CHƯƠNG 6

Một số ứng dụng

GV: TS. Nguyễn Thị Phương Trâm

Tài liệu tham khảo

- Slide bài giảng ThS. Trần Quốc Việt
- Slide bài giảng ThS. Lê Phi Hùng
- Nguyễn Cam –Chu Đức Khánh, *Lý thuyết đồ thị* – NXB Trẻ Tp. HCM, 1998.
- Kenneth H. Rosen: *Discrete Mathematics and its Applications*, 7 Edition, McGraw Hill, 2010.

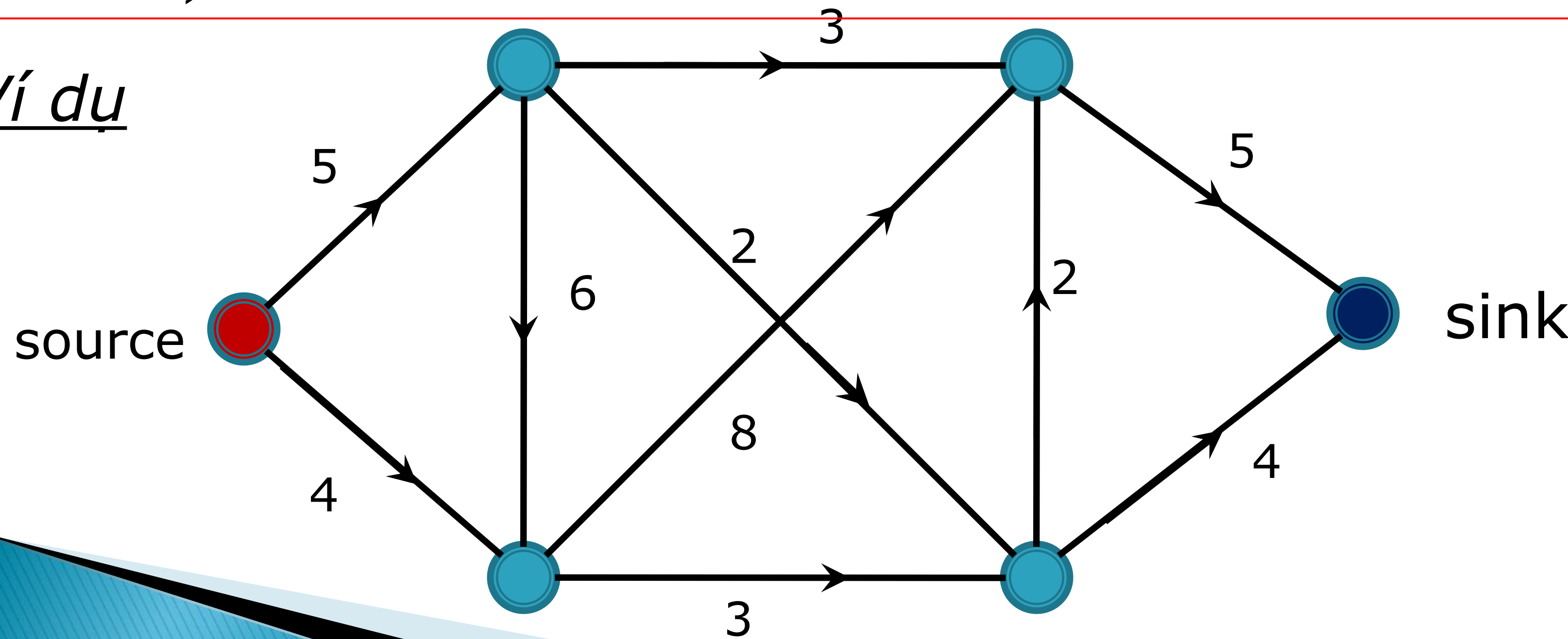
Bài toán luồng cực đại

(Max flow problem)

Định nghĩa

- Mạng (network) là một đồ thị có hướng có trọng số G = (V,E) trên đó ta chọn một đỉnh gọi là đỉnh phát (*source vertex*) và 1 đỉnh gọi là đỉnh thu (*sink vertex*).

Ví dụ

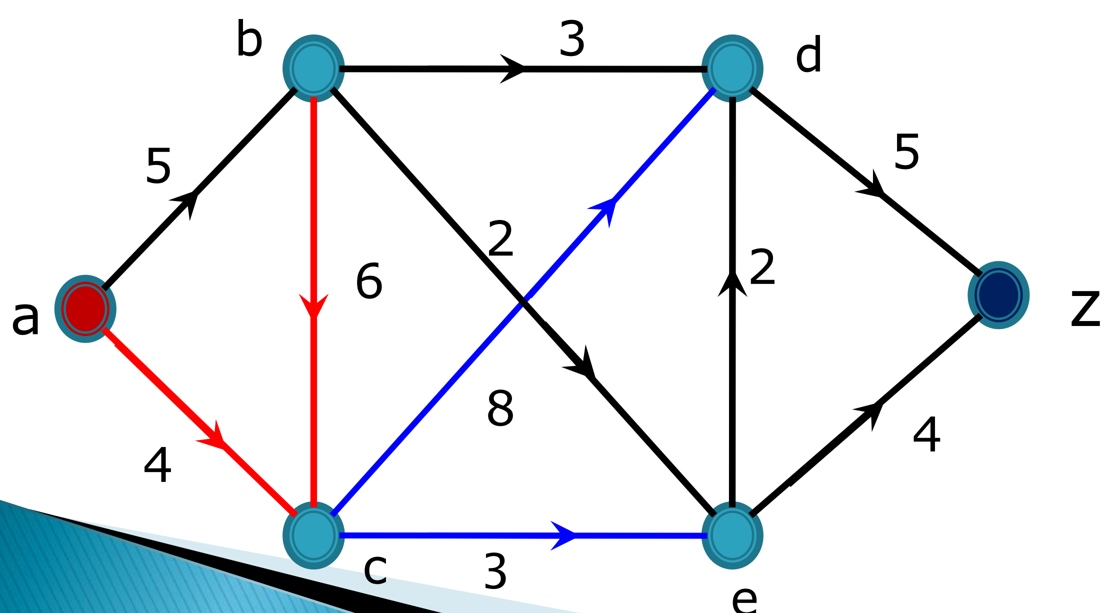


Định nghĩa

- Một mạng G = (V,E) với đỉnh phát là a, đỉnh thu là z, c(e) ∈ N là trọng số của cung e. Với mỗi đỉnh x, ta đặt:

  In(x) = {e ∈ E | e tới trong x}

  Out(x) = {e ∈ E | e tới ngoài x}



$In(c)=\{\overrightarrow{ac}, \overrightarrow{bc}\}$

$Out(c)=\{\overrightarrow{cd}, \overrightarrow{ce}\}$

Định nghĩa

❑ Một hàm tải (*flow function*) trên G được định nghĩa bởi ánh xạ:

$$\varphi: E \rightarrow N$$

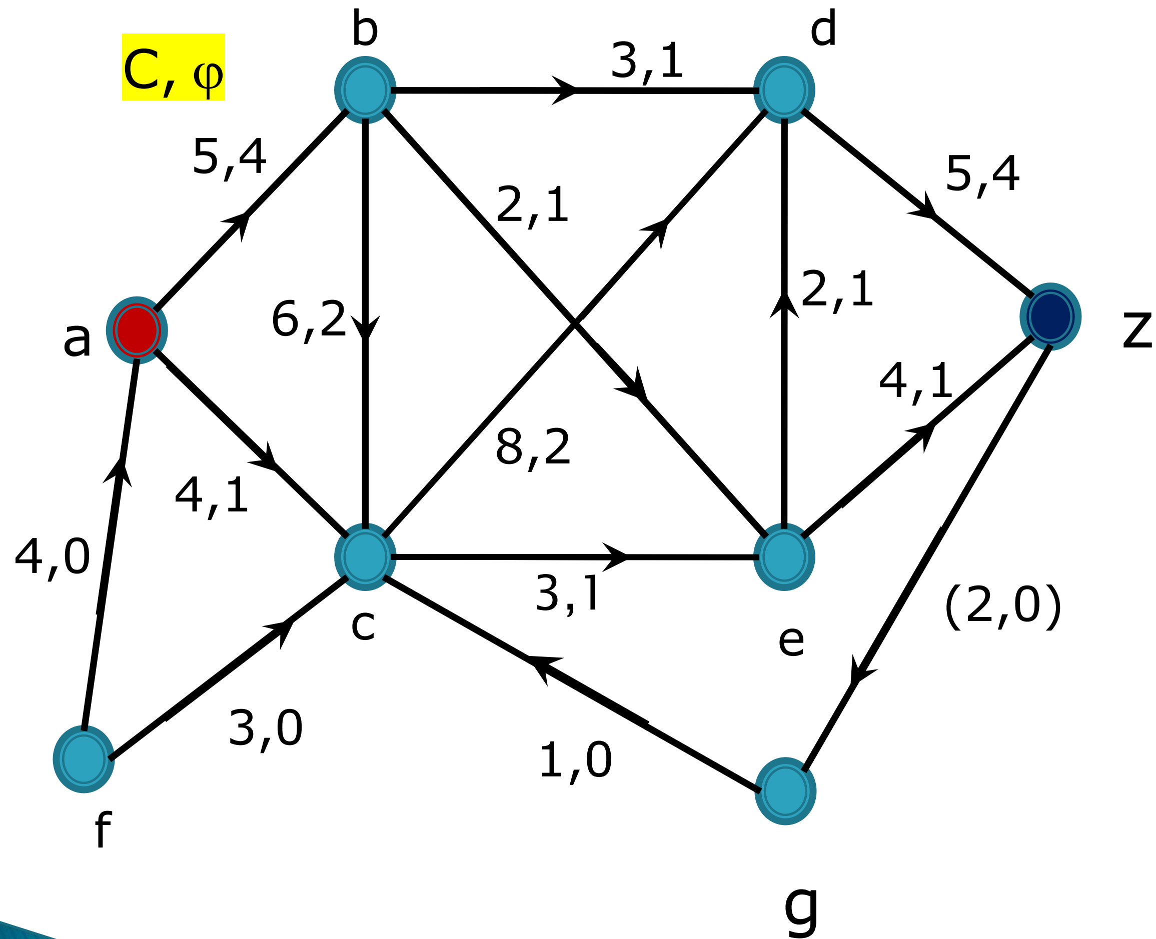
thỏa các điều kiện

(i) $\varphi(e) \le c(e), \forall e \in E$

(ii) $\varphi(e) = 0, \forall e \in In(a) \cup Out(z)$

(iii) $\sum_{e \in In(x)} \varphi(e) = \sum_{e \in Out(x)} \varphi(e), \forall x \in V \setminus \{a, z\}$

Ví dụ về hàm tải a:source, z:sink



$\varphi(\overrightarrow{fa}) = 0$

$\varphi(\overrightarrow{zg}) = 0$

$\varphi(\overrightarrow{ab}) = 4$

$\varphi(\overrightarrow{ac}) = 1$

$\varphi(\overrightarrow{fc}) = 0$

$\varphi(\overrightarrow{gc}) = 0$

$\varphi(\overrightarrow{bd}) = 1$

$\varphi(\overrightarrow{be}) = 1$

$\varphi(\overrightarrow{bc}) = 2$

$\varphi(\overrightarrow{cd}) = 2$

$\varphi(\overrightarrow{ce})=1$

$\varphi(\overrightarrow{dz}) = 4$

$\varphi(\overrightarrow{ez}) = 1$

$\varphi(\overrightarrow{ed}) = 1$

Định nghĩa

- Một phép cắt (*cut*) xác định bởi 1 tập hợp con P của V, ký hiệu $(P, \overline{P})$ là tập hợp:

$$(P, \overline{P}) = \{ \overrightarrow{xy} \mid x \in P \text{ và } y \in \overline{P} \}$$

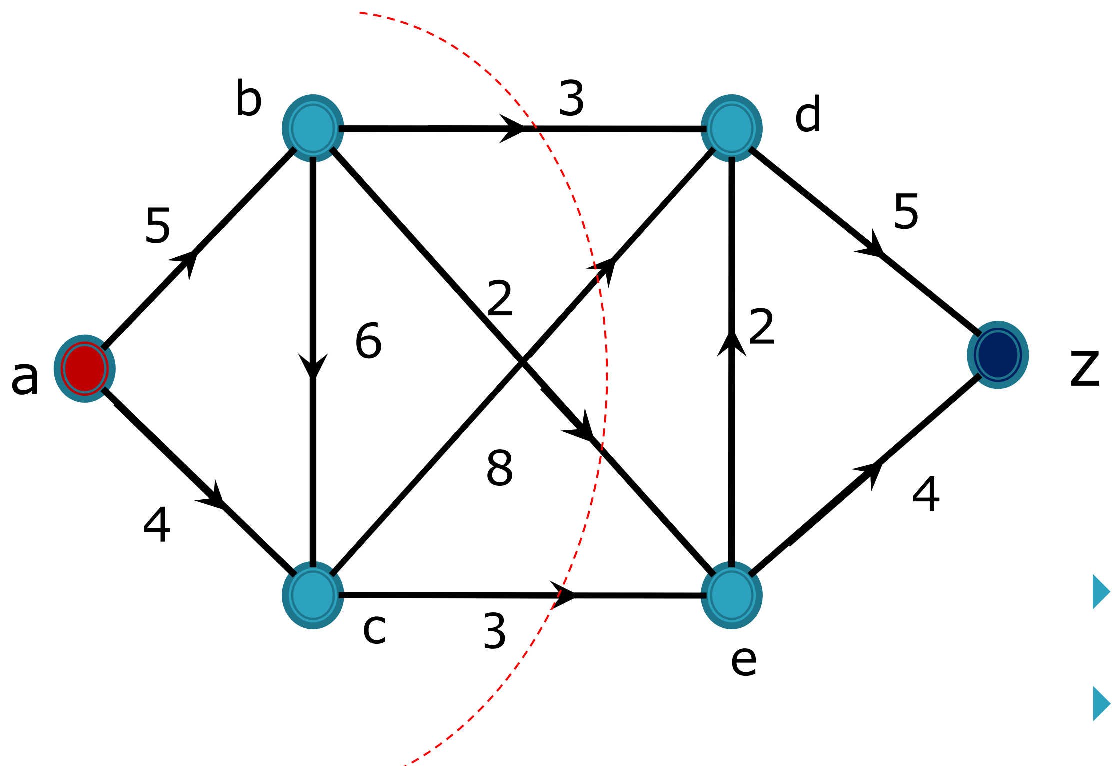
Trong đó $\overline{P} = V \setminus P$

- Phép cắt $(P, \overline{P})$ gọi là 1 phép cắt a-z nếu $a \in P$ và $z \in \overline{P}$

- Trọng số (*capacity*) của một phép cắt được định nghĩa là:

$$c(P, \overline{P}) = \sum_{e \in (P, \overline{P})} c(e)$$

Ví dụ:



- $P = \{a, b, c\}$
- $\overrightarrow{P} = \{d, e, z\}$
- $(P, \overline{P}) = \{\overrightarrow{bd}, \overrightarrow{be}, \overrightarrow{cd}, \overrightarrow{ce}\}$
- $c(P, \overline{P}) = 16$

Định lý 6.1

Gọi φ là một hàm tải trên mạng G và P ⊂ V\{a,z} thì:

$$\sum_{e \in (P,\overline{P})} \varphi(e) = \sum_{e \in (\overline{P},P)} \varphi(e)$$

