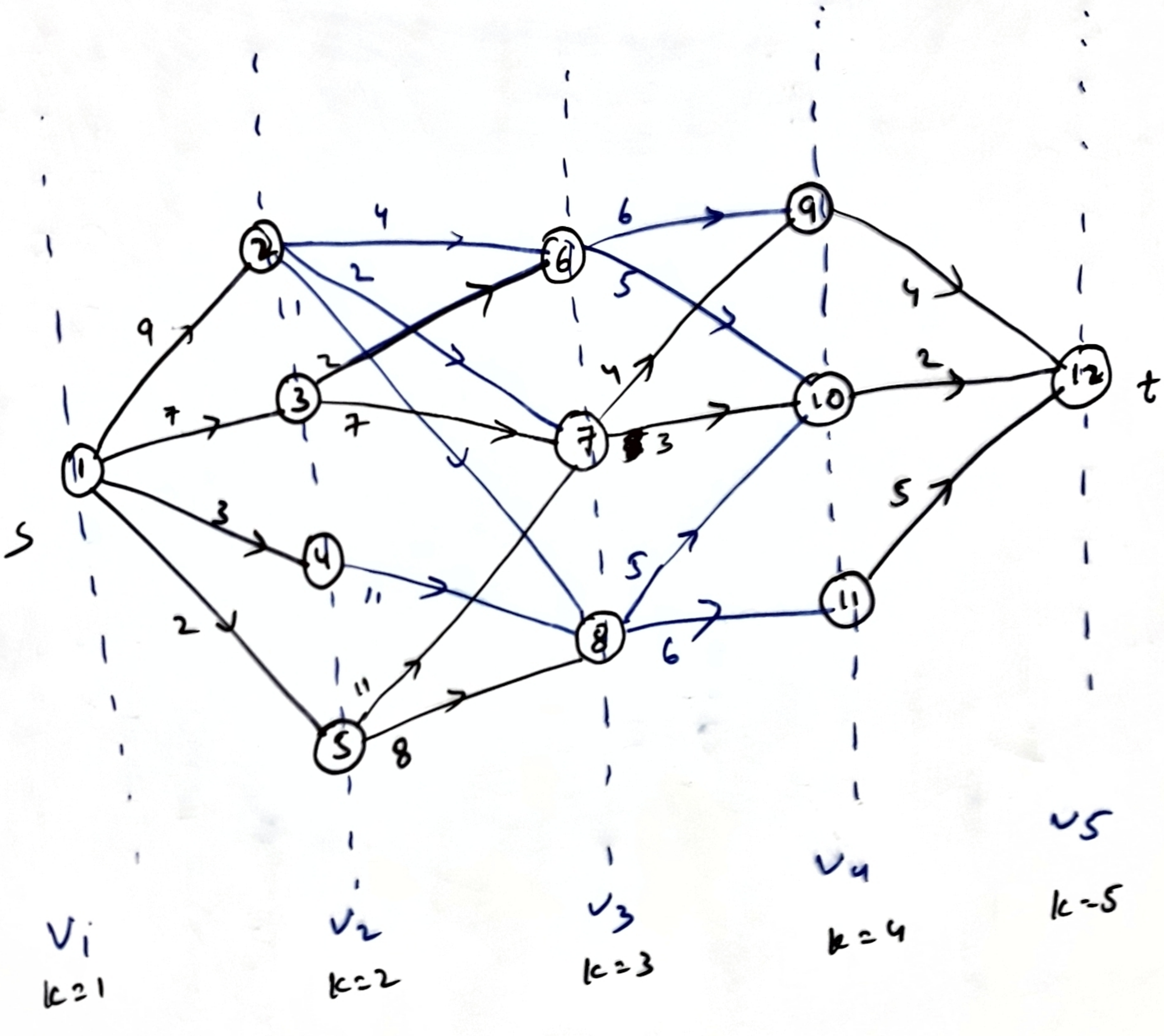
Multistage Graph

A multistage graph $G = (V, E)$ is a directed graph in which vertices are partitioned $k >= 2$ disjoint set (set V_i) where $[1 <= i <= k]$.
(u, v) is an edge E then $u \in V_i$, $v \in V_{i+1}$

