

Μέγιστη Ροή – Ελάχιστη Τομή

Δημήτρης Φωτάκης

Σχολή Ηλεκτρολόγων Μηχανικών
και Μηχανικών Υπολογιστών

Εθνικό Μετσόβιο Πολυτεχνείο



Δίκτυα και Ροές

□ **Δίκτυο** : κατευθυνόμενο γράφημα $G(V, E)$.

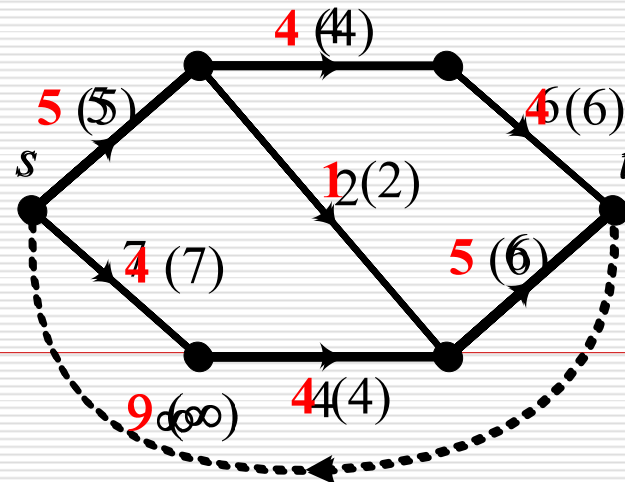
■ Πηγή s , προορισμός t , χωρητικότητα ακμής b_e .

□ $s - t$ ροή μεγέθους d : $f : E \mapsto \mathbb{R}_+$:

■ Χωρητικότητα: $\forall e \in E \ f_e \leq b_e$

■ Διατήρηση ροής: $\forall v \in V \ \sum_{e \in \text{in}(v)} f_e = \sum_{e \in \text{out}(v)} f_e$

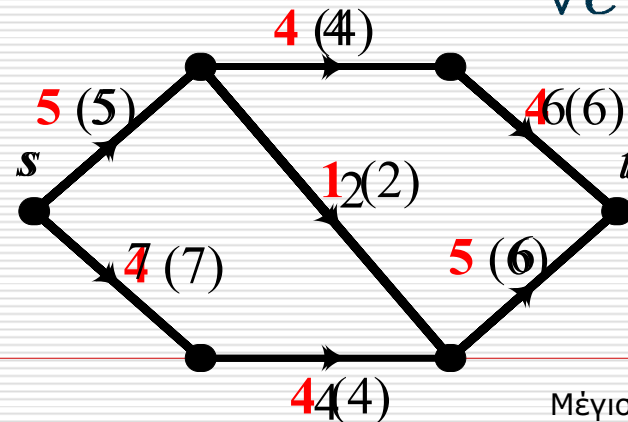
■ Μέγεθος: $f_{ts} = d$



Μέγιστη $s - t$ Ροή

- Πρόβλημα **Μέγιστης $s - t$ Ροής** (Max-Flow):
- Δεδομένου δικτύου $G(V, E, s, t, b)$
 - Υπολόγισε $s - t$ ροή με μέγιστη τιμή.

$$\begin{array}{ll} \max & f_{ts} \\ \text{s.t.} & f_e \leq b_e \quad \forall e \in E \\ & \sum_{e \in \text{in}(v)} f_e - \sum_{e \in \text{out}(v)} f_e \leq 0 \quad \forall v \in V \\ & f_e \geq 0 \quad \forall e \in E \end{array}$$



