Practice 3.3: Shortest path

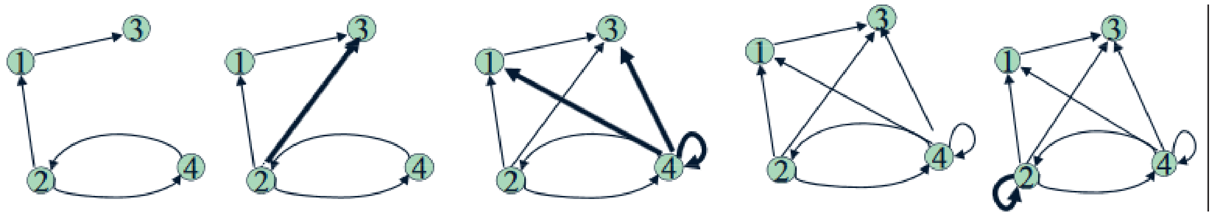
Artificial Intelligence

G.Guérard

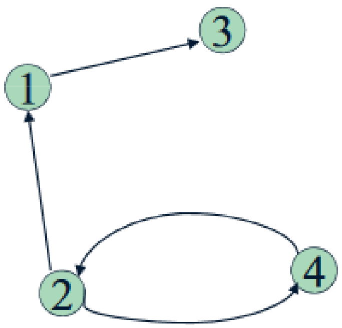
Floyd-Warshall's algorithm

Remind the principle of this algorithm. A path exists between two vertices I and j iff:

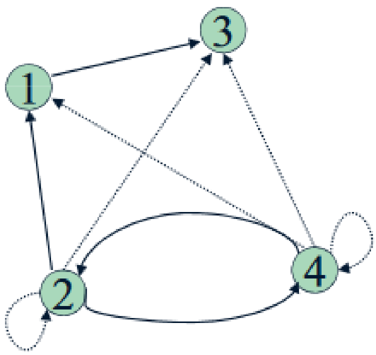
- There is an edge from i to j or
- There is a path from i to j going through vertex 1 or
- There is a path from i to j going through vertices 1 and/or 2 or
- Etc.
- There is a path from i to j going through any of the other vertices.



This principle is called "transitive closure": if a path from i to j exists, add an edge from i to j

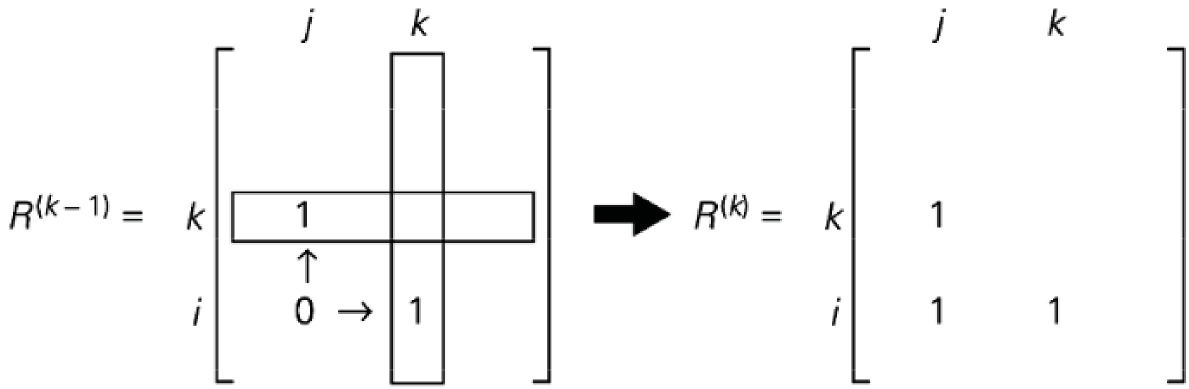


0	0	1	0
1	0	0	1
0	0	0	0
0	1	0	0



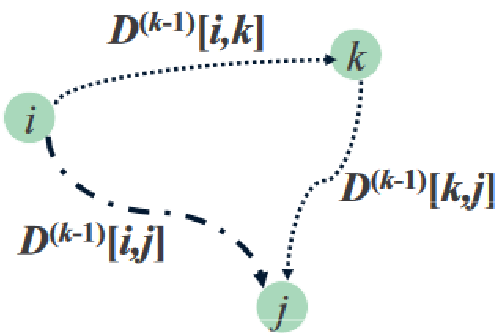
0	0	1	0
1	**1**	**1**	1
0	0	0	0
1	1	**1**	**1**

The transitive closure compute: $R^k[i,j] = \{R^{k-1}[i,j] or \left(R^{k-1}[i,k] \; and \; R^{k-1}[k,j]\right)\}$. The R matrix (transitive closure reproduce its previous iteration and add 1 to [I,j] iff –see the following scheme).

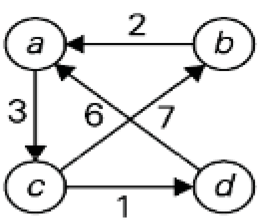


On the k-th iteration, the algorithm **determine if a path exists** between two vertices I, j using just vertices among 1,…,k allowed as intermediate.

The Floyd-Warshall's algorithm is based on the transitive closure. It computes $W^k[i,j] = \min\{W^{k-1}[i,j], W^{k-1}[i,k] + W^{k-1}[k,j]\}$. This function is similar to Dijkstra or Bellman-Ford formula.



On the k-th iteration, the algorithm **determine shortest path** between two vertices I, j using just vertices among 1,…,k allowed as intermediate.



$D^{(0)}$ =

	a	b	c	d
a	0	∞	3	∞
b	2	0	∞	∞
c	∞	7	0	1
d	6	∞	∞	0

➔ Lengths of the shortest paths with no intermediate vertices (aka the weight matrix).

$D^{(1)}$ =

	a	b	c	d
a	0	∞	3	∞
b	2	0	**5**	∞
c	∞	7	0	1
d	6	∞	**9**	0

➔ Lengths of the shortest paths with intermediate vertices numbered not higher than 1 (here node a)

$$D^{(2)} = \begin{array}{c} \\ a \\ b \\ c \\ d \end{array} \begin{array}{c} \begin{array}{cccc} a & b & c & d \end{array} \\ \begin{bmatrix} 0 & \infty & 3 & \infty \\ 2 & 0 & 5 & \infty \\ \mathbf{9} & 7 & 0 & 1 \\ 6 & \infty & 9 & 0 \end{bmatrix} \end{array}$$

➔ Etc…

$$D^{(4)} = \begin{array}{c} \\ a \\ b \\ c \\ d \end{array} \begin{array}{c} \begin{array}{cccc} a & b & c & d \end{array} \\ \begin{bmatrix} 0 & 10 & 3 & 4 \\ 2 & 0 & 5 & 6 \\ \mathbf{7} & 7 & 0 & 1 \\ 6 & 16 & 9 & 0 \end{bmatrix} \end{array}$$

Exercise

Use the Floyd-Warshall's algorithm on the following graph: