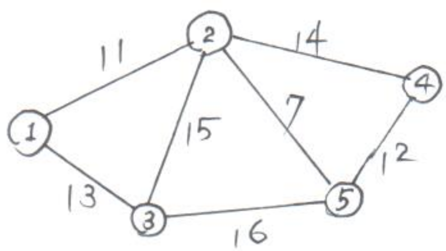
Chapter 6 Network Models 網路模式

§ Scope of Network Application

很多重要的 optimization 問題都可以用圖形或網路代表。

例如，你有五個城市要鋪設地下電纜電線，使得這五個城市連接，而目標是

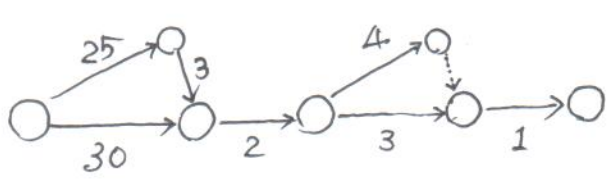


○ 節点 (node)

— 節線 (arc)

使得所有使用的電線最短，你可能有 minimum spanning tree problem 最小鋪設問題

例如，你從台北開車到學校，可能經由高速公路或走新台五線，進入基隆後可能選擇走中正路或祥豐街而你的目的是找尋一條最短路徑，那你有一 shortest path problem 最短路徑問題



Maximal-flow problem 最大流量問題

最大流量(川流)問題，有一單一源點(source)及單一滙點(sink)，這一問題在於透過整個網路找尋單位時間下的最大流量(原油，水，交通量，資訊)。而每一條路線都有它的流量限制，例如油管的口徑。較常見的應用自來水供水量，例如由青潭堰至台北地區，這些已鋪設管路每天最大供水量有多少？

Minimum-cost-capacitated 最小成本模式

例如有油，從油田運到煉油廠，然後再運至配送中心，而原油和汽油產品的運送，可透過油輪、管道與卡車決定如何安排運送並使運送成本最低，就是屬於這問題.

§網路的定義§

A graph or network is defined by two sets of symbols: node and arc.

First, we define a set of point or vertex (vertices). The vertices of a graph or network are called nodes. 以 N (或是 V) 表示。

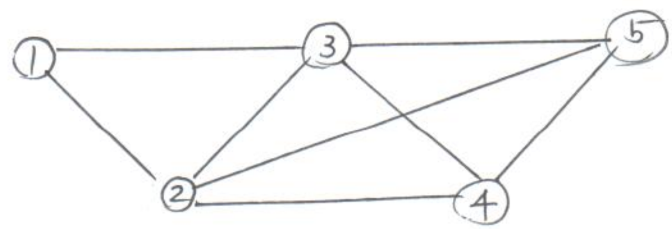
We define a set of arc. (以 A 表示) 或是 E 表示,

Definition: An arc consists of an ordered pair of vertices and represents a possible direction of motion that may occur between vertices.

The standard notation to describe a network G

$$G = (N, A)$$

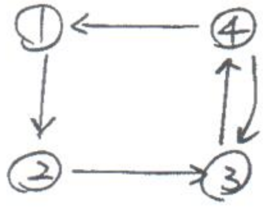
例如

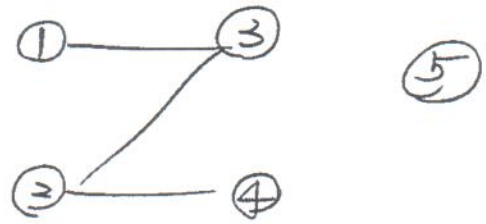


$$N = \{1, 2, 3, 4, 5\}$$

$$A = \{(1,3), (1,2), (2,3), (2,4), (2,5), (3,4), (3,5), (4,5)\}$$

For the arc (i, j) node i is the initial node of the arc, node j is the terminal node of the arc.





Definition : a sequence of arcs such that every arc has exactly one vertex in common with the previous arc is called a chain

Definition : A path is a chain in which the terminal node of each arc is identical to the initial node of the first arc

(1.2) - (2.3) - (4.3) is a chain, but not a path.

(1.2) - (2.3) - (3.4) is a chain, and a path.

The path (1.2) - (2.3) - (3.4) represents a way to travel from node 1 to node 4.

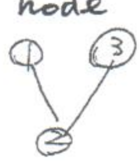
一般而言, 流量 in an arc 是受限於他的容量
可能是有限, 也可能是無限.

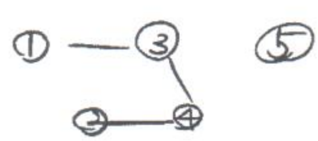
arc 有二种 - 种是 directed (oriented) arc 也就是只
允許單方向的流量 ⟶ ; 另一种是 undirected
arc 也就是允許双方向的流動.

A path will form a loop or a cycle if it connects a node to itself. A directed loop is a loop in which all arcs have the same direction.

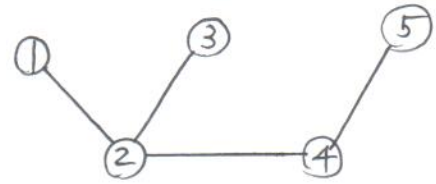
A connected network is a network in which every two distinct nodes are linked by a path.

