Algoritmi Lezione 12/12/2023 Faro

CAMMINI MINIMI DA SORGENTE SINGOLA (SSSP)

La scorsa lezione abbiamo trattato del label correcting

$\forall v \in V$ troviamo $d[v] \rightarrow$ stima di cammino minimo

$d[v] = \delta(s,v)$ la stima del cammino minimo converge verso questo valore

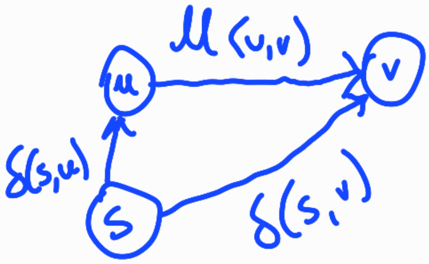
Le operazioni che consentono la convergenza sono operazioni di $\boxed{\text{relax}}$

RELAX$(u, v, w) \rightarrow$ crea delle variabili temporali che contengono il valore minimo

PROPRIETÀ LABEL CORRECTING

1) Disuguaglianza triangolare

Abbiamo: $\delta(s,v) \leq \delta(s,u) + w(u,v)$



Si dimostra facilmente poiché $\delta(s,v)$ è un cammino minimo.

2) Proprietà del limite superiore

$d[v] \geq \delta(s,v)$

La stima del cammino minimo non scende sotto il valore del cammino minimo.

3) Proprietà dell'assenza del cammino

Se manca un cammino $d[v] \leftarrow +\infty$

I nodi non raggiungibili rimarranno $\delta(s,v) \leftarrow +\infty$

con valore $+\infty$

4) Proprietà della convergenza

Se $d[u] = \delta(s,u)$

$\downarrow$

RELAX (u, v, w)

$\downarrow$

$d[v] = \delta(s,v)$

La proprietà di convergenza si propaga tra i nodi adiacenti

5) Proprietà della convergenza del cammino

Seguendo il cammino minimo si possono rilassare gli archi man mano e mandare i nodi del cammino in convergenza sfruttando la proprietà 4

DAG - SP (usato nei grafi aciclici)

Arriva molto vicino al caso ottimale

Si sfrutta l'ordinamento topologico dei grafi aciclici.

Sfrutta a cascata la proprietà 5

Parte dal nodo sorgente e rilassa gli archi sfruttando l'ordinamento topologico

In questo modo con un solo ciclo si portano a convergenza tutti i nodi

PSEUDO-CODICE

DAG - SP(G, s)

// calcola un ordinamento topologico di G $O(V+E)$

$\forall v \in V$ (seguendo l'ord. topologico) $O(V)$

FOR EACH $u \in$ ADJ(v) $O(E)$

FOR EACH $u \in Adj[v]$ $O(E)$

RELAX (v, u, w) Costante

Le complessità è di $O(V+E)$ è un algoritmo ottimo, di meglio non si può fare, ma non si può usare in presenza di cicli

Andiamo a trattare del caso generale ovvero in presenza di cicli

Bellman - Ford

L'algoritmo deve calcolare i cammini minimi e verificare la presenza di cicli di peso negativo

Un cammino minimo non ha cicli, quindi la lunghezza massima di un cammino è di $|P| \leq |V|-1$

L'unica cosa che sappiamo è che s è andato in convergenza, quindi Bellman-Ford va alla cieca e rilassa tutti gli archi così u_0 sicuramente va a convergenza. Applica lo stesso procedimento per tutti gli archi e i nodi, fa questa procedura per $|V|-1$ volte, mandando a cascata tutti i nodi a convergenza.

PSEUDO-CODICE

```
      BELLMAN - FORD (G, s)
O(V)  ∀ v ∈ V DO   d[v] ← +∞
        d[s] ← 0  // sorgente a convergenza
O(V)  FOR i ← 1 TO |V| - 1 DO
O(E)    FOR EACH (u, v) ∈ E DO
            RELAX (u, v, w)
```

Possiamo aggiungere un test che verifica se sono state effettuate relax e ...

RELAX (u, v, w) in caso terminarla prima

$\Theta(E)$ FOR EACH $(u, v) \in E$ DO

IF $(d[v] > d[u] + w(u, v))$ THEN RETURN FALSE

RETURN TRUE Se risulta vero il grafo presenta cicli negativi ed esce poiché il problema è impossibile da risolvere

La complessità risulta $O(V \cdot E)$, non si può fare di meglio.