

# 最小生成树算法及其应用

段东东

(西安高等电力专科学校 基础部, 陕西 西安 710000)

**摘要:** Kruskal 算法和 Prim 算法是求最小生成树的常用算法, 设计了这两种算法的 C 语言程序, 并通过实例研究了这两种算法的实际应用价值。

**关键词:** 最小生成树; 图; Kruskal 算法; Prim 算法

**中图分类号:** O29; TB11 **文献标识码:** A **文章编号:** 1008-9233(2010)01-0055-03

## 1 引言

求最小生成树问题, 就是要在一个连通赋权网络中, 寻找最小权数的支撑树. 给定网络  $G = (V, E, W)$ , 设  $T = (V, E')$  为  $G$  的一个支撑树, 令  $w(T) = \sum_{e \in E'} w(e)$  表示  $T$  的权.  $G$  中权最小的支撑树成为  $G$  的最小生成树:  $w(T^*) = \min\{w(T)\}$ . 例如, 某市区有七个住宅小区需要铺设天然气管道, 各个小区的位置及它们之间可修建管道路线与费用如图 1(a) 所示. 现要设计一个管道铺设路线, 要求天然气能输送到各个小区并且修建的总费用为最小, 这就是求最小生成树的问题. 这里图 1(b) 是图 1(a) 的一棵最小生成树, 即天然气能输送到各个小区, 若按图 1(b) 铺设, 所需费用最小<sup>[1]</sup>.

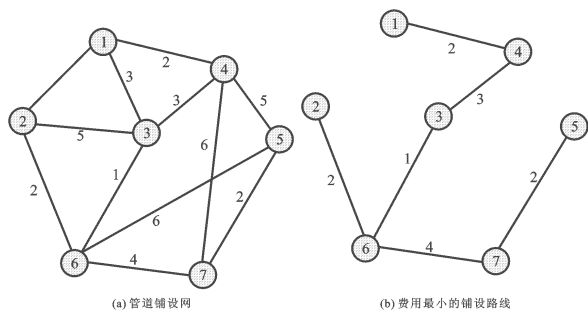


图 1 管道铺设网

目前, 求最小生成树的算法已经相当成熟, 常用的经典算法<sup>[2]</sup> 有避圈法, 边割法, 破圈法, Sollin 算法, Dijkstra 算法等. 本文在对普里姆(Prim)算法和克鲁斯卡尔(Kruskal)算法分析的基础上, 设计了这两种算法的 C 语言程序, 通过实例说明了算法的应用, 并给出了相应的图解。

## 2 算法思想及步骤

### 2.1 Kruskal 算法

也称避圈法, 是由 Kruskal 于 1956 年提出的。

#### 2.1.1 算法思想<sup>[3]</sup>

每次选取权值最小的边  $e$ , 加入  $T$  中, 如果此时构成回路, 那么它一定是这个回路中的最长边, 删之. 直至达到  $n-1$  条边为止. 这时  $T$  中不含任何回路, 因此是树, 而且是最小生成树。

#### 2.1.2 算法步骤<sup>[4]</sup>

Step1. 把图  $G$  的边按权值由小到大排序, 即  $w(e_1) \leq w(e_2) \leq \dots \leq w(e_m)$ , 令  $T_0 = (V, \emptyset)$ ,  $i = 1, j = 0$ ;

Step2. 若  $T_j + e_{j+1}$  含有回路, 则执行 Step3; 否则执行 Step4;

Step3. 置  $i = i + 1$ , 若  $i \leq m$ , 则执行 Step2; 否

收稿日期: 2009-09-22

作者简介: 段东东(1967—), 女, 浙江省宁波市人, 陕西师范大学数学系毕业, 学士, 讲师, 主要从事高等数学、离散数学方面的教学和研究工作。

则停止,  $G$  中不存在最小生成树;

- Step4. 令  $T_{j+1} = T_j + e_{j+1}$ , 并置  $j = j + 1$ ;
  - Step5. 若  $j = n - 1$ , 则结束,  $T_j$  是最小生成树;
- 否则执行 Step3;

2.2 Prim 算法:

也称边割法, 是 Prim 于 1957 年提出的。

2.2.1 算法思想

任意选取一节点  $v$  构成集合  $V'$ , 然后不断在  $V - V'$  中选一点  $v_k$  到  $V'$  中某点(如:  $v_i$ ), 权最小的边( $v_k, v_i$ ) 加入树  $T$  中, 并令  $V' = V' \cup \{v_k\}$ , 直至  $V' = V$ 。

2.2.2 算法步骤<sup>[4]</sup>

- Step1. 设  $v$  为  $V$  的任一顶点, 令  $S_0 = \{v\}$ ,  $E_0 = \emptyset$ ,  $k = 0$ ;
- Step2. 若  $S_k = V$  结束, 以  $S_k$  为顶点集,  $E_k$  为边集的图即是  $G$  的最小生成树; 否则执行 Step3;
- Step3. 构造 $[S_k, S']$ , 若 $[S_k, S'] = \emptyset$ , 则  $G$  不连通, 停止; 否则, 设  $w(ek) = \min_{e \in [S_k, S_k]} w(e)$ ,  $ek = v_k v'_k$ ,  $v_k \in S_k$ , 令  $S_{k+1} = S_k \cup \{v'_k\}$ ,  $E_{k+1} = E_k \cup \{ek\}$ , 置  $k = k + 1$ , 执行 Step2.

