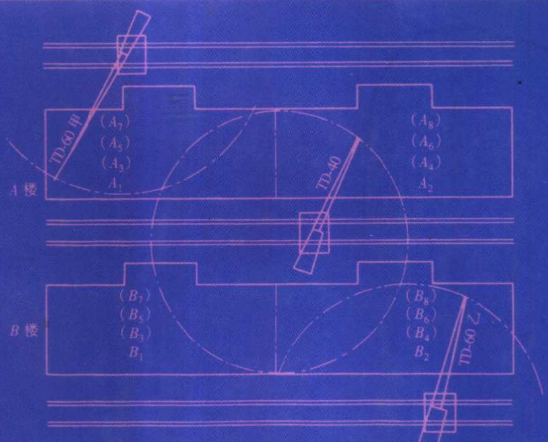
网络计划技术与施工组织设计

主　编　曹吉鸣　徐　伟
副主编　张维国　赵　雷
　　　　蔡雪峰　林跃忠



同济大学出版社

网络计划技术与施工组织设计

主　编　曹吉鸣　徐　伟
副主编　张维国　赵　雷　蔡雪峰　林跃忠

同济大学出版社

图书在版编目(CIP)数据

网络计划技术与施工组织设计/曹吉鸣,徐　伟 主编 .—上海:同济大学出版社,2000.6

ISBN 7-5608-2140-5

Ⅰ.网…　Ⅱ.① 曹… ② 徐 Ⅲ.网络计划技术-应用-建筑工程-施工组织　Ⅳ.TU721

中国版本图书馆 CIP 数据核字(2000)第 16120 号

网络计划技术与施工组织设计

主　编　曹吉鸣　徐　伟

副主编　张维国　赵　雷　蔡雪峰　林跃忠

同济大学出版社出版发行

(上海四平路 1239 号　邮编 200092)

全国新华书店经销

崇明晨光印刷厂印刷

开本:787×1092　1/16　印张:16.25　字数:410 千字

2000 年 6 月第 1 版　　2000 年 6 月第 1 次印刷

印数:1—4000　　定价:25.00 元

ISBN7-5608-2140-5/TU·360

前　言

随着我国建设事业的飞速发展，土木工程施工领域的现代化技术水平日新月异，我们的施工技术已能建造属于世界上一流的超高层的建筑物、超大跨度的桥梁、最新颖结构的公共建筑，同样我们的施工管理也达到了世界上大型建筑企业的先进水平。近年来，如此巨大的进步促使我们必须在已经取得的成绩的基础上，进一步研究和完善土木工程施工领域的管理科学、技术理论和实用方法。本书正是在此前提下应用当前较新的网络技术成果，结合土木工程施工领域的实践，阐述服务于施工第一线的网络技术应用理论和实用方法，并引录了一些编制实例，为网络计划技术在施工组织设计中的更好应用提供方便。本书可作为高等院校土木类及相关专业本科和研究生的教材，也可作为生产第一线的土木工程施工技术人员的工作参考书。限于我们的水平，书中不足之处，恳请读者给予批评指正。

全书共分 10 章。第 1 章至第 6 章由曹吉鸣、徐伟编写；第 7 章由曹吉鸣、李映红、林跃忠编写；第 8 章由曹吉鸣、林跃忠、阮永辉编写；第 9 章由赵雷、曹吉鸣编写；第 10 章由曹吉鸣、蔡雪峰、朱圣妤、于晓音编写。最后由曹吉鸣、徐伟、张维国进行统纂定稿。

在本书编写过程中曾得到了各有关单位的大力支持，在此表示衷心的感谢。

谨以此书献给新世纪的到来。

编者　于同济大学

1999 年 11 月

目　录

第 1 章　网络图的基本理论

50 年代以来，为了适应生产发展和关系复杂的科学研究工作开展的需要，国外陆续采用了一些计划管理的新方法，网络计划技术是其中之一，它是由箭线和节点组成，用来表达各项工作的先后顺序和相互关系。这种方法逻辑严密，主要矛盾突出，有利于计划的优化调整和电子计算机的应用，因此在工业、农业、国防和关系复杂的科学研究计划管理中，都得到了广泛的应用.

现代化施工需要及时准确地收集、整理、贮存和检索各类信息，反映实际生产状况，这不仅要求我们迅速编制施工计划，而且在实施过程中对施工计划不断进行动态控制、调整和优化，合理安排各种资源，从而缩短工期，降低成本，这些都离不开网络计划技术的发展和计算机的应用，两者结合是实现计划管理科学化、现代化的重要手段。

网络图和网络分析作为图论的重要内容，已广泛地应用于实际生活、生产和科学研究的各个领域。借助于网络图的知识来研究组织管理、安排工程计划的优化问题，将庞大复杂的工程系统和管理问题用图来描述，已成为当代管理科学的重要手段。例如，在组织生产中，为完成某项生产任务，各工序之间怎样衔接，才能使生产任务完成的既快又好；一个邮递员送信，要走完他负责投递的全部街道，完成任务后回到邮局，应该按照怎样的路线走，所走的路程最短；各种通信网络的合理架设，交通网络的合理分布；完成工程任务的时间最少、距离最短、费用最省等问题，应用网络图的基本理论和方法求解，形象直观，且很简便。随着科学技术的发展以及电子计算机的广泛应用，网络图受到了工程技术人员和经营管理者等各方面人士的越来越广泛的重视。