A tree is a connected network that include only a subset of the node





a spanning tree is a connected network that inclues all nodes in the network with no loops.



§ minimum spanning tree § 最小鋪設問題.
最小展延樹問題

step 1. Begin at any node i, and join node i to the node in the network (call it node j) that is closest to node i. The two nodes i and j now form a connected set of nodes $C = \{i, j\}$ and arc (i, j) will be in the minimum spanning tree. The remaining nodes in the network (call them C') are referred to as the unconnected set of nodes.

step 2. Now choose a member of C' (call it n) that is closest to some nodes in C. Let m represent the node in C that is closest to n. Then the arc (m, n) will be in the MST. Now update C and C'. C now equals $\{i, j, n\}$ and we must eliminate node n from C'.

step 3. Repeat this process until a minimum spanning tree is found. Ties for closest node and arc to be inclued in the MST may be broken arbitrarily.

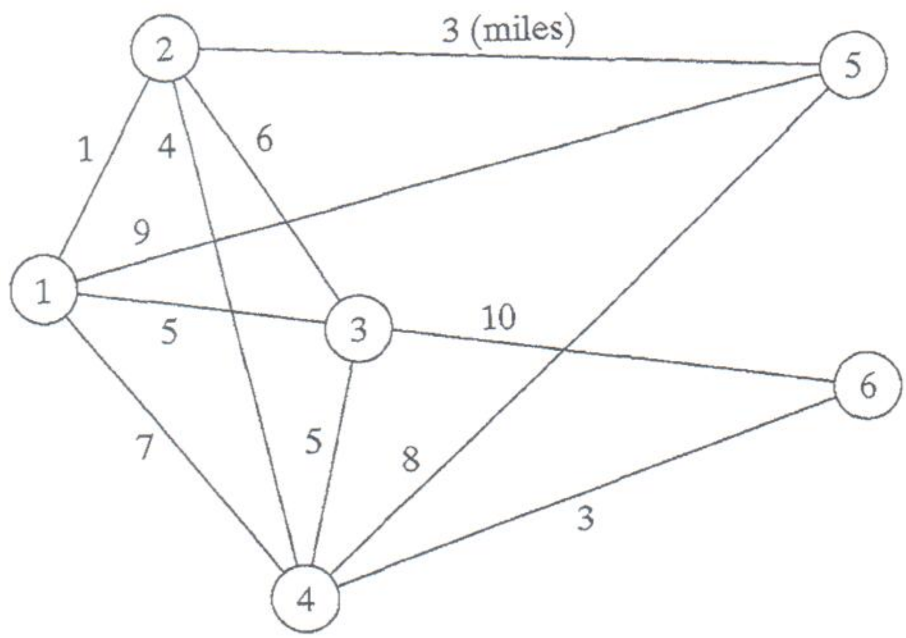


Figure 6.6
Cable connections for Midwest TV Company.

