

基于 GIS 的城市道路网最短路径算法探讨

严寒冰 刘迎春

(浙江工程学院电子与自动化系 杭州 310033)

摘 要 文中从城市道路网络的特点出发,分析了道路路段间的连通关系,得出一种求城市道路网络两节点间的最短路径的算法。算法的复杂度与网络节点数 N 成线性关系,即 $O(N)$ 。最后通过在 GIS 软件 Map Info 中编制的算法程序实例,验证了算法的实用性和可靠性。

关键词 最短路径, 城市道路网络, GIS, 算法

中图法分类号 TP391

A New Algorithm for Finding Shortcut in a City's Road Net Based on GIS Technology

YAN Han-Bing LU Ying-Chun

(Department of Electronics and Automation, Zhejiang Institute of Science and Technology, Hangzhou 310033)

Abstract This article discusses a new algorithm which gets the shortest path between two points in a city's road net. This algorithm is based on the geographic relevance relationship among roads in the road net. The presented algorithm's complexity is just proportion to the number of nodes in the road net while conventional shortcut algorithm's complexity is proportion to the power of the number of nodes. At the end of the article, it provides some instance which prove that the algorithm is applied and reliable.

Keywords shortcut, city's road net, GIS, arithmetic

1 引 言

通常的最短路径算法,往往是建立在抽象的数学模型之上,即网络模型。在这种网络模型上,实际的路径被抽象为网络中的一条边,实际路径的长度与网络边的长度可以不成比例,以边的权值来表征路径的长度(或其它特征),在该网络上求某点到其它任一点的最短路径的方法,被称为最短路径算法。最短路径问题^[1]在数学中被认为是NP问题,各种算法即使是较优的狄克斯特累算法(Dijkstra's

algorithm)在求解时都有可能准备搜索所有的网络节点,在网络节点数较大情况下,其算法的时间花费成倍甚至幂次增长,很难满足实际运算的需要。

本文在对城市道路网络进行大量分析、实验的基础上^[2],认为过于抽象的网络模型忽略了道路网络中某些内在的拓扑信息,如边之间的相对位置关系等。在对道路网络进行一定数学抽象的基础上,建立了一个基于地理相对关系的数学模型,提出了几种较优的求城市道路网络最短路径算法,适用于不同的求解条件。通过算法检验^[3],给出了算法的时间花费,并在实际应用中证明了算法的实用性和可靠性。

2 模型的建立

城市交通枢纽主要由街道和河流组成, 但重点为街道的集合。交通图主要由众多街道相交、相连而构成, 并组成纵横交织、错综复杂的城市交通网络图。在交通网络图中, 街道间的地理位置关系相当复杂, 一条街道可能与若干条街道相交、相连, 且其相交、相连的模式复杂。为了避免过多地考虑街道间的拓扑关系, 在本文中抽取交通网络图中街道交叉路口作为分析的对象之一, 并对分析的另一对象——街道, 以交叉路口为点进行分割, 成为路段。这样, 整个网络图将由交叉路口点和路段组成, 并定义交叉路口点为网络的节点, 路段为网络的边。在一张城市交通网络图上, 从自然的角度可得出以下特点:

- (1) 网络中节点和边的数目都远在100个以上;
- (2) 在城市1:1000(或以上)的地图上, 路段(网络边)是近似直线或弧度数不大的(特别在经过规划的现代大都市);
- (3) 网络的拓扑关系复杂;
- (4) 边通常是双向可通的。

根据以上城市交通网络的特点, 可做如下的分析假设:

(1) 所有的路段(边)都是直线。对于弧度数稍大的路段, 为了求解弧段间夹角的方便并使所得的夹角能正确反映弧段间的相对位置关系, 可通过在路段的拐点处加一个节点的方法来减小弧度数, 如图1所示, 节点1、2之间的路段的弧度较大, 在路段上加一节点3, 把原来的路段分割为两个弧度相对较小的路段。即在网络中增加一个节点, 把一条边分为两条边, 每条边都可看做直线进行求角度的运算, 但边的其它属性(如长度)仍由弧段的属性(如曲线长度)来表征;