1.1　图、树及其性质

图论是数学的一个古老分支。从本质上讲，图论就是一维拓扑学，也就是一维的抽象几何学。它借助于点、线为任何一个包含了一种二元关系的系统提供了一个数学模型。

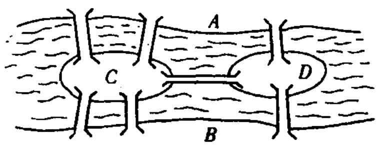
1.1.1　图

图是由点和连线构成的图形，它指的是由若干个点和连接这些点中的某些“点对”的连线所组成的图形。具体地说，它可以用来表示城市与公路、电话与电话线、电子元件与导线、甚至人与关系等二元关系。

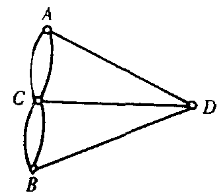
欧拉在 1736 年解决了著名的哥尼斯堡七桥问题。哥尼斯堡城中有一条河叫普雷格尔河，该河中有两个岛，河上有七座桥。如图 1-1(a)所示。

当时那里的居民热衷于这样的问题：一个散步者能否走过七座桥，且每座桥只走过一次，最后回到出发点。

1736 年欧拉将此问题归结为如图 1-1(b)所示图形的一笔画问题。即能否从某一点开始一笔画出这个图形，最后回到原点，而不重复。欧拉证明了这是不可能的，因为图 1-1(b)



(a)



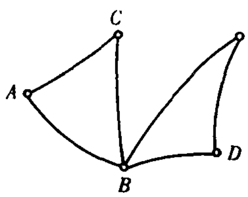
(b)

图 1-1　哥尼斯堡七桥问题

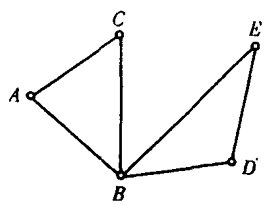
中的每个点都只与奇数条线相关联,不可能将这个图不重复地一笔画成。

这里所说的图与数学中的几何图形不完全相同。图 1-1(b)中两点间连接的长短不与实际距离成比例,连线只能反映两点间具有某种关系。反之,无连线只能说明这两点间不具有某种关系。例如,在交通路线图上,产地和销地通常用点来表示,有连线就意味着这两地直接可以到达,无连线就说明不能直接到达。至于两点间连线是直的还是曲的无关紧要。

例如,图 1-2(a)表示某地区的公路交通图,A,B,C,D,E 表示五个城镇,两城之间的连线表示公路。若 A,B 间有公路相通,则两城间有一条连线(即边,它表示 A 和 B 之间的特定关系),我们感兴趣的是 A,B 之间是否有这种特定关系。也就是说,两点间有无连线是重要的,而连结的方式无关紧要。因此,图 1-2(a)所示的两城间公路也可以用直线表示,如图 1-2(b)所示。



(a)



(b)

图 1-2　某地区公路交通图

为了叙述方便,首先介绍有关图的几个基本概念和术语。

所谓图,是指由有 P 个点的非空有限集 $V=\{v_1,v_2,\cdots,v_p\}$ 和由 E 中 q 条边 $E=\{e_1,e_2,\cdots,e_q,\}$,其中

$$e_i=(v_i,v_j)\text{或 } e_i=(v_j,v_i),v_i,v_j\in V$$

构成的图形,记为 $G=(V,E)$。

如图 1-3 所示,图 G 中,

$$V=\{v_1,v_2,v_3\},\ p=3$$

$$E=\{e_1,e_2,\cdots,e_5\},\ q=5$$

其中　$e_1=(v_1,v_1),e_2=(v_1,v_2),e_3=(v_2,v_3),e_4$ 表示法同 $e_3,e_5=(v_1,v_3)$。

v_i,v_j 又称为 e_i 的两个端点。又称 v_i,v_j 与 e_i 是关联的。若边的两个端点相同,称其为**环**。如图 1-3 中 e_1 为环。

两点间多于一条边的称为多重边。如图 1-3 中 e_3, e_4 都是以 v_2, v_3 为端点的边。含多重边的图为多重图，无环且无多重边的图称为简单图，如图 1-4 所示。以后研究的图一般指的都是简单图。这种图中，任两点间只有一条边，所以通常用(v_i, v_j)即可表示某条边了。如图 1-4 中，v_2 和 v_3 所确定的边即可表示为(v_2, v_3)或(v_3, v_2)，不必另外给出边的记号了。

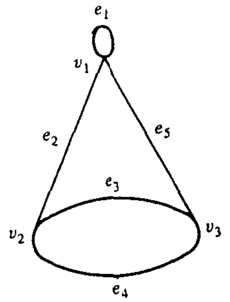


图 1-3　图 G

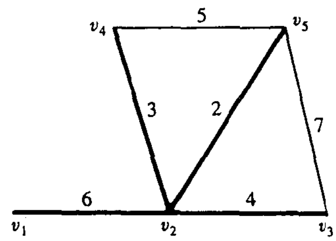


图 1-4　简单图

以点 v_i 为端点的边的个数称该点的**次**，记为 $d(v_i)$，次为偶数的点称为偶点，次为奇数的点称为奇点。如图 1-4 中，$d(v_2)=4, d(v_5)=3$ 等，其中偶点有 v_2, v_3, v_4，奇点有 v_1 和 v_5。请注意这样一个事实：这五个点的次的总和为 12，恰是边数 6 的 2 倍，这并不是偶然的巧合，而是对于任一无环的图中，所有的点的次数和都等于边数的两倍，即