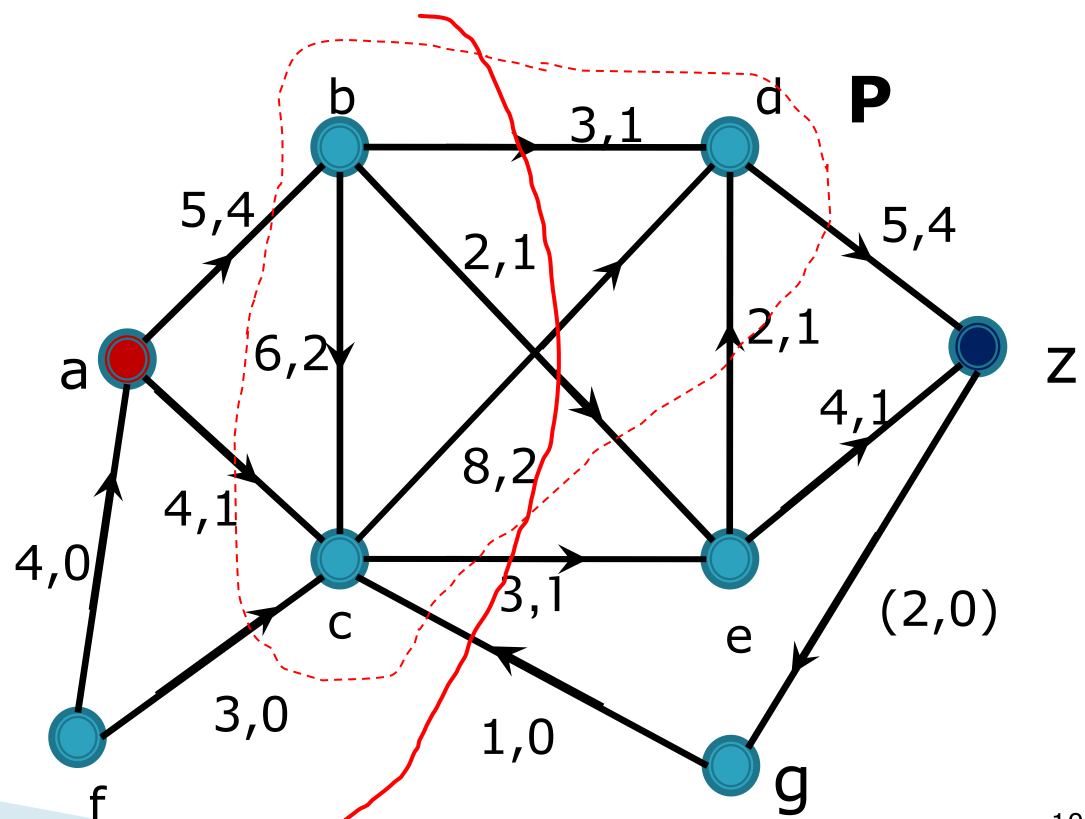
Ví dụ:

P={b,c,d}

$$\sum_{e \in (P,\overline{P})} \varphi(e) = ?$$

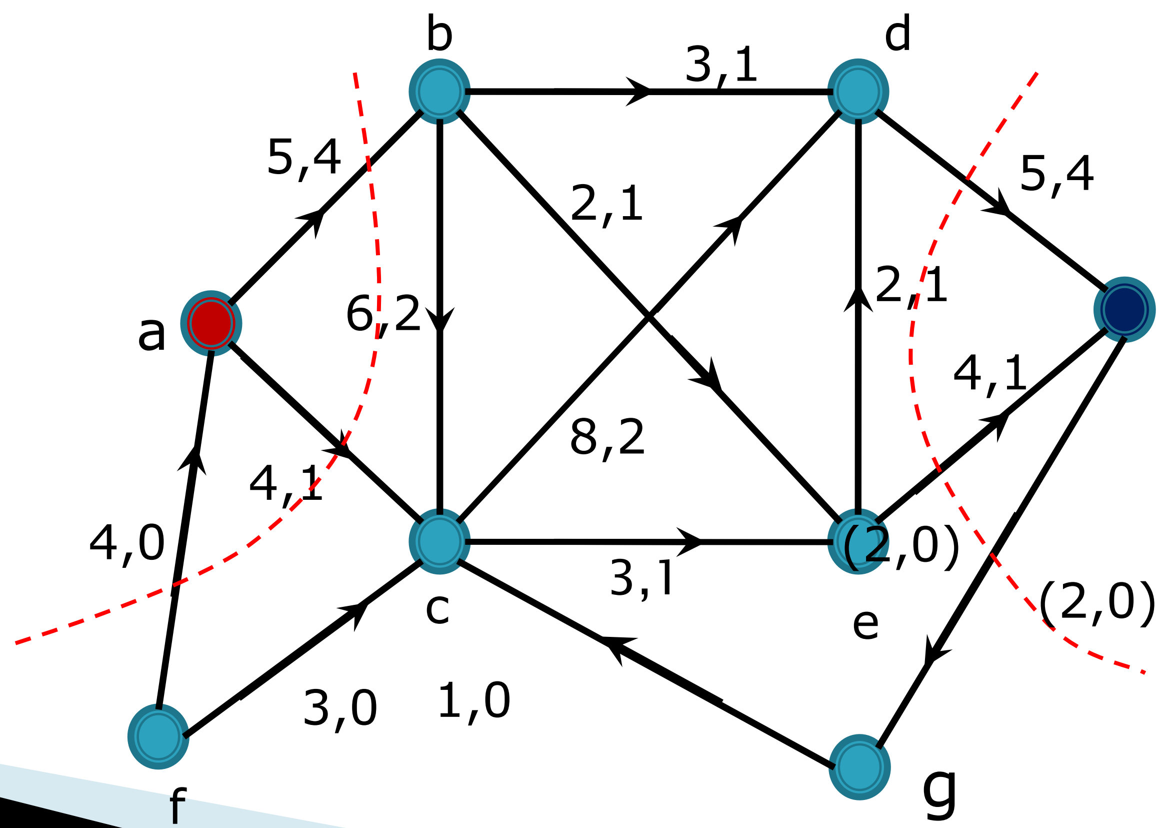
$$\sum_{e \in (\overline{P},P)} \varphi(e) = ?$$

$$\sum_{e \in (P,\overline{P})} \varphi(e) = \sum_{e \in (\overline{P},P)} \varphi(e)?$$



Định lý 6.2

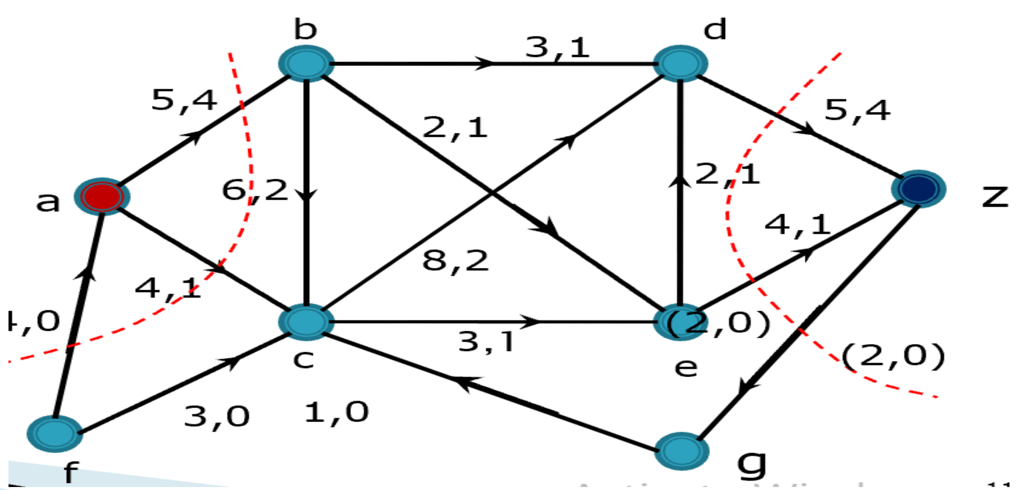
- Với mọi hàm tải φ trên mạng G, lượng tải khỏi a bằng lượng tải vào z, nghĩa là: $\sum\limits_{e \in Out(a)} \varphi(e) = \sum\limits_{e \in In(z)} \varphi(e) = |\varphi|$
- Kí hiệu: |φ| gọi là tải trọng của hàm tải φ



Chứng minh định lý 6.2

- Không mất tính tổng quát, giả sử G không chứa cung (a,z). Đặt P = V \ {a,z}, khi đó:

$$\sum_{e\in Out(a)} \varphi(e) = \sum_{e\in(\overline{P},P)} \varphi(e) = \sum_{e\in(P,\overline{P})} \varphi(e) = \sum_{e\in In(z)} \varphi(e)$$



Định lý 6.3

- Với mọi hàm tải φ và với mọi phép cắt a-z trong mạng G, ta có:

$$\boxed{|\varphi| \leq c(P, \overline{P})}$$

Hệ quả

- Với mọi hàm tải φ và mọi phép cắt a-z trong mạng G. $|\varphi| = c(P, \overline{P})$ nếu và chỉ nếu thỏa 2 điều kiện:

(i) $\forall e \in (\overline{P}, P), \varphi(e) = 0$

(ii) $\forall e \in (P, \overline{P}), \varphi(e) = c(e)$

- Khi $|\varphi| = c(P, \overline{P})$ thì φ là hàm tải có tải trọng lớn nhất và $(P, \overline{P})$ là phép cắt a-z có trọng số nhỏ nhất

