

Contents

1	通信网结构3图论基础	3
1.1.1	一笔画问题	3
1.1.2	图的基础知识	3
1.1.2.1	图的联结性	5
1.1.2.2	链、径和还	5
1.1.2.3	连接图	5
1.1.2.4	几种特殊的连接图	6
1.2	树	8
1.2.1	基本性质	8
1.2.2	树的一些概念	9
1.2.3	主树	9
1.2.4	树枝与连枝	9
1.2.5	图 G 的阶及其空度	9
1.2.5.1	阶	9
1.2.5.2	空度	9
1.2.5.3	主林与林补	9
1.3	割和环	10
1.3.1	割	10
1.3.1.1	割端与割端集	10
1.3.1.2	割边集和割集	10
1.3.2	联结度、结合度、连通度	10
1.3.3	基本割集	10
1.3.4	基本环和所有环的方法	12
1.4	平面性和对偶性	12
1.4.1	平面型	12
1.4.2	对偶性	12
1.5	图阵	13
1.5.1	关联阵	13
1.5.1.1	全关联阵	13
1.5.1.2	关联阵	14
1.5.2	割阵	15
1.5.3	邻接阵	16
1.6	最短径问题	16
1.6.1	最短主树	16
1.6.1.1	无限制条件	16
1.6.1.2	有限制条件下最短树的求取	18
1.6.2	端间的最短径	18
1.6.2.1	D 算法	18
1.6.2.2	F 算法	18
1.6.3	次短径和可用径	19
1.6.3.1	找与最短径边分离的次短径	19
1.6.3.2	找与最短径端分离的最短径。	19
1.6.4	限制条件下的可用径	19
1.7	网的中心和中点	20
1.7.1	网中心	20
1.7.2	网中点	21

1.8 站址问题	22
1.8.1 单中点问题	22
1.8.2 k 中点问题	24
1.8.3 设站问题	25
2 通信网中的流量优化	26
2.1 一般性问题	26
2.1.1 可行流及其优化问题	26
2.1.1.1 可行流	26
2.1.1.2 问题	26
2.2 最大流问题	26
2.2.1 M 算法	26
2.2.1.1 M 算法的几点推广	27
2.2.2 最佳流问题	27
2.2.2.1 N 算法	27
2.3 线性规划	28
3 通信网的可靠性	29
3.1 可靠性理论概要	29
3.1.1 不可修复系统的可靠度	29
3.1.1.1 理论分析	29
3.1.1.2 寿命分析	30
3.1.2 可修复系统的可靠度	30
3.1.3 复杂系统的分解	32
3.1.3.1 串接系统	32
3.1.3.2 并接系统	33
3.1.4 串并混合	35
3.1.5 桥式电路	35
3.1.6 可靠性设计	37
3.2 通信网的可靠性	38
3.3 通信网的联结性	38
3.4 局间通信和综合可靠度	38
通信网结构	

1 图论基础

1.1 一笔画问题

奇点：与该条边连接的个数有奇数个

偶点：与该条边连接的个数有偶数个

一笔画定理：

- 中间点一定是偶数点
- 最多有两个奇点
- 若由偶点组成的连通图，一定可以一笔画，以任意点为起点一以这个点为终点
- 只有两个奇点的连通图，一定可以一笔画完；画时以一个奇点为起点，另一个奇点为终点。

1.2 图的基础知识

1.2.0.1 1 基础概念

- 相邻点: v_i 与 v_j 互为相邻点 (v_i 和 v_j 是一条边的两个端点)
- 相邻边: 两条边与同一端点相关联, 则这两条边为相邻边

- 度数 (或次数): 与同一端点相关联的边的个数
- 两个端点重合为一点的边成为自环。
- 平行边, 与同一队端点关联的两条边或两条以上的边
- 简单图与复杂图, 没有自环和并行边的图称为简单图, 否则称为复杂图
- 空图: 没有点
- 孤立点: 有点但无边
- 平面图和非平面图: 非平面图画在平面上时, 至少有两条边要相交。平面图则不想交

1.2.0.2 2 图的运算

- 并图

可定义图的运算, 设有两个图 G_a 、 G_b , 则

并图 $G_c = G_a \cup G_b$

G_c 中的端集和边集分别是 G_a 和 G_b 中的端集和边集的并, 即 $V_c = V_a \cup V_b$ 以及 $E_c = E_a \cup E_b$ 。

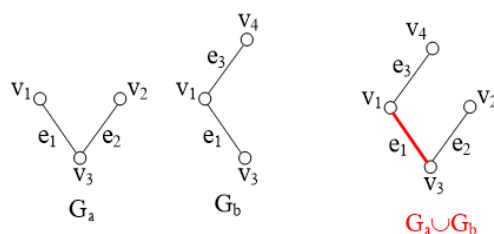


Figure 1:

- 交图

交图 $G_c = G_a \cap G_b$

G_c 中的端集和边集分别是 G_a 和 G_b 中的端集和边集的交, 即 $V_c = V_a \cap V_b$ 以及 $E_c = E_a \cap E_b$ 。

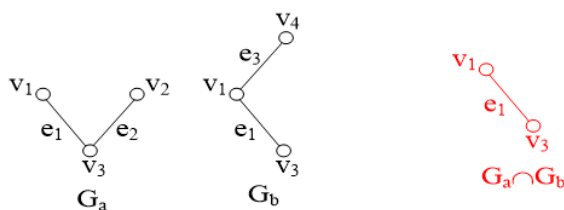


Figure 2:

- 差图

❖ 差图-- $G_c = G_a \sim G_b$

从 G_a 中去掉 G_a 和 G_b 的共有边和共有端，但保留与未去掉的边关联的端。

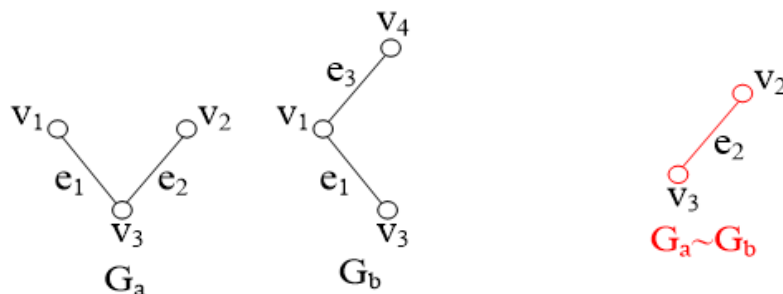


Figure 3:

一般来说: $G_a \cap G_b = G_a - G_a \cap G_b$, 所以该运算要分方向

- 环合图

❖ 环和图-- $G_c = G_a \oplus G_b$

从 $G_a \cup G_b$ 中去掉 G_a 和 G_b 的共有边 e_1 。

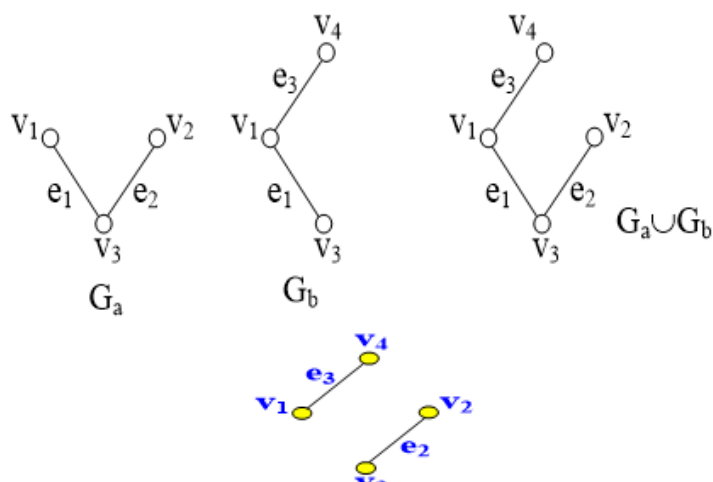


Figure 4:

1.2.1 图的联结性

1. 端的度数 与端相关的变数为该端的度数，自环度数 +2. 在有向图中 $d^+(v_i)$ 表示离开 v_i 的边数， $d^-(v_i)$ 表示进入 v_i 的度数。

图的度数性质: 对于有 n 个端， m 条边的无向图

$$\sum_{i=1}^n d(v_i) = 2m \quad (1)$$

若 G 为有向图

$$\sum_{i=1}^n d^+(v_i) = \sum_{i=1}^n d^-(v_i) = m \quad (2)$$

任意图中，度为奇数数的端的数目必为偶数。

1.2.2 链、径和还

边序列 有限条边的一种串序排列称为边序列，要求相邻两边有公共端。(边可重复)

链 没有重复边的链

环 起点和终点为同一端的链

径 无重复边、无重复端的边序列

1.2.3 连接图

联结图的一般定义：图内任何 2 个端之间至少有一条径，这图就称为联结图（或称连通图）。否则就是非联结图（或非连通图）

非连接图总可以分为几个部分，所谓部分是指原图的一个子图，该子图是一个**最大连接图**

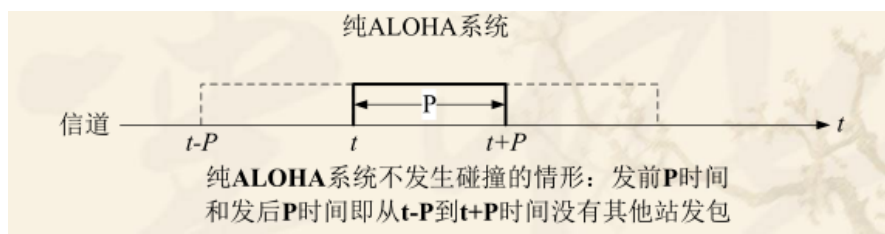


Figure 5:

1.2.4 几种特殊的连接图

1. **全连接图**：任意两端都有边的无向图成为全连接图各端的度数均为 $d(v_i)=n-1$ 。其边 m 和端 n 的关系

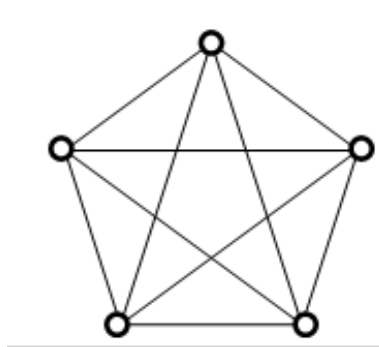


Figure 6:

$$m = C_n^2 = \frac{n(n-1)}{2} \quad (3)$$

2. **两部图**：端点集合可分为 2 个部分，所有边的 2 个邻端分别在这 2 个集合中。特别，**完全两部图** $K_{m,n}$ 的端点集合有 2 个部分，分别有 m 和 n 个端点；从 2 个端集合中各任取一个端，它们之间都有一条边，共有 mn 条边。