Let $c(i, j)$ be the cost of edge (i, j) the cost of a path from (S to T) is the sum of costs of the edges on the path (S and T are the vertices V_1 and V_k)

The multistage graph problem is to find the minimum cost path from S to T.

$$cost(i, j) = \min_{\substack{l \in V_{i+1} \\ \langle j, l \rangle \in E}} \{ c(j, l) + cost(i+1, l) \}$$


s
t
V1
k=1
V2
k=2
V3
k=3
V4
k=4
V5
k=5

$$cost(i,j) = \min_{\substack{l \in V_{i+1} \\ \langle j,l \rangle \in E}} \{ c(j,l) + cost(i+1,l) \}$$

Stage 4-5

Step 1 :- $\quad i = 4,\ j = 9, 10, 11$

$$cost(4,9) = 4$$
$$cost(4,10) = 2$$
$$cost(4,11) = 5$$

Stage 3-4

Step 2 :- $\quad i = 3,\ j = 6, 7, 8$

$$cost(3,6) = \min \begin{cases} c(6,9) + cost(4,9) = 6+4 = 10 \\ c(6,10) + cost(4,10) = 5+2 = 7^* \end{cases}$$

$$cost(3,7) = \min \begin{cases} c(7,9) + cost(4,9) = 4+4 = 8 \\ c(7,10) + cost(4,10) = 3+2 = 5^* \end{cases}$$

$$cost(3,8) = \min \begin{cases} c(8,10) + cost(4,10) = 5+2 = 7^* \\ c(8,11) + cost(4,11) = 6+5 = 11 \end{cases}$$

Stage 2-3

Step 3 :- $\quad i = 2,\ j = 2, 3, 4, 5$

$$cost(2,2) = \min \begin{cases} c(2,6) + cost(3,6) = 4+7 = 11 \\ c(2,7) + cost(3,7) = 2+5 = 7^* \\ c(2,8) + cost(3,8) = 11+7 = 16 \end{cases}$$

$$\text{cost}(2,3) = \min \begin{cases} c(3,6) + \text{cost}(3,6) = 2+7 = 9 \\ c(3,7) + \text{cost}(3,7) = 7+5 = 12 \end{cases}$$

$$\text{cost}(2,4) = \min \{ c(4,8) + \text{cost}(3,8) = 11 + 7 = 18$$

$$\text{cost}(2,5) = \min \begin{cases} c(5,7) + \text{cost}(3,7) = 11 + 5 = 16 \\ c(5,8) + \text{cost}(3,8) = 8 + 7 = 15 \end{cases}$$