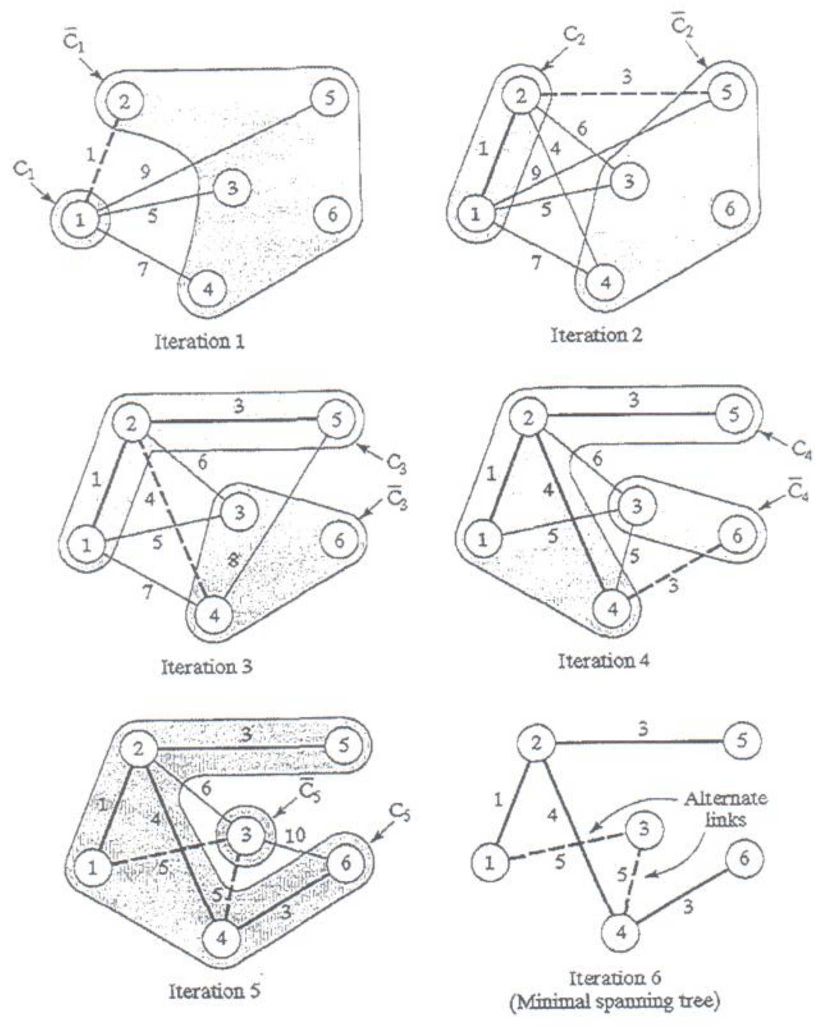
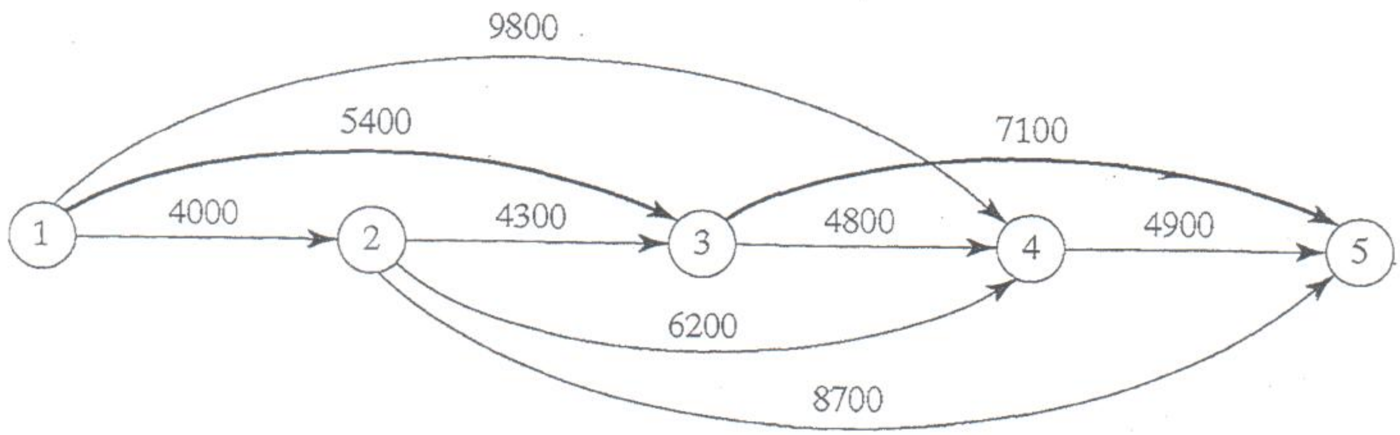


Figure 6.7

6.3-1 設備重置

一租車公司發展一重置計劃，車子必須使用一年以上，方得考慮重置，下表摘要車子的重置成本，成本為營運年數與時間之函數。重置問題可以建構成最短路徑問題

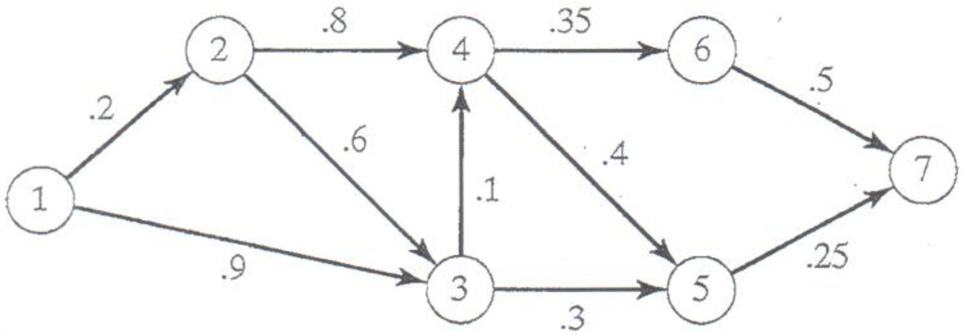


Equipment acquired at start of year	Replacement Cost($) for given years in operation		
	1	2	3
1	4000	5400	9800
2	4300	6200	8700
3	4800	7100	—
4	4900	—	—

使用軟体求解 optimal 1 → 3 → 5（上圖中之粗線）

第1年年初買車，使用2年後，在第3年年初重置。重置的車保持服務直到第4年年底（第5年年初）。總重置成本 $12,500 (5400+7100)

6.3-2 最可靠的路徑，I.Q. Smart 想找最短路徑可是路上有很多警察。經調查 Smart 發現在路上不會被攔下等於構成所有路徑和分配在路段上機率的乘積。



綫段上機率為不會被攔下之機率

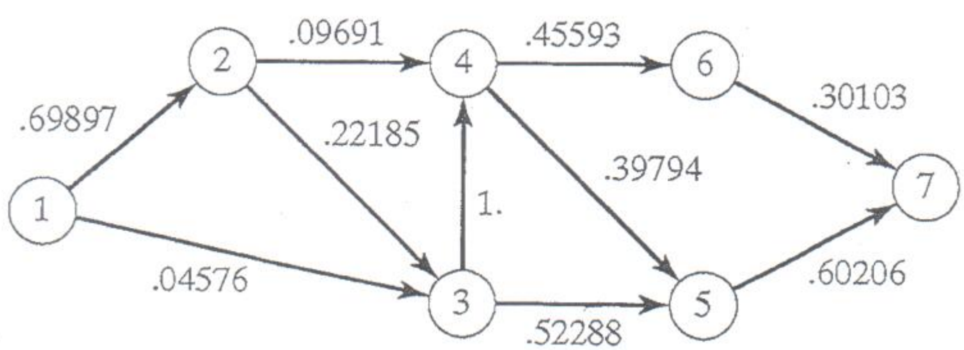
例如 $1 \rightarrow 2 \rightarrow 3 \rightarrow 5 \rightarrow 7$ $\quad 0.2 \times 0.6 \times 0.3 \times 0.25 = 0.009$

