grant No.0424220159)。

作者简介: 王玉琨 (1960-), 男, 副教授, 硕士生导师, 主要研究领域为计算机图形学与 CAD; 吴锋 (1981-), 男, 硕士研究生, 研究方向为计算机图形学。

收稿日期: 2007-11-19 修回日期: 2008-03-03

## 2 基于 Dijkstra 算法的 GIS 路径优化

### 2.1 改进的方面

系统算法的优劣很大程度上在于运行时所占用的内存资源及执行的时间效率, 文中在优化时间与空间效率方面进行了以下的重要改进:

- (1) 采用了一种新型的网络存储模型;
- (2) 将修改结点的权值与搜索最小权值的结点同步进行, 以达到提高时间效率的目的. 也就是说文中所描述的算法将从空间与时间两个方面来改进基于 GIS 的路径分析。

### 2.2 网络路径优化基础模型的建立

在网络分析过程中, 实际上只需关心网络边的信息, 如边的权值、起点、终点。这样就可采用只存储边的网络拓扑信息, 不存储实际网络点的拓扑信息, 减少空间分析时所要检索的数据量。根据该思想, 采用表 1 及图 1 的网络边存储模型代替传统的矩阵或邻接表的存储模型, 使内存的占用量大大减少。

表 1 网络边存储模型

边号	起点	终点	边的权值
1	$V_1$	$V_0$	10
2	$V_0$	$V_5$	6
3	$V_5$	$V_4$	9
4	$V_3$	$V_4$	6
5	$V_2$	$V_3$	3
6	$V_1$	$V_2$	2
7	$V_2$	$V_3$	8
8	$V_4$	$V_1$	9

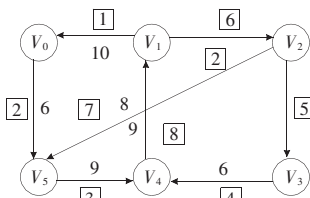


图 1 网络结构图

对于嵌入式系统, 考虑硬件计算环境的限制, 不需要一安装所有的网络模型信息。文中采用的方法为:

- (1) 载入用户设置的起点、终点、途经点、障碍点, 确定 Bound 范围, 装载 Bound 范围内的数据, 可以减少空间分析时所要检索的临时结点的数量, 基于这些数据建立上述结构的网络模型;

- (2) 如果在(1)中设置的 Bound 范围内没有找到一条最优的路径, 再适当扩大 Bound 范围, 并采用分块装入数据的方法。

这个 Bound 的初始大小可以根据用户实际应用情况定义。当然同样可以先装载全部数据, 然后, 基于起点与终点的位置进行预先排序处理, 排序的原则是离起点与终点近的边位于上述结构网络模型较近的存储位置, 再在这个网络模型的基础上进行分块优化, 但该方法具有占用存储开销较大的缺点。

### 2.3 基于上述网络模型的路径优化过程

基于上面的网络存储模型, 采用 Dijkstra 算法原理对网络进行路径优化。假设图  $D=(V, A, W)$ , 其中  $V$  为结点集,  $A$  为边集,  $W$  为权值集,  $V_i \in V, A_i \in A, W_{ij} \in W, 1 \leq i \leq N, 1 \leq j \leq N$ , 求解  $V_i$  至  $V_j$  的一条最短路径, 则采用的步骤如下所述:

**步骤 1** 用  $U_i$  表示当前结点  $V_i$  关联网络边的最短权值, 初始设置  $U_i=0, U_j=W_{ij} (2 \leq j \leq N), S=\{V_i\}, R=\{V_1, V_2, \dots, V_n\}$ 。

**步骤 2** 在  $R$  中取一个顶点  $V_k$ , 使  $U_k=\min(U_{ij}) (2 \leq j \leq N)$ 。如果  $U_k=+\infty$ , 则停止, 从  $V_i$  到  $R$  中各顶点都没有路径, 否则转步骤 3。

**步骤 3** 置  $S=S \cup \{V_k\}, R=R-S$ , 若  $R=\emptyset$ , 结束, 所有最短路径  $(V_i, V_j)$  的长  $U_j$  都已经求出, 否则转步骤 4。

**步骤 4** 对一切  $V_j \in R$ , 设置:  $U_j=\min\{U_i, U_k+W_{kj}\}$ , 再转步骤 1。

以上算法过程中, 在步骤 2 同时记录下个最小  $W$  的结点编号  $MinStopID$ , 如果最小结点编号  $MinStopID$  等于  $V_j$  终结点号, 那么说明找到了最短路径, 如果  $MinStopID$  与原来的相同, 则说明没有找到最短路径。在这两个条件下, 可以转到步骤 4, 则转到步骤 2。

### 2.4 算法效率分析

为了验证该算法的执行效率, 笔者采用以下系统测试环境对程序进行测试, 即操作系统为 Windows CE, CPU 为 Intel 2.0 GHz, 内存为 1 GB。

图 2 是将该算法应用于焦作市地理信息公众查询系统中的一个图示(小于 2000 条弧段, 查询道路时间 < 1/8 s)。



图 2 基于 Dijkstra 的 GIS 路径分析算法结果示例图

表 2 网络分析效率表

边数	占用内存/B		花费时间/s	
	改进后算法	传统算法	改进后算法	传统算法
5 000	<100 K	<300 K	<0.25	<2
10 000	<200 K	<600 K	<1	<3
20 000	<400 K	<1.2 M	<1	<4
30 000	<600 K	<1.8 M	<1	<4

## 3 结语

空间分析是地理信息系统区别于一般信息系统的标志, 它以空间数据和属性数据为基础, 回答真实地理客观世界的有关问题, 网络分析是空间分析的重要组成部分, 文中通过改变图的存储结构及搜索方法, 与传统 Dijkstra 相比不仅在存储空间上, 而且在时间上具有非常高的效率。

### 参考文献:

- [1] 严蔚敏, 吴伟民. 数据结构[M]. 北京: 清华大学出版社, 1997.
- [2] 陆锋, 卢冬梅, 崔伟宏. 基于四叉堆优先级队列及逆邻接表的改进型 Dijkstra 算法[J]. 中国图象图形学报, 1999, 4(12): 1039-1045.
- [3] 王开义, 赵春江. GIS 领域最短路径搜索问题的一种高效实现[J]. 中国图象图形学报, 2003(8).
- [4] 蔡少华. GIS 图形空间关系的研究与实践[D]. 郑州: 解放军测绘学院, 1999.
- [5] 古凌岚. GIS 最短路径分析中 Dijkstra 算法的优化[J]. 计算机与数字工程, 2006(12).
- [6] 王宏勇, 卢战伟. 嵌入式 GIS 最短路径分析中 Dijkstra 法改进[J]. 测绘学院学报, 2005(3).