2020-21 (Φεβρ)

Θ1 16

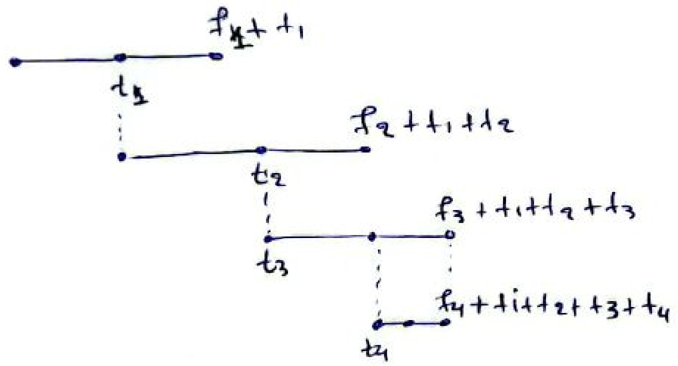
s ~~~~ t   L χλμ

(α) Ελέγχω αν $\xrightarrow{O(m)}$ υπάρχουν ακμές με $w(e) > L$. Αν ναι την αφαιρώ. Κάνω Dijkstra ελέγχοντας αν έφτασα στην t με αφετηρία την s. $O(m\log n)$

(β) • $O(m\log n)$ → Kruskal → max $w(e)$ του mst

• O(γραμμ.) ?

Θ2 (α) Ταξινομώ κατά φθίνουσα σειρά τις εργασίες ως προς $\frac{f_i}{t_i}$ ⇝ $i_1,\dots,i_n$ $\boxed{\text{όχι}}$ ~~άπληστο~~ ⇝ DP $\boxed{\text{όχι}}$



ΚΑΤΩ ΦΡΑΓΜΑ $\sum_{i=1}^{n} t_i$

(γ) ? τίποτα γιατί αλλάζουν στην πραγματικότητα τα t_i

Συμβολίζω $T^*(i)$: χρόνος (βέλτιστος) αν έχω τελειώσει i εργασίες

$T^*(n) =$ $\cancel{T^*(n-1) + \max_{1\le i\le n}\{f_i + \sum_{j=1}^{i} t_j\}}$?

Ταξινομώ κατά φθίνουσα σειρά τις εργασίες ως προς f_i ⇝ $i_1,\dots,i_n$ $O(n\log n)$

Εκτελώ τις εργασίες με τη σειρά αυτή

$T^*(n) =$ $\cancel{T^*(n-1) + f_{i_n} + \sum_{i=1}^{n} t_i \quad \sum_{i=1}^{n} t_i \quad \max_{1\le i\le n}\{f_i +}$ $\quad n \geq 1$

$\max_{1\le k\le n} \min\{f_{i_k} + \sum_{j=1}^{k} t_j, \sum_{j=1}^{k} t_j\} = \max\{T^*(n-1), \min\{f_{i_n} + \sum_{j=1}^{n} t_{i_j}, \sum_{j=1}^{n} t_j\}\} \to O(n)$

σύμφωνα με ταξινόμ.

$T^*(1) = t_1 + f_1$

ορθότητα

Βάση ✓ Επ. Β Έστω ότι ισχύει η αναδρ. σχέση για τις πρώτες $k \leq n-1$ εργασίες. Θ.δ.ο. ισχύει για k+1.

Υποθέτω προς άτοπο ότι η (k+1)-οστή f_j εργασία έπρεπε να γίνει αργότερα (αφού οι k πρώτες είναι σωστές). Άρα αν υποθέσω ότι έχω ανταλλάξει τις k+1, k+2 (ή μετά με επαγωγή αποδεικνύεται για περισσότερα swaps διαδοχικών)

k k+1 k+2

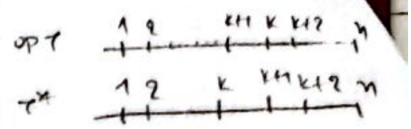
Συμβολίζω

OPT(k+1) τη βέλτιστη λύση

$\tau^*(k+1)$ τη λύση της αναδρ. εξίσ.