$P_{1k} = P_1 \times P_2 \times \cdots \times P_k$ 是特定路綫 $(1, k)$ 上不會被攔之機率 取對數 $\log P_{1k} = \log P_1 + \log P_2 + \cdots + \log P_k$

P_{1k} 最大化 $\cong \log P_{1k}$ 最大化

既然 $\log P_j \leq 0$, $j=1,2,\ldots,k$, maximize

maximize the sum of $\log P_j \cong$ minimize the sum of $(-\log P_j)$



使用軟体 最短路徑 1−3−5−7 長度 1.1707
於是不會被攔下之最大機率為 0.0675
$(0.9 \times 0.3 \times 0.25)$

Example 6.3-3 (Three-Jug Puzzle)

給一裝滿 8 gallon 水的水瓶,與二個空瓶 5 gallon 和 3 gallon,水瓶上面沒有刻度,可在這三個水瓶間倒來倒去,希望得到 2 份相等 4 gallon 的水。令(a, b, c)代表(8 gallon, 5 gallon, 3 gallon)這些瓶中裝的水,則一開始為(8,0,0)

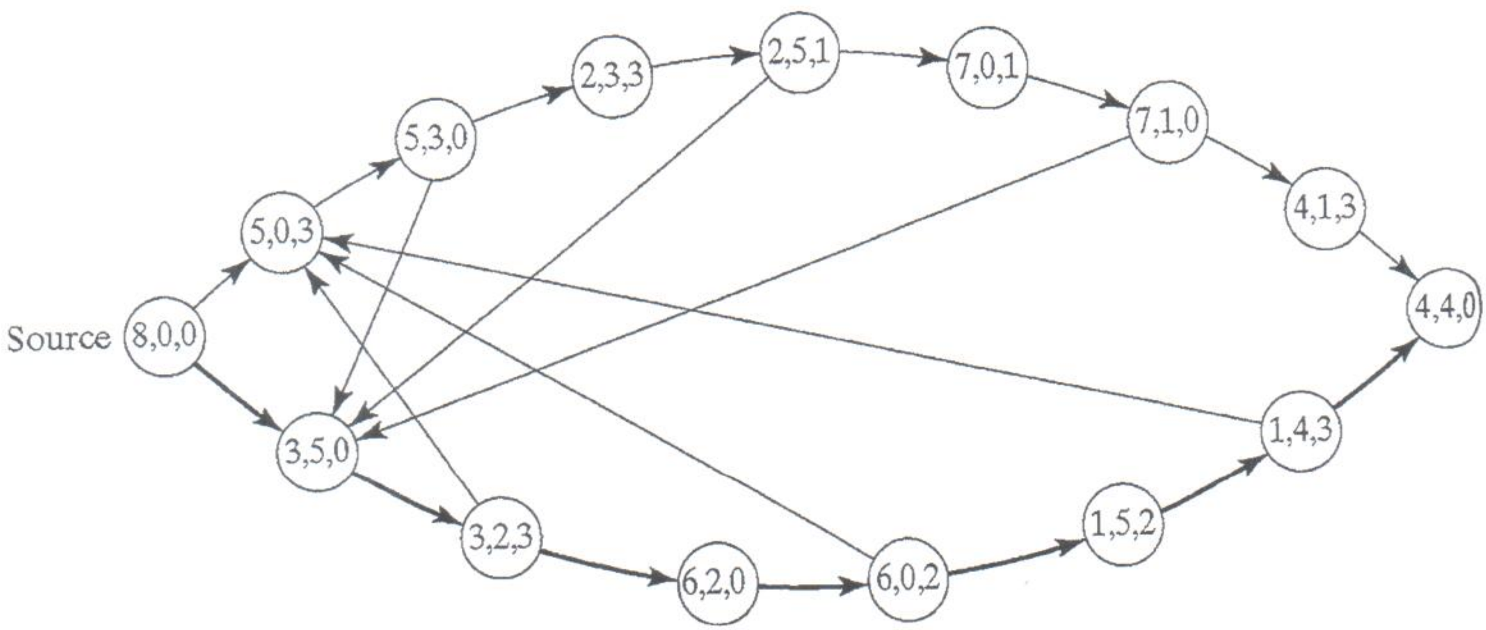


Figure 6.13
Three-jug puzzle representation as a shortest-route model.