$$\sum d(v_i) = 2q,\ v_i \in V$$

事实上，这是因为在计算各点的次数时，每条边在它的两个端点各计算了一次的缘故。

同时，还可知对于任一个无环图，奇点个数必为偶数。

1.1.2　连通图

任何一个图的最基本的性质之一是它是否连通，要给它下定义首先必须学习下面几个概念。

一个图 G 的某一点、边交替序列$\{v_1, e_1, v_2, e_2, \cdots, v_{n-1}, e_{n-1}, v_n\}$，其中 $e_i=(v_i, v_{i+1})$，$i=1, \cdots, n-1$，称为从点 v_1 到 v_n 的链。也可简记作$\{v_1, v_2, \cdots, v_n\}$。其中 v_1, v_n 称为链的两个端点。

形象地说，链即是从一个端点到另一点间的一条通道。例如图 1-4 中$\{v_1, v_2, v_5\}$，$\{v_1, v_2, v_3\}$，$\{v_2, v_3, v_5, v_2\}$都是链，链$\{v_1, v_2, v_5\}$可理解为从 v_1 到 v_5 的一条通道，也可理解为从 v_5 到 v_1 的一条通道。

一条链中若两个端点是同一点，即 $v_1=v_n$ 称为闭链或圈，上面所说的链$\{v_2, v_3, v_5, v_2\}$就是一个圈。

一条链的各边若均不相同则称此链为简单链。一条链的各点、各边均不相同则称此链为初等链。例如图 1-4 中$\{v_1, v_2, v_5, v_4, v_2, v_3\}$是简单链，但不是初等链。$\{v_1, v_2, v_5, v_4\}$是初等链，也是简单链。由定义可知，一个链若是初等链，必是简单链，是简单链却不一定是初等链。

对两个图间的关系定义如下：

两个图 $G=(V, E)$与 $G'=(V', E')$中，若满足条件 $V' \subseteq V, E' \subseteq E$，即 G'的点集和边集

分别是图 G 点集和边集的子集，称图 G' 是图 G 的一个子图。若满足条件 $V' = V, F' \subseteq E$，即点集不变，仅 G' 的边集是 G 边集的子集，称 G' 是 G 的支撑子图。

如图 1-4 中 $G' = \{v_2, v_3, v_5, v_4\}$ 是 G 的子图，但不包含 v_1 点，所以不是 G 的支撑子图。$G' = \{v_1, v_2, v_3, v_5, v_4\}$ 是图 G 的支撑子图，可见支撑子图是包含图 G 的所有点的子图。顾名思义“支撑”即是图的大架子，在图形上来说就是点集不变，只是少了几条边而已。

如果图 G 中任两点间至少有一条链则称 G 为连通图。反之图 G 为不连通。所以不连通图 G 中至少有一点与其他点间无通道，

任何图都有它的支撑子图，连通图的支撑子图尤为重要，特别是不成圈的支撑子图。

1.1.3 树

在各式各样的图中，有一类图是简单而重要的，就是树。我们把无圈的连通图称为树。

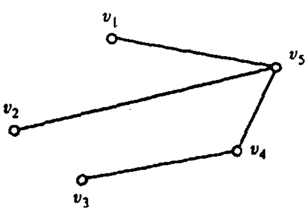


图 1-5 五城市电话网线图

例如有五个城市，要在它们之间架设电话线网，要求任何两个城市都可以彼此通话（允许通过其他城市），并且电话线的根数最少。

我们把电话线网用图来表示。为了使任何两个城市都可以通话，这样的图必须是连通的。其次，若图中有圈的话，从圈上任意去掉一条边，余下的仍是连通图，这样可以省去一根电话线。因此，满足要求的电话线网的图必定是不含圈的连通图，如图 1-5 所示。

某施工企业的组织机构如图 1-6(a)所示，如果用树表示，如图 1-6(b)所示。

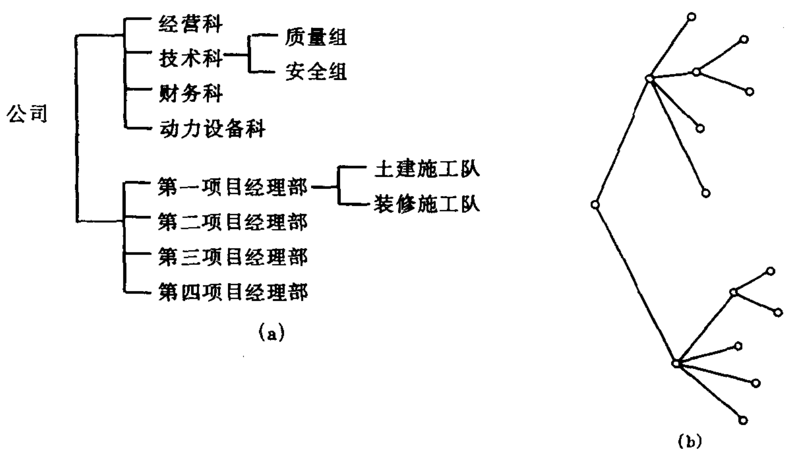


图 1-6 施工企业组织机构图

树具有许多显而易见的性质：

(1) 树中任两顶点间必有一条且仅有一条链。

(2) 在树的两个不相邻的顶点间添上一条边，就得到一个圈。反之，若去掉树中任一边，图就不连通了。

(3) 有 P 个顶点的树有 $P-1$ 条边。

由树的定义，我们可以看出图 1-4 中支撑子图 $\{v_1, v_2, v_3, v_5, v_4\}$ 就是一棵树，这种特殊的树我们称为支撑树。一般地，图 $G=(V,E)$ 的支撑子图 $G'=(V,E')$ 若是树，则图 G' 称作