$$OPT(n) \overset{1}{<} \tau^*(n) = \max\left\{\tau^*(n-1)\ f_{in} + \sum_{j=1}^{n} t_{ij},\ \sum_{j=1}^{n} t_{ij}\right\}$$

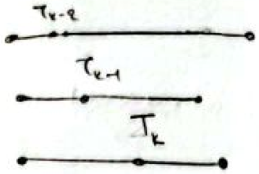
όλα τα άλλα είναι στη ίδια θέση



$$(\Pi_1)\quad \tau^*(n-1) \leq \tau^*(n) = f_{in} + \sum_{j=1}^{n} t_{ij}$$

$$OPT(n) = \max_{1\leq\lambda\leq n}\left\{\sum_{j=1}^{n} t_{ij},\ \underset{\lambda \neq k,k+1}{f_{i\lambda} + \sum_{j=1}^{\lambda} t_{ij}},\ f_{ik+1} + \sum_{j=1}^{k-1} t_{ij} + t_{ik+1},\ f_{ik} + \sum_{j=1}^{k+1} t_{ij}\right\}$$

$T_k = \sum_{i=1} t_{j_i}$



$$\max_{1\le k\le n} \{T_k + f_{ik}\} = \Big\{ \underbrace{\max_{1\le k\le n-1} \{T_k + f_{ik}\}}_{T^*(n-1)}, T_n + f_{in} \Big\}$$

$T^*(1) = t_{i_1} + f_{i_1}$ $\qquad T^*(k) = \max \{T^*(k-1), T_k + f_{ik}\}$

$OPT(n) < T^*(n)$

$\le \max_{j\in[n]} \Big\{ \underset{j\neq k,k+1}{T_j + f_{ij}}, \overbrace{T_{k-1} + t_{ik+1} + f_{ik+1}}^{C(k)}, \overbrace{T_{k+1} + f_{ik}}^{C(k+1)} \Big\}$

- Αν $OPT(n) = T_j + f_{ij}$ για $j \neq k,k+1 \Rightarrow$

$T^*(n) > T_j + f_{ij} \ge C(k), C(k+1)$

$\;\|$

$T_k + f_{ik}$ ή $T_{k+1} + f_{ik+1}$

$T^*(n) = T_k + f_{ik} > T_{k-1} + f_{ik+1}$

$\boxed{t_k + f_{ik} > f_{ik+1}}$

$T_k + f_{ik} > T_{k+1} + f_{ik} \Rightarrow$

$0 > t_{k+1}$ άτοπο

ομοίως αν $T^*(n) = T_{k+1} + f_{ik+1}$

$T_{k+1} + f_{ik+1} > T_{k-1} + t_{ik+1} + f_{ik+1}$

$t_k > 0$

$T_{k+1} + f_{ik+1} > T_{k+1} + f_{ik}$

$f_{ik+1} > f_{ik}$ άτοπο από φθίνουσα σειρά f_{ij}

- Αν $OPT(n) = C(k)$ ή $C(k+1) \ge T_j + f_{ij} \ \forall\, j \neq k,k+1$

Αν $T^*(n) = T_j + f_{ij}$ για $j \neq k,k+1$ θα έχω άτοπο άρα

$T^*(n) = T_k + f_{ik}$ ή $T_{k+1} + f_{ik+1}$

Αν $T^*(n) = T_k + f_{ik} > \left\{ \begin{array}{l} C(k) = T_{k-1} + t_{ik+1} + f_{ik+1} \\ \text{και} \\ C(k+1) = T_{k+1} + f_{ik} \end{array} \right\} \Rightarrow$

$\Rightarrow \left\{ \begin{array}{l} t_k > t_{ik+1} + f_{ik+1} \\ 0 > t_{k+1} \text{ άτοπο} \end{array} \right\}$

Αν $T^*(n) = T_{k+1} + f_{ik+1} > \left\{ \begin{array}{l} C(k) = T_{k-1} + t_{ik+1} + f_{ik+1} \\ \text{και} \\ C(k+1) = T_{k+1} + f_{ik} \end{array} \right\} \Rightarrow f_{ik+1} > f_{ik}$ άτοπο από φθ. σειρά

(Θ3) (α)