最佳解可由上圖中下面粗線路徑代表,需要7次。

6.3.2 Shortest-Route Algorithms

1. Dijkstra's algorithm

2. Floyd's algorithm

Dijkstra's algorithm 用以找 source 與網路其他一點之最短距離. Floyd's algorithm 更一般化可找網路中任意兩點之最短路徑。

Dijkstra's algorithm: d_{ij} 是 node i 與 j 間的距離.

u_i 是由 source node 1 至 node i 的最短距離。

$$[u_j, i] = [u_i + d_{ij}, i], \quad d_{ij} \ge 0$$

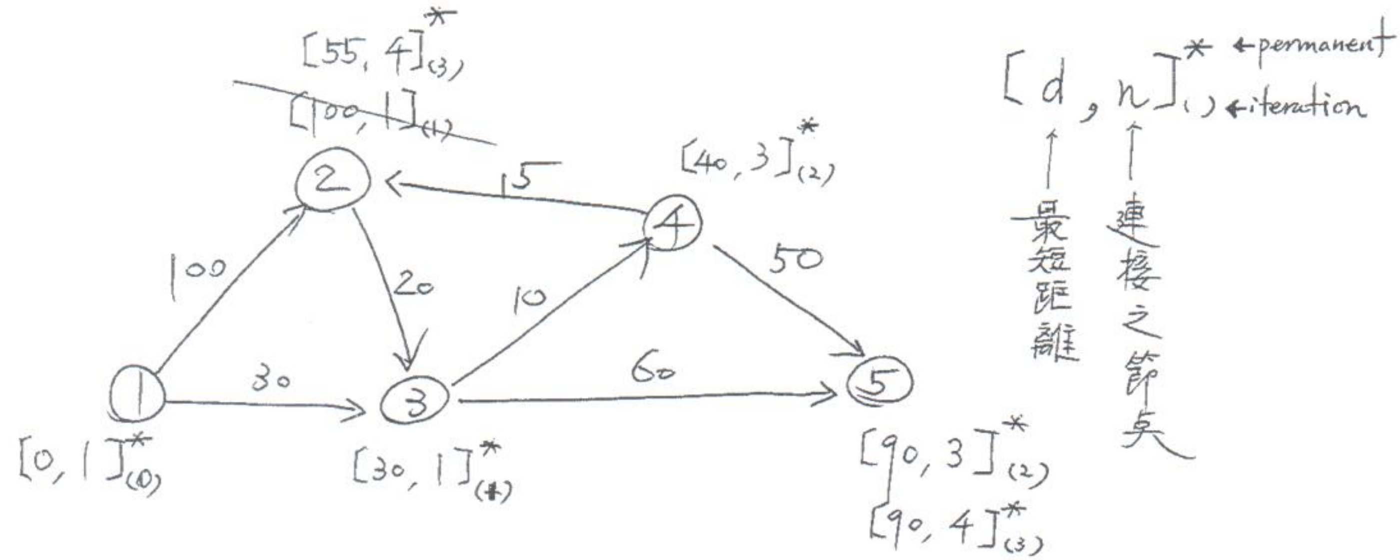
j 為由 i 連接之節點.

有兩种標籤: 臨時 (temporary) 與 永久 (permanent label)

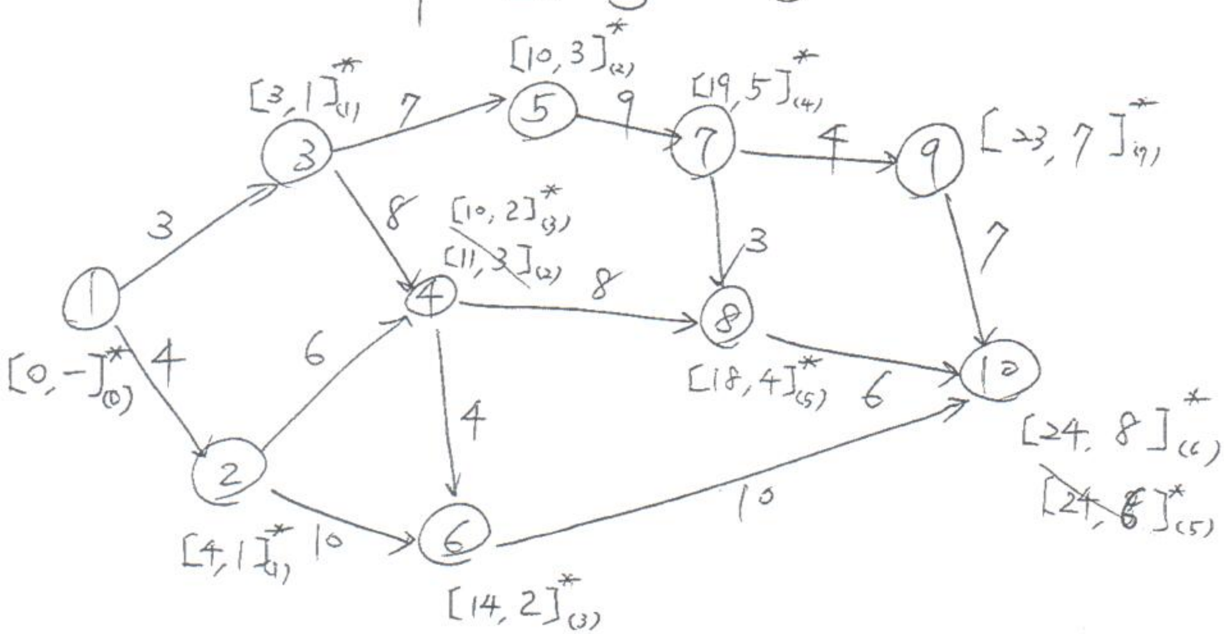
step 0: 令 node 1 為一 permanent label $[\underset{\text{距離}}{0}, \underset{\text{predecessor}}{-}]$, set $i=1$.