3 实例应用

某公司规模不断扩大, 在全国各地设立了好多分公司, 为了提高公司的工作效率使各分公司之间的信息可以更快、更准确的进行交流, 该公司决定要在各分公司之间架设网络, 由于地理位置和距离等其它因素, 使各分公司之间架设网线的费用不同, 公司想使各分公司之间的网络畅通并且把费用降到最低, 那么应该怎样来设计各分公司及总公司间的线路? 该公司的所有分公司及总公司的所在位置如图 2 所示。顶点代表位置及名称, 边代表可以架设网线的路线, 边上的数字代表架设该网线所需要的各种花费的总和。这样就构成了一个带权连通图, 从而就转化为求所得到的带权连通图的最小生成树的问题。

3.1 利用 Kruskal 算法求其最小生成树

Kruskal 的计算机实现过程<sup>[5]</sup> 如下:

- 1. 输入图的顶点数及边数;
- 2. 将图的各边端点及权值输入;
- 3. 程序的运行结果为:

the set of the minimum spanning tree of the

graph:

- edge( 5, 8), right: 1
  - edge( 2, 4), right: 2
  - edge( 3, 6), right: 3
  - edge( 4, 7), right: 4
  - edge( 1, 2), right: 5
  - edge( 6, 7), right: 5
  - edge( 4, 5), right: 8
- the sum of right of the graph: 28
- 所得最小生成树如图 3。

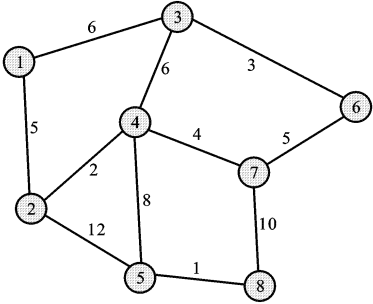


图 2 各分公司及总公司间架设网络的布线图

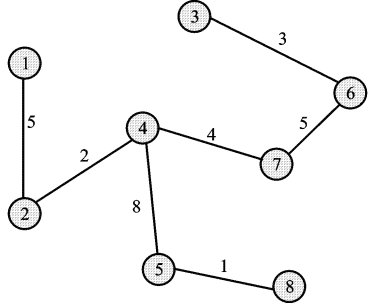


图 3 Kruskal 算法得到的最小生成树

3.2 利用 Prim 算法求其最小生成树

Prim 算法的计算机实现过程<sup>[5]</sup> 如下:

- 1. 输入图的顶点数;
- 2. 输入图的邻接矩阵(即两点之间有边的输入权值);
- 3. 最后得到的结果为:

the set of eht minimum spanning tree of the graph:

- path 1: from vertex 1 to vertex 2, right: 5
- path 2: from vertex 2 to vertex 4, right: 2
- path 3: from vertex 4 to vertex 7, right: 4
- path 4: from vertex 7 to vertex 6, right: 5
- path 5: from vertex 6 to vertex 3, right: 3
- path 6: from vertex 4 to vertex 5, right: 8
- path 7: from vertex 5 to vertex 8, right: 1

the sum right of the minimum spanning tree of the graph: 28  
所得最小生成树如图 4。

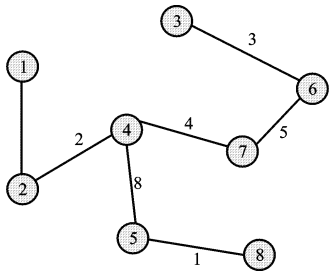


图 4 算法得到的最小生成树

3.3 算法分析

从两种算法的实现步骤可以看出:  
(1)Prim 算法实现起来简洁清晰易懂, 并且使用的判断语句较少, 而 Kruskal 算法实现起来相对来说步骤多、判断语句多。  
(2)由两种算法的图解可以看出 Prim 相对来说更清晰、更容易实现、更易理解。  
(3)从结果来看, 利用 Prim 算法从顶点 start 出发求出用邻接矩阵 G 来表示图的最小生成树与利用 Kruskal 算法求结构体数组的最小生成树是一致的<sup>[5]</sup>。  
(4)由程序本身可以看出 Kruskal 算法的时间复杂度为  $o(e^2)$ , 与顶点数无关, 所以适合求边稀疏

的网的最小生成树, 而 Prim 算法的时间复杂度为  $o(n^3)$ , 与边数无关, 所以适合求边稠密的网的最小生成树。

4 结语

本文对 Kruskal 算法和 Prim 算法进行分析研究, 设计了他们的 C 语言程序。此程序可以寻找任意带权连通图的最小生成树, 给出了相应的图解, 最后说明了算法的应用; 但此 C 语言程序中缺少判断图连通性的部分, 所以此程序只适用于求任意带权连通图, 而对于带权非连通图无法判断, 此程序有待于进一步完善。

参 考 文 献

[ 1 ] Fred Buckley, MaryLewinter. 图论简明教程[ M] . 李慧霸, 王凤芹, 译. 北京: 清华大学出版社, 2005.  
[ 2 ] Cormen TH, Lleiserson CE, Rivest RL. Introduction to Algorithms[ M] . USA: MIT Press, 2001.  
[ 3 ] 高一凡.《数据结构》算法实现及解析[ M] . 西安: 西安电子科技大学出版社, 2002.  
[ 4 ] 刘瓚武. 应用图论[ M] . 长沙: 国防科技大学出版社, 2006.  
[ 5 ] 侯识忠. 数据结构算法程序集[ M] . 北京: 中国水利水电出版社, 2005.

[ 责任编辑、校对: 李小光]

Minimal Spanning Tree Algorithm & Its Application

DUAN Dong-dong

(Department of Basic Courses, Xi'an Electric Power College, 710000, Xi'an, Shaanxi, China)

**Abstract:** Kruskal and Prim are the ordinary algorithms for the computing of minimal spanning tree. The article designs C—language program of the two algorithms, and studies the application of the two algorithms through practical examples.  
**Key Words:** Minimal spanning tree; Kruskal; Prim; Algorithm