Định nghĩa:

- Cho một mạng G, đỉnh phát a và đỉnh thu z, φ là một hàm tải trên G, $(P, \overline{P})$:một phép cặt a–z
- Một chuyền a–z K là một đường đi vô hướng nối a với z
- Độ lệch tải của cung e: s(e) = c(e)–φ(e)
- Xem:

$$\varphi_K(e) = \begin{cases} 0 & : e \notin K \\ 1 & : e \in K \text{ và có hướng từ a đến z} \\ -1 & : e \in K \text{ và e có hướng từ z đến a} \end{cases}$$

Thuật toán Ford-Fulkerson (Tìm một phép cắt a-z tối thiểu)

Input: Mạng G, đỉnh phát a và đỉnh thu z

Output: Tập P của phép cắt a-z tối thiểu $(P, \overline{P})$

Bắt đầu bằng 1 hàm tải φ bất kỳ trên G

1. Đánh dấu mọi đỉnh đều chưa xét, gán nhãn cho a là (-, $\Delta(a)$) với $\Delta(a)=\infty$. Đặt $p_0=a$.
2. Xét p_0.
   a. Cạnh $e=\overrightarrow{p_0q}$ với q chưa có nhãn và $s(e)>0$ thì gán nhãn cho q là $(p_0^+, \min(\Delta(p_0), s(e)))$
   b. Cạnh $e=\overrightarrow{qp_0}$ với q chưa có nhãn và $\varphi(e)>0$ thì gán nhãn cho q là $(p_0^-, \min(\Delta(p_0), \varphi(e)))$
3. Nếu đỉnh z đã được gán nhãn → 4, ngược lại → 5.
4. Xác định một dây chuyền (vô hướng) từ a đến z dựa vào thành phần thứ 1 của nhãn. Cập nhật lại φ như sau:

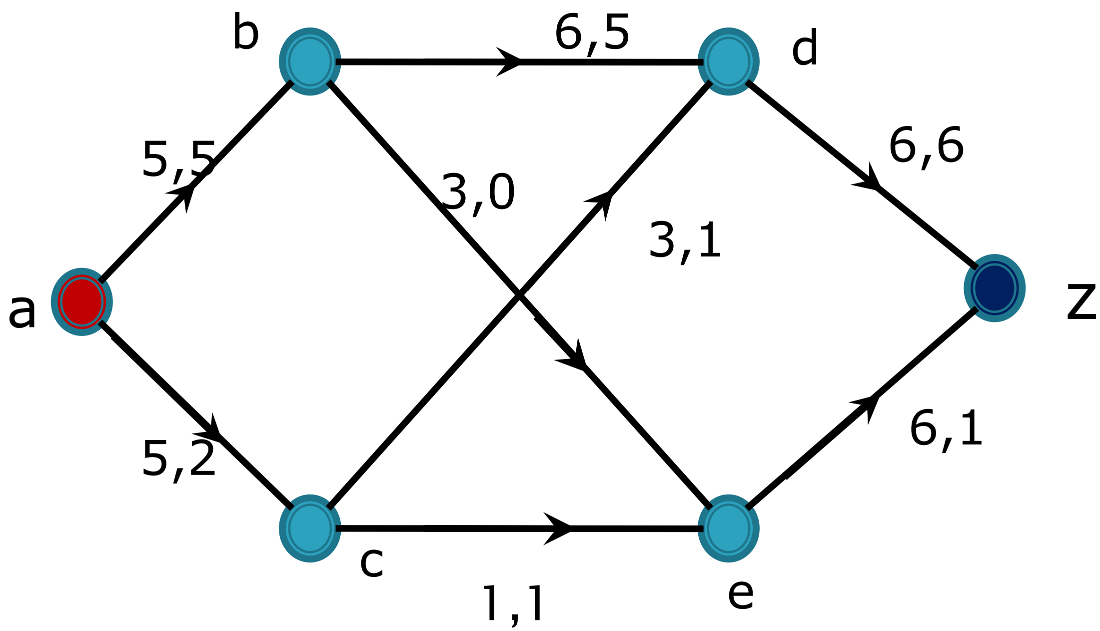
   $\varphi(e) = \varphi(e) + \Delta(z) \times \varphi_K(e)$. Về bước 1.
5. Tìm 1 đỉnh p đã có nhãn nhưng chưa xét. Nếu tồn tại p, đặt $p_0 = p$, → bước 2. Ngược lại dừng.

Thuật toán Ford-Fulkerson

- Sau khi thuật toán kết thúc. P là tập hợp các đỉnh đã có nhãn và đã xét.

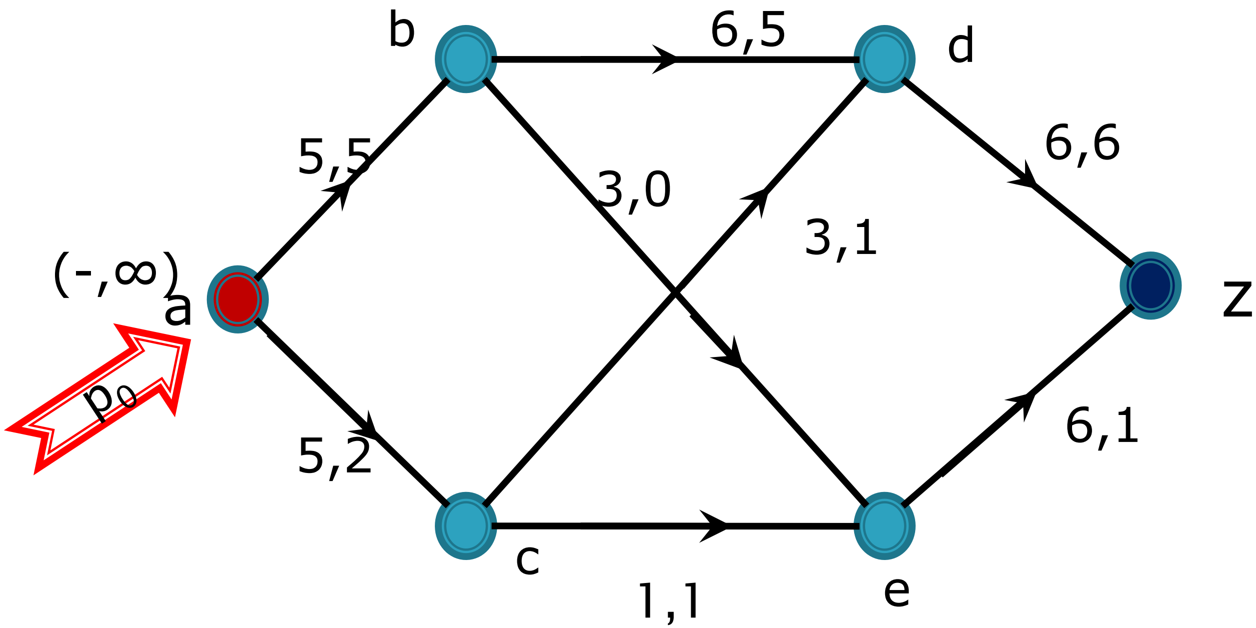
Ví dụ: Tìm một hàm tại cực đại trên mạng G

➢G với hàm tải ban đầu:



☞Lặp lần 1:

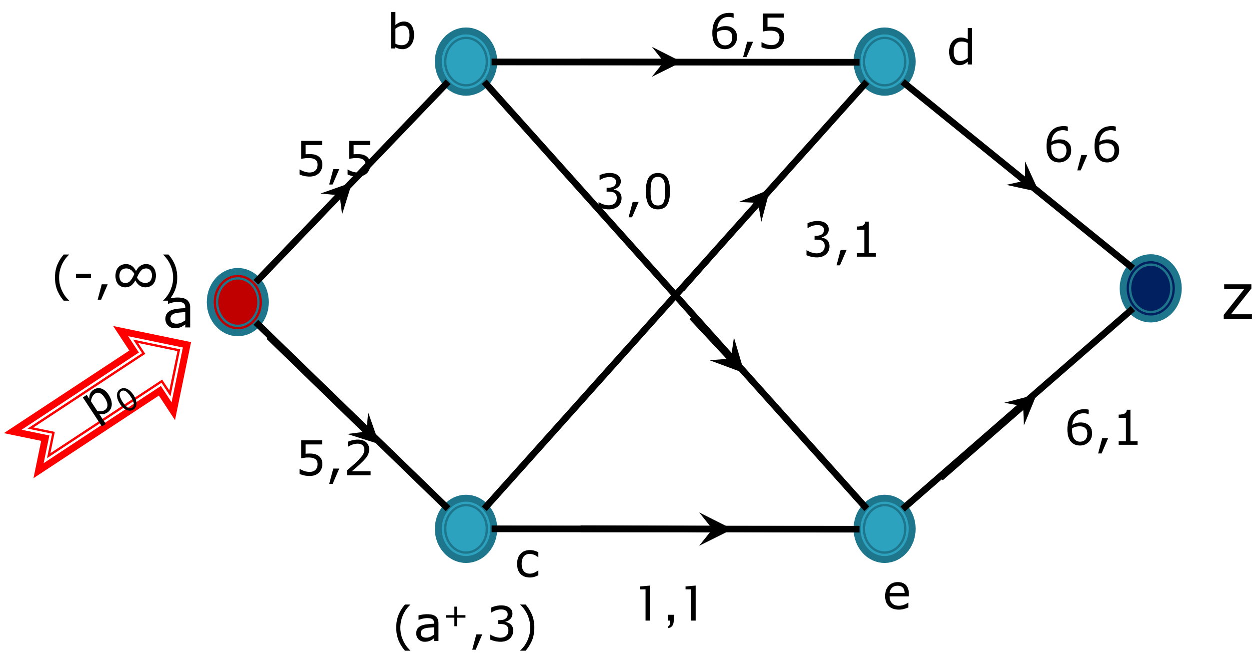
- Gán nhãn cho đỉnh a là (-,Δ(a)), với Δ(a)=∞
- Đặt p_0=a



Xét các đỉnh kề với p_0:

➢Cạnh $e_1=(\overrightarrow{a,b})$ có $s(e_1)=0$ nên không xét

➢Cạnh $e_2=(\overrightarrow{a,c})$ có $s(e_2)=3>0$ nên gán nhãn cho đỉnh c là: **$(a^+,\min\{\Delta(p_0),s(e_2)\}) =(a^+,3)$**

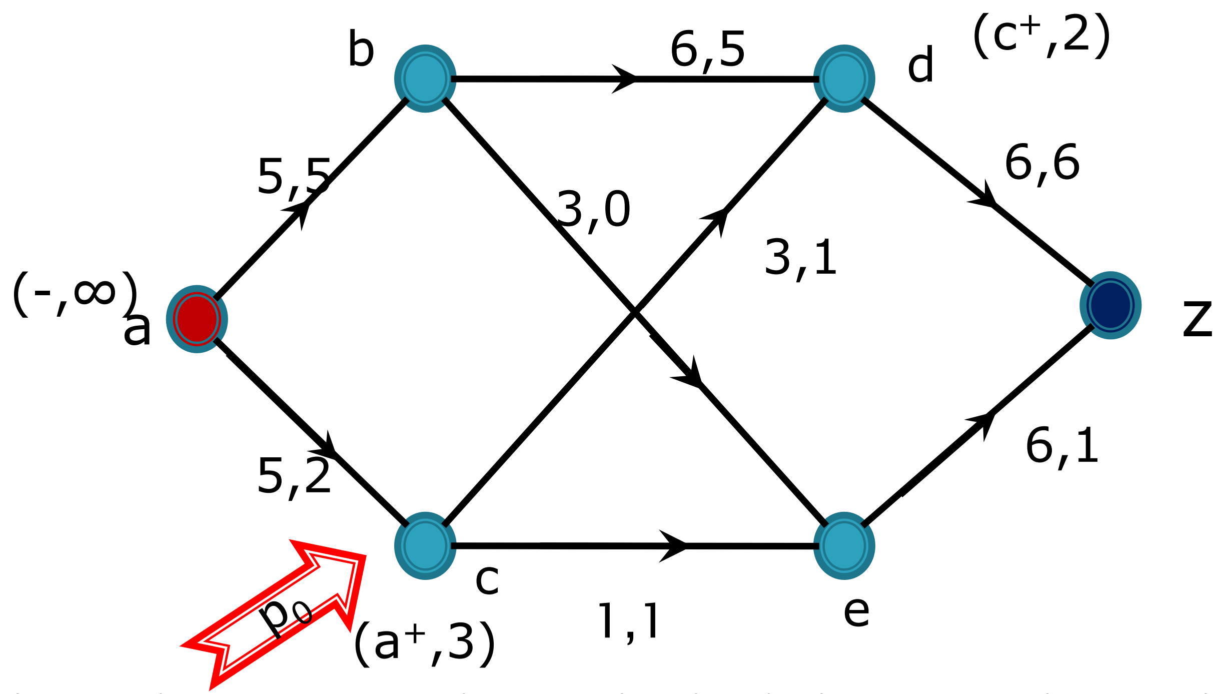


➢Đỉnh z chưa được gán nhãn, đỉnh c đã gán nhãn nhưng chưa xét, đặt $p_0=c$

Xét các đỉnh kề với p_0:

➢Cạnh $e_3=(\overrightarrow{c,e})$ có $s(e_3)=0$ nên không xét

➢Cạnh $e_4=(\overrightarrow{c,d})$ có $s(e_4)=2>0$ nên gán nhãn cho đỉnh d là: **$(c^+,\min\{\Delta(p_0),s(e_4)\})=(c^+,2)$**

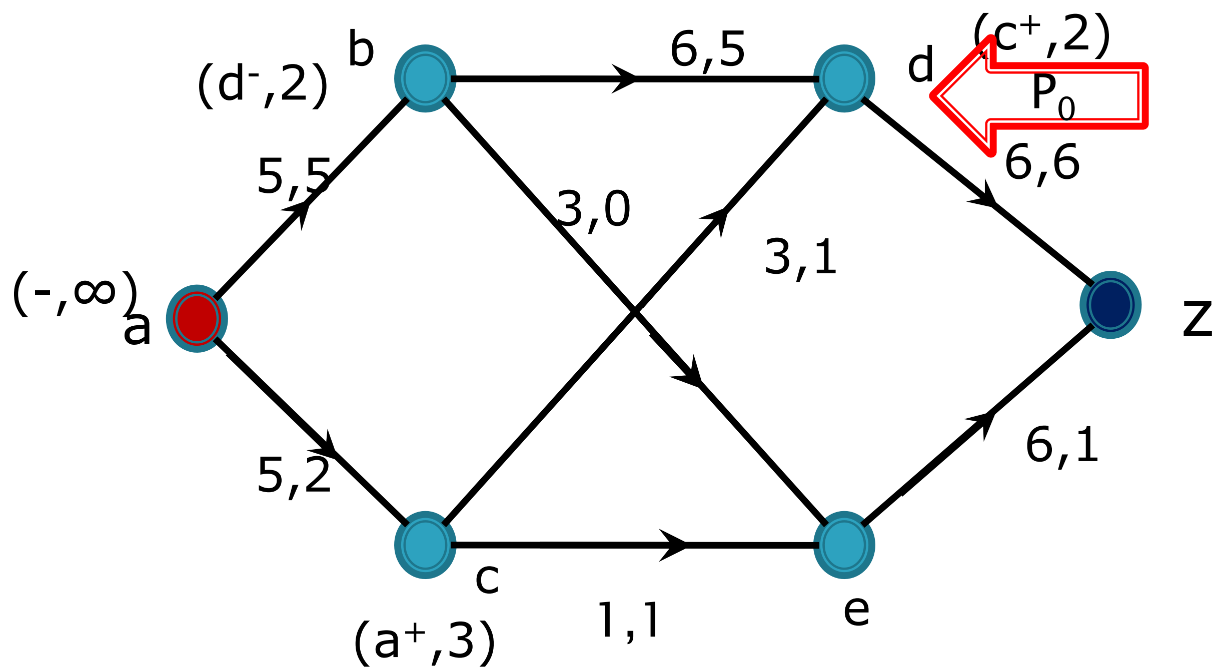


➢Đỉnh z chưa được gán nhãn, đỉnh d đã gán nhãn nhưng chưa xét, đặt $p_0=d$

$p_0 = d$:

➢Cạnh $e_5=(\overrightarrow{d,z})$ có $s(e_5)=0$ nên không xét

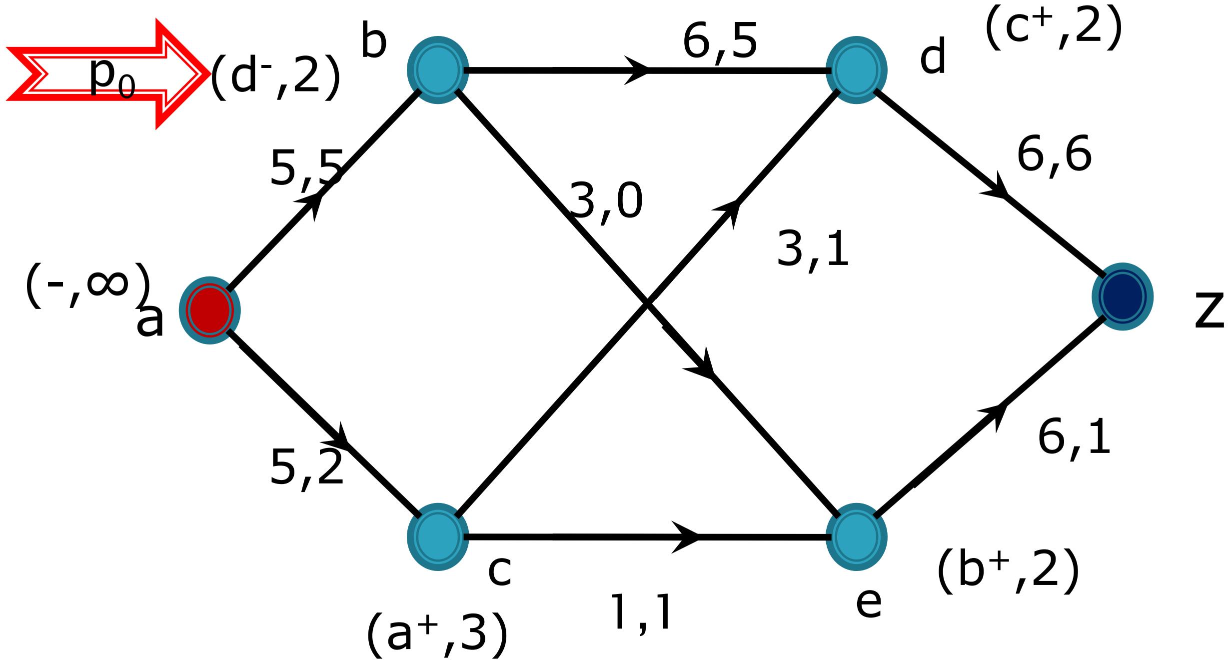
➢Cạnh $e_6=(\overrightarrow{b,d})$ có $\varphi(e_6)=6>0$ nên gán nhãn cho đỉnh b là: $\mathbf{(d^-,\min\{\Delta(p_0), \varphi(e_6)\}) =(d^-,2)}$



➢Đỉnh z chưa được gán nhãn, đỉnh b đã gán nhãn nhưng chưa xét, đặt $p_0=b$

p_0=b:

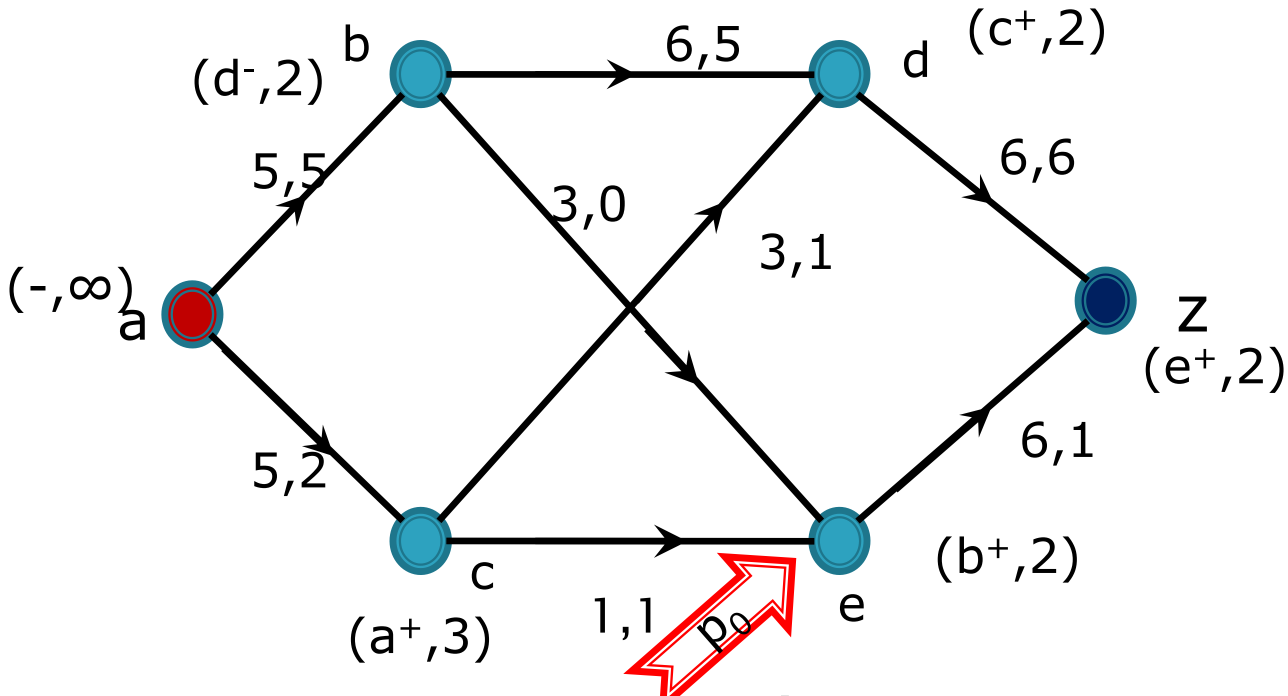
➢Cạnh $e_7=(\overrightarrow{b,e})$ có $s(e_7)=3>0$ nên gán nhãn cho đỉnh e là: **$(b^+,\min\{\Delta(p_0), s(e_7)\}) = (b^+,2)$**



➢Đỉnh z chưa được gán nhãn, đỉnh e đã gán nhãn nhưng chưa xét, đặt p_0=e

p_0=e:

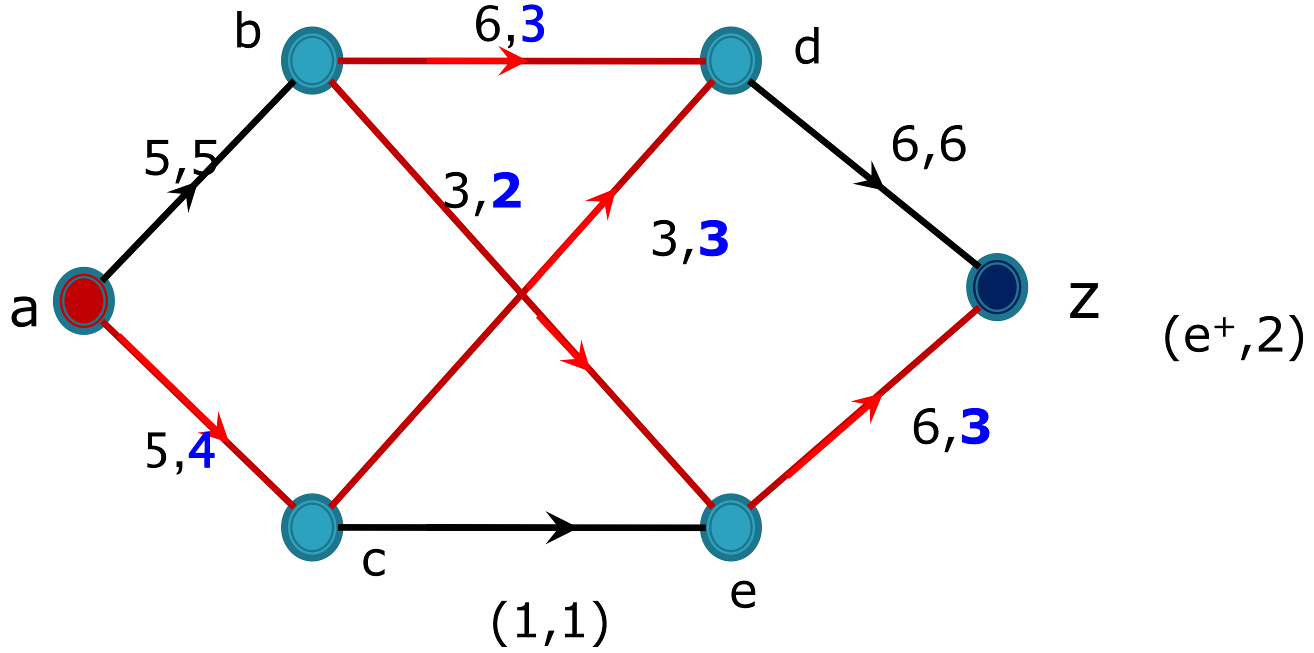
➢Cạnh $e_8=\overrightarrow{(e,z)}$ có $s(e_8)=5>0$ nên gán nhãn cho đỉnh z là: **$(e^+, \min\{\Delta(p_0), s(e_8)\}) = (e^+,2)$**