step i: (a) 計算所有 temporary labels $[u_i + d_{ij}, i]$, for each node j 可由 node i 連接。如果 node j 已經標示為 $[u_j, k]$ 透過其他 node k, 且 $u_i + d_{ij} < u_j$ 將 $[u_j, k]$ 換成 $[u_i + d_{ij}, i]$

(b) 如果所有節點為 permanent labels, 停止; 否則在所有 temporary labels 選一 label $[u_r, s]$ 有最短距離。令 $i = r$. 和重覆 step i.



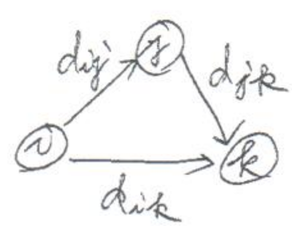
1 − 3 − 4 − 5
1 − 3 − 5



1 − 2 − 4 − 8 − 10
1 − 2 − 6 − 10

Floyd's algorithm

其 idea



如果從 i 至 k，

$d_{ij}+d_{jk}<d_{ik}$，則將路徑 i-k 換成 i-j-k.

步驟 0：定義開始矩陣 D_0 和 節點順序矩陣 S_0

D_0 與 S_0 的对角線 (—) 代表不通，令 k=1

步驟 k：定義 k 列 和 k 欄 為 樞軸列 與 樞軸欄

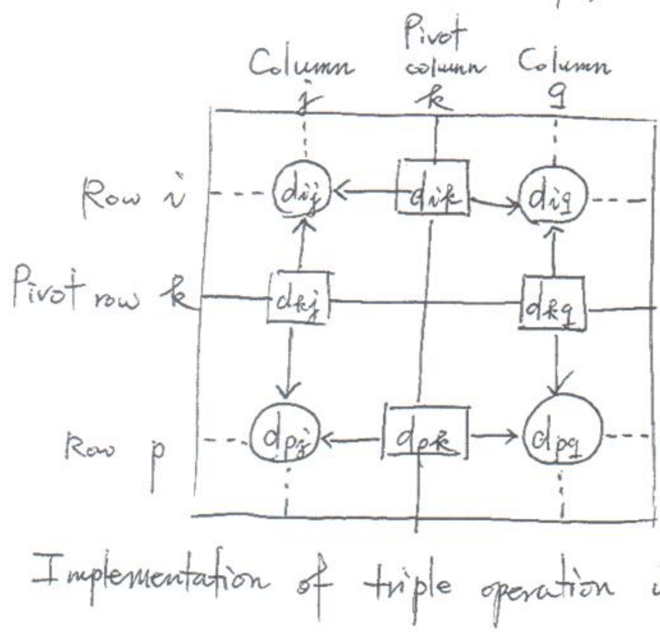
應用 triple operation ~~於每~~ 一 d_{ij} 在 D_{k-1}，对所有 i, j

如果 $d_{ik}+d_{kj}<d_{ij}$ ($i\neq k, j\neq k, i\neq j$) 滿足，則做下列改變.

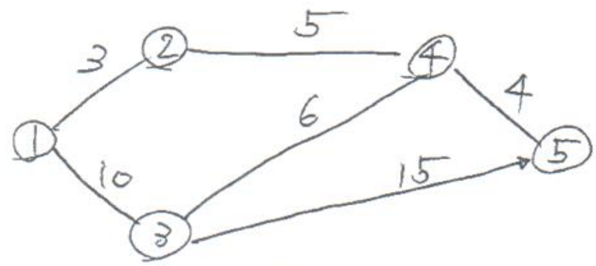
(a) 創造 D_k，藉著將 $d_{ik}+d_{kj}$ 取代 D_{k-1} 中的 d_{ij}

(b) 創造 S_k 藉著將 k 取代 S_{k-1} 中的 S_{ij}

令 k=k+1，重覆 k 步驟.



Implementation of triple operation in matrix form e.g. 6.3-5



Iteration 0

完成 D_0 與 S_0

D_0

	1	2	3	4	5
1	—	3	10	∞	∞
2	3	—	$\boxed{\infty}$	5	∞
3	10	$\boxed{\infty}$	—	6	15
4	∞	5	6	—	4
5	∞	∞	∞	4	—

S_0

	1	2	3	4	5
1	—	2	3	4	5
2	1	—	3	4	5
3	1	2	—	4	5
4	1	2	3	—	5
5	1	2	3	4	—

Iteration 1. $k=1$, pivot row 和 pivot column $=1$ 在 D_0 matrix

畫方框的 d_{23} 和 d_{32} 是唯一可以改進

d_{23} 以 $d_{21}+d_{13}=3+10=13$ 取代；令 $S_{23}=k=1$

d_{32} 以 $d_{31}+d_{12}=10+3=13$ 取代；令 $S_{32}=k=1$

於是得到 D_1 與 S_1

D_1

	1	2	3	4	5
1	—	3	10	$\boxed{\infty}$	∞
2	3	—	13	5	∞
3	10	13	—	6	15
4	$\boxed{\infty}$	5	6	—	4
5	∞	∞	∞	4	—

S_1

	1	2	3	4	5
1	—	2	3	$\boxed{4}$	5
2	1	—	1	4	5
3	1	1	—	4	5
4	$\boxed{1}$	2	3	—	5
5	1	2	3	4	—

iteration 2 $k=2$. pivot row 和 pivot column $=2$ 在 D_1 matrix

畫方框的 d_{14} 與 d_{41} 可以改善.