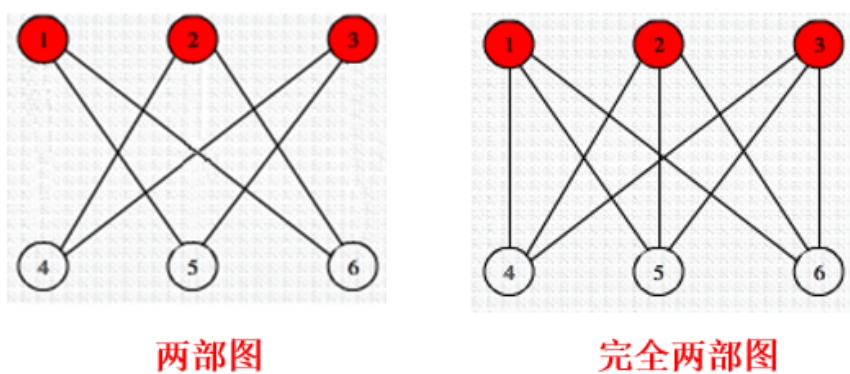


Figure 7:

3. 正则图：所有节点的度数都相等. 正则图的联结性最均匀; 无重边和自环的全联结图是正则图。

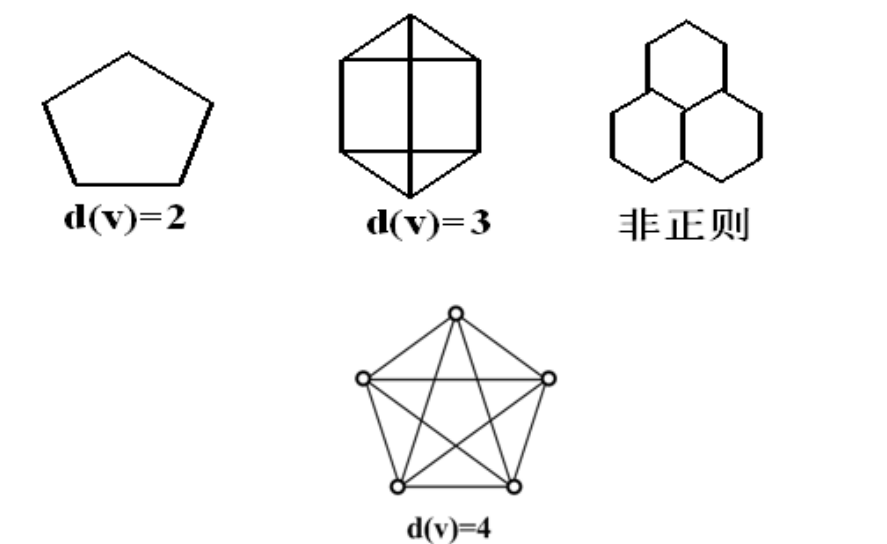


Figure 8:

4. 欧拉图：端度数均为偶数（不一定为连接图）。
连接欧拉图 \Leftrightarrow 存在一个含全边的环

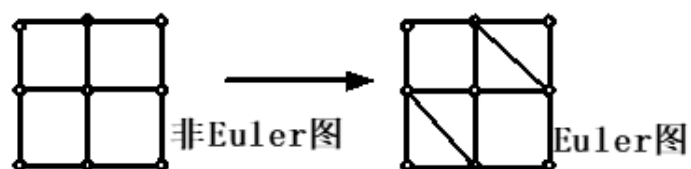
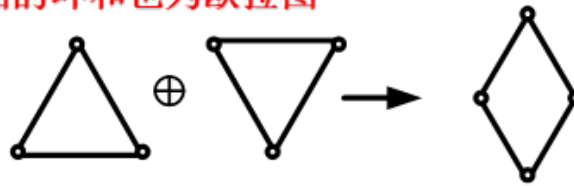


Figure 9:

二欧拉图的环和也为欧拉图



环和:从 $G_a \cup G_b$ 中去掉 G_a 和 G_b 的共有边。

Figure 10:

5. M 图: 如果图中只有 2 个度数为奇数的端, 则此图称为 M 图。
M 图可以是联结图, 也可以是非联结图, 但此时各部分除一个是 M 图外, 其他都是欧拉图。

M图:只有二个奇度数端的图

- 不一定是联结的
- 不存在含有所有边的环
- **联结M图的充要条件:**存在一个含有所有边的开链, 其起点和终点就是度数为奇数的2个端

E图去任何一边---**M图**

M图奇度端点间加一边---**E图**

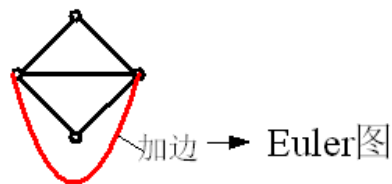


Figure 11:

6. 汉密尔顿 (Hamilton) 图: 当图中至少存在一个含有所有端的环, 这个图称为汉密尔顿图 (也称哈密顿图), 上述的环称为汉密尔顿环。

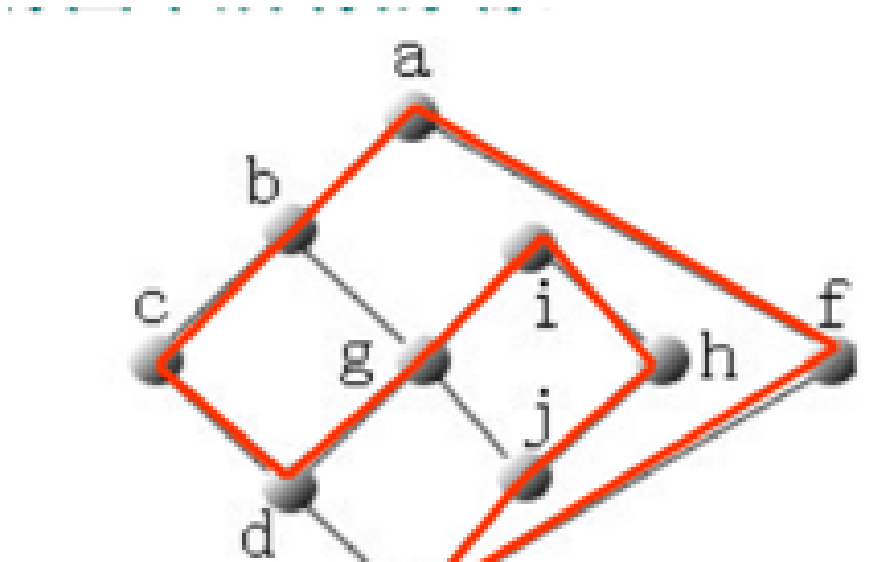


Figure 12:

2 树

定义：任何二端间有径且只有一条径的图

2.1 基本性质

1. 树是无环的连接图
2. 树是最小连通图。去掉任意一边就变成非连接图
3. 若树有 m 条边及 n 个端，则有 $m=n-1$
4. 除单点树外，树至少有 2 个端的度数为 1

2.2 树的一些概念

- ☞ 树枝—树中的边（包括树干和树尖）
- ☞ 树干—两个端点都至少与两条边关联的树枝
- ☞ 树尖、树叶—树枝的一个端点仅与此边关联，此树枝为树尖，该端点为树叶

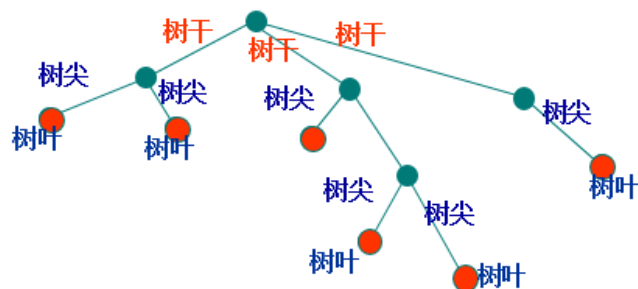


Figure 13:

2.3 主树

生成树：包含所有端点，一定为连通图，可能不止一个

2.4 树枝与连枝

对于图的某一棵主树而言，主树上的边称为树枝，非树枝的边称为连枝。主树就是树枝集；连枝的边集称为连枝集或称为树补。

2.5 图 G 的阶及其空度

2.5.1 阶

联结图 G 的主树 T 的树枝数称为图 G 的阶。若图 G 有 n 个端，则它的阶 ρ 为 $\rho(G)=\rho=|T|=mT=n-1$

2.5.2 空度

联结图 G 的连枝集的连枝数称为图 G 的空度，记为 μ 。当 G 有 m 条边时，有 $\mu(G) = |G - T| = m - n + 1$ 且 $\rho + \mu = m$

- μ 越大，连枝数越多，图 G 的联结性越好。
- $\mu = 0$ 表示最低联结性，即 G 是最小连接图。

2.5.3 主林与林补

对一个非联结图 G，它可分成 k 个部分，也就是 k 个最大联结图。每个部分至少有一棵主树。这可找到 k 棵主树，所形成的集称为主林。余下的边所形成的集称为林补。此时，G 的阶可定义为主林的边数，G 的空度为林补的边数。

$$\rho(G) = (n_1 - 1) + (n_2 - 1) + \dots + (n_k - 1) = n - k \quad (4)$$

$$\mu(G) = m - n + k \quad (5)$$

3 割和环

3.1 割

割是指图的某些子集，去掉这些子集就使图的部分数增加。若图是联结的，去掉这种子集就成为非联结图。根据这种子集的元素不同，可分为割端集和割边集。

3.1.1 割端与割端集

3.1.1.1 1 割端

割端：令 v 是 G 的一个端，在去掉 v 和与之关联的边后，若使 G 的部分数增加，则称 v 是 G 的割端。

3.1.1.2 2 割端集

割端集：去掉几个端后，部分数增加，则这些端的集称为割端集。

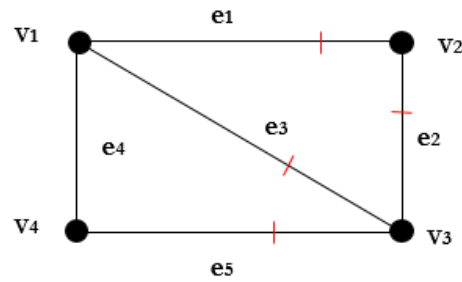
最小割端集中的端数，称为图的联结度，表示要破坏图的联结性的难度；联结度愈大，联结性愈不易被破坏。

3.1.2 割边集和割集

割边集：令 S 是联结图 G 的边子集，如果在 G 中去掉 S 能使 G 成为非联结图，则称 S 是 G 的割边集

割集，最小割边集：若 S 的任何真子集都不是割边集，称 S 是割集。实际上，**割集是最小割边集。**

最小割集的边数称为图的结合度，表示图的联结程度。



右图的割边集是 $\{e1, e2, e3, e5\}$ ，但不是割集，因为其真子集 $\{e1, e2\}$ 是割边集

割集为：

$\{e1, e3, e5\}, \{e1, e2\}, \{e4, e5\}, \{e1, e3, e4\}, \{e2, e3, e5\}, \{e2, e3, e4\}$

Figure 14:

3.2 联结度、结合度、连通度

- 联结度：最小割端集的端数称为图的联结度。
- 结合度：最小割集的边数称为图的结合度。
- 连通度：联结度和结合度统称为连通度。

联结度是点连通度；结合度是边连通度对于通信网来说，连通度越高，可靠性越好

3.3 基本割集

设 T 是联结图 G 的一棵主树，取一条树枝与某些连枝一定能构成一个割集，这种割集称为基本割集。

基本割集只含一条树枝

若 G 有 n 个端，则主树有 $n-1$ 条树枝，所以有 $n-1$ 个基本割集

基本割集有 $n-1$ ，由基本割集及其环和共形成 $2^{(n-1)} - 1$ (二项式求和)，然后排除重复的地方。

基本割集

- ❖ 设 T 是联结图 G 的一棵主树，取一条树枝与某些连枝一定能构成一个割集，这种割集称为基本割集。
- ❖ 基本割集只含一条树枝
- ❖ 若 G 有 n 个端，主树的树枝有 $n-1$ 条，所以 G 有 $n-1$ 个基本割集。

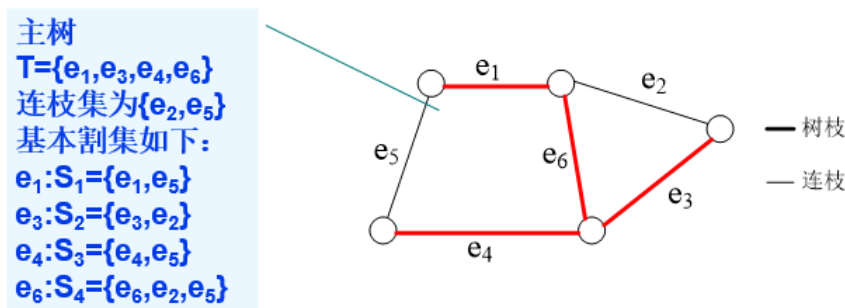


Figure 15:

基本割集共 $n-1$ 个，由基本割集及其环和共形成 $2^{n-1}-1$ 个元：

$$\binom{n}{0} + \binom{n}{1} + \dots + \binom{n}{n} = 2^n$$

$$\binom{n-1}{0} + \binom{n-1}{1} + \dots + \binom{n-1}{n-1} = 2^{n-1}$$

$$\text{所以: } \binom{n-1}{1} + \binom{n-1}{2} + \dots + \binom{n-1}{n-1} = 2^{n-1} - 1$$

$$\begin{aligned} S_5 &= S_1 \oplus S_2 = S_1 \cup S_2 \\ S_6 &= S_1 \oplus S_3 = \{e_1, e_4\} \\ S_7 &= S_1 \oplus S_4 = \{e_1, e_2, e_6\} \\ S_8 &= S_2 \oplus S_3 = S_2 \cup S_3 \\ S_9 &= S_2 \oplus S_4 = \{e_3, e_5, e_6\} \\ S_{10} &= S_3 \oplus S_4 = \{e_2, e_4, e_6\} \end{aligned}$$

$$\begin{aligned} S_{11} &= S_1 \oplus S_2 \oplus S_3 = S_6 \cup S_2 \\ S_{12} &= S_1 \oplus S_2 \oplus S_4 = \{e_1, e_3, e_6\} \\ S_{13} &= S_1 \oplus S_3 \oplus S_4 = S_6 \cup S_4 \\ S_{14} &= S_2 \oplus S_3 \oplus S_4 = \{e_3, e_4, e_6\} \\ S_{15} &= S_1 \oplus S_2 \oplus S_3 \oplus S_4 = S_6 \cup S_9 \end{aligned}$$

Figure 16:

1. C_4^1
2. C_4^2
3. C_4^3
4. C_4^4

再去掉所有基本割集的并或者重复的。

3.4 基本环和所有环的方法

取一条连枝可与某些树枝构成闭径或环。这种仅包含有一条连枝的环称为联结图的基本环。显然，基本环的数目等于连枝数 $m-n+1$ 。基本环的环和可组成 $2^{m-n+1}-1$ 个元，每个元或为环，或为环的并。

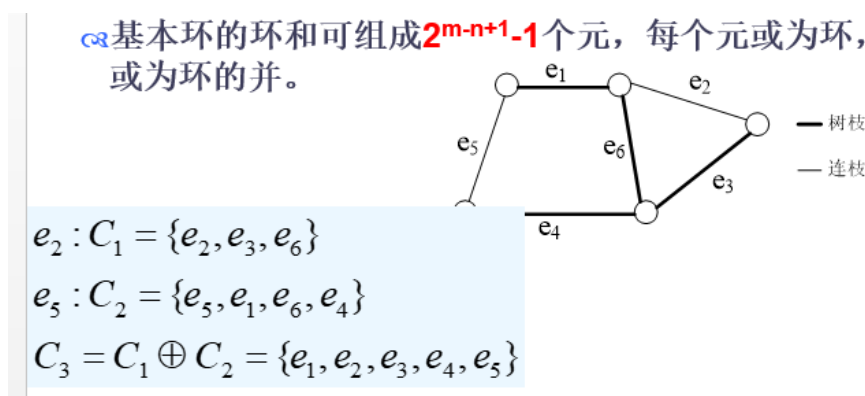


Figure 17:

4 平面性和对偶性

平面图，任意两条边无交点（除端点）。

4.1 平面型

- 设一个联结的平面图有 m 条边， n 个端，把平面分成 S 个区域（包括开区域），它们之间有 $S = m - n + 2$ 。
- 对于无重边、无自环的联结图，具有平面性的必要条件是 $m \leq 3n - 6$ （即平面图必定有 $m \leq 3n - 6$ ，但 $m \leq 3n - 6$ 不一定是平面图）。

4.2 对偶性

设有两个边集 E 相同的图 G_1 和 G_2 ，若 G_2 中每个无重复端的环（闭径）都对应 G_1 中的一个割集，反之亦然，则 G_1 和 G_2 互为对偶图或具有对偶性。

- 平面图的对偶图总是存在的，而非平面图是没有对偶图的。
- 倘若一个图 G 的对偶图就是自己，则称 G 为自对偶图。

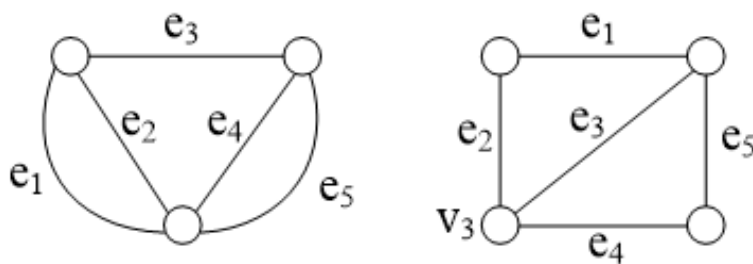


Figure 18:

❖ 对偶图的构造方法：如图，**G1**把平面分成了**3**个区域，在每个区域内各取一点作为对偶图的**3**个端点，把**隔开2个区域的边**用作对偶图中对应**2**个端的联结边

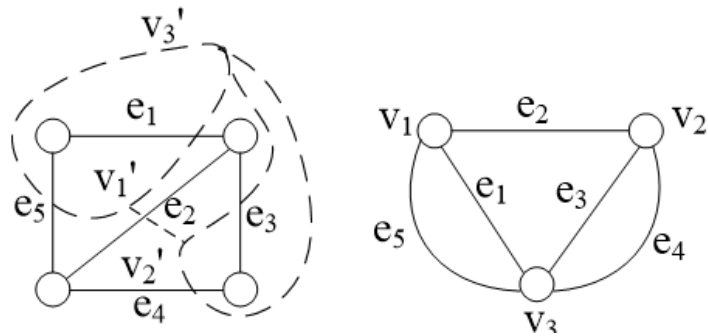


Figure 19:

5 图阵

5.1 关联阵

5.1.1 全关联阵

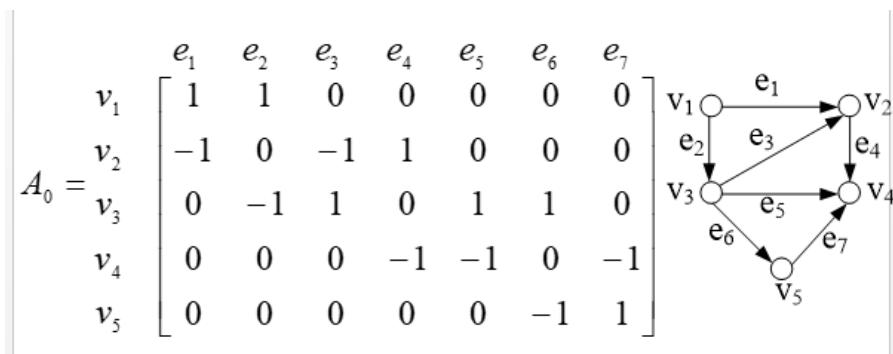


Figure 20:

$$\text{对无向图 } a_{ij} = \begin{cases} 1 & \text{若 } e_j \text{ 与 } v_i \text{ 关联} \\ 0 & \text{若 } e_j \text{ 与 } v_i \text{ 不关联} \end{cases}$$

$$\text{对有向图 } a_{ij} = \begin{cases} 1 & \text{若 } e_j \text{ 是 } v_i \text{ 的射出边} \\ -1 & \text{若 } e_j \text{ 是 } v_i \text{ 的射入边} \\ 0 & \text{若 } e_j \text{ 与 } v_i \text{ 不关联} \end{cases}$$

Figure 21:

- 每行中非零元素的个数等于该端的度数
- 每一列元素之和为对无向图为 2，对有向图为 0

5.1.2 关联阵

去掉全关联阵中的任一行即得关联阵

- 联结图的关联矩阵中，必存在至少一个 $(n-1) \times (n-1)$ 的方阵是非奇异的，这个方阵所对应的边集就是一棵主树。
- 若关联矩阵中有一个 $(n-1) \times (n-1)$ 的方阵是奇异的，即它的行列式值为零，则这方阵所对应的边集中必存在环。
- 当关联矩阵的阶小于 $n-1$ 时，它所对应的图必为非联结图，因为没有 $(n-1) \times (n-1)$ 的方阵是非奇异的，也就是不存在主树。

联结图主树数目 S 与全关联阵的关系

$$S = |AA^T| \quad (6)$$

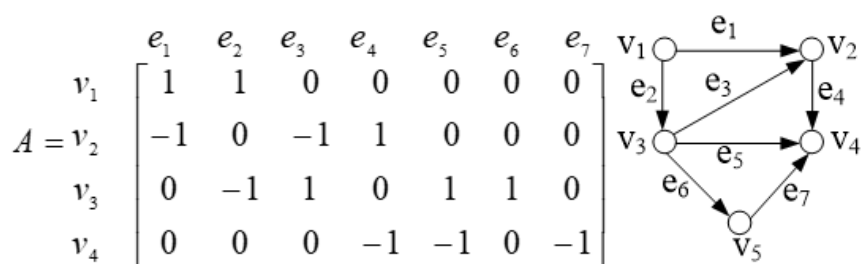


Figure 22:

- AA^T (无重复边) 对角线上的元是端的度数 (AA^T 中没有删的行所对应的端), 其他元非-1 即零 (若两个端之间没有边, 为零, 有边为-1)。
- 对于无向图, 可以在图上任意加箭头, 得相应的有向图, 这样就可以求得无向图的主树数目。

5.2 割阵

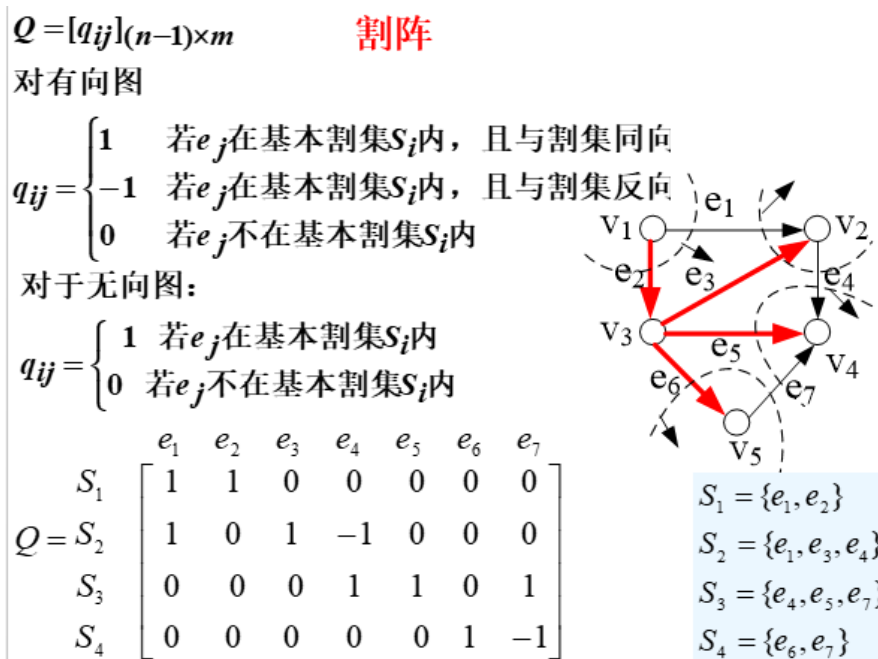


Figure 23:

S_1, S_2, S_3, S_4 为基本割集。正负 1 看树枝所指端，如果边枝和树枝指向相同则为正，反之为负

割阵

❖ 重排 Q 阵中边的次序，将树枝放在前面，可得

$$Q = \begin{matrix} & e_2 & e_3 & e_5 & e_6 & e_1 & e_4 & e_7 \\ \begin{matrix} S_1 \\ S_2 \\ S_3 \\ S_4 \end{matrix} & \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} & \begin{bmatrix} 1 & 0 & 0 \\ 1 & -1 & 0 \\ 0 & 1 & 1 \\ 0 & 0 & -1 \end{bmatrix} \end{matrix} = [I Q_t]$$

❖ I 是 $(n-1) \times (n-1)$ 单位矩阵

❖ Q_t 是 $(n-1) \times (m-n+1)$ 的矩阵

❖ Q_t 对应于连枝集的边。

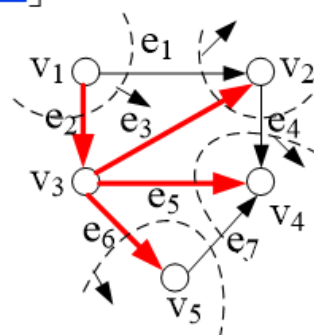


Figure 24:

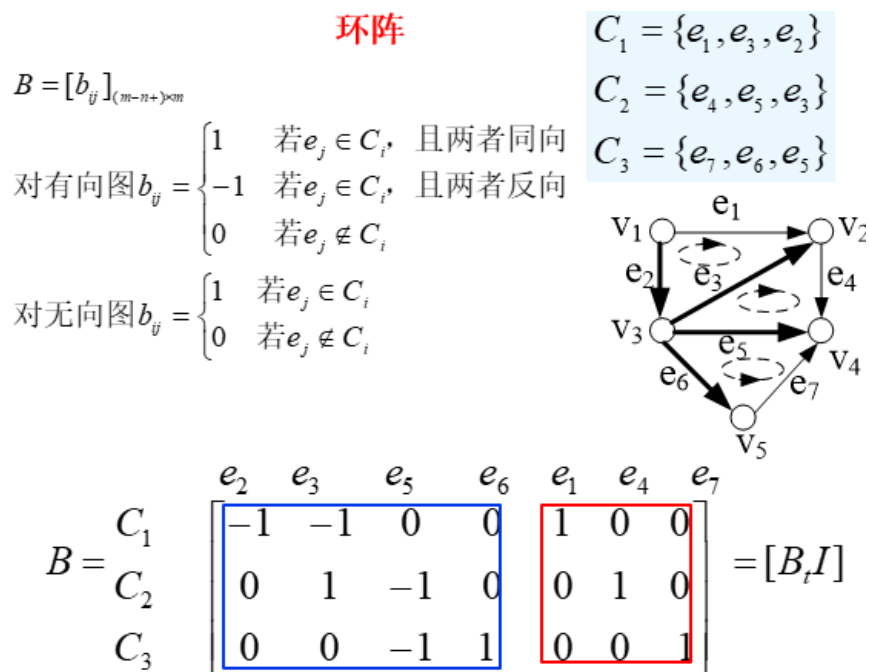


Figure 25:

联结图的环阵 B 和割阵 Q 以下关系, 若边的次序是一样的: $BQ^T = 0$

5.3 邻接阵

示例

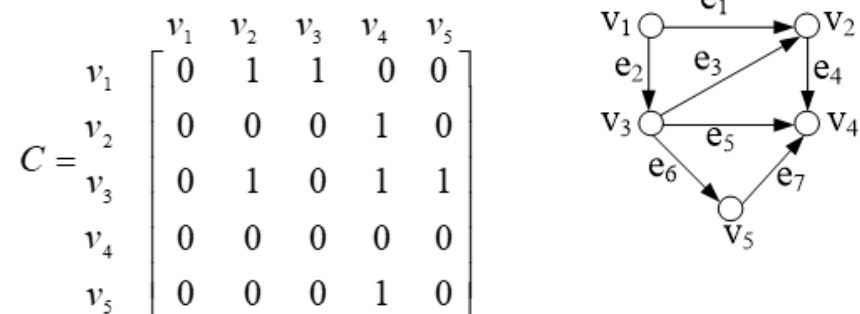


Figure 26:

6 最短径问题

6.1 最短主树

6.1.1 无限制条件

6.1.1.1 1 P 算法

示例

❖ 依算法可顺序得:

❖ $G_1 = \{v_1\}$

❖ $G_2 = \{v_1, v_2\}$

d_{12} 最小

❖ $G_3 = \{v_1, v_2, v_5\}$

d_{25} 最小

❖ $G_4 = \{v_1, v_2, v_5, v_4\}$

d_{54} 最小

❖ $G_5 = \{v_1, v_2, v_5, v_4, v_3\}$

d_{23} 最小

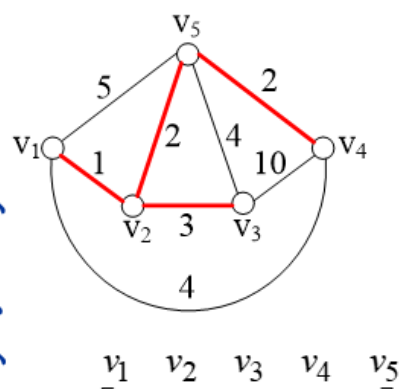


Figure 27:

6.1.1.2 2 K 算法

【方法例题1】

示例:

❖ $G_1 = \{v_1, v_2\}$

❖ $G_2 = G_1 \cup \{v_2, v_5\} = \{v_1, v_2, v_5\}$

❖ $G_3 = G_2 \cup \{v_5, v_4\} = \{v_1, v_2, v_5, v_4\}$

❖ $G_4 = G_3 \cup \{v_2, v_3\} = \{v_1, v_2, v_5, v_4, v_3\}$

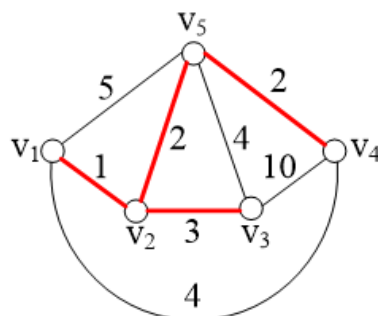


Figure 28:

6.1.1.3 3 破圈法

无限制条件下最短主树求取法

(3) 破圈法

破圈法：从联结图中寻找圈，然后在圈中删去权值最大边，最后剩下的无圈联结图为最短主树。其步骤如下：

- ❖ **Q₀：**设 $G^{(k)}$ 是 G 的联结子图， $G^{(0)}=G$ ；
- ❖ **Q₁：**若 $G^{(k)}$ 中包含圈，设 μ 是 $G^{(k)}$ 中的一个圈，取 μ 上的一条权最大的边 $e^{(k)}$ ，令 $G^{(k+1)}=G^{(k)}-e^{(k)}$ ；
- ❖ **Q₂：**若 $G^{(k)}$ 中不再含圈（即上述 **Q1** 步中的 $G^{(k+1)}$ 不再含圈），则它是最短主树，算法终止；否则，重复 **Q₁**。

Figure 29:

6.1.2 有限制条件下最短树的求取

6.1.2.1 1 穷举法（可用置换法）

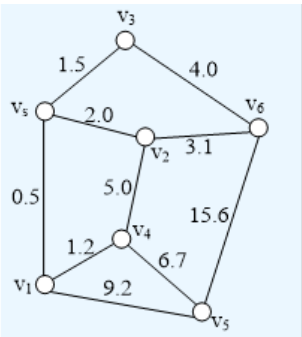
6.1.2.2 2 E-W 算法

6.2 端间的最短径

6.2.1 D 算法

示例

v_s	v_1	v_2	v_3	v_4	v_5	v_6	置定	最短径长
0	∞	∞	∞	∞	∞	∞	v_s	$w_s=0$
	0.5	2	1.5	∞	∞	∞	v_1	$w_1=0.5$
		2	1.5	1.7	9.7	∞	v_3	$w_3=1.5$
		2		1.7	9.7	5.5	v_4	$w_4=1.7$
		2			8.4	5.5	v_2	$w_2=2$
					8.4	5.1	v_6	$w_6=5.1$
					8.4		v_5	$w_5=8.4$