- (2) 边都是双向可通的;
- (3) 节点都为实节点

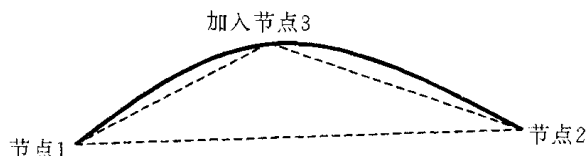


图1 加节点减小弧度数

城市交通网络图是一张矢量化的城市道路交通地图。在桌面地理信息系统软件Map Info中对城市道路交通地图的扫描图进行矢量化处理, 抽取

城市交通网络的节点和边, 即地图中的路段交叉点和路段。网络中节点和边的拓扑关系通过节点和边的属性字段值来定义, 即在节点的属性字段中有相交边的信息, 以及节点的相对地理坐标(X, Y); 在边的属性字段中有边的起、止节点的信息以及该边的权值(如长度)。这些属性值的赋予, 可通过Map-Info 的编程语言MapBasic^[4]编程实现。在Map Info中打开地图窗口, 即可看到一张清晰的城市交通电子地图。

3 准备算法(求最短路径的最差解 most weak solution)

3.1 算法原理与描述

从几何学中知道, 两点间直线最短。在错综复杂的道路网上, 任选两点, 其最短的路径可暂设为一条从起点到终点的直线, 但实际上这条直线作为一段道路存在的可能性极小, 但这条从起点到终点的直线代表了一个路线的趋势, 顺着这个方向的某条道路是最短路径的可能性较大。

在图2中, 求A点到B点的最短路径, AB是连接起点A和终点B的直线, AB的长度应为A点到B点的最短路径长的下限值, 如果AB直线作为网络的一条边存在, 则AB直线即为A点到B点的最短路径; 如AB不是网络中的一条边, 则在准备算法中, 从起点A出发找到一条边, 该边与AB的夹角最小, 即取

$$|f(AB, AP)| = \min_{k \in D} (|f(AB, AP_k)|) \quad (1)$$

其中, D 为所有与A点相连的节点的数目, $f(l_1, l_2)$ 为求直线 l_1, l_2 的夹角的函数。

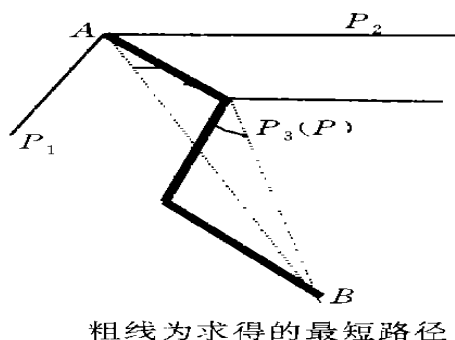


图2 求从A到B最短路径的准备算法

取P为当前起点, 替换A点, 找下一个与PB直线夹角最小的边, 如此反复直至所得当前起点P与终点B重合。这样得到的一组路段即是起、止点间最短路径的一个解。

3.2 算法分析

在准备算法中,从起点开始,每经过节点时,只有一条边、一个节点被选中,直到选中的节点与终点重合,算法结束。所以算法所要搜索的节点、边数极少,从而时间花费很少。而所得的这条路径具有了起、止点的方向性趋势特征,所以在很大程度上可作为最短路径的参考。但由于算法条件考虑过于简单,造成该路径可能因振荡而增加了路径总长度(如图3所示),因而其可信度不高。本算法的目的就是得到最短路径的一个最差解,作为下步优化算法的启发性准备知识。

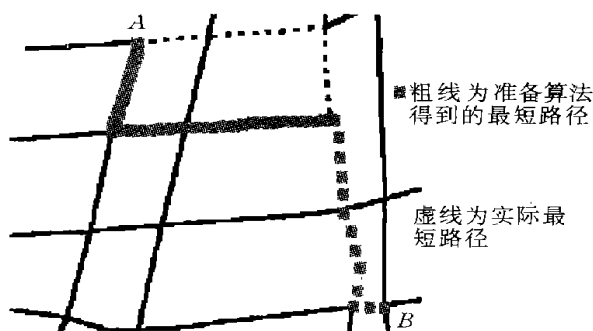


图3 准备算法得到的有振荡的最短径

4 优化算法(最短路的优化解 optimized solution)