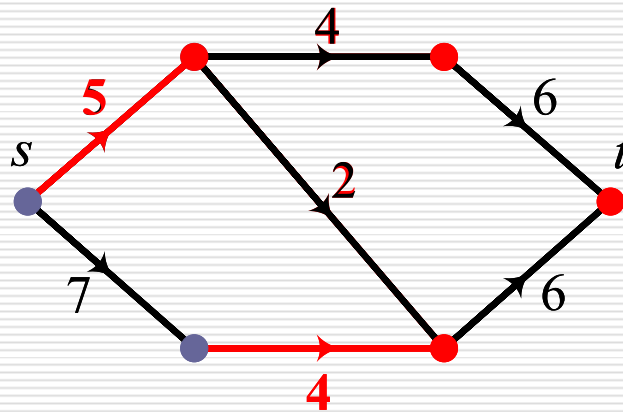
s - t Τομή

□ s - t τομή χωρητικότητας d :

■ Διαμέριση $(S, V \setminus S)$ με $s \in S$ και $t \in V \setminus S$.

■ Χωρητικότητα $b(S, V \setminus S) = \sum_{(u,v):u \in S, v \notin S} b_{uv} = d$

■ Ακμές χωρητικότητας d που χωρίζουν s από t .



Ελάχιστη $s-t$ Τομή

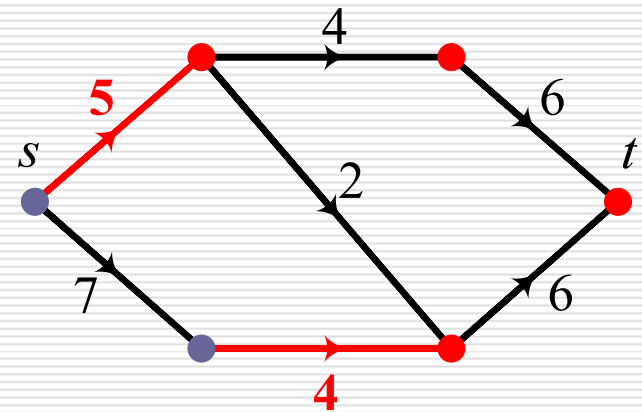
- Πρόβλημα **Ελάχιστης $s-t$ Τομής** (Min $s-t$ Cut):
 - Δεδομένου δικτύου $G(V, E, s, t, b)$
 - Υπολόγισε $s-t$ τομή με ελάχιστη χωρητικότητα.

$$\min \sum_{(u,v) \in E} d_{uv} b_{uv}$$

$$\text{s.t. } d_{uv} - p_u + p_v \geq 0 \quad \forall (u, v) \in E$$

$$p_s - p_t \geq 1$$

$$d_{uv}, p_v \geq 0$$



Ροές και Τομές

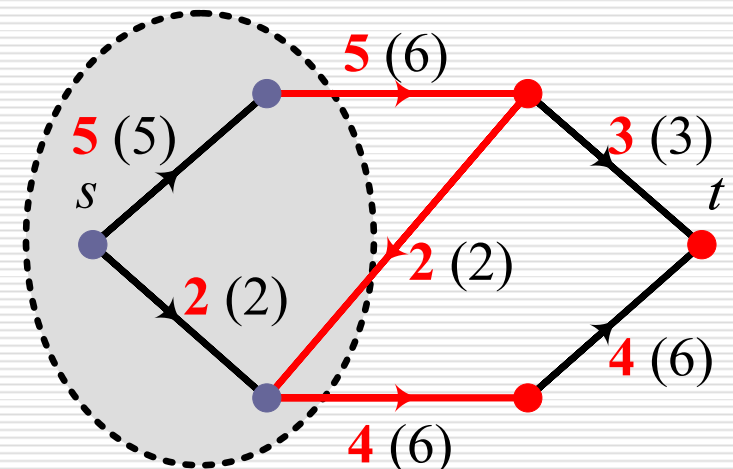
- Έστω ροή f και τομή $(S, V \setminus S)$.

$$f(S, V \setminus S) = \sum_{v \in S, u \notin S} f_{vu} - \sum_{v \in S, u \notin S} f_{uv}$$

- Κάθε $s - t$ ροή f και $s - t$ τομή $(S, V \setminus S)$:

$$f_{ts} = f(S, V \setminus S) \leq b(S, V \setminus S)$$

- **Μέγιστη $s - t$ ροή**
 \leq ελάχιστη $s - t$ τομή.