Stage 1-2

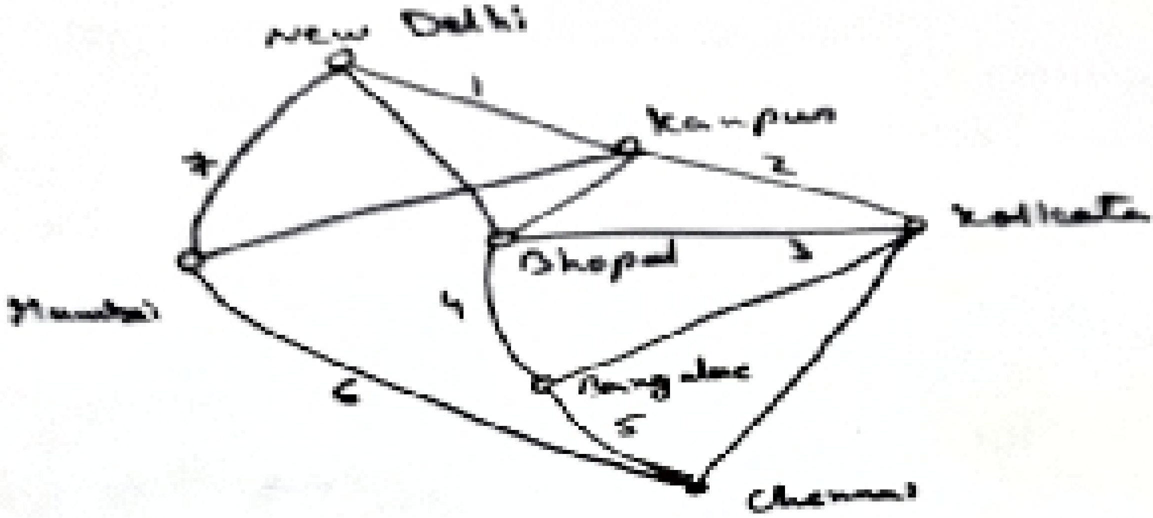
Step 4 :- $i=1,\ j=1$

$$\text{cost}(1,1) \overset{\min}{=} \begin{cases} c(1,2) + \text{cost}(2,2) = 9 + 7 = 16 \\ c(1,3) + \text{cost}(2,3) = 7 + 9 = 16 \\ c(1,4) + \text{cost}(2,4) = 3 + 18 = 21 \\ c(1,5) + \text{cost}(2,5) = 2 + 15 = 17 \end{cases}$$

Path :-
1. $1 \to 2 \to 7 \to 10 \to 12$
2. $1 \to 3 \to 6 \to 10 \to 12$

```
Algorithm Fgraph(G, k, n, p)
// the input is a k stage graph G=(V,E)
// with n vertices indexed in order of stages.
// E is a set of edges and c[i,j] is the
// cost of <i,j>. p[1:k] is a minimum
// cost path.
{
    cost[n] := 0.0;
    for j := n-1 to 1 step -1 do
    {   // compute cost[j]
        Let r be a vertex such that <j,r>
        is an edge of G and c[j,r]+
        cost[r] is minimum.
        cost[j] := c[j,r] + cost[r];
        d[j] := r;
    }
    // Find a minimum cost path.
    p[1] := 1; p[k] := n;
    for j := 2 to k-1 do
        p[j] := d[p[j-1]];
}
```

Dynamic Programming - Traveling Salesperson problem ①



Starts from one city and deliver parts in all other cities and come back.
One can go from any city to any city but the traveling cost varies. Find the minimum cost path.

Let $G(V, E)$ ~ directed graph

Edge cost $= C_{ij}$

$= \infty$ if cities are not connected.

→ A tour of G is a directed simple cycle that includes every vertex in V.

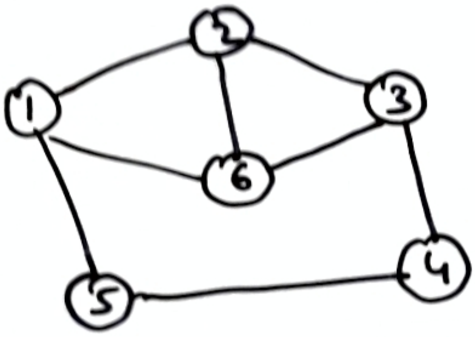
→ The cost of tour is sum of weights of edges on tour.

→ Traveling Salesman problem is the minimum cost tour.

→ Without loss of generality, let the tour start at vertex 1.

Each tour consists of an EDGE <1, k> for some k ∈ V - {1} and a path from vertex k to 1 going through all the vertices in V - {1, k} exactly once.

→ If k = 2, Edge from 1 to 2 and path from 2 to 1, which consists of all vertices 3, 4, 5 and 6



→ If k is optimal (tour is optimal), then path from k to 1 through all vertices in V - {1, k} should be shortest. Hence, the principle of optimality holds.

Let $g(i, S)$ = length of shortest path from node 1 and going through all vertices in S and terminating at 1.

$\therefore g(1, V-\{1\})$ = optimal length tour.

$$g(1, V-\{1\}) = \min_{2 \le k \le n} \{ c_{1k} + g(k, V-\{1,k\}) \}$$

$$\boxed{\begin{array}{l} \text{For } i \notin S \\ g(i, S) = \min\limits_{j \in S} \{ c_{ij} + g(j, S-\{j\}) \} \end{array}}$$

