

# Cammini minimi tra tutte le coppie

Vittorio Maniezzo - Università di Bologna

1

## Cammini minimi tra tutte le coppie

Dato un grafo (orientato o non orientato)  $G = (V, E, W)$  con funzione di peso  $w: E \rightarrow \mathbf{R}$ , trovare per ogni coppia di vertici  $u, v \in V$  il minimo peso di un cammino da  $u$  a  $v$ .

Verrà calcolata anche una **matrice di predecessori**  $\Pi(\pi_{uv})$  dove  $\pi_{uv}$  è NIL se  $u=v$  o se non c'è un cammino da  $u$  a  $v$ , altrimenti è un predecessore di  $v$  su di un cammino minimo da  $u$ .

Il sottografo indotto dall' $i$ -esima riga della matrice  $\Pi$  sarà un **albero di cammini minimi** con radice in  $i$ .

Vittorio Maniezzo - Università di Bologna

2

2

# Algoritmo di Floyd-Warshall

E' un algoritmo di **programmazione dinamica**, può gestire archi di peso negativo ma si assume che **non ci siano cicli negativi**.

Idea:

$d_{s,t}(i)$ : cammino minimo da  $s$  a  $t$  contenente solo i vertici intermedi  $v_1, \dots, v_i$

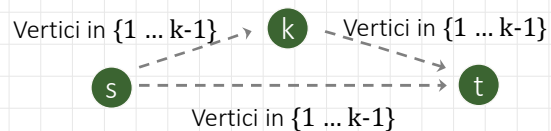
$$d_{s,t}(0) = w(s,t)$$

Vittorio Maniezzo - Università di Bologna

3

3

## Floyd-Warshall, idea



$d_{s,t}(k-1)$ : cammino minimo da  $s$  a  $t$  contenente solo i vertici intermedi  $v_1, \dots, v_{k-1}$

$$d_{s,t}(0) = w(s,t)$$

$$d_{s,t}(k) = \begin{cases} w(s,t) & \text{se } k = 0 \\ \min\{d_{s,t}(k-1), d_{s,k}(k-1) + d_{k,t}(k-1)\} & \text{se } k > 0 \end{cases}$$

$k$  non è nel cammino minimo  $s$ -

$k$  è nel cammino minimo  $s$ - $t$

Vittorio Maniezzo - Università di Bologna

4

4

## Determinazione $\Pi$

$\Pi$  è ottenibile da  $D$  in  $O(n^3)$ .

Oppure, "on line":

$$\pi_{ij}(0) = \begin{cases} \text{NIL} & \text{se } i=j \text{ o } w_{ij} = \infty \\ i & \text{se } i \neq j \text{ e } w_{ij} < \infty \end{cases}$$

poi

$$\pi_{ij}(k) = \begin{cases} \pi_{ij}(k-1) & \text{se } d_{ij}(k-1) \leq d_{ik}(k-1) + d_{kj}(k-1) \\ \pi_{kj}(k-1) & \text{se } d_{ij}(k-1) > d_{ik}(k-1) + d_{kj}(k-1) \end{cases}$$

## Algoritmo di Floyd-Warshall

Floyd-Warshall( $W$ )

$n = \text{rows}[W]$

$D(0) = W$

for  $k = 1$  to  $n$  do

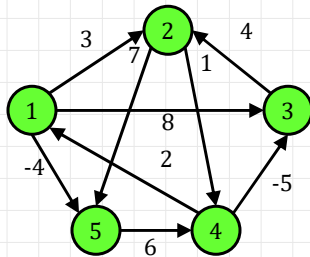
  for  $i = 1$  to  $n$  do

    for  $j = 1$  to  $n$  do

$d_{ij}(k) = \min(d_{ij}(k-1), d_{ik}(k-1) + d_{kj}(k-1))$

return  $D(n)$

# Floyd-Warshall, esempio



W

0	3	8	$\infty$	-4
$\infty$	0	$\infty$	1	7
$\infty$	4	0	$\infty$	$\infty$
2	$\infty$	-5	0	$\infty$
$\infty$	$\infty$	$\infty$	6	0

Vittorio Maniezzo - Università di Bologna

7

7

# Floyd-Warshall, esempio

$D^{(0)}$

0	3	8	$\infty$	-4
$\infty$	0	$\infty$	1	7
$\infty$	4	0	$\infty$	$\infty$
2	$\infty$	-5	0	$\infty$
$\infty$	$\infty$	$\infty$	6	0

$\Pi^{(0)}$

	1	1		1
			2	2
	3			
4		4		
			5	

Vittorio Maniezzo - Università di Bologna

8

8

## Floyd-Warshall, esempio

$D^{(1)}$

0	3	8	$\infty$	-4
$\infty$	0	$\infty$	1	7
$\infty$	4	0	$\infty$	$\infty$
2	5	-5	0	-2
$\infty$	$\infty$	$\infty$	6	0

$\Pi^{(1)}$

	1	1		1
			2	2
	3			
4	1	4		1
			5	

Vittorio Maniezzo - Università di Bologna

9

9

## Floyd-Warshall, esempio

$D^{(2)}$

0	3	8	4	-4
$\infty$	0	$\infty$	1	7
$\infty$	4	0	5	11
2	5	-5	0	-2
$\infty$	$\infty$	$\infty$	6	0

$\Pi^{(2)}$

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

Vittorio Maniezzo - Università di Bologna

10

10

## Floyd-Warshall, esempio

$D^{(3)}$

0	3	8	4	-4
$\infty$	0	$\infty$	1	7
$\infty$	4	0	5	11
2	-1	-5	0	-2
$\infty$	$\infty$	$\infty$	6	0

$\Pi^{(3)}$

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

Vittorio Maniezzo - Università di Bologna

11

11

## Floyd-Warshall, esempio

$D^{(4)}$

0	3	-1	4	-4
3	0	-4	1	-1
7	4	0	5	3
2	-1	-5	0	-2
8	5	1	6	0

$\Pi^{(4)}$

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

Vittorio Maniezzo - Università di Bologna

12

12

## Floyd-Warshall, esempio

$D^{(5)}$

0	3	-3	2	-4
3	0	-4	1	-1
7	4	0	5	3
2	-1	-5	0	-2
8	5	1	6	0

$\Pi^{(5)}$

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

13

## Floyd-Warshall: complessità

Determinata dai tre cicli for.

Ogni esecuzione dell'istruzione interna è  $O(1)$ , quindi:

$$T(V,E) = \Theta(n^3) = \Theta(V^3)$$

14

## Stampa dei cammini

```
Print-all-pairs-shortest-path( $\Pi$ , i, j)
  if (i=j)
  then STAMPA i
  else if  $\pi_{ij}$  = NIL
    then STAMPA "Non esiste il cammino"
    else Print-all-pairs-shortest-path( $\Pi$ , i,  $\pi_{ij}$ )
      STAMPA j
```