ĐẠI HỌC BÁCH KHOA HÀ NỘI
VIỆN CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG

Toán rời rạc

Phần thứ hai

LÝ THUYẾT ĐỒ THỊ

Graph Theory

Nội dung phần 2: Lý thuyết đồ thị

Chương 1. Các khái niệm cơ bản

Chương 2. Biểu diễn đồ thị

Chương 3. Duyệt đồ thị

Chương 4. Cây và cây khung của đồ thị

Chương 5. Bài toán đường đi ngắn nhất

Chương 6. Bài toán luồng cực đại trong mạng

Nội dung chi tiết

6.1. Phát biểu bài toán và các ứng dụng

6.2. Lát cắt

6.3. Đồ thị tăng luồng và Đường tăng luồng

6.4. Thuật toán Ford-Fulkerson

6.5. Thuật toán Edmond-Karp

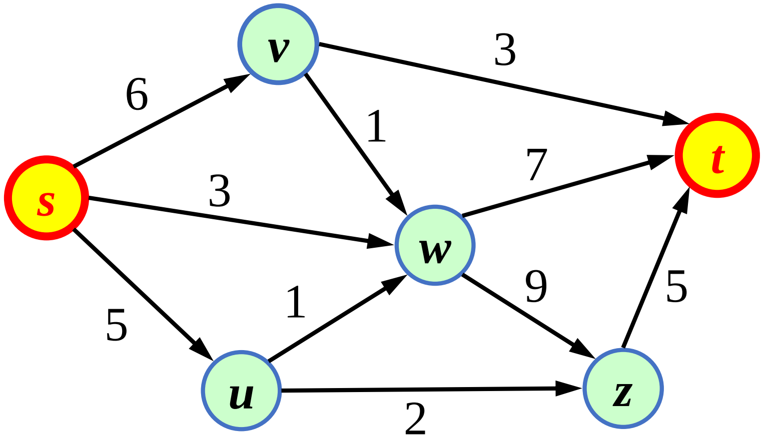
6.6. Một số ứng dụng

MẠNG (Network)

Mạng là đồ thị có hướng G = (V,E) :

- Có duy nhất một đỉnh *s* không có cung đi vào gọi là *đỉnh phát* (nguồn) và duy nhất một đỉnh *t* không có cung đi ra gọi là *đỉnh thu* (đích).
- Mỗi cung *e* của *G* được gắn với một số không âm *c*(*e*) được gọi là *khả năng thông qua* của *e*.

Ví dụ:



LUỒNG TRONG MẠNG

Định nghĩa. Luồng f trong mạng $G=(V,E)$ là phép gán số $f(e)$ cho mỗi cạnh e ($f(e)$ được gọi là luồng trên cạnh e) thoả mãn các điều kiện:

1) Hạn chế về khả năng thông qua (Capacity Rule):

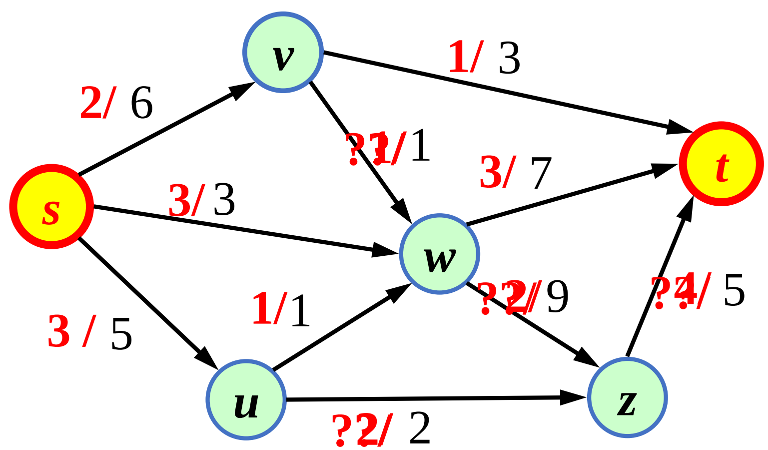
Với mỗi cung e, $0 \le f(e) \le c(e)$

2) Điều kiện cân bằng luồng (Conservation Rule): Với mỗi $v \neq s, t$

$$\sum_{e \in E^-(v)} f(e) = \sum_{e \in E^+(v)} f(e)$$

trong đó $E^-(v)$ và $E^+(v)$ tương ứng là tập các cung đi vào và đi ra khỏi đỉnh v.

Các điều kiện 1) và 2) được thoả mãn => f là luồng trên mạng.



LUỒNG TRONG MẠNG

Định nghĩa. Luồng f trong mạng $G=(V,E)$ là phép gán số $f(e)$ cho mỗi cạnh e ($f(e)$ được gọi là luồng trên cạnh e) thoả mãn các điều kiện:

1) Hạn chế về khả năng thông qua (Capacity Rule):

Với mỗi cung e, $0 \le f(e) \le c(e)$

2) Điều kiện cân bằng luồng (Conservation Rule): Với mỗi $v \neq s, t$

$$\sum_{e \in E^-(v)} f(e) = \sum_{e \in E^+(v)} f(e)$$

trong đó $E^-(v)$ và $E^+(v)$ tương ứng là tập các cung đi vào và đi ra khỏi đỉnh v.

