ES 31

Per scoprire se G ha un ciclo di costo < 0, dobbiamo

Eseguire l'algoritmo di Bellman e Ford (nella prima versione)

per calcolare i valori $OPT(i, v)$ $\forall \in V$ e per $i = 1 .. n$

Non esistono cicli se e solo se esiste qualche valore

$i \leq n$ per cui $OPT(i, v) = OPT(i-1, v)$ $\forall v \in V$

In G non esistono cicli di costo negativo da cui si può raggiungere

t se e solo se vale che $OPT(n, v) = OPT(n-1, v)$ per ogni nodo v.

Se ogni cammino da s a t non contiene alcun ciclo di costo totale

minimo, allora esiste un cammino totale minimo da s a t e contiene al

più n-1 archi ($n = |V|$)

E questo implica che $OPT(n, v) = OPT(n-1, v)$. Ciò implica

che $OPT(n, v) = OPT(n-1, v)$ perché l'aggiunta di più archi non

può evidentemente portare a cammini di costo totale inferiore al costo del

cammino minimo di minimo costo

ES 32

Eseguire l'algoritmo di Bellman e Ford (prima versione) per calcolare

i valori $OPT(i, v)$ $\forall v \in V$ e per $i = 1 .. n$

Sia v un nodo per cui $OPT(n, v) < OPT(n-1, v)$ (c'è altrimenti non ci sono cicli)

Il cammino da v a t di costo $OPT(n, v)$ ha esattamente n archi,

tale cammino ha n + 1 nodi e quindi un ciclo, il ciclo ha

costo < 0 altrimenti lo potremmo eliminare, ottenendo un

cammino di $\leq n-1$ archi di costo $< OPT(n, v) < OPT(n-1, V)$ e

un assurdo.