图 G 的支撑树。

图 G 的性质就往往可通过它的支撑树来研究。由于我们知道支撑树首先必须是支撑子图,也就是说支撑树必须是包含图 G 的所有点的树。前面说过对图 1-4 的 G 来说 $G' = \{v_1, v_2, v_5\}$ 是一棵树,但却不是图 G 的支撑子图,所以它不是图 G 的支撑树。而图 1-7(a),(b)都是图 G 的支撑树,这是因为它们都是图 G 的支撑子图且都是树。

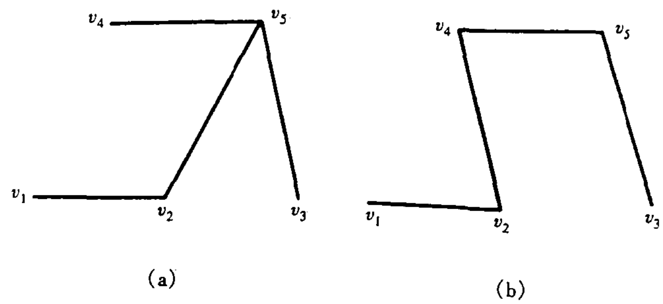


图 1-7　图 G 的支撑树

那么是不是每一个图都有支撑树呢? 图 G 有支撑树的必要且充分条件是 G 是连通图。

这个结论的必要性是显然的。充分性是由于:图 G 是连通的,只要能找出它的一个支撑树即可。我们分作两种情况来考虑,若图 G 无圈,显然它自己就是一个支撑树;若 G 有圈,我们可用“破圈”的方法,先选一个圈,去掉一条或几条边,使剩余图要包含这个圈内的所有点(即保持图的连通性)但无圈,然后陆续用这样的方法使得每个圈都受到破坏,最后只剩下包含所有点但不成圈的图,这就是 G 的一棵支撑树。

求连通图的支撑树的方法有两种:

(1) 破圈法

在一个连通图中,破掉所有的圈,剩下不含圈的连通图就是图的一棵支撑树,这种方法形象地称为“破圈法”。破圈原则:取一个圈,从圈中丢去任一边,对余下的图重复这个步骤,直到无圈为止,即可得到一棵支撑树。

例 1　用破圈法求图 1-8 的支撑树。

解　在图中任取一圈 $\{v_1, v_2, v_3, v_1\}$,从圈中去掉边 e_3;从圈 $\{v_1, v_2, v_4, v_3, v_1\}$ 中,去掉边 e_4;在圈 $\{v_3; v_4; v_5; v_3\}$ 中,去掉边 e_6;在圈 $\{v_1, v_3, v_4, v_5, v_2, v_1\}$ 中去掉 e_8;于是得到 G 的一棵支撑树,如图 1-8 中粗线所示。

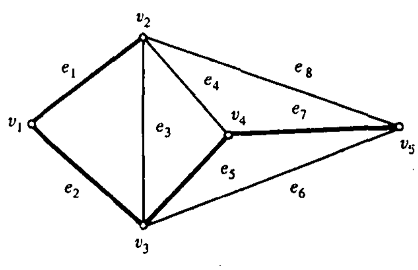


图 1-8　用破圈法求支撑树

从破圈法可看出,支撑树是连结图 G 全部顶点的最少边数的连通图,若再去掉一条边就不再连通了。这就使得支撑树有了它的实际价值,如 G 中的五个点表示煤气用户,现要铺设煤气管道,若按图铺设,就比原图要少铺几条路,同样可达到各户都能用上煤气的目的。

(2) 避圈法

避圈法的思维顺序刚好与破圈法相反。在图中任意取一条边 e_1,找一条不与 e_1 构成圈

的边 e_2,然后再找一条与$\{e_1,e_2\}$不构成圈的边 e_3,继续找下去,直到这个过程不能进行时为止。这时所得到的图就是一棵支撑树。这种方法称为"避圈法"。

避圈原则:每步选取与已选边不构成圈的边,直到不能进行时为止。

例 2 用避圈法求图 1-9 的支撑树。

解 任取 e_1;因为 e_2 与 e_1 不构成圈,所以取 e_2;同理,e_4 与$\{e_1,e_2\}$不构成圈,e_6 与$\{e_1,e_2,e_4\}$不构成圈,e_8 与$\{e_1,e_2,e_4,e_6\}$不构成圈,因此取$\{e_1,e_2,e_4,e_6,e_8\}$五条边,算法结束。得到图 G 的一棵支撑树,如图 1-9 粗线部分所示。

1.1.4 最小支撑树

首先要介绍一个统计概念——权,权是用来表示某种特定指标的数,这里可表示距离、时间、费用等指标的数,边(v_i,v_j)上的权用 W_{ij}表示,即 W_{ij}可表示点 v_i 到 v_j 的距离,也可表示这段工序所用的时间或完成从 v_i 到 v_j 运输任务所付出的费用。图 1-7 的(a),(b)中,若各点间距离知道,那么图 1-7(a)与图 1-7(b)的总长就不同,按上面赋于的实际意义来铺设管道,所用管子多少就不同,自然我们要选取总长最短的方案,这样就可以节省开支。

