Bellman Ford

Που χρησιμοποιείται;

- Σε κατευθυνόμενα εμβαρή γραφήματα με αρνητικά βάρη
- Όπως και ο Dijkstra υπολογίζει το συντομότερο μονοπάτι για κάθε κορυφή του γραφήματος, με κορυφή εκκίνησης την s
- Σημαντική είναι η εύρεση αρνητικού κύκλου στο γράφημα

Είσοδος: $G(V,E,w)$, s
Έξοδος: Διανύσματα d, p

Αλγόριθμος

d_v^{k-1} = Απόσταση του κόμβου v από την κορυφή s, που έχει υπολογιστεί στην k-1 επανάληψη.

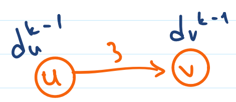
void **Bellman-Ford**$(G(V,E,w), s)$
for all $v \in V$ **do**
  $d_v^0 \leftarrow \infty; p_v^0 \leftarrow NULL;$
end for
$d_s^0 \leftarrow 0;$
for $k \leftarrow 1; k \leq |V|-1; k++$ **do**
  for all $v \in V$ **do**
    $d_v^k \leftarrow d_v^{k-1}; p_v^k \leftarrow p_v^{k-1};$
  end for
  for $(u,v) \in E$ **do**
    if $d_v^k > d_u^{k-1} + w(u,v)$ **then**
      $d_v^k \leftarrow d_u^{k-1} + w(u,v);$
      $p_v^k \leftarrow u;$
    end if
  end for
end for

Αρχικοποίηση των διανυσμάτων d,p

→ $|V|-1$ επαναλήψεις (όπως και Dijkstra)

Η επανάληψη k μεταφράζεται σε:
το πολύ k ακμές στο μονοπάτι κάθε κορυφής

Για κάθε ακμή (u,v) επαναυπολογίζει το d_v^k στην k επανάληψη ως εξής

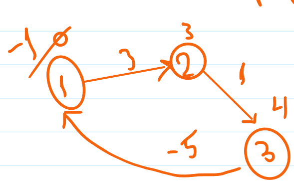


Εάν $d_v^{k-1} > d_u^{k-1}$ + βάρος ακμής
⋮

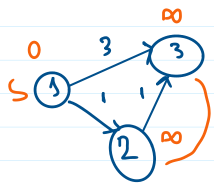
////check for a negative cycle
for $(u,v) \in E$ **do**
  if $d_v^{|V|-1} > d_u^{|V|-1} + w(u,v)$ **then**
    Print "Negative Cycle"
  end if
end for

τελευταία επανάληψη

Εκτελεί έναν τελευταίο επαναυπολογισμό των αποστάσεων κάθε κορυφής.



Παράδειγμα



$k=1$

Ακμή (1,3): $d_3^1 \leftarrow d_3^0 = \infty$, $d_3^1 > d_1^0 + w(1,3) \Rightarrow d_3^1 \leftarrow 3$

Ακμή (1,2): $d_2^1 \leftarrow d_2^0 = \infty$, $d_2^1 > d_1^0 + w(1,2) \Rightarrow d_2^1 \leftarrow 1$

Ακμή (2,3): $d_3^1 = 3$, $d_3^1 > d_2^0 + w(2,3)$ ✗

$k=2$

Ακμή (1,3): $d_3^2 \leftarrow d_3^1 = 3$, $d_3^2 > d_1^1 + w(1,3)$ ✗

Ακμή (1,2): $d_2^2 \leftarrow d_2^1 = 1$, $d_2^2 > d_1^1 + w(1,2)$ ✗

Ακμή (2,3): $d_3^2 \leftarrow d_3^1 = 3$, $d_3^2 > d_2^1 + w(2,3)$, $d_3^2 \leftarrow 2$

Διαφορές με Dijkstra

Σε κάθε επανάληψη k:

argmin $\{d_v : v \in V \setminus S\}$

- Ο Dijkstra επαναυπολογίζει τις αποστάσεις των γειτόνων τις κορυφής v^*, όπου μόλις έγινε μόνιμη.
- Ο Bellman-Ford επαναυπολογίζει τις αποστάσεις όλων των κόμβων (βάσει των αποστάσεων στην $k-1$ επανάληψη) χάρη στην extra δομή επανάληψης for $(u,v) \in E$. Άρα μεγαλύτερη πολ/τητα $O(|V| \cdot |E|)$

$|V|$ = πλήθος κόμβων (από for ($i=1$; $i \le |V|-1$; i++))

$|E|$ = πλήθος ακμών (από for $(u,v) \in E$)