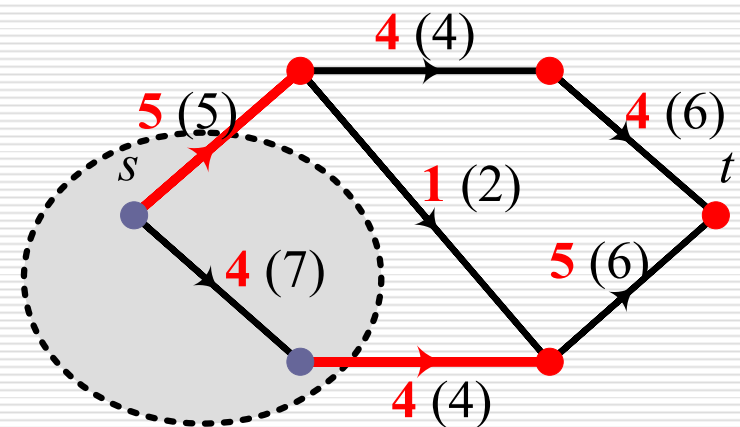


Μέγιστη Ροή και Ελάχιστη Τομή

□ Μέγιστη $s - t$ ροή = Ελάχιστη $s - t$ τομή!

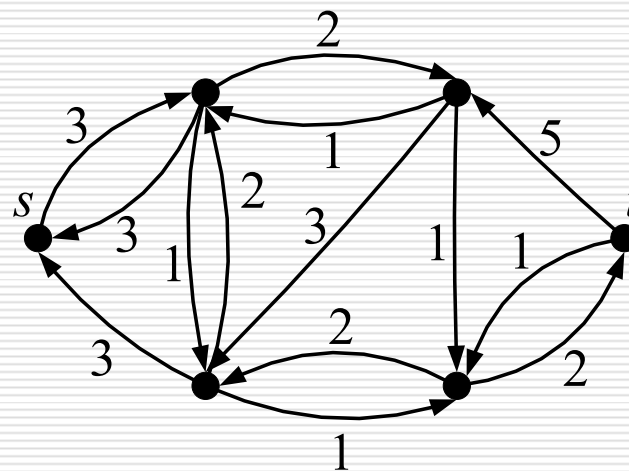
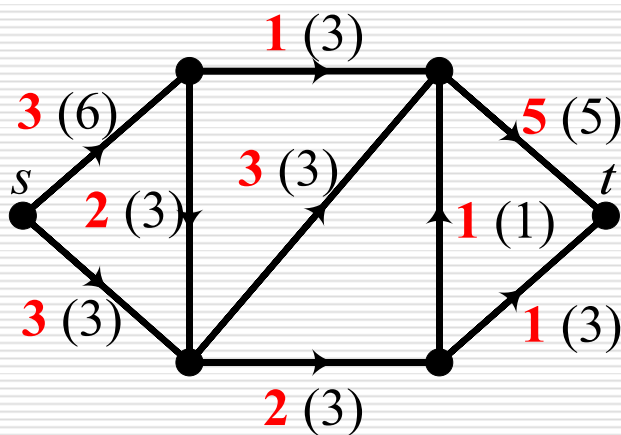
- Max-Flow – Min-Cut Θεώρημα.
- Ακμές ελάχιστης τομής
κορεσμένες σε μέγιστη ροή.

□ Μέγιστη ροή, ελάχιστη τομή: συνεκτικότητα / μεταφορική ικανότητα δικτύου.