我们将每边都有权的图称为赋权图。图 1-4 的赋权图若为图 1-10 中所示,它的所有支撑树都有一个总长用 $S_a,S_b,S_c\cdots$表示,如图 1-7 中(a),(b)分别用 S_a,S_b 表示,有

$$S_a=6+2+5+7=20,\ S_b=6+3+5+7=21$$

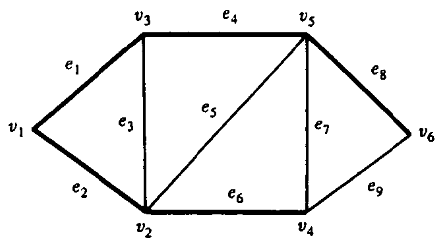


图 1-9 用避圈法求支撑树

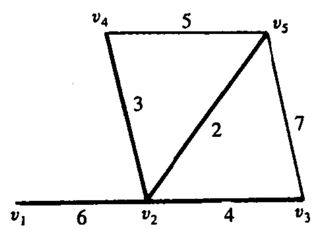


图 1-10 赋权图

但这些支撑树中定有一总长最短者,称其为最小支撑树,简称最小树。图 1-10 的最小支撑树显然应是$\{(v_1,v_2),(v_2,v_3),(v_2,v_4),(v_2,v_5)\}$,如图中粗线部分,记为$(c)$,其 $S_c=6+4+3+2=15$。

读者要问最小树是怎么找到的呢?我们观察一下图 1-10 中粗线部分是将(v_4,v_5)及(v_3,v_5)两条边去掉所得,而(v_3,v_5)是图$\{v_2,v_3,v_5,v_2\}$中权最大的边,(v_4,v_5)是图$\{v_2,v_5,v_4,v_2\}$中权最大的边,即

$$W_{35}=\max\{4,7,2\}=7;$$

$$W_{45}=\max\{2,5,3\}=5.$$

有这样的事实:在每一个圈内去掉的边定大于支撑树所有的边,这一事实仅限于这个圈内,超出这个圈,去掉的边就不一定大于支撑树其他边了,如 $W_{45}\ngtr W_{12}$。所以要想得到图 G 的最小树,利用前面讲过的求支撑树的"破圈法"时,只须注意去掉的边为所在圈内最大权即可。下列两种是常用的求最小树的算法。

算法 1(破圈法)

任取一圈,从圈上去掉一条最大权的边,在余下的图中,重复这个步骤,直到无圈时为止,即可求出最小树。

例 3 在图 1-11(a)中 v_1 为电信局所在地,$v_2,v_3,\cdots,v_{10}$为九个需用电话用户所在地,它们间距离如图上数字所示,请您为电信局安排一个用线最省的架线方案。

解 此题实为求由 v_1 出发的最小支撑树问题,用"破圈法"解法如下:

在图 1-11(a)中,将圈$\{v_1,v_2,v_6,v_1\}$中去掉 $W_{26}=3$;在圈$\{v_6,v_5,v_9,v_{10},v_6\}$中去掉 $W_{56}=5$;在圈$\{v_2,v_3,v_8,v_7,v_2\}$中去掉 $W_{23}=5$,在圈$\{v_3,v_4,v_5,v_3\}$中去掉 $W_{35}=4$;在圈$\{v_7,v_8,v_9,v_{10},v_7\}$中去掉 $W_{98}=3$,得图 1-4(b)。

在图 1-11(b)中,将圈$\{v_1,v_2,v_7,v_{10},v_6,v_1\}$中任选一权为 2 的边,我们去掉 $W_{27}=2$;在圈$\{v_7,v_8,v_3,v_4,v_5,v_9,v_{10},v_7\}$中去掉 $W_{9,10}=3$,得图 1-11(c)。

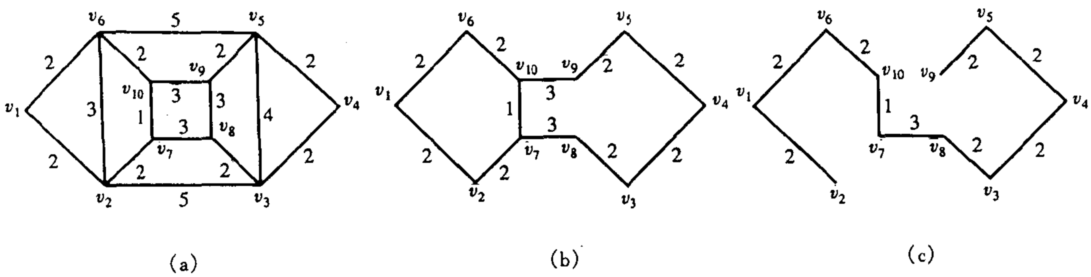


图 1-11 电信局架线方案

此时已无圈,即为所求之最小树,全长为 18。显然,最小树不是唯一的,但总长却都应为 18。

算法 2(Kruskal 算法)

Kruskal 算法的指导思想为保留最小边,用避圈的办法找出其他点到这边中任一端点的最短链来,从而达到去掉最大边的目的。

首先要介绍一下何为最短链?在现实生活中存在这样的问题:在一赋权图 $G=(V,E)$中,对于任意的$(v_i,v_j)\in E$ 有 $W_{ij}\geq 0$, W_{ij}表示距离。若有一点 $v_1\in V$,表示某公司仓库所在地,其他点 $v_j\in V, j=2,\cdots,p$ 表示该公司的直属商店所在地,就存在一个求从仓库到各商店送货所走路程的最短距离问题。两点间的路就可看成一条链,求以这两点为端点的所有链中权数最小的就称为这两点的最短链。如图 1-12 中,v_1 是仓库所在地,该公司有三个商店 v_2,v_3 和 v_4。