$M = 100$

(β) πίνακας

	1	2	3	4
Α/0	1	2	1	2
Θ/1	2	1	2	1

ΚΟΣΤΟΣ ΑΠΛΗΣΤΟ : $1+100+1+100+1+100+1 = 304$

ΕΛ. ΚΟΣΤΟΣ : $1+2+1+2 = 6$

(β) $c(n,k)$: ελάχιστο cost για k πρώτους μήνες $k \le n-1$ αν πριν είχε επιλέξει την πόλη $n \in \{0, 1\}$ (0 → A, 1 → Θ)

(*) $c(n,k+1) = \min \begin{cases} c(n,k) + D[n,k+1], \\ c(n',k) + M + D[n,k+1] \end{cases}$ με $n' = \begin{cases} 1 & \text{αν } n=2 \\ 2 & \text{αν } n=1 \end{cases}$

$\boxed{c(n,1) = D[n,1] \quad n \in \{1,2\}}$

ελάχ. κόστος $c(n,2) = \min \begin{cases} c(n,1) + D[n,2], \\ c[n',1] + M + D[n,k+1] \end{cases}$ $n' = \begin{cases} 1 & \text{αν } n=2 \\ 2 & \text{αλλιώς} \end{cases}$

$\boxed{c^*(n) = \min\{C(0,n), C(1,n)\}}$

$\boxed{\text{OPT\_COS}(p, i)}$ # $p = 1$ ή 0

```
p' = p + 1
if i = 1
    c(p, i) ← D[n,1]
else
    c(p,i) ← c(p,i-1) + D[p, i]
    if c(p', i) + M + D[p, i] > c(p,i)
        c(p,i) ← c(p', i-1) + M + D[p, i]
    endif
endif
```

Τρέχω OPT_COS (p, n) για p=0 και για p=1 και παίρνω το ελάχιστο.

(δ) ορθότ. + υπολ. πολ.

(δ) γενίκευση για $m \ge 2$ πόλεις με $C_k(i)$: ελάχιστο κόστος για i-οστή πόλη τον k-οστό μήνα

$M(i,j)$: κόστος μετακίνησης από i → j πόλη

πολυωνυμικό.

ΟΡΘΟΤΗΤΑ 1) Το πρόβλ. παρουσιάζει βέλτ. υποδομή 2) Η (*) υπολογίζει σωστά το ελ. κόστος 3) Ο OPT_COS υπολογ. σωστά την (*)

ΧΡΟΝ ΠΟΛ. ↔ $O(n)$

~~$T(1) = 1$~~

~~$T(n) = T(n-1) + O(1)$~~

με επαγωγή στο # μηνών k

Έστω ότι $\boxed{k=1}$ $\min \begin{cases} C[0,1] = D[0,1] \\ C[1,1] = D[1,1] \end{cases} = \text{MIN COST}$ ✓

Αν ισχύει η αναδρ. σχέση για λ μήνες θ.δ.ο ισχύει για λ+1

$$C^*(\lambda) = \min \{ C(0,\lambda), C(1,\lambda) \}$$

Αν τον (λ+1)-οστό μήνα είμαι στην πόλη π μπορεί να συνέβη 1 από τα 2

- να άλλαξα πόλη για το μήνα → $C(\pi,\lambda+1) = C(\pi',\lambda) + M + D[\pi,\lambda+1]$
- να έμεινα στην ίδια → $C(\pi,\lambda+1) = C(\pi,\lambda) + D[\pi,\lambda+1]$

Άρα διαλέγω το $\min_{\pi\in\{0,1\}} \{C(\pi,\lambda+1)\}$ ✓

3) Βάση ✓

Ε.Β το else υπολογίζει το min κ αποθηκεύει το αποτέλεσμα σ' έναν 2×n πίνακα (για 1 πόλη κάθε φορά)