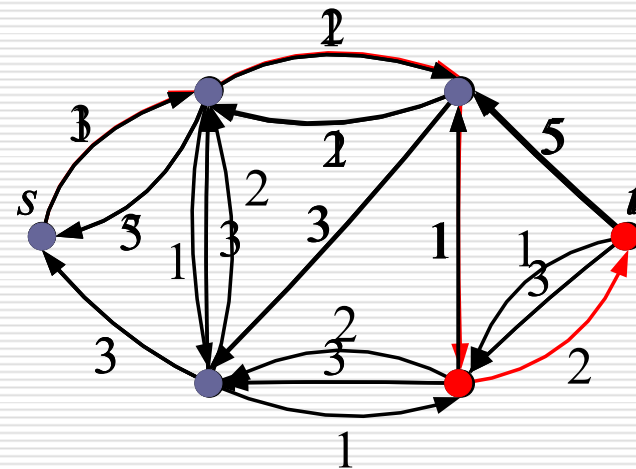
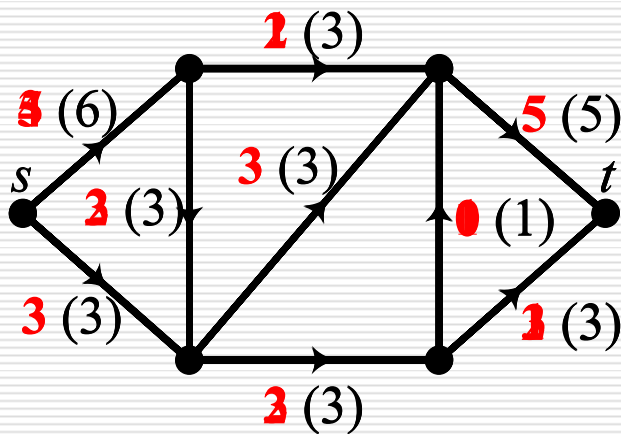
Υπολειμματικό Δίκτυο

- Δίκτυο $G(V, E, b)$ και ροή f .
- Υπολειμματικό δίκτυο $G_f(V, E_f, r_f)$:
 - Χωρητικότητα (μπρος-ακμές): $\forall(u, v) \in E \ r_{uv} = b_{uv} - f_{uv}$
 - Ροή (πίσω-ακμές): $\forall(u, v) \in E \ r_{vu} = f_{uv}$
- $s - t$ μονοπάτι στο υπολειμματικό: **επαυξητικό μονοπάτι.**



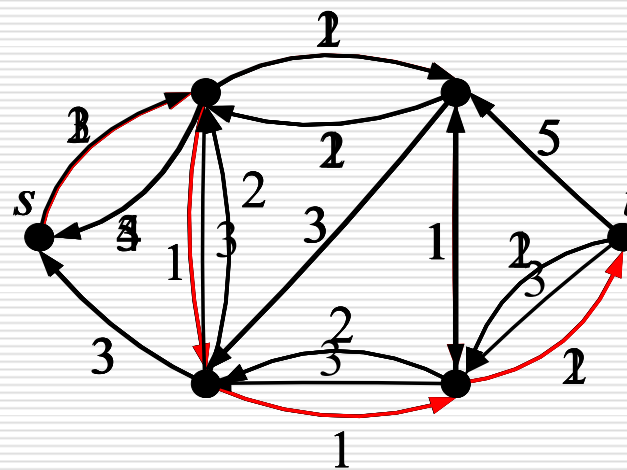
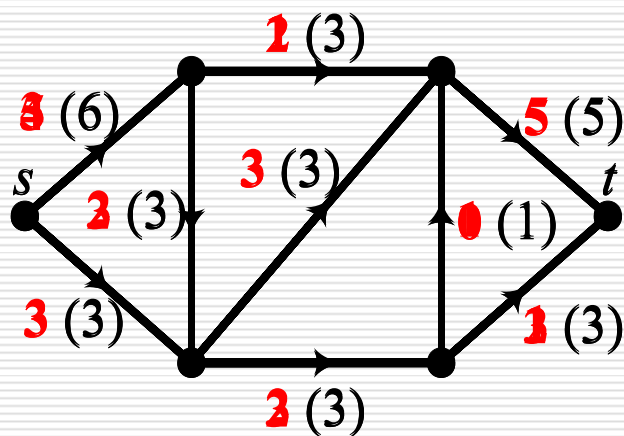
Χαρακτηρισμός Μέγιστης Ροής