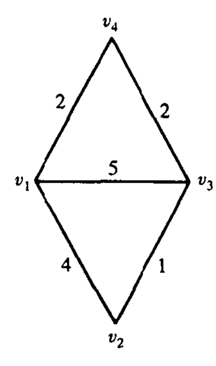


图 1-12 仓库到各商店的最短距离

v_1 到 v_4 最短链(v_1,v_4), $W_{14}=2$;

v_1 到 v_3 有链$\{v_1,v_4,v_3\}$, $W_{14}+W_{43}=2+2=4$;链(v_1,v_3), $W_{13}=5$;链$\{v_1,v_2,v_3\}$, $W_{12}+W_{23}=4+1=5$;故最短链为$\{v_1,v_4,v_3\}$;

v_1 到 v_2 最短链(v_1,v_2), $W_{12}=4$。

显然最短链和最小树是两个不同概念,不应混淆。最短链是局部概念,最小树是全局概念,是针对整个图 G 而言,总权数 S 最小。

Kruskal 算法的具体步骤：

(a) 选图 G 中最小权边的一个端点 v_{i_0}，连结 v_{i_0} 与另一端点 v_{j_0}，使得

$$w_{i_0j_0} = \min\{v_{i_0j},\ (v_{i_0}, v_j) \in E\},\ S_1 = w_{i_0j_0}$$

(b) 找出其他 $p-2$ 个点到 v_{i_0} 或 v_{j_0} 的最短链，不成圈且使 $S_2, \cdots, S_{P-1}$（扩大后树总长）最小。连结这些点即得最小树。

例 4 某工厂沿图 1-13(a)所示的厂区道路架设高压供电线路，将七个车间连接成网，现已知各条道路的长度，试考虑线路架设方案，使总距离最短。

解 显然这是一个最小支撑树问题。有七个车间（即七个点），需经过六步才能选出最小支撑树。

取最小权边一端点 v_7，连 v_2 与 v_7，$W_{27}=1$，$S_1=1$。连 v_6 与 v_7，$W_{67}=2=\min\{W_{67}, W_{61}+W_{17}, W_{65}+W_{57}\}$，$S_2=2+S_1=3$。

连 v_3 与 v_7，$W_{37}=3=\min\{W_{37}, W_{32}+W_{27}, W_{35}+W_{57}\}$，$S_3=3+S_2=6$。

考察 v_5 到 v_7，有两条链，一为 $\{v_7, v_3, v_5\}$，因 S_3 已计算到 v_3，故 $S_4=S_3+W_{35}=6+1=7$；另一为 $\{v_7, v_5\}$，$W_{75}=4$，$S_4'=6+4=10$。显然 $S_4<S_4'$，所以 v_5 到 v_7 最短链取 $\{v_7, v_3, v_5\}$。

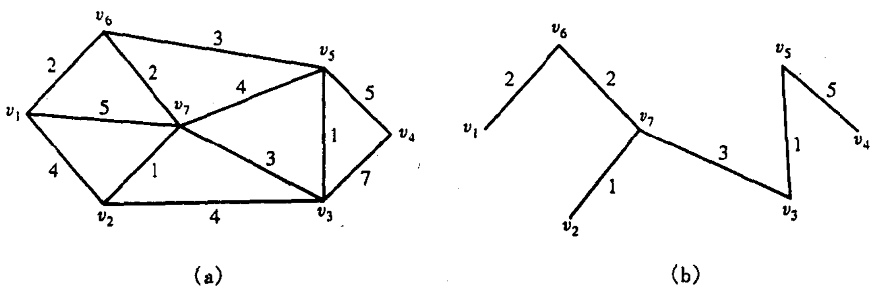


图 1-13 厂区高压供电线路

连结 v_4 与 v_5，$W_{45}=5$，$S_5=5+S_4=12$。

连结 v_1 与 v_6，$W_{16}=2$，$S_6=2+S_5=14$。

即有图 1-13(b)为所求之最小树。

当然若取 v_2 点，找出其他点到 v_2 点的最短链来，亦可求出最小树来。

1.2 最短路程问题

1.2.1 有向图

前一节所讨论的图，是由点 V 和边 E 组成的，没有标明某点到另一点的方向，即 $[v_i, v_j]$ 和 $[v_j, v_i]$ 是相同的。这种图称为无向图。但在实际生活中，很多问题用无向图描述不清楚。例如运输问题中有产地与销地之分，交通网络中的单行道，一项工程中各工序之间的先后关系，竞赛中的胜负关系等等。显然这些关系仅用边是反映不出来的。这时，可以用一条带箭头的线 $v_i \rightarrow v_j$ 反映 v_i 与 v_j 之间的这种关系：例如工序 v_j 必须在工序 v_i 完成之后才能开始；v_i, v_j 两地之间的交通线是从 v_i 到 v_j 的单行道；运动队 v_i 胜了运动队 v_j 等等。这种点与点

之间有方向的线称为弧。

由点集 V 和弧集 A 组成的图 $D=(V,A)$称为有向图。有向图中的弧,例如图 1-14 中的弧 a,记为 $a=(v_i,v_j)$(注意(v_i,v_j)与(v_j,v_i)是不同的)。v_i 称为弧 a 的始点,v_j 称为弧 a 的终点,并称弧 a 是从v_i 指向 v_j 的。

$$V=\{v_1,v_2,v_3,v_4,v_5,v_6,v_7\}$$

$$A=\{(v_1,v_2),(v_1,v_3),(v_3,v_2),(v_3,v_4),(v_2,v_4),(v_4,v_5),(v_4,v_6),(v_5,v_3),(v_6,v_4),(v_5,v_6),(v_6,v_7),(v_7,v_7)\}$$

若弧(v_i,v_j)的始点 v_i 和终点 v_j 相同,即 $v_i=v_j$,则称为环。如图 1-15 中的(v_7,v_7)就是一个环。

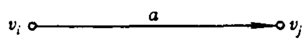


图 1-14 弧

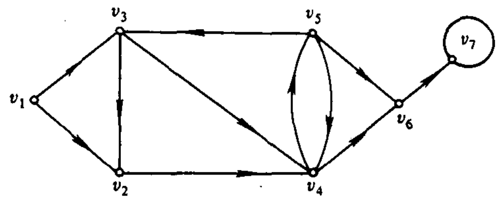


图 1-15 有向图

如果从一个有向图 D 中去掉箭头,得到一个无向图,这个无向图称为 D 的基础图,记之为 $G(D)$。图 1-16 就是图 1-15 的基础图。

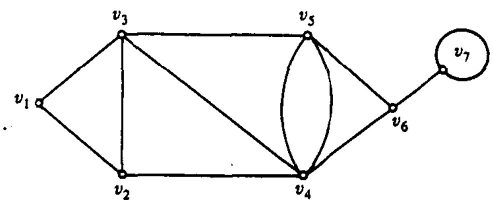


图 1-16 基础图

给出 D 中点弧的交替序列$\{v_{i_1},a_{i_1},v_{i_2},a_{i_2},\cdots,a_{i_{k-1}},v_{i_{k-1}}\}$,如果在基础图 $G(D)$中是一个链,则这个点弧的交替序列就称为 D 的一条链。

类似地定义圈和初等链(圈)。

方向是构成有向图的一个要素。下面给出与方向有关的一些概念。

若$\{v_{i_1},a_{i_1},v_{i_2},a_{i_2},\cdots,a_{i_{k-1}},v_{i_k}\}$是一条链,并且对 $t=1,2,\cdots,k-1$, $a_{i_t}=(v_{i_t}),v_{i_{t+1}}$则称之为从 v_{i_1}到 v_{i_k}的路。图 1-15 中的$\{v_1,(v_1,v_2),v_2,(v_2,v_4),v_4,(v_4,v_6),v_6,(v_6,v_7),v_7\}$就是从 v_1 到 v_7 的一条路。若路的第一个点与最后一个点相同,则称为回路。图 1-15 中的$\{v_3,(v_3,v_2),v_2,(v_2,v_4),v_4,(v_4,v_5),v_5,(v_5,v_3),v_3\}$就是一条回路,而$\{v_1,(v_1,v_3)v_3,(v_5,v_3),v_5,(v_5,v_6),v_6\}$是一条链,但不是路。同样地,以后也以点的序列来记路或回路。

1.2.2 最短路算法

例 1 从油田铺设管道,把原油运到原油加工厂。要求管道必须沿图 1-17 中所给定的道路铺设。设图中的 v_1 点为油田,v_9 为原油加工厂,每条弧旁的数字表示这条道路的长度,要求使管道总长最短的铺设方案。

可见满足条件的铺设方案是很多的,例如沿$\{v_1,v_4,v_7,v_8,v_9\}$或沿$\{v_1,v_2,v_3,v_6,v_9\}$等等。不同的方案,管道长是不同的。比如按第一方案铺设,管道长为 $4+6+4+2=16$ 单位,

按第二方案铺设为 $2+4+4+4=14$ 单位，等等。

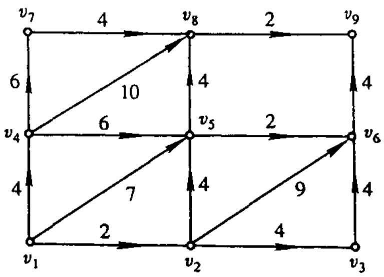


图 1-17 油田管道铺设方案

用图的语言来描述，一个方案对应着一条从 v_1 到 v_9 的路。若定义一条路的长是这条路上各条弧的长之和，那么，上述问题显然是要求一条从 v_1 到 v_9 的路，使路的长度最短。

从这个例子中，可以得出一般情况下最短路问题的叙述：

在有向图 $D=(V,A)$ 中，给定一个始点 v_1 和终点 v_p，对每条弧 $(v_i,v_j)\in A$ 相应地有一个权 w_{ij}（类似地称 D 为赋权有向图）。最短路问题，就是要求从始点 v_1 到终点 v_p 的一条路，使其在所有从 v_1 到 v_p 的路中，它是总权最小的一条。

最短路问题可以直接应用于解决生产实际的很多问题，诸如各种管道铺设、线路安排、厂区布局、设备更新时间安排等等。

目前公认的最短路较好的算法是由 E. W. Dijkstra 在 1959 年首先提出来的，它不仅求出 v_1 到 v_p 的最短路，而且可以得到从始点 v_1 到各点的最短路。