$d_{14} = d_{12} + d_{24} = 3+5 = 8$; 令 $S_{14} = k = 2$

$d_{41} = d_{42} + d_{21} = 5+3 = 8$; 令 $S_{41} = k = 2$.

於是得到 D_2 與 S_2

D_2

	1	2	3	4	5
1	—	3	10	8	$\boxed{\infty}$
2	3	—	13	5	$\boxed{\infty}$
3	10	13	—	6	15
4	8	5	6	—	4
5	∞	∞	∞	4	—

S_2

	1	2	3	4	5
1	—	2	3	2	$\boxed{5}$
2	1	—	1	4	$\boxed{5}$
3	1	1	—	4	5
4	2	2	3	—	5
5	1	2	3	4	—

iteration 3 $k=3$. pivot row 和 pivot column $=3$ 在 D_2 matrix

畫方框的 d_{15} 與 d_{25} 可以改善

$d_{15} = d_{13} + d_{35} = 10 + 15 = 25$; 令 $S_{15} = 3$

$d_{25} = d_{23} + d_{35} = 13 + 15 = 28$; 令 $S_{25} = k = 3$

於是得到 D_3 與 S_3

D_3

	1	2	3	4	5
1	—	3	10	8	$\boxed{25}$
2	3	—	$\boxed{13}$	5	$\boxed{28}$
3	10	$\boxed{13}$	—	6	$\boxed{15}$
4	8	5	6	—	4
5	$\boxed{\infty}$	$\boxed{\infty}$	$\boxed{\infty}$	4	—

S_3

	1	2	3	4	5
1	—	2	3	2	$\boxed{3}$
2	1	—	$\boxed{1}$	4	$\boxed{3}$
3	1	$\boxed{1}$	—	4	$\boxed{5}$
4	2	2	3	—	5
5	$\boxed{1}$	$\boxed{2}$	$\boxed{3}$	4	—

iteration 4. Set $k=4$ pivot row 和 pivot column $=4$ 在 D_3 matrix

畫方框的均可以改善

$d_{15}=d_{14}+d_{45}=8+14=12$; $S_{15}=4$

$d_{25}=d_{24}+d_{45}=5+4=9$; $S_{25}=4$

$d_{23}=d_{24}+d_{43}=5+6=11$; $S_{23}=4$

$d_{32}=d_{34}+d_{42}=6+5=11$; $S_{32}=4$

$d_{35}=d_{34}+d_{45}=6+4=10$; $S_{35}=4$

$d_{51}=d_{54}+d_{41}=4+8=12$; $S_{51}=4$

$d_{52}=d_{54}+d_{42}=4+5=9$; $S_{52}=4$

$d_{53}=d_{54}+d_{43}=4+6=10$; $S_{53}=4$.

於是得到

D_4

	1	2	3	4	5
1	—	3	10	8	12
2	3	—	11	5	9
3	10	11	—	6	10
4	8	5	6	—	4
5	12	9	10	4	—

S_4

	1	2	3	4	5
1	—	2	3	2	4
2	1	—	4	4	4
3	1	4	—	4	4
4	2	2	3	—	5
5	4	4	4	4	—

iteration 5 Set $k=5$, pivot row 與 pivot column $=5$ 在 D_4 matrix

沒有可改善的可能。

如果要找 node 1 至 node 5 之最短路徑, 看 D_4 $(1,5)=12$

可知距離為 12, 再看 S_4 中

$S_{15}=4$, 不是直接連接 4-5

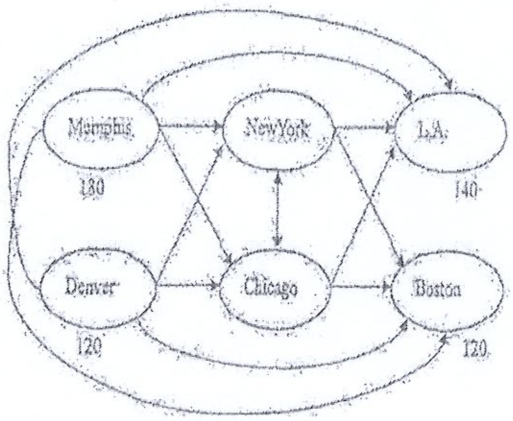
再看 $S_{14}=2$, 不是直接連接 2-4-5

再看 $S_{12}=2$, 是直接連接 1-2-4-5

§ 轉運問題 § THE TRANSSHIPMENT MODEL

標準運輸模式假設供給點與運送點之間的路徑為最短路徑，而轉運問題結合運輸模式與最短路徑成為一方法。

轉運問題乃基於運輸上及管理上的需求而產生。例如某地区定期航線的港口之間，經常運用轉運方式，以湊足開船的載量，經常有甲港口輸往非洲地区之貨物集中到乙港口以便集中一船運送。