4.1 算法原理

准备算法中每经过一个节点只选中与起、止直线夹角最小的一条边,虽然考虑了路段的方向性特征,但取的是局部最优解而非全局最优解,从而造成结果不够理想。如果在经过每一个节点的同时进行起、止点直线左右两边各一条夹角最小的边的搜寻,即搜索一条二叉树,就会提高解的精确性,但同时增加了搜索的时间。

求A点到B点的最短路径,设有一路径队列,最先放有起点A。取出A作为当前节点, D_1 为所有与A点相连且 AD_1 与 AB 的夹角为负的节点, D_2 为所有与A点相连且 AD_2 与 AB 的夹角为正的节点,即

$$(AB, AP_{K_1}) < 0, \quad K_1 \in D_1 \quad (2)$$

$$(AB, AP_{K_2}) \geq 0, \quad K_2 \in D_2 \quad (3)$$

在 D_1 和 D_2 中分别找出 P_1 和 P_2 ,满足式(1),它们的父亲都为A,把 P_1, P_2 放入路径队列。依次取出队列中的节点作为考虑对象,找出它们的子节点并

放入队列,如此反复,直至队列为空。每个当前节点将最多有两个子节点。为了减少时间花费,并使构造的二叉树最小,采用以下几种方法:

(1) 用准备算法所得的最差解作为优化算法的最短路径长的初始值 L_{\max} ;

(2) 采用标号法,使对路径的距离体现在节点上,即用起点到该点的当前累加距离标注该点,如初始时A点的累加长度为0, B点的累加长度为 L_{\max} ;

(3) 对二叉树的遍历,采用广度优先的方法,尽量减少二叉树的结点遍历数;

(4) 对子节点 P_1, P_2 ,在加入路径队列前,进行判断。如果为终点B或为已经访问过的节点,则不加入路径队列,但要对其影响的节点和子节点的累加长度标志进行改变;如果该子节点当前的累加长度加上该点到终点B的直线距离大于终点B当前的累加长度,则也不加入路径队列,即不作为二叉树的分支。

通过以上算法策略和限制,实际得到的二叉树节点数较少,图4中的粗点即是用优化算法求A到B最短路径生成的二叉树的所有结点。

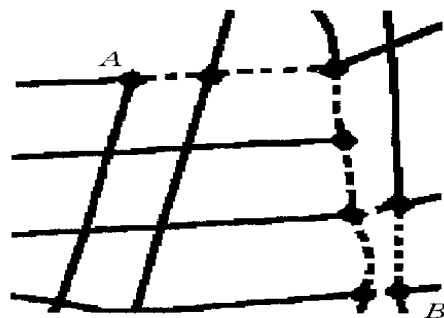


图4 优化算法得到的二叉树

4.2 算法描述

(1) 定义

$N ode[]$ 为具有以下结构的数组:

$N ode[] . no$ 节点的编号

$N ode[] . len$ 节点的当前累加长度

$N ode[] . father$ 节点的父节点

$turn$ 为当前访问到的 $N ode[]$ 数组的序号

num 为 $N ode[]$ 中加入的节点的总数

(2) 初始化

$N ode[beg in] . no =$ 起点编号;

$N ode[beg in] . len = 0$;

$N ode[beg in] . father = 0$;

$N ode[end] . no =$ 终点编号;

$N ode[end] . len =$ 准备算法中得到的最短路最差解(已做过准备算法)或为无穷大;

turn= 0;
num = 1;
(3) 终止条件

如 $num < turn$, 表示 $N ode[]$ 数组中的节点都已被处理了, 终止循环 此时的 $N ode[end] \cdot len$ 就是所得的最短路径长, 输出最短路径长并通过 $father$ 找到求得的最短路径, 在地图上显示, 程序终止 否则循环执行以下步骤

(4) 循环执行

turn 加1; 找与 $N ode[turn] \cdot no$ 相交的边; 计算每条边与起、止点直线 L 的夹角, 找出直线两边各自夹角最小的边; 得到这些边的端节点的编号, 判断所得的端节点为以下情况:

为终点 如该端节点的累加长度小于 $N ode[end] \cdot len$, 则把该端节点的累加长度赋给 $N ode[end] \cdot len$, 并且 $N ode[end] \cdot father = N ode[turn] \cdot no$ 此端节点不加入 $N ode[]$ 数组中