(δ) γενίκευση m×n πίνακας

$C(\pi,1) = D(\pi,1) \quad \pi\in[m]$

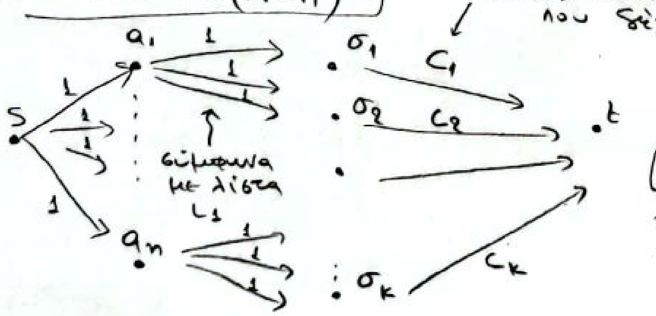
m αθροίσματα $\forall \pi$ $\forall \lambda$ $\rightsquigarrow C(\pi,\lambda+1) = \min_{\substack{\pi'\neq\pi \\ \pi'\in[m]}} \begin{cases} C(\pi,\kappa) + C_{\kappa+1}(\pi) & \text{ίδια πόλη} \\ C(\pi',\kappa) + M[\pi,\pi'] + C_{\kappa+1}(\pi') \end{cases}$

$$C^*(\lambda+1) = \min_{\pi\in[m]} C(\pi,\lambda+1)$$

χρον. πολ. $O(m^2 \cdot n)$

Θ4

G



$\boxed{C=n}$

$m \leq n + n\cdot k + k$

Ford-Fulkerson: Αν βρω μια μέγιστη ροή

Το πρόβλημα (Π) με τα αυτοκίνητα έχει λύση αν και τα n αυτοκίνητα βρουν συνεργείο σύμφωνα με τους περιορισμούς

Ισχυρισμός

Το G έχει ροή όγκου n αν το Π έχει λύση

(⇒) n μονάδες φεύγουν από s ⇒ κάθε αυτοκίνητο στέλνει 1 μονάδα σε 1 συνεργείο

1) Βέλτιστη υποδομή

Αν έχω $C^*(\lambda)$ το ελάχ. κόστος για λ μήνες κ (προς άτοπο) υπάρχει διάστημα $i,\dots,j$ μηνών $1\leq i\leq j\leq\lambda$ τ.ω το συνολ. κόστος για αυτούς να μην είναι ελάχιστο, τότε:

έχω επιλογές πόλεων $a_i \dots a_j$ διαφορετ. από αυτές της βελτ. λύσης (το υπόλ. ίδιο) $b_i \dots b_j$

(Π1) $a_i = b_i$ ή $a_j = b_j$ παίρνω υποπρόβλημα με διαφορετικά και τα 2 άκρα → (Π2)

b_{i-1} a_i a_j b_{j+1} (Π2) $a_i \neq b_i$ κ $a_j \neq b_j$

b_{i-1} b_i b_j b_{j+1}

(α) Αν $a_i = b_{i-1} \Rightarrow b_i \neq b_{i-1}$

(β) Αν $a_j = b_{j+1} \Rightarrow b_j \neq b_{j+1}$

Αν ισχύουν (α) + (β)

συνολ. κόστος = κόστος$(b_1,\dots,b_{i-1})$ + κόστος$(a_i,\dots,a_j)$ + κόστος$(b_{j+1},\dots,b_\lambda)$ $< C^*(\lambda)$ άτοπο

Αν έχω μόνο (α) $a_j \neq b_{j+1} \overset{2\text{επιλ.}}{\Rightarrow} b_j = b_{j+1}$

Άρα συνολ. κόστος = κόστος$(b_1,\dots,b_{i-1})$ + κόστος$(a_i,\dots,a_j)$ + M + κόστ$(b_{j+1},\dots,b_\lambda)$ < κόστος$(b_1,\dots,b_{i-1})$ + M + κόστος$(b_i,\dots,b_j)$ + κόστος$(b_{j+1},\dots,b_\lambda)$ άτοπο

ομοίως για μόνο β.