若 $D=(V,A)$，且 $W_{ij}\geqslant 0, v_i, v_j\in V$，不妨假设求 v_1 到 v_j 的最短路，具体计算步骤如下：

(1) 将所有点集分为两类：P 标号点和 T 标号点。

说 v_i 是 P 标号点，是指已找到了 v_1 到 v_i 的最短路，其路长记为 $P(v_i)$，已得到 P 标号的点，不再改变。凡是没有标上 P 标号的点为 T 标号点。开始时，除第一个点 v_1 是 P 标号点，$P(v_1)=0$，其余点都是 T 标号点，$T(v_j)=+\infty$。

要求 v_1 到 v_j 的最短路，只要将 v_1 到 v_j 所有路上所经过 T 标号点变为 P 标号点，就可找到 v_1 到 v_i 的最短路了。所以要想求有向图 D 中 v_1 到所有点的最短路，只须将 $P-1$ 个 T 标号点全部变为 P 标号点，求解终止。若 v_j 不能成为 P 标号点，即 v_1 到 v_j 无最短路可求。

(2) T 标号点变为 P 标号点。

由 $P(v_1)=0$，若与 v_1 相邻的点为 $v_{j1}, v_{j2}, \cdots, v_{jr}\in V$，且有 $T(v_j)=+\infty, j=j_1,\cdots,j_r$。$T$ 标号变为 P 标号实为逐步缩短 v_1 到 v_j 的路长，即由无穷大逐步缩短为 $P(v_j)$，如下取法可达目的：

$$T(v_j)=\min(+\infty, P(v_1)+W_{ij}),\ j=j_1,\cdots,j_r$$

令

$$T(v_{jk})=\min(T(v_{j1}),\cdots,T(v_{jr})),\ 1\leqslant k\leqslant r$$

此时 v_1 到 v_{jk} 的最短路已找到，故 v_{jk} 成为 P 标号点，有 $P(v_{jk})$。

再从 v_{jk} 点开始，重复上述作法，又得到了新的 P 标号点。如此进行下去，经有限次 P 标号，即可得到所求之最短路。这里要注意与 v_{jk} 相邻的点中有可能是与 v_1 相邻的 $v_{j1},\cdots, v_{jr}$ 中的点，而这些点已是缩短了的 T 标号点，即已不是 $+\infty$，而是具体的有限数 $T(v_j)$，所以

$$T(v_j)=\min(T(v_j), P(v_{jk}+W_{jkj})),\ j=j_1,\cdots,j_t$$

例 2 求图 1-18 中从 v_1 到 v_7 的最短路，弧旁数字表示该弧的权。

解 开始给点 v_1 标上 P 标号 0，即 $P(v_1)=0$，表示从 v_1 的最短路权为零。其他点（v_2

至 v_1)标上 T 标号,$T(v_j)=\infty$,$(j=2,3,\cdots,7)$。

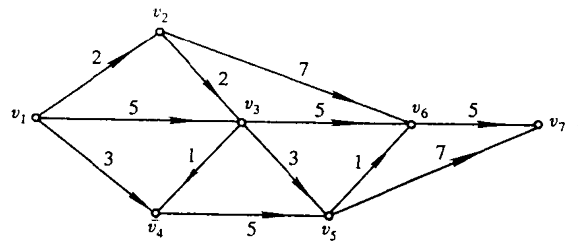


图 1-18　最短路算法

第一步:

(1) 因为 (v_1,v_2),(v_1,v_3),$(v_1,v_4)\in A$,且 v_2,v_3,v_4 是 T 标号点,则修改这三个点的 T 标号分别为

$$T(v_2)=\min[T(v_2),P(v_1)+w_{12}]$$
$$=\min[\infty,0+2]=2$$
$$T(v_3)=\min[T(v_3),P(v_1)+w_{13}]$$
$$=\min[\infty,0+5]=5$$
$$T(v_4)=\min[T(v_4),P(v_1)+w_{14}]$$
$$=\min[\infty,0+3]=3$$

(2) 在所有 T 标号中,$T(v_2)=2$ 最小,于是令 $P(v_2)=2$。

第二步:

(1) v_2 是刚刚得到 P 标号的点,故考察 v_2。因为$(v_2,v_3)(v_2,v_6)\in A$,且 v_3 和 v_6 是 T 标号,故 v_3 和 v_6 新的 T 标号为

$$T(v_3)=\min[T(v_3),P(v_2)+w_{23}]$$
$$=\min[\infty,2+2]=4$$
$$T(v_6)=\min[T(v_6),P(v_2)+w_{26}]$$
$$=\min[\infty,2+7]=9$$

(2) 在 D 的所有 T 标号中,$T(v_4)=3$ 最小,故令 $P(v_4)=3$。

第三步:

(1) 考虑点 v_4,类似上述过程,

$$T(v_5)=\min[T(v_5),P(v_4)+w_{45}]$$
$$=\min[\infty,3+5]=8$$

所有 T 标号中 $T(v_3)=4$ 最小,令 $P(v_3)=4$。

第四步:

考察 v_3,

$$T(v_5)=\min[T(v_5),P(v_3)+w_{35}]$$
$$=\min[8,4+3]=7$$
$$T(v_6)=\min[T(v_6),P(v_3)+w_{36}]$$
$$=\min[9,4+5]=9$$

所有 T 标号中,$T(v_5)=7$ 最小,令 $P(v_5)=7$。

第五步:

考察 v_5,

$$T(v_6)=\min[T(v_6),P(v_5)+w_{56}]$$
$$=\min[9,7+1]=8$$