为已访问过的点 如该端节点的累加长度小于原已访问的点, 则把该端节点的累加长度赋给已访问过的点的 len , 把 $N ode[turn] \cdot no$ 赋给已访问过的点的 $father$, 并改变所有已访问过点的子节点的累加长度, 该端节点不加入 $N ode[]$ 数组

该端节点的累加长度加上该端节点到终点的直线距离大于 $N ode[end] \cdot len$, 则不加入 $N ode[]$ 数组

不符合以上3点的端节点, 加入 $N ode[]$ 数组, 使 num 加1, 并把 $N ode[num]$ 的 $no, len, father$ 赋为相应的值

5 理想算法(最短路径精确解 perfect solution)

在理想情况下, 每经过一个节点, 与其相连的所有节点都要考虑 设 D_i 为所有与当前节点(如 A) 相连的端节点, 在优化算法的循环执行步骤中, 取所有的 D_i 进行运算, 其它步骤和优化算法相同, 这时一个节点的子节点数和与之相连的边数相等, 如此求得的最短路径是精确解, 但时间花费将是最大的

6 算法检验

6.1 实验结果

本文介绍的基于 GIS 的最短路径算法, 已在 GIS 软件平台 Map Info 的支持下, 在 Delphi 4.0 中编程实现 实验所用的城市道路图之一是杭州市区的部分街道图(图5), 其路口节点数 $N = 129$, 路段(边)数 $R_N = 209$, 在该路网上任取 A, B, C, D, E, F 六点分别用准备算法、优化算法、理想算法求 AB, CD, EF 的最短路径, 结果如表1、表2; 实验示例之二是一张某县乡(镇)、村之间的公路网(图6), 其中网络的边数 $R_N = 69$, 节点数 $N = 38$, 分别求 QP, SR, TZ 之间的最短路径, 实验结果如表3、表4所示

表 1 初始值为无穷大时3种算法的比较

项目	路 径											
	AB				CD				EF			
	N_1	N_2	$T(s)$	$L(m)$	N_1	N_2	$T(s)$	$L(m)$	N_1	N_2	$T(s)$	$L(m)$
准备	13	13	10	14181.9	9	9	6	8066.6	13	13	9	13458.3
优化	14	46	44	13403.1	9	22	22	8066.6	11	21	20	9003.1
理想	14	80	74	13403.1	9	59	60	8066.6	11	89	78	9003.1

表 2 初始值为准备算法所得的最差解时算法比较

项目	路 径								
	AB			CD			EF		
	N_1	N_2	$T(s)$	N_1	N_2	$T(s)$	N_1	N_2	$T(s)$
优化	总计	14	30	31	9	15	15	11	22
			43	41		24	21		35
理想	总计	14	34	32	9	15	17	11	32
			47	42		24	23		45

表 3 初始值为无穷大时3种算法比较

项目	路 径											
	QP				SR				TZ			
	N_1	N_2	$T(s)$	$L(Km)$	N_1	N_2	$T(s)$	$L(Km)$	N_1	N_2	$T(s)$	$L(Km)$
准备	6	6	5	49.6	3	3	3	29.6	6	6	6	42.1
优化	6	23	22	41.5	3	8	8	29.6	6	51	49	42.1
理想	6	50	46	41.5	3	9	9	29.6	6	69	67	42.1

表 4 初始值为准备算法所得的最差解时算法比较

项目	QP			SR			TZ		
	N_1	N_2	$T(s)$	N_1	N_2	$T(s)$	N_1	N_2	$T(s)$
优化	6	13	12	3	3	3	6	9	9
	总计	19	17	6	6	6	15	15	15
理想	6	25	22	3	3	3	6	11	10
	总计	31	27	6	6	6	17	16	16

注: N_1 为组成最短路径的路段个数; N_2 为求解过程中搜索的节点数; T 为求解所需时间, 单位为 s (实验所用计算机配置为: 586/100MHz, 16M 内存); L 为最短路径的长度, 单位为 m 或 km. 表 2 和表 4 中的“总计”表示包括准备算法的时间