某点最小值由哪一个节点的最小值计算得到，则该点与那一点相连，如 v_5 的最小值 8.4 是由 v_4 的最小值 1.7 加上 6.7 得到，所以，在最短树中， v_5 与 v_4 相连。

- ❖ **最短径：** v_s-v_1 ; v_s-v_2 ; v_s-v_3 ; $v_s-v_1-v_4$; $v_s-v_1-v_4-v_5$; $v_s-v_2-v_6$

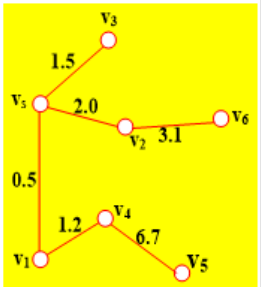


Figure 30:

6.2.2 F 算法

十字交叉法，加上一个中转矩阵（每次更新时，更新中转矩阵即可）

6.3 次短径和可用径

❖ P_1 与 P_2 没有公共边，但有公共端，称 P_1 与 P_2 为**不共边径或边分离径**。

❖ P_1 与 P_3 除了起点外，没有公共端，称其为**不共端径或端分离径**。

☞ 端分离径必为边分离径，反之则不一定。

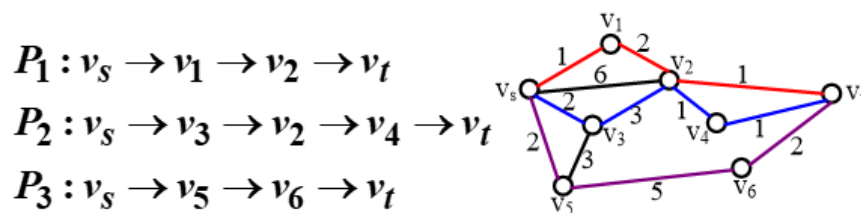


Figure 31:

6.3.1 找与最短径边分离的次短径

当用 F 算法或 D 算法得到最短径后，从原图中去掉此径的所有边（保留这边的两个端），然后在剩下的图中用 D 算法求 v_s 和 v_t 间的最短径。这就是所要求的次短径。这方法还可继续下去。

6.3.2 找与最短径端分离的最短径。

在这种情况下，求得最短径后，应把径中的所有中间端去掉（同时也去掉与之关联的边），在余下的图中求 v_s 和 v_t 间最短径，就得到与最短径分离的次短径。这方法也可继续下去。

6.4 限制条件下的可用径

1. 用 F 算法求得图的最短径长矩阵 W 和转接矩阵 R
2. DFS 遍历各点，在进行判定。

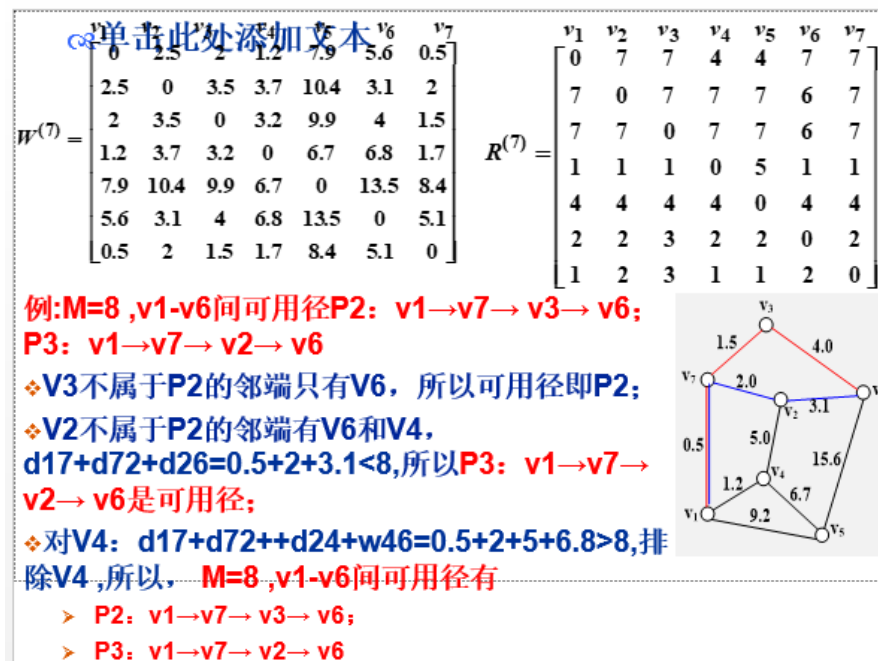


Figure 32:

7 网的中心和中点

7.1 网中心

Steps:

- w_{ij} 有一最大值; 称为最长的最短径. 求每个端点到其它端点的最大长度
- t_i 的最小值所对应的端 v_i^* 称为网的中心. 求上述长度的最小值, 所对应的点设置为中心。

❖ 网络内各端间均存在一最短径; 例如 v_i 端与网内的各端均有一最短径 w_{ij} ;

❖ w_{ij} 有一最大值; 称为最长的最短径, 即: $t_i = \max_j w_{ij}$

❖ t_i 的最小值所对应的端 v_i^* 称为网的中心, 即:

$$t_{i^*} = \min_i t_i$$

❖ 在距离的意义上来说, 若按最短径走, 从网中心到最远端去所走的路程比其他端为短, 所以网中心作为维修中心和服务中心是最有利的。

$$W^{(7)} = \begin{bmatrix} v_1 & v_2 & v_3 & v_4 & v_5 & v_6 & v_7 \\ 0 & 2.5 & 2 & 1.2 & 7.9 & 5.6 & 0.5 \\ 2.5 & 0 & 3.5 & 3.7 & 10.4 & 3.1 & 2 \\ 2 & 3.5 & 0 & 3.2 & 9.9 & 4 & 1.5 \\ 1.2 & 3.7 & 3.2 & 0 & 6.7 & 6.8 & 1.7 \\ 7.9 & 10.4 & 9.9 & 6.7 & 0 & 13.5 & 8.4 \\ 5.6 & 3.1 & 4 & 6.8 & 13.5 & 0 & 5.1 \\ 0.5 & 2 & 1.5 & 1.7 & 8.4 & 5.1 & 0 \end{bmatrix}$$

Figure 33:

7.2 网中点

Steps:

- 按行求和
- 上述的最小值所对应的点

网的中点可用作全网的交换或控制中心

网的中心和中点-网中点

网的中点 v_{i^*} 可定义为平均最短径长最小的端，即

$$s_i = \sum_j w_{ij}$$

❖如果边的权值均取1，网的中点就是平均转接次数最少的端，所以网的中点可用作全网的交换或控制中心。

$$s_{i^*} = \min_i s_i$$

网的直径 D 定义为网内两端间最短径长的最大值，即

$$D = \max_{i,j} w_{ij}$$

$$w^{(7)} = \begin{bmatrix} v_1 & v_2 & v_3 & v_4 & v_5 & v_6 & v_7 \\ 0 & 2.5 & 2 & 1.2 & 7.9 & 5.6 & 0.5 \\ 2.5 & 0 & 3.5 & 3.7 & 10.4 & 3.1 & 2 \\ 2 & 3.5 & 0 & 3.2 & 9.9 & 4 & 1.5 \\ 1.2 & 3.7 & 3.2 & 0 & 6.7 & 6.8 & 1.7 \\ 7.9 & 10.4 & 9.9 & 6.7 & 0 & 13.5 & 8.4 \\ 5.6 & 3.1 & 4 & 6.8 & 13.5 & 0 & 5.1 \\ 0.5 & 2 & 1.5 & 1.7 & 8.4 & 5.1 & 0 \end{bmatrix}$$

Figure 34:

例题：

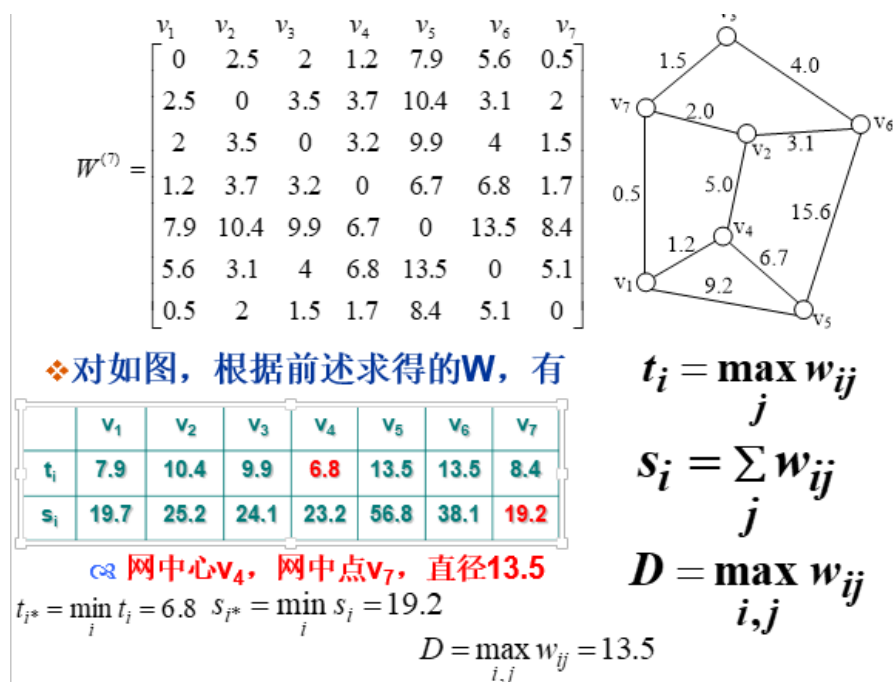


Figure 35:

8 站址问题

8.1 单中点问题

设有 n 个用户点，它们的平面坐标分别为 (x_i, y_i) ($i=1, 2, \dots, n$)。又设各点的加权系数为 w_i ，代表用户所需的联线数或其它需求的大小。单中点问题就是要找到一个中点的坐标 (x_q, y_q) ，使代价 L 最小。

$$L = \sum w_i d_i \quad (7)$$

d_i 为距离的测度

1. 欧式距离，平方和求根
2. 距离平方，欧式距离的平方。适合电磁场传播
3. 矩形距离，直角边的距离。适合城市街道铺设

欧式距离或者平方下，中心站点的最小距离几何确定法：

可以证明，中点的位置，是距离三个点位置最近的中心点，按照初等几何的原理，应该是该点与三个点的连线形成的夹角，均为 120° 。

(1) 单中点问题-图解法求中点

- 方法：以 v_1 、 v_2 为边，作正三角形 $v_1v_2v_0$ ；
- 作正三角形 $v_0v_1v_2$ 及其外接圆。联结 v_0 和 v_3 ，与圆相交于 Q ；此 Q 点就是所示的中点。
- 增加此新端 Q 后，形成的树 v_1Q 、 v_2Q 和 v_3Q 为比以前更短的主树，称其为斯顿树。

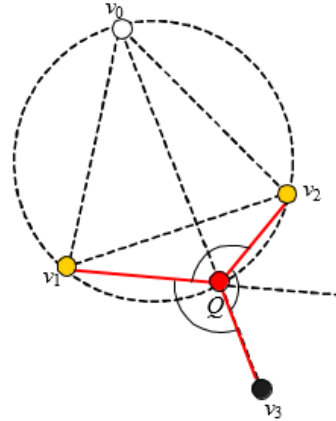


Figure 36:

斯顿树：如容许增加新的端点，包括中心点，存在比以前的最短主树更短的主树。这种主树称为斯顿树。
矩形线距离的情况理论基础：

$$\begin{cases} x_q : \sum_{i: x_q > x_i} w_i = \sum_{i: x_q < x_i} w_i = \frac{1}{2} \sum_{i=1}^n w_i \\ y_q : \sum_{i: y_q > y_i} w_i = \sum_{i: y_q < y_i} w_i = \frac{1}{2} \sum_{i=1}^n w_i \end{cases}$$

Figure 37:

几种情况：

☞ 当 $\sum w_i$ 是偶数，且在上下和左右均可分割成相等的两部分时，可用上述式来求中点。这时解并不唯一。

❖ 如图示，中点可为长方形ABCD内的任一点。

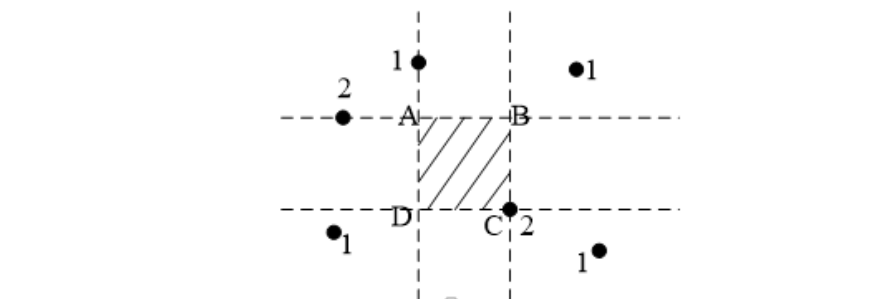


Figure 38:

☞ 若 $\sum w_i$ 是奇数，或虽为偶数但不能均分，则 x_q 必与某些 x_i 相同， y_q 也是这样。

❖ 有时，中点可在一条线上的任一点，也可能有一点作为中点。

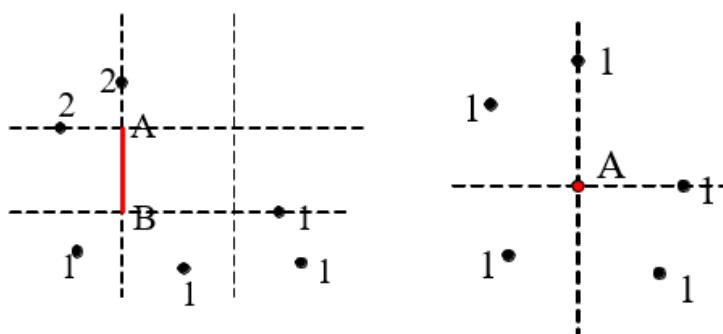


Figure 39:

8.2 k 中点问题

一般所谓的 k 中点问题，是指 k 为预给值，不计各中点的中继线代价，求这些中点的位置，以使总代价最小。

$$L = \sum_{i,j} c_{ij} w_j d_j \quad (8)$$

c_{ij} 表示 i, j 是否有连接。类似 **k-means** 算法。先随机确定初始点，之后根据测度进行 k 聚类，更新中点，判定误差。

8.3 设站问题

设站问题的目标函数为

$$L = \sum_{i=1}^k f_i + \sum_{i=1}^k \sum_{j=1}^n c_{ij} w_j d_{ij}$$

- ❖ f_i : 设立第*i*个站的费用;
- ❖ w_j : 第*j*个用户点的权值;
- ❖ d_{ij} : 第*i*个站与第*j*个用户点联线的距离。

$$c_{ij} = \begin{cases} 1 & j \text{ 点属于 } i \text{ 站, 即与 } i \text{ 站有连线} \\ 0 & j \text{ 点不属于 } i \text{ 站} \end{cases}$$

Figure 40:

单，双，三中点分别计算。得到一个关于设站成本 f 的图形。再进一步分析

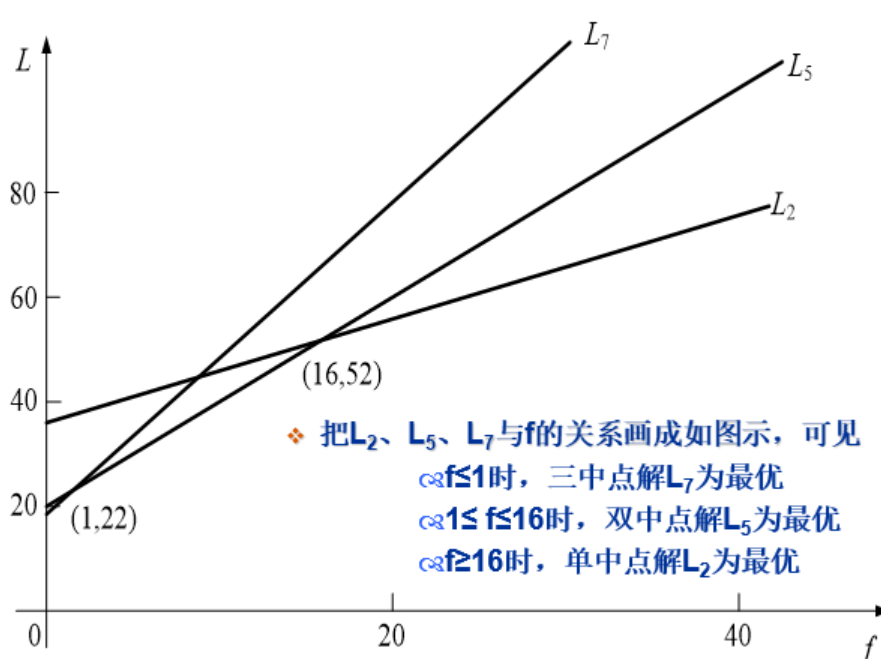


Figure 41:

通信网中的流量优化

9 一般性问题

9.1 可行流及其优化问题

9.1.1 可行流

流量的两个特性

- 非负性和有限性
- 连续性

满足前述限制条件的流称为可行流

9.1.2 问题

- 最大流问题
- 最佳流或最小费用流问题

10 最大流问题

源宿端达到最大流量的充分必要条件, 从 v_s 到 v_t 的每一条边上至少有一个饱和的前向边或一个零流的反向边

10.1 M 算法

采用 DFS 和 BFS 均可做。

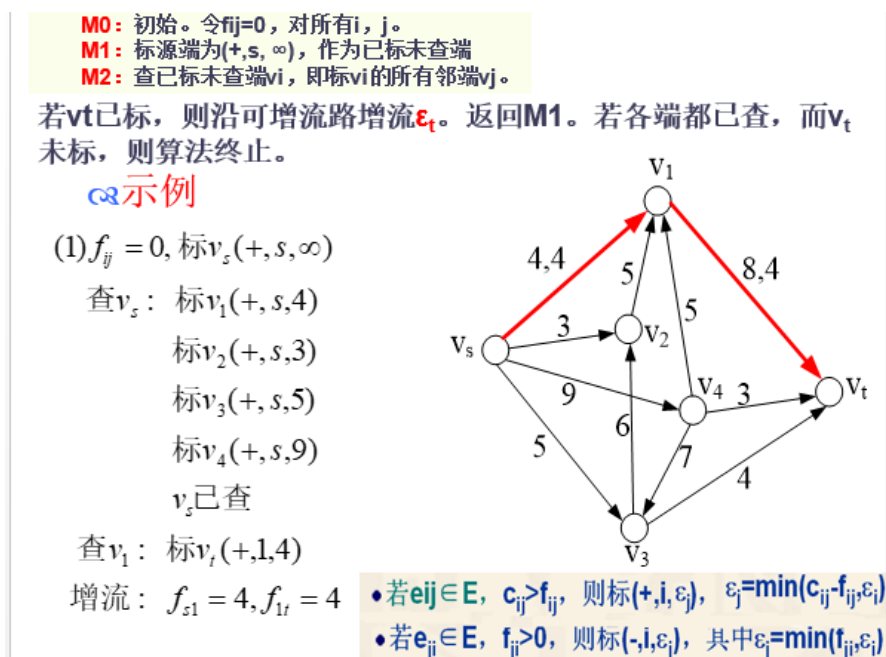


Figure 42:

10.1.1 M 算法的几点推广

- 端容量问题
- 多源多宿情况
- 求结合度 (最小割边集的边数)

10.2 最佳流问题

如果每条边 e_{ij} 各赋予各自的费用系数 a_{ij} , 那么当总流量 F_{st} 相同时, 各种可行流的费用可以不同; 因此, 有时需寻找满足流量要求的最小费用的可行流, 例如传送某一信息流时寻找最小费用的路由, 以达到最佳的流量分配。

每条边具有费用系数, 当总流量固定时总会有费用最小的路由

10.2.1 N 算法

负价环: 补图上若存在一个有向环, 环上各边的费用 a_{ij} 之和是负数, 则称此环为负价环。

如何补图? 元祖结构

正向: 剩容量, a_{ij}

反向: 流量, $-a_{ij}$

步骤:

1. 随便找个满足总流量的流量图
2. 按照补图元祖进行补图
3. 在补图上寻找负价环, 增流值为 $\min\{c_{ij}\}$, 但是注意增流方向, 将这个值增到可行流图上。
4. 重复前两步, 直到找不到负价环

示例

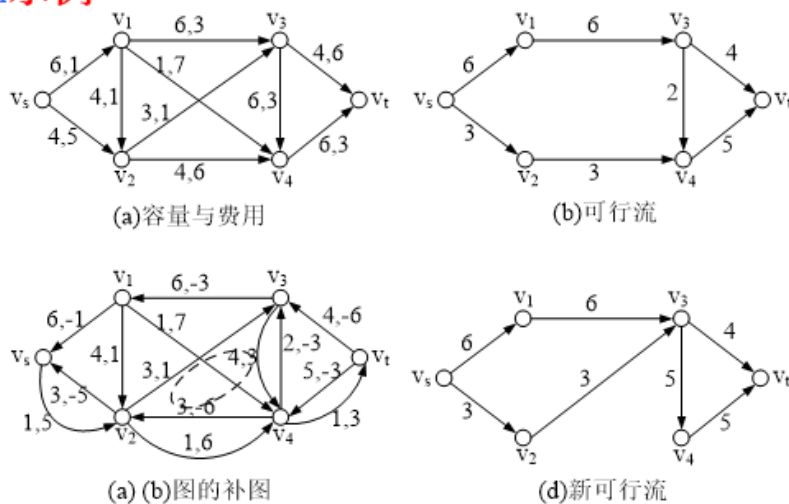


Figure 43:

图的容量和费用图是不会发生改变的, 改变的是可赠流。

11 线性规划

求解线性规划问题就是系统地搜索超平面多面体的顶点, 以达到 f 最大。这种方法通常称为单纯形法。通信网的可靠性

12 可靠性理论概要

12.1 不可修复系统的可靠度

所谓不可修复系统是该系统一旦启用, 直到损坏或失效为止, 一旦失效, 就不会再回到运行状态。这种系统只有两个状态: 一为运行, 另一为失效; 而且只有运行状态向失效状态转移一种可能。

12.1.1 理论分析

定义可靠度:

$$R(t) = P[\text{工作时间} > t] = P[t \text{ 时刻仍正常工作}], R[0] = 1 \quad (9)$$

定义不可靠度:

$$F(t) = 1 - R(t) = P[\text{工作时间} > t] = P[t \text{ 时刻不能正常工作}] \quad (10)$$

故障率, 失效率 (平均) α

$$dt \text{ 失效的概率} = \alpha dt \quad (11)$$

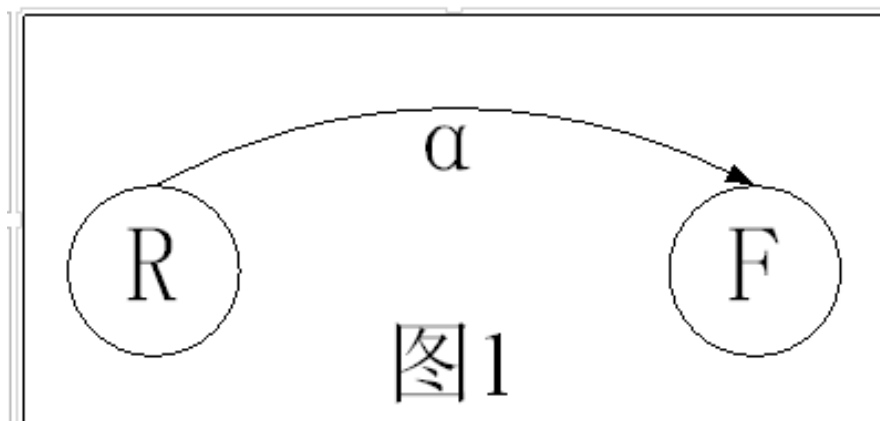


Figure 44:

t 时刻正常, 经 Δt 还没有失效有: $R(t + \Delta t) = R(t) \cdot (1 - \alpha \Delta t)$, 解一个微分方程即可。求得:

$$R(t) = e^{-\alpha t} \quad (12)$$

所以

$$F(t) = 1 - R(t) = 1 - e^{-\alpha t} \quad (13)$$

因此寿命 t 的概率密度函数 $f(t)$ 为:

$$f(t) = F'(t) = -R'(t) = \alpha R(t) \quad (14)$$

平均寿命:

$$T = \int_0^{\infty} R(t) dt = \int_0^{\infty} e^{-\alpha t} dt = \frac{1}{\alpha} \quad (15)$$

❖ 可靠度 $R(t)$ 和失效率 $\alpha(t)$ 的估计

🔗 用实际测量来估计可靠度 $R(t)$ 和失效率 $\alpha(t)$

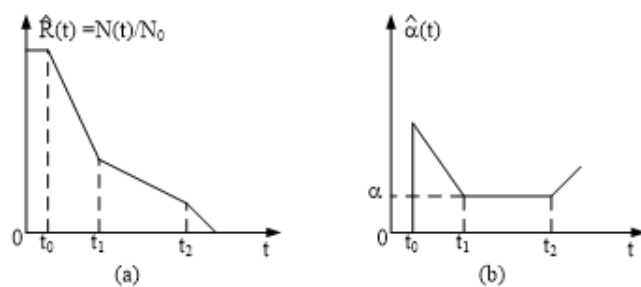
❖ 设有 N_0 个系统同时开动, 随着时间的推移, 有些系统必将失效。到 t 时若尚余 $N(t)$ 个系统运行, 则 $R(t)$ 和 $\alpha(t)$ 的估值如下:

$$\hat{R}(t) = \frac{N(t)}{N_0}$$

$$\hat{\alpha}(t) = -\frac{\hat{R}'(t)}{\hat{R}(t)}$$

Figure 45:

12.1.2 寿命分析



元件可靠度和失效率的实测曲线示意

- ❖ 在 $t=0$ 到 t_0 之间，无元件失效，失效率为0，可靠度为1。
- ❖ 在 t_0 到 t_1 之间，元件失效较多，通常称为多故障期，失效率逐渐下降，可靠度急剧下降。实际上这是试用期或老化期，以便筛选合格的元件。
- ❖ 在 t_1 到 t_2 之间，元件失效较少，失效率接近常量，这可作为正常使用期。
- ❖ 到 t_2 之后，失效元件又较多，失效率上升，是这衰竭期。实际上元件这时已过了正常寿命。

Figure 46:

12.2 可修复系统的可靠度

对于大型设备，不能一出故障就丢弃，而是要把它修复再使用，这类系统称为可修复系统。此时，不但能从正常运行状态转移到失效状态，而且还能从失效状态转移到正常运行状态。前者仍可用失效率 a 来表示转移概率；后者可相仿地定义修复率 b 。

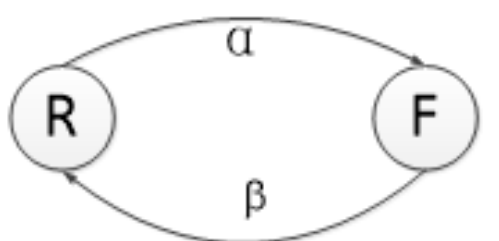


Figure 47:

当在 t 时处于失效状态的情况下，在 t 到 $t + dt$ 内修复的概率为 βdt 。

$$\begin{aligned}
R(t + \Delta t) &= R(t)(1 - \alpha\Delta t) + [1 - R(t)]\beta\Delta t \\
&= R(t) - R(t)\alpha\Delta t + \beta\Delta t - R(t)\beta\Delta t \\
&= R(t) + \beta\Delta t - (\alpha + \beta)R(t)\Delta t
\end{aligned}$$

$$R'(t) = \lim_{\Delta t \rightarrow 0} \frac{R(t + \Delta t) - R(t)}{\Delta t} = \beta - (\alpha + \beta)R(t)$$

Figure 48:

求解上述微分方程:

$$\begin{aligned}
R(0) = 1 \quad R(t) &= \frac{\beta}{\alpha + \beta} + \frac{\alpha}{\alpha + \beta} e^{-(\alpha + \beta)t} \\
R(0) = 0 \quad R(t) &= \frac{\beta}{\alpha + \beta} [1 - e^{-(\alpha + \beta)t}]
\end{aligned}$$

Figure 49:

$t \rightarrow \infty$ 时

$$R(t) = \frac{\beta}{\alpha + \beta} \quad (16)$$

MTBF Mean Time Between Failures

$$MTBF = \frac{1}{\alpha} \quad (17)$$

MTTR Mean Time To Repair

$$MTTR = \frac{1}{\beta} \quad (18)$$

$$R(t) = \frac{MTBF}{MTBF + MTTR} = \frac{\beta}{\alpha + \beta} \quad (19)$$

12.3 复杂系统的分解

12.3.1 串接系统

$$R = \prod_{r=1}^n R_r \quad (20)$$

$$F = 1 - R = 1 - \prod_{r=1}^n R_r \quad (21)$$

如果是 n 个不可修复系统, 则系统可靠度为

$$R = e^{-nat} \quad (22)$$

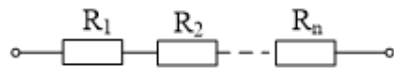
寿命为

$$T = \frac{1}{na} \quad (23)$$

若各子系统的可靠度相差悬殊, 则作为近似计算, 高可靠度的失效率较小, 可以忽略不计。也就是计算总的可靠度或平均寿命, 常只选几个薄弱环节, 亦即几个低可靠度的子系统来计算, 这样可简化。

如果是 n 个独立可修复系统:

串接系统：各子系统都是可修复且相互独立



趋于无穷大时，单系统 $= \frac{\beta}{\alpha + \beta}$

各子系统都是可修复且相互独立

❖ 若每个子系统的失效率和修复率分别为 α_r 和 $\beta_r (r=1, 2, \dots, n)$ ，则

$$R_r = \frac{\beta_r}{\alpha_r + \beta_r}$$

$$R = \prod_{r=1}^n R_r = \prod_{r=1}^n \frac{\beta_r}{\alpha_r + \beta_r} \quad \alpha = \sum_{r=1}^n \alpha_r$$

❖ 系统的平均故障间隔时间为

$$MTBF = 1 / \sum_{r=1}^n \alpha_r = 1 / \sum_{r=1}^n \frac{1}{MTBF_r}$$

Figure 50:

❖ 系统的平均修复时间为

$$MTTR = MTBF \frac{1-R}{R} = \frac{1}{\sum_{r=1}^n \alpha_r} \frac{1 - \prod_{r=1}^n \frac{\beta_r}{\alpha_r + \beta_r}}{\prod_{r=1}^n \frac{\beta_r}{\alpha_r + \beta_r}}$$

Figure 51:

如果是 n 个非独立可修复系统：

❖ 各子系统都是可修复但不相互独立

❖ 加权平均可得全系统的平均修复时间

$$MTTR = \sum_{r=1}^n \frac{1}{\beta_r} p_r = \left(\sum_{r=1}^n \frac{\alpha_r}{\beta_r} \right) / \sum_{r=1}^n \alpha_r$$

❖ 平均故障时间仍为

$$MTBF = 1 / \sum_{r=1}^n \alpha_r$$

❖ 系统可靠度为

$$R_{\text{不}} = \frac{MTBF}{MTBF + MTTR} = \frac{1}{1 + \sum_{r=1}^n \frac{\alpha_r}{\beta_r}} > R = \prod_{r=1}^n \frac{\beta_r}{\alpha_r + \beta_r}$$

各子系统不独立

因为许多子系统停止了不必要的运行

各子系统相互独立

Figure 52:

子系统中有可修复系统和不可修复系统：则整个系统的平均寿命为：

$$R = T_{\text{不可修复}} R_{\text{可修复}} \quad (24)$$

$$T' = \frac{1}{\sum_{r=1}^m \alpha_r} \prod_{i=m+1}^n \frac{\beta_i}{\alpha_i + \beta_i}$$

不可修复系统串接时的平均寿命

各可修复系统串接时的运行率。

Figure 53:

12.3.2 并接系统

$$F = \prod_{r=1}^n F_r = \prod_{r=1}^n (1 - R_r) \quad (25)$$

$$R = 1 - F = 1 - \prod_{r=1}^n (1 - R_r) \quad (26)$$

并接系统

各子系统均为不可修复

全系统的可靠度为

$$R(t) = 1 - \prod_{r=1}^n [1 - R_r(t)] = 1 - \prod_{r=1}^n (1 - e^{-\alpha_r t})$$

$$= \sum_{r=1}^n e^{-\alpha_r t} - \sum_{r_1 \neq r_2} e^{-(\alpha_{r_1} + \alpha_{r_2})t} + \sum_{r_1 \neq r_2 \neq r_3} e^{-(\alpha_{r_1} + \alpha_{r_2} + \alpha_{r_3})t} - \dots$$

平均寿命

$$T = \int_0^{\infty} R(t) dt = \sum_{r=1}^n \frac{1}{\alpha_r} - \sum_{r_1 \neq r_2} \frac{1}{\alpha_{r_1} + \alpha_{r_2}} + \Lambda$$

Figure 54:

$$T = \sum_{r=1}^n \frac{1}{\alpha_r} - \sum_{r_1 \neq r_2} \frac{1}{\alpha_{r_1} + \alpha_{r_2}} + \Lambda$$

并接系统

若各子系统的失效率均为 α ，则有

$$T = \frac{n}{\alpha} - \frac{1}{2\alpha} \binom{n}{2} + \frac{1}{3\alpha} \binom{n}{3} - \Lambda + (-1)^n \frac{1}{n\alpha}$$

n	2	3	4	5	6
αT	1.5	1.83	2.08	2.28	2.45
$\alpha T/n$	0.75	0.62	0.52	0.46	0.41

可以看出，随着子系统的增多，平均寿命并不按正比例增大。

效率 $\alpha T/n$ 是逐渐下降的，这是因为各系统相互独立，从一开始均一直在工作，都在消耗各自的寿命。在实际中，这相当于热备份，一个坏了，另一个可立刻填补，不会造成短时间的中断运行。

Figure 55:

上述过程为热备份。

冷备份（旁置备份）：某个时间段内只开启一个系统，只有当这个系统故障了才开启另一个，这会造成启动时间差。

半热备份，一个子系统在运行，另一个处于半工作状态，如预热。

寿命对比：

$$T_{\text{冷}} > T_{\text{半热}} > T_{\text{热}}$$

列写状态转移时，出为负，进为正。

并接系统

$$R = 1 - F = 1 - \prod_{r=1}^n (1 - R_r)$$

各子系统均为可修复

❖ 全系统失效也就是各子系统在进行修复，它们的修复率分别为 $\beta_r (r=1, 2, \dots, n)$ ，则平均修复时间为

$$MTTR = 1 / \sum_{r=1}^n \beta_r$$

❖ 平均故障时间为

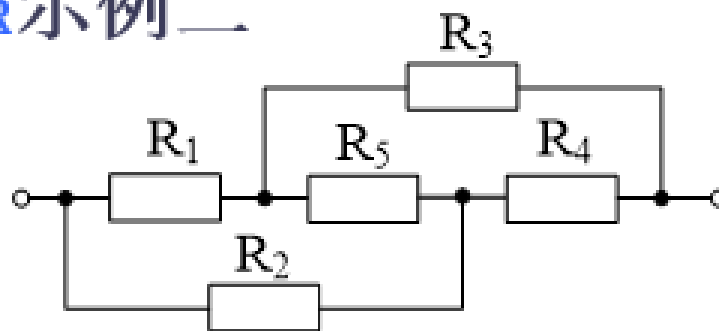
$$MTBF = \frac{1 - \prod_{r=1}^n (1 - R_r)}{\prod_{r=1}^n (1 - R_r)} MTTR = \frac{\prod_{r=1}^n (1 + \beta_r / \alpha_r) - 1}{\sum_{r=1}^n \beta_r}$$

Figure 56:

12.4 串并混合

12.5 桥式电路

示例二



桥式系统

Figure 57:

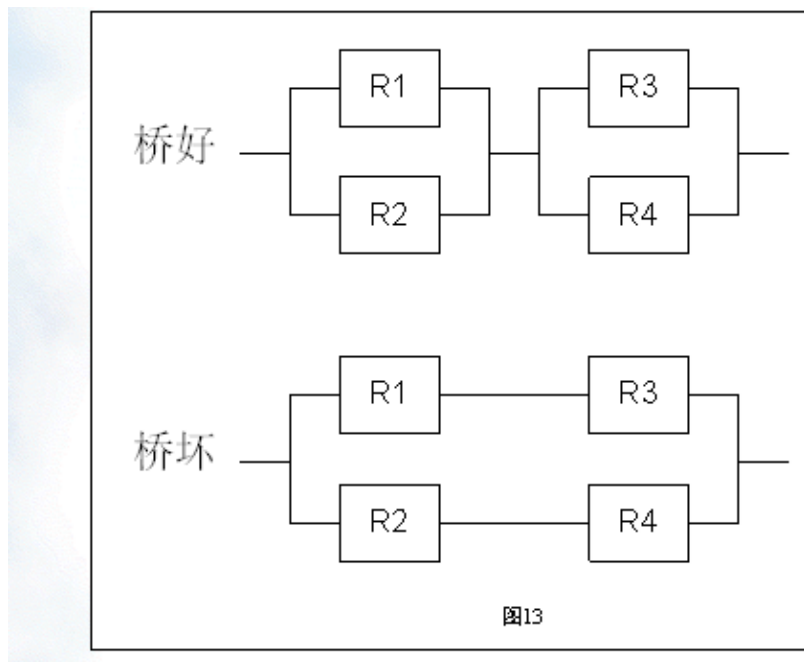


Figure 58:

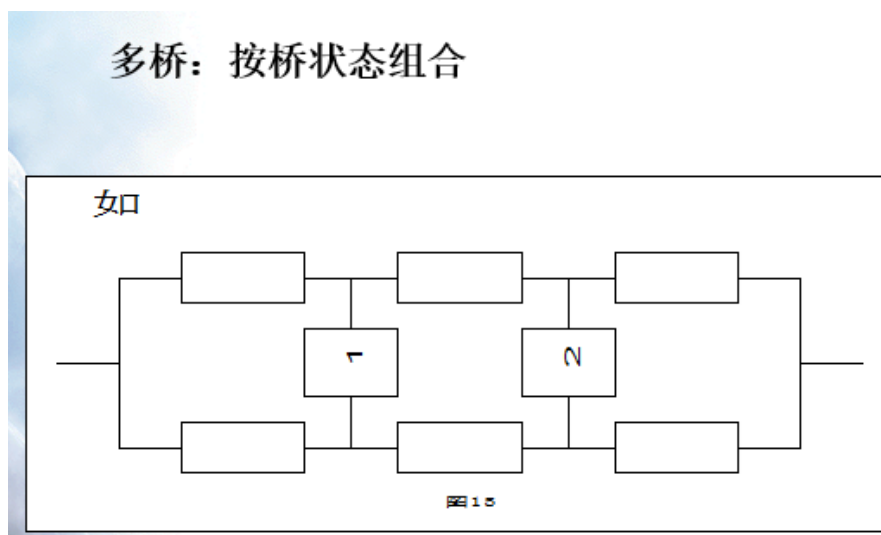


Figure 59:

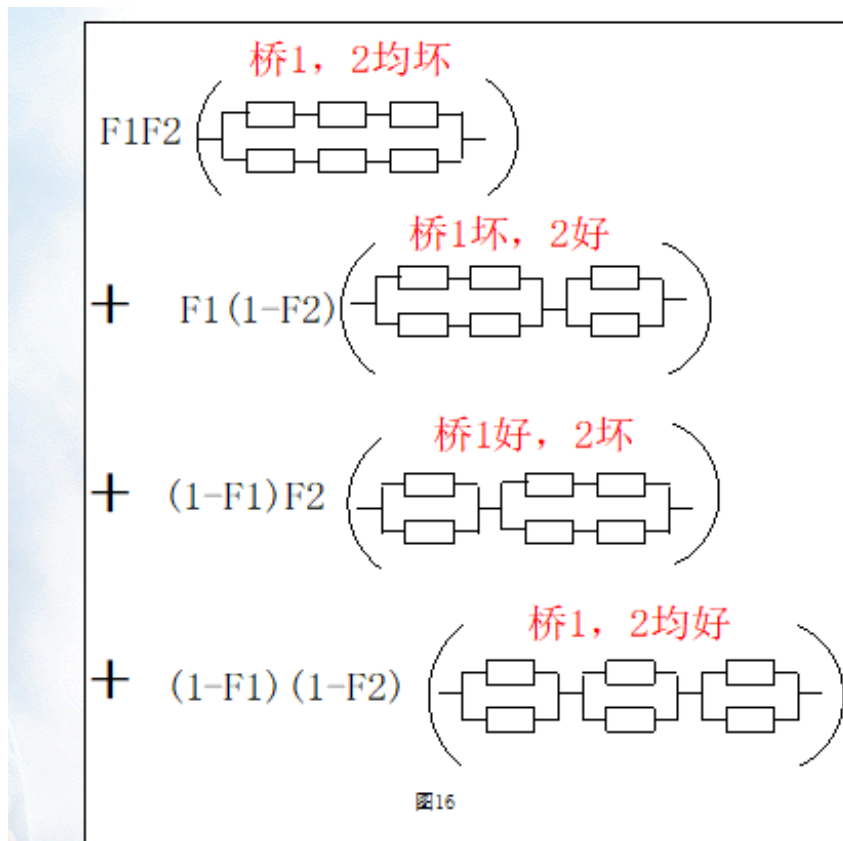


Figure 60:

12.6 可靠性设计

- 避免串接的子系统过多
- 必要时采用备份以形成并接系统
- 尽量减小各子系统或部件的故障率
- 尽量增加修复率

☞全系统的可靠度将为这些 R_i 的函数，即

$$R = f(R_1, R_2, \Lambda, R_n)$$

☞总费用为 $X = \sum_{i=1}^n x_i$

☞最佳设计就是保证 R 达到要求的情况下，以总费用 X 最小的准则来分配各子系统的 R_i 。

❖这样的问题，一般用拉格朗日乘子法求解，即

$$\frac{\partial X}{\partial R_i} + \lambda \frac{\partial R}{\partial R_i} = 0$$

Figure 61:

13 通信网的可靠性

1. 第 1 种定义是从整个网出发，图可能成为不联结的，那么从整体上说，网已失效。
2. 着眼于网内某些端
3. 从随机图出发，设代表网的图并不是确定的，任两端之间有边与否是以概率规定。

14 通信网的联结性

- α 最小割端集的端数
- β 最小割边集的边数，当各端度数相等时，可使 β 等于 $2m/n$ 。正则图
- γ 上述的混合
- $2m/n$, 冗余度

$$\gamma = \alpha \leq \beta \leq 2m/n \quad (27)$$

边数最小的连接图为树，中的任两端之间只有一条径，去掉一条边或一个端，必使图成为不联结的。

15 局间通信和综合可靠度