λόγω των περιορισμών χωρητικότητας το πολύ c_i μονάδες ροής φεύγουν από κάθε σ_i (άρα το πολύ c_i αυτοκίνητα εξυπηρετεί το σ_i)

Το πρόβλημα έχει λύση, σύμφωνα με τους περιορισμούς (+ μόνο $\sigma_j \in L_i$ δέχεται ροή από a_i)

($\Leftarrow$)

Το πρόβλημα έχει λύση άρα

- <u>κάθε</u> αυτοκίνητο a_i βρίσκει $\sigma_j \in L_i$ για επισκευή
- κάθε σ_j συνεργείο εξυπηρετεί το πολύ c_j αυτοκίνητα

$\Rightarrow$
- Στέλνω n μονάδες ροής από s προς τα a_i
- Στέλνω από κάθε a_i 1 μονάδα ροής προς το σ_j που ~~θα~~ το εξυπηρετεί
- οι μονάδες ροής από κάθε σ_j είναι $\leq c_j$

<u>χρόνος</u> $O(m^2 \log_2 C)$ με $C = n$, $m \leq n + k + n \cdot k$

$\downarrow$

$O(n^2 k^2 \log_2 n)$

Θε

(5β) 3DM $\longrightarrow$ 5β

$\langle A\times B\times\Gamma, F\rangle \longmapsto \langle X, S, P, B, k\rangle$ $\quad X = A\cup B\cup \Gamma$

$$s(x)=\begin{cases} 1 & x\in A\\ 2 & x\in B \\ 3 & x\in\Gamma\end{cases}$$

$k = 3|A|$

θέλω $S\subseteq X$: $\sum_{x\in S} p(x) \geq k$, $\sum_{x\in S} s(x) = B$

$p(x) = 1 \quad \forall\, x\in X$

$B = 6|A|$

- $\langle A\times B\times\Gamma, F\rangle \in$ 3DM $\Rightarrow \exists\, H\subseteq F$ με $|H| = |A|$, ~~κάθε~~ καμία τριάδα στο H δεν έχει ίδια συντεταγμ. με άλλη τριάδα

$\Rightarrow C(H) = \{ \alpha, \beta, \gamma : (\alpha,\beta,\gamma)\in H\}$

$\sum_{x\in C(H)} s(x) = \sum_{(\alpha,\beta,\gamma)\in H} 6 = 6|A|$, $\quad \sum_{x\in C(H)} p(x) = 3|A|$

- $\langle A\times B\times\Gamma, F\rangle \notin$ 3DM $\Rightarrow \nexists\, H\subseteq F$ με $|H| = |A|$ & καμία τριάδα στο H δεν έχει ίδια συντεταγμ. με άλλη τριάδα

$\forall H$ $\Rightarrow |C(H)| < 3|A| \Rightarrow \sum_{x\in C(H)} p(x) < 3|A|$ & $\sum_{x\in C(H)} s(x) < 6|A|$

(5α) Βλ. διαφάνειες «reduction examples»

LONGEST PATH $\longrightarrow$ 5α

? αν δεν είναι συνεκτικό το G του L.P. $\rightarrow$ είσοδος στο L.P. πάντα συνεκτικό

? πως ξέρω ποια κορυφή να κάνω s

προσθέτω μια νέα και τις ενώνω όλες με αυτή

$\Rightarrow k = |V|+1$

L.P → 5α

$\langle G \rangle \mapsto \langle G', \frac{|V|}{4}+1 \rangle$

$\parallel$

(V,E)

G' : G όπου έχω μια κορυφή s που συνδέεται με όλες τις κορυφές του G

$\langle G \rangle \in L.P \Rightarrow$ G : συνεκτικό κ έχει μονοπάτι τουλ. $|V|/4$ μήκους

~~τοτ~~ $\Rightarrow$ G' : συνεκτικό κ έχει μονοπάτι μήκους τουλ $|V|/4$

$\Rightarrow$ έχω δέντρο συνδετικό με ρίζα την s, περιλαμβάνει το μονοπάτι P παραπάνω κ κλαδιά μήκους 1 από s για όλες κορυφές δεν ανήκουν στο P.

$\langle G \rangle \in L.P \Rightarrow$ (π_1) G όχι συνεκτικό $\Rightarrow$ δεν έχει συνεκτικό δέντρο

(π_2) G ~~...~~ συνεκτικό κ κάθε μονοπάτι έχει λιγότερο από $|V|/4$ μήκος $\Rightarrow$ κάθε κλάδι σε συνδετικό δέντρο του G' έχει μήκος $< \frac{|V|}{4}+1$