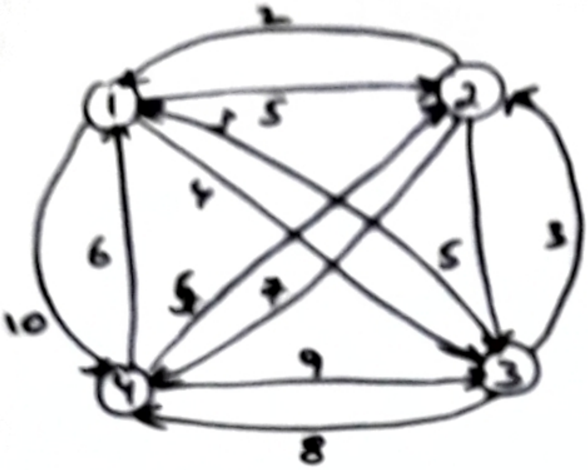
→ 1

To solve $g(1, V-\{1\})$ we need to know $g(k, V-\{1,k\})$ for all choices of k subproblems.

$g(i, \phi) = c_{i1} \quad | \quad 1 \le i \le n$

when $|S| < n-1$, the values of i and S for which $g(i, S)$ is needed are $i \neq 1$, $1 \notin S$, $i \notin S$.

Dynamic Programming - Traveling Salesperson Problem 24



Distance matrix d:

$$\begin{array}{c|cccc} & 1 & 2 & 3 & 4 \\ \hline 1 & 0 & 5 & 3 & 10 \\ 2 & 2 & 0 & 5 & 7 \\ 3 & 4 & 3 & 0 & 8 \\ 4 & 6 & 5 & 9 & 0 \end{array}$$

$g(i, \phi) = c_{i1}$, $1 \le i \le n$

$|s| = 0$

1. $g(2, \phi) = c_{21} = 2$
   $g(3, \phi) = c_{31} = 4$
   $g(4, \phi) = c_{41} = 6$

2. $|s| = 1$
   1 node in the set. Using eq. 1

$$g(2, \{3\}) = \min_{j \in s} \{c_{23} + g(3, \phi)\}$$
$$= 5 + 4 = 9$$

$g(2, \{4\}) = c_{24} + g(4, \phi)$
$= 7 + 6 = 13$

$g(3, \{2\}) = c_{32} + g(2, \phi)$
$= 3 + 2 = 5$

$g(3, \{4\}) = c_{34} + g(4, \phi)$
$= 8 + 6 = 14$

$g(4, \{2\}) = c_{42} + g(2, \phi)$
$= 5 + 2 = 7$

$g(4, \{3\}) = c_{43} + g(3, \phi)$
$= 9 + 4 = 13$

3 For $|S| = 2$
2 nodes in the set

$$g(2, \{3, 4\}) = \min_{j \in S} \left\{ \begin{array}{l} c_{23} + g(3, \{4\}), \\ c_{24} + g(4, \{3\}) \end{array} \right\}$$

$= \min\{5 + 14, 7 + 13$
$= \min\{19, 20\} = 19$

$$g(3, \{2, 4\}) = \min_{j \in S} \left\{ \begin{array}{l} c_{32} + g(2, \{4\}), \\ c_{34} + g(4, \{2\}) \end{array} \right.$$

$= \min\{3 + 13, 8 + 7$
$= \min\{16, 15\}$
$= 15$

$$g(4, \{2,3\}) = \min_{j \in s} \begin{cases} c_{42} + g(2, \{3\}), \\ c_{43} + g(3, \{2\}) \end{cases}$$

$= \min\{5+9, 9+5$

$= \min\{14, 14 \quad = 14$

(4) $|s| = 3$, 3 nodes in the set

$g(1, \{2,3,4\})$

$$= \min \begin{cases} c_{12} + g(2, \{3,4\}) \\ c_{13} + g(3, \{2,4\}) \\ c_{14} + g(4, \{2,3\}) \end{cases}$$

$= \min\{5+19, 3+15, 10+14$

$= \min\{24, 18, 24 \quad = 18$

$\therefore$ optimal tour length $= 18$

Reconstruct the path:

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

$1 \rightarrow 3$

$g(3, \{2,4\}) = 4$

$1 \rightarrow 3 \rightarrow 4$

$g(4, \{2,3\}) = 2$

$\therefore$ the cycle is: $1 \rightarrow 3 \rightarrow 4 \rightarrow 2 \rightarrow 1$