Định nghĩa. Giá trị của luồng f là

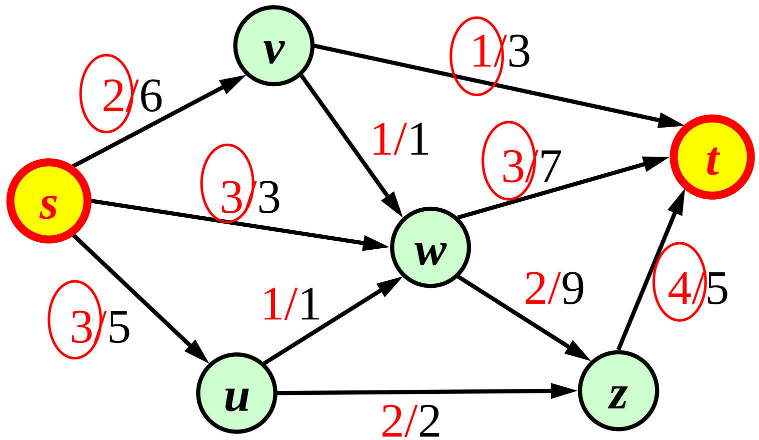
$$val(f) = \boxed{\sum_{e \in E^+(s)} f(e)} \overset{(*)}{=} \boxed{\sum_{e \in E^-(t)} f(e)}$$

Các cạnh đi ra khỏi đỉnh s Các cạnh đi vào đỉnh t

(Đẳng thức (*) thu được bằng cách cộng tất cả các điều kiện cân bằng luồng.)

Luồng trong mạng – Ví dụ

Ví dụ:



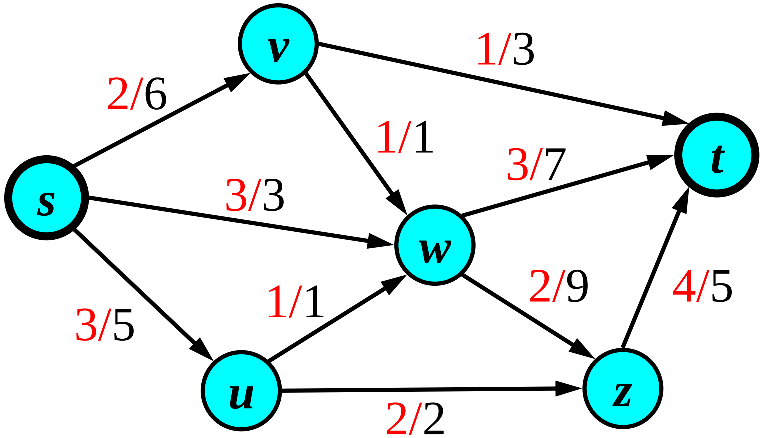
- Trong 2 số viết bên mỗi cạnh: giá trị luồng trên cạnh là số màu đỏ, số còn lại là khả năng thông qua.
- Các điều kiện 1) và 2) được thoả mãn => *f* là luồng trên mạng.
- Giá trị luồng là: $val(f) = \sum_{e \in E^+(s)} f(e) \overset{(*)}{=} \sum_{e \in E^-(t)} f(e)$

$$8 = f(s,v) + f(s,u) + f(s,w) = f(v,t) + f(w,t) + f(z,t)$$

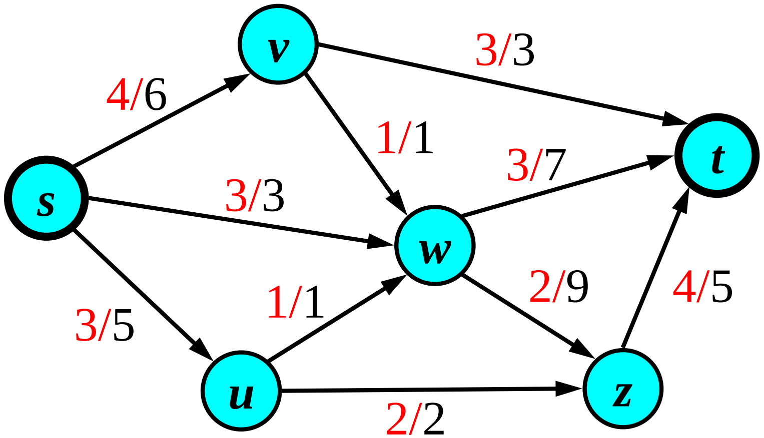
Bài toán luồng cực đại

Luồng trong mạng G được gọi là luồng cực đại nếu trong số tất cả các luồng trong mạng G nó là luồng có giá trị lớn nhất