实验效果图如图 5



图 5 杭州市路网求最短路径示例

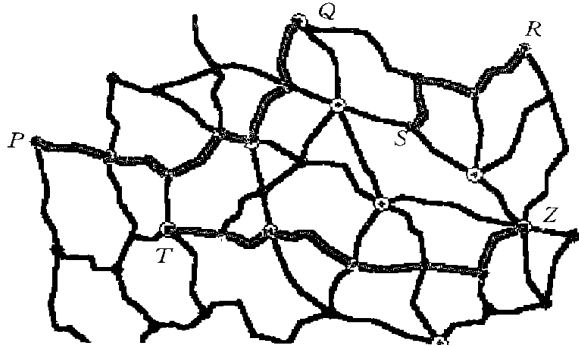


图 6 某县公路网求最短路径示例

6.2 实验讨论

从实验结果中的表 1—表 4, 可看出:

1) 求解的时间与算法求解过程中搜索的节点数 N_2 大致成正比;

2) 准备算法中所得的路长比实际最短路径长略大或相等;

3) 优化算法所得路长基本等于实际路长, 时间花费较多. 其中有准备算法最差解作为初始值的优化算法所用的时间总花费 (包括求最差解的时间) 比初始值为无穷大的优化算法的时间花费要少, 并且有最差解作为初始值的优化算法求最短路径在准确性和时间花费上都是最优的

4) 理想算法简单稳定且结果精确, 在最坏的情况下, N_2 等于 N , 时间花费相对较多. 有准备算法最差解作为初始值的理想算法所用的时间总花费 (包括求最差解的时间) 比初始值为无穷大的理想算法的时间花费明显减少.

5) 3 种算法在最坏的情况下, 每经过某个节点, 所做的比较和运算次数最多为该节点所连的边数, 并且, 访问过的节点将不会再被访问. 设网络中的节点总数为 N ; 节点的出度为 $d[n]$, $n = 1, 2, \dots, N$; $D = \max(d[n])$; 则算法的复杂度与 $D \times N$ 成正比, 其中 D 为有限的常数, 所以算法复杂度可表示为 $O(N)$.

6.3 实验结论

传统的求最短路径算法, 算法的复杂度与节点数 N 的幂次方成正比, 如狄克斯特累算法, 复杂度为 $O(N^3)$, 当节点数较大时, 算法的时间花费极为庞大. 本文探讨的几种算法的复杂度仅与节点数 N 成线性关系, 因此在算法的时间复杂度上的优越性是极其明显的, 而这几种算法与传统算法的精确度、可靠性也相差不大, 实践证明: 对于具体问题, 本文所探讨的算法具有更大的实用性和可操作性.

7 总 结

本文探讨的最短路径算法, 是基于网络中边 (路段) 与节点的地理相关性的拓扑特征. 其中准备算法利用两点间直线最短的原理, 求解过程简单, 速度快, 得到的最短路径长具有一定的近似性. 在优化算法中, 准备算法得到的最短路径长作为实际最短路径长的最差解被赋为初始值, 优化算法构造一个包括最短路径的二叉树, 在构造过程中受到初始最短路径长的限制而简化了算法的复杂度. 优化算法得到的最短路径依然是一个近似解, 按照最理想的情况, 对每个节点都要搜寻与其相连的边, 这样得到的最短路径将是理想的最短路径, 但实际搜寻的节点数较多. 优化算法充分应用准备算法得到的最短路径

的方向性特性的经验, 在理想最短路径的算法思想中搜寻最有可能的两条路径, 减少了算法的复杂度, 是介于准备算法和理想算法之间的一种较优算法

参 考 文 献

- 1 Wang Chaorui. Graph Theory, Second Edition. Beijing: Beijing University of Technology Press, 1997(in Chinese)
(王朝瑞 图论 第二版 北京: 北京理工大学出版社, 1997)
- 2 Liu Yingchun. The research for application of geography information system [Master dissertation]. Zhejiang Institute of Silk Texture, Hangzhou, 1999(in Chinese)
(刘迎春 地理信息系统的应用研究[硕士学位论文] 浙江丝绸工学院, 杭州, 1999)
- 3 Xu Zhuoqun, Zhang Naixiao *et al*. Data Structure. Beijing: Higher Education Press, 1981(in Chinese)
(许卓群, 张乃孝等 数据结构 北京: 高等教育出版社, 1981)
- 4 Map Info Corporation. MapBasic Reference Guide. Map Info Corporation, 1997