	Memphis	Denver	N.Y.	Chicago	L.A.	Boston
Memphis	$0	—	$8	$13	$25	$28
Denver	—	$0	$15	$12	$26	$25
N.Y.	—	—	$0	$6	$16	$17
Chicago	—	—	$6	$0	$14	$16
L.A.	—	—	—	—	$0	—
Boston	—	—	—	—	—	$0

Memphis 與 Denver 為供給地，L.A.與 Boston 為需求地，其相關供給量、需求量、轉運點與轉運成本如上所述，試解決上述問題，列出轉運問題模式，並求解

	New York	Chicago	Boston	L.A.	
Memphis	8	13	25	28	180
Denver	15	12	26	25	120
New York	0	6	16	17	300
Chicago	6	0	14	16	300
	300	300	120	140	

We now describe how the optimal solution to a transshipment problem can be found by solving a transportation problem. Given a transshipment problem, we create a balanced transportation problem by the following procedure (assume that total supply exceeds total demand):

Step 1 If necessary, add a dummy demand point (with a supply of 0 and a demand equal to the problem's excess supply) to balance the problem. Shipments to the dummy and from a point to itself will, of course, have a zero shipping cost. Let $s =$ total available supply.

Step 2 Construct a transportation tableau as follows: A row in the tableau will be needed for each supply point and transshipment point, and a column will be needed for each demand point and transshipment point. Each supply point will have a supply equal to its original supply, and each demand point will have a demand equal to its original demand. Let $s =$ total available supply. Then each transshipment point will have a supply equal to (point's original supply) $+ s$ and a demand equal to (point's original demand) $+ s$. This ensures that any transshipment point that is a net supplier will have a net outflow equal to the point's original supply, and, similarly, a net demander will have a net inflow equal to the point's original demand. Although we don't know how much will be shipped through each transshipment point, we can be sure that the total amount will not exceed s. This explains why we add s to the supply and demand at each transshipment point. By adding the same amounts to the supply and demand, we ensure that the net outflow at each transshipment point will be correct, and we also maintain a balanced transportation tableau.

Transshipment occurs in the network in Figure 5.7 because the entire supply amount of 2200 (= 1000 + 1200) cars at nodes $P1$ and $P2$ could conceivably pass through any node of the network before ultimately reaching their destinations at nodes $D1$, $D2$, and $D3$. In this regard, each node of the network with both input and output arcs ($T1$, $T2$, $D1$, and $D2$) acts as both a source and a destination and is referred to as a **transshipment node.** The remaining nodes are either **pure supply nodes** ($P1$ and $P2$) or **pure demand nodes** ($D3$).

The transshipment model can be converted into a regular transportation model with six sources ($P1, P2, T1, T2, D1$, and $D2$) and five destinations ($T1, T2, D1, D2$, and $D3$). The amounts of supply and demand at the different nodes are computed as

$$\text{Supply at a } \textit{pure supply node} = \text{Original supply}$$
$$\text{Demand at a } \textit{pure demand node} = \text{Original demand}$$
$$\text{Supply at a } \textit{transshipment node} = \text{Original supply} + \text{Buffer amount}$$
$$\text{Demand at a } \textit{transshipment node} = \text{Original demand} + \text{Buffer amount}$$

The buffer amount should be sufficiently large to allow all of the *original* supply (or demand) units to pass through any of the *transshipment* nodes. Let B be the desired buffer amount; then

$$\begin{aligned} B &= \text{Total supply (or demand)} \\ &= 1000 + 1200 \text{ (or } 800 + 900 + 500) \\ &= 2200 \text{ cars} \end{aligned}$$

Using the buffer B and the unit shipping costs given in the network, we construct the equivalent regular transportation model as in Table 5.43.

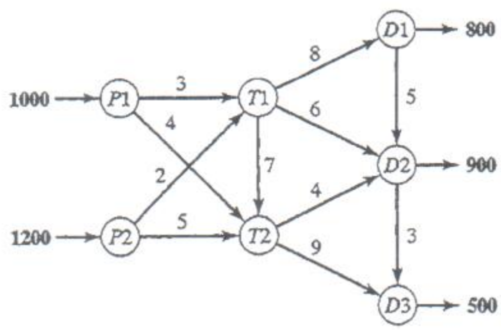


FIGURE 5.7

Transshipment network between plants and dealers

TABLE 5.43 Transshipment Model

	$T1$	$T2$	$D1$	$D2$	$D3$	
$P1$	3	4	M	M	M	1000
$P2$	2	5	M	M	M	1200
$T1$	0	7	8	6	M	B
$T2$	M	0	M	4	9	B
$D1$	M	M	0	5	M	B
$D2$	M	M	M	0	3	B
	B	B	$800 + B$	$900 + B$	500	

The solution of the resulting transportation model (determined by TORA) is shown in Figure 5.8. Note the effect of transshipment: Dealer $D2$ receives 1400 cars, keeps 900 cars to sat-