- Μέγιστη ροή ανν όχι επαυξητικό μονοπάτι.
- Επαυξητικό μονοπάτι \Rightarrow αύξηση ροής \Rightarrow όχι μέγιστη ροή.
- Όχι επαυξητικό μονοπάτι :
 - Κορυφές προσπελάσιμες από s ορίζουν τομή χωρητικότητας ίσης με ροή.
 - Μέγιστη ροή και ελάχιστη τομή λόγω Θ . Max-Flow-Min-Cut!



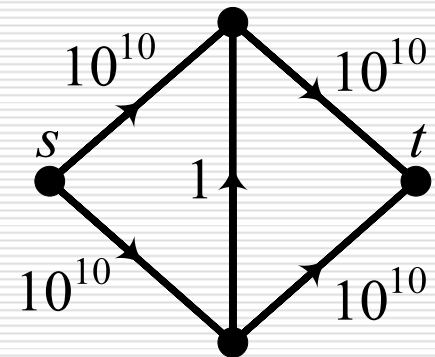
Αλγόριθμος Ford-Fulkerson

- Ενόσω **επαυξητικό μονοπ. p** στο υπολειμματικό,
 - Χωρητικότητα επαυξητικού $\delta \leftarrow \min_{e \in p} \{r_e\}$
 - **Αύξηση ροής κατά δ** στο p και ενημέρωση υπολειμματικού δικτύου.
- Επαυξητικό μονοπάτι με π.χ. **DFS, BFS.**
 - Επαύξηση σε χρόνο $O(m)$.



Χρόνος Εκτέλεσης

- **Ακέραιες χωρητικότητες $\leq U$:**
 - Επαύξηση αυξάνει ροή τουλάχιστον κατά 1.
 - Χρόνος εκτέλεσης $O(m^2 U)$.
- Δίκτυο με ακέραιες χωρητικότητες έχει **ακέραιη μέγιστη ροή**.
- Μπορεί εκθετικός χρόνος για **μεγάλες χωρητικότητες!**
- Μπορεί να **μην τερματίσει** για **άρρητες χωρητικότητες**.



Βελτιώσεις Edmonds-Karp

- Επαυξητικό μονοπάτι με **μέγιστη χωρητικότητα**.
 - $2m$ επαυξήσεις \Rightarrow **μέγιστη χωρητικότητα στο μισό**.
 - Αντί «μέγιστης», «αρκετά μεγάλης» χωρητικότητας:
 - Υπολειμματικό γράφημα μόνο με **χωρητικότητες $\geq \Delta$** .
 - Αν όχι επαυξητικό μονοπάτι, $\Delta \leftarrow \Delta / 2$.
 - Χρόνος εκτέλεσης $O(m^2 \log U)$.
- Επαυξητικό μονοπάτι **ελάχιστου μήκους** (ακμών).
 - Υπολογισμός με BFS σε χρόνο $O(m)$.
 - #επαυξήσεων $O(nm)$, χρόνος εκτέλεσης $O(nm^2)$.
 - Βελτίωση Dinic: υπολογισμός με BFS σε **χρόνο $O(n)$** !
 - Χρόνος εκτέλεσης $O(n^2m)$.
- Καλύτεροι αλγόριθμοι με **blocking-flow** και **push-relabel** τεχνικές έχουν χρόνους **$O(nm \log n)$** και **$O(n^3)$** αντιστ.

Μέγιστο Ταίριασμα

- Διμερές γράφημα: υπολογισμός **μέγιστου** αριθμού **ακμών χωρίς κοινά άκρα** (ταίριασμα).
- **Μέγιστη ροή**: πηγή s , προορισμός t , προσανατολισμός $s \rightarrow t$, **χωρητικότητα 1**.