➢Đỉnh z đã được gán nhãn, tìm chuyền a-z K: acdbez

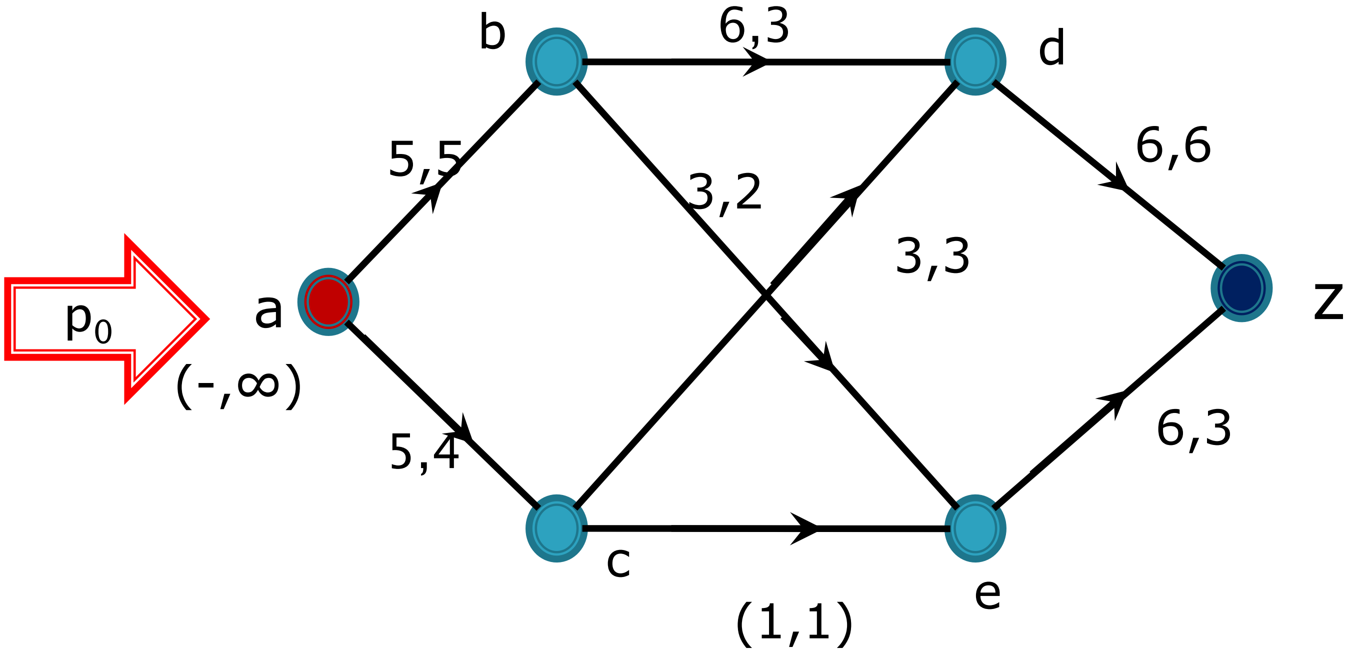
➢Cập nhật lại hàm tải: $\varphi = \varphi + \Delta(z)\varphi_K$

$$\varphi_K(e) = \begin{cases} 0 & : e \notin K \\ 1 & : e \in K \text{ và có hướng từ a đến z} \\ -1 & : e \in K \text{ và e có hướng từ z đến a} \end{cases}$$



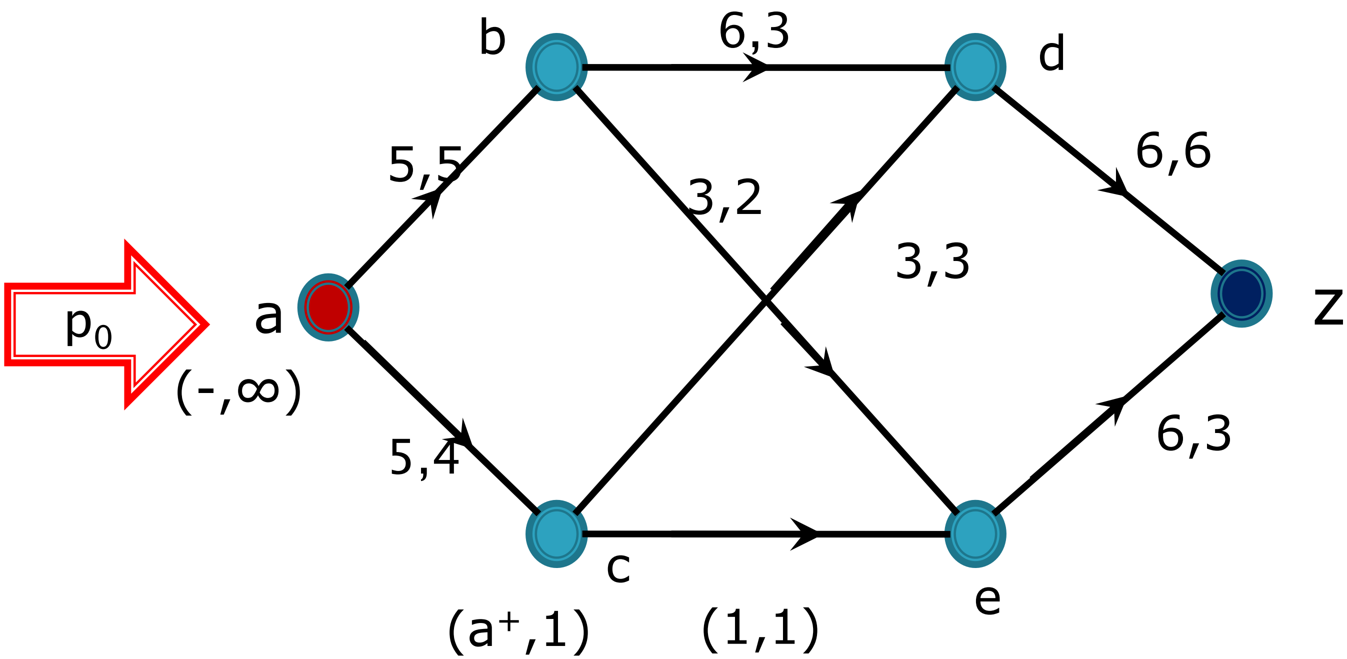
☞ Lặp lần 2:

- Gán nhãn cho đỉnh a là (-,Δ(a)), với Δ(a)=∞
- Đặt p_0=a



☞ Lặp lần 2:

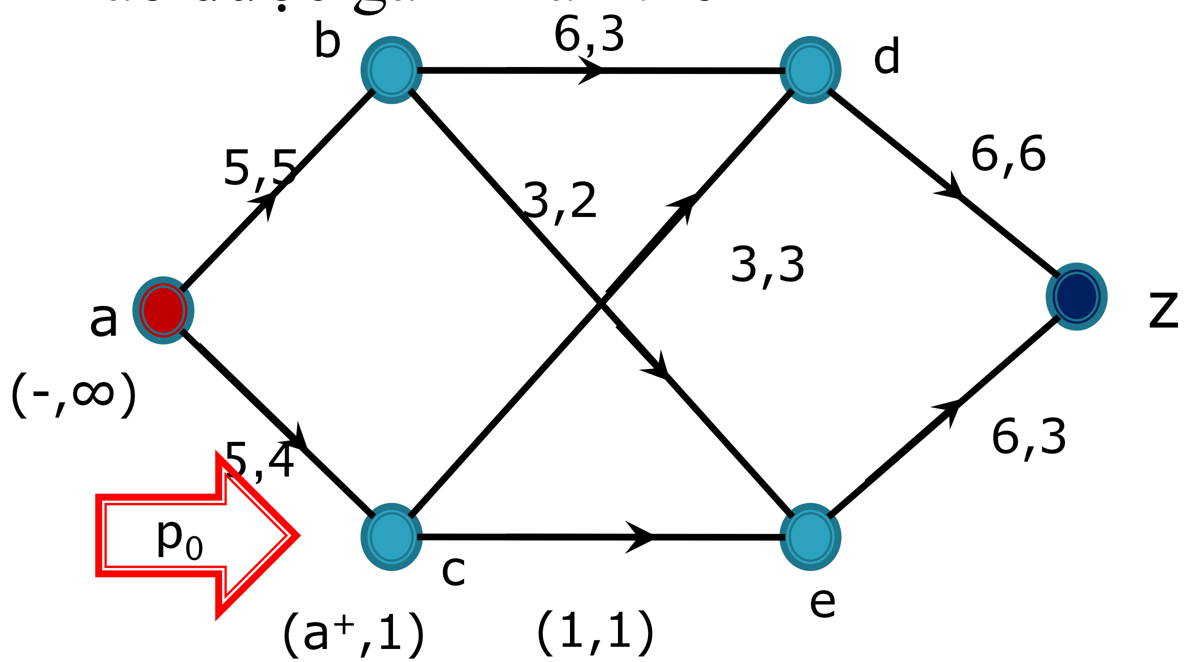
- Gán nhãn cho đỉnh a là (-,Δ(a)), với Δ(a)=∞
- Đặt p_0=a



- Đỉnh z chưa gán nhãn, đỉnh c đã gán nhãn nhưng chưa được xét, đặt p_0=c

☞Lặp lần 2:

- p_0=c
- Không có đỉnh nào được gán nhãn thêm



- Đỉnh z chưa gán nhãn, , không tìm được đỉnh p_0 khác. Dừng. Tập P={a,c} là tập tương ứng với phép cắt tối thiểu $(P, \overline{P})$

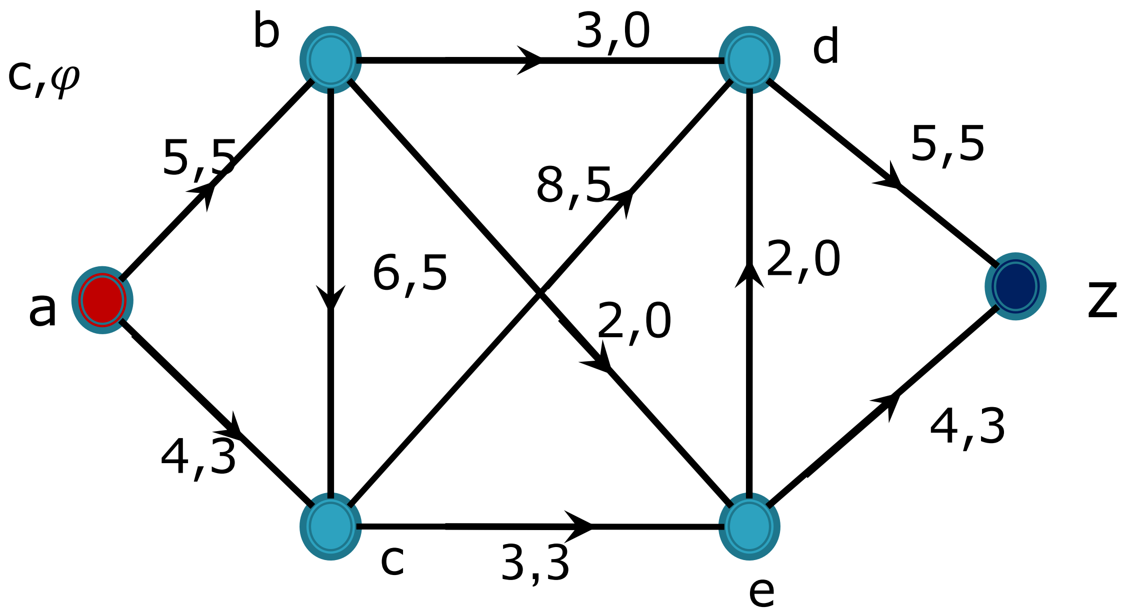
Định lý 6.4

- Khi kết thúc thuật toán Ford-Fulkerson thì φ là 1 hàm tải tối đại và $(P, \overline{P})$ là 1 phép cắt a-z tối tiểu.

Định lý 6.5

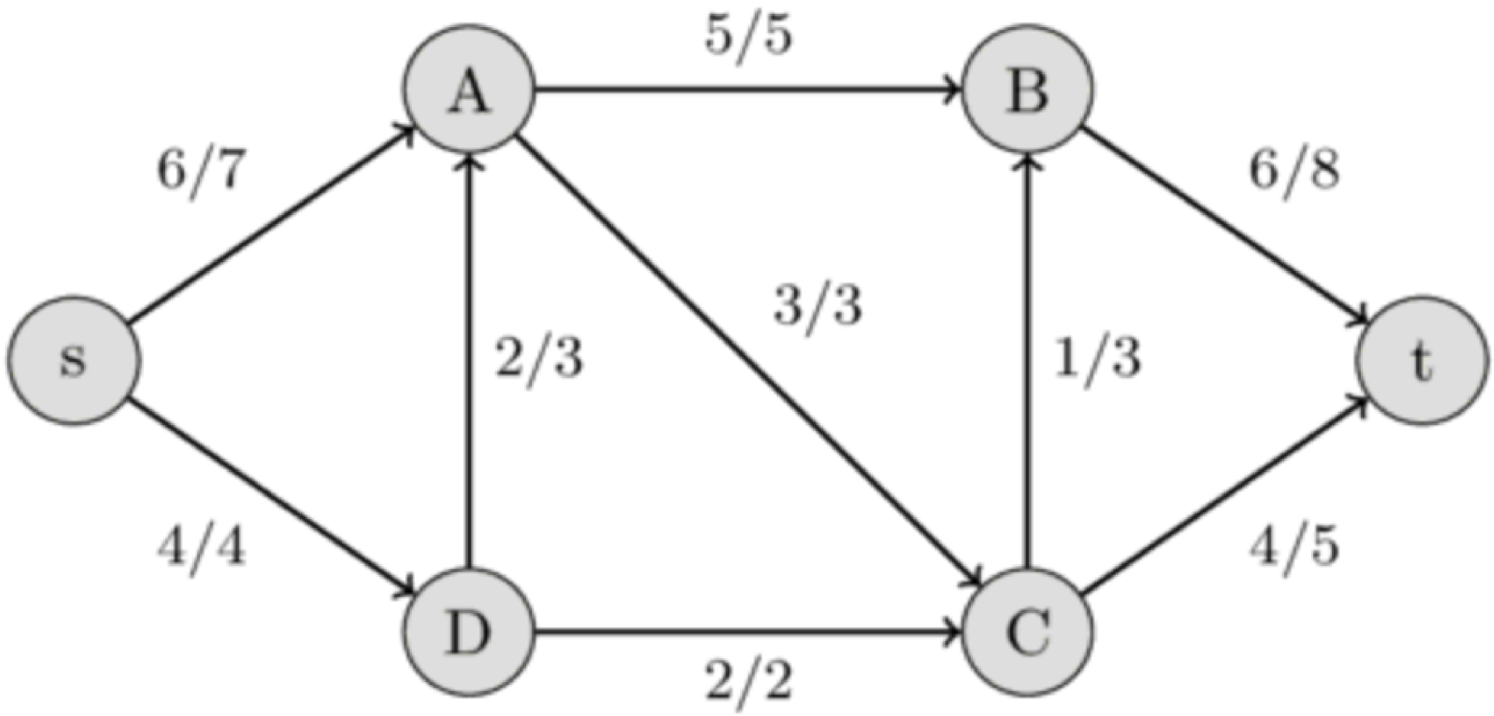
- Trong một mạng G, tải trọng của 1 hàm tải tối đại bằng trọng số của một phép cắt a-z tối tiểu.

Bài tập: Tìm một phép cắt a-z tối thiểu trên các mạng sau:



Đỉnh phát: a
Đỉnh thu: z

Bài tập: Tìm một phép cắt a–t tối thiểu trên các mạng sau:



Đỉnh phát: a
Đỉnh thu: t