Bài toán tìm luồng cực đại trong mạng G được gọi là bài toán luồng cực đại



Luồng với giá trị 8 = 2 + 3 + 3 = 1 + 3 + 4



Luồng cực đại có giá trị 10 = 4 + 3 + 3 = 3 + 3 + 4

Các ứng dụng trực tiếp

Mạng	Đỉnh	Cung	Luồng
truyền thông	trạm giao dịch, máy tính, vệ tinh	cáp nối, cáp quang,	voice, video, packets
mạng điện	cổng, registers, processors	dây dẫn	dòng điện
cơ khí	joints	rods, beams, springs	heat, energy
thuỷ lợi	hồ chứa, trạm bơm, nguồn nước	đường ống	dòng nước, chất lỏng
tài chính	nhà băng	giao dịch	tiền
giao thông	sân bay, ga tàu, giao lộ	đường cao tốc, ray, đường bay	hàng hoá, phương tiện, hành khách
hoá học	sites	bonds	energy

Nội dung chi tiết

6.1. Phát biểu bài toán và các ứng dụng

6.2. Lát cắt

6.3. Đồ thị tăng luồng và Đường tăng luồng

6.4. Thuật toán Ford-Fulkerson

6.5. Thuật toán Edmond-Karp

6.6. Một số ứng dụng

Lát cắt (Cuts)

Lát cắt là cách phân hoạch tập đỉnh của đồ thị thành 2 tập S và T sao cho $s \in S$, $t \in T$.

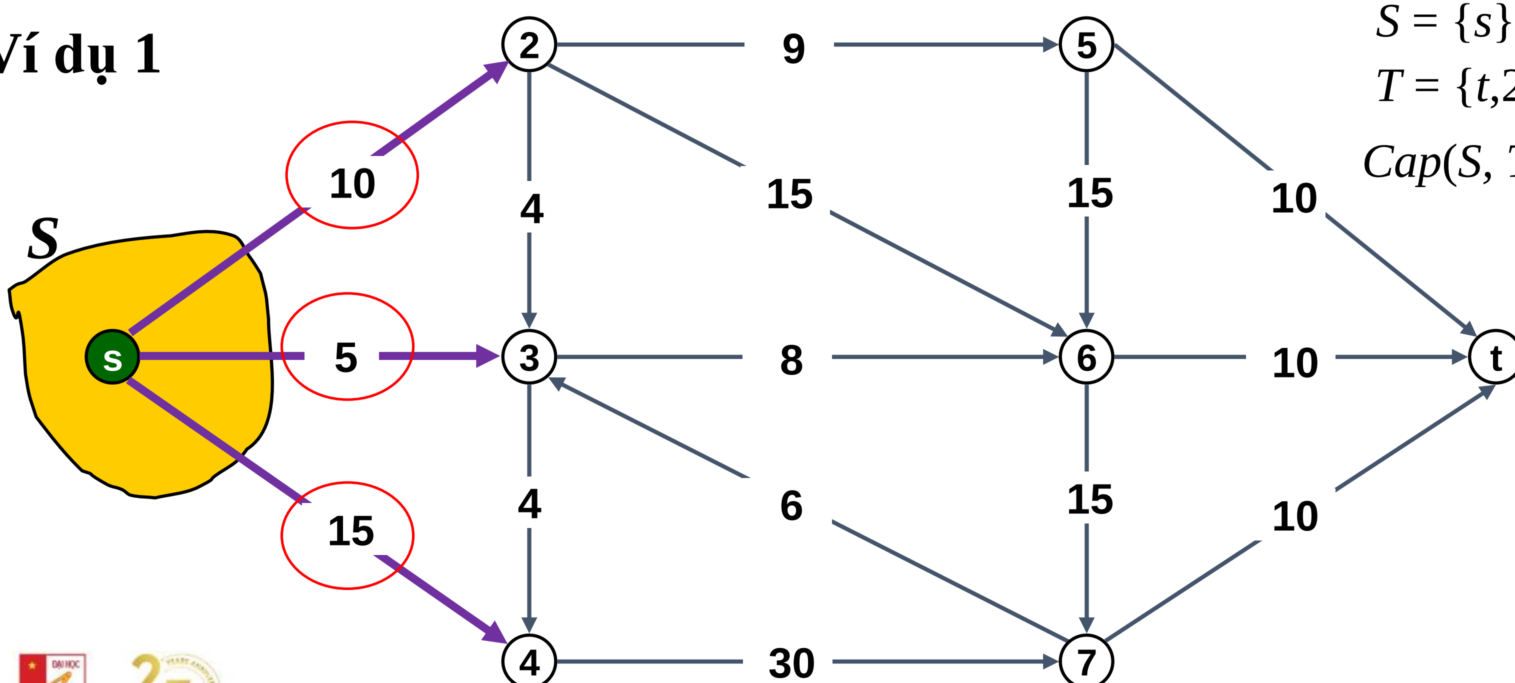
- Khả năng thông qua $cap(S,T)$ của lát cắt (S, T) là số:

$$cap(S,T) = \sum_{e \in S \to T} c(e),$$

trong đó $S \to T := \{(v,w) \in E : v \in S, w \in T\}$

Lát cắt nhỏ nhất (hẹp nhất) là lát cắt với kntq nhỏ nhất.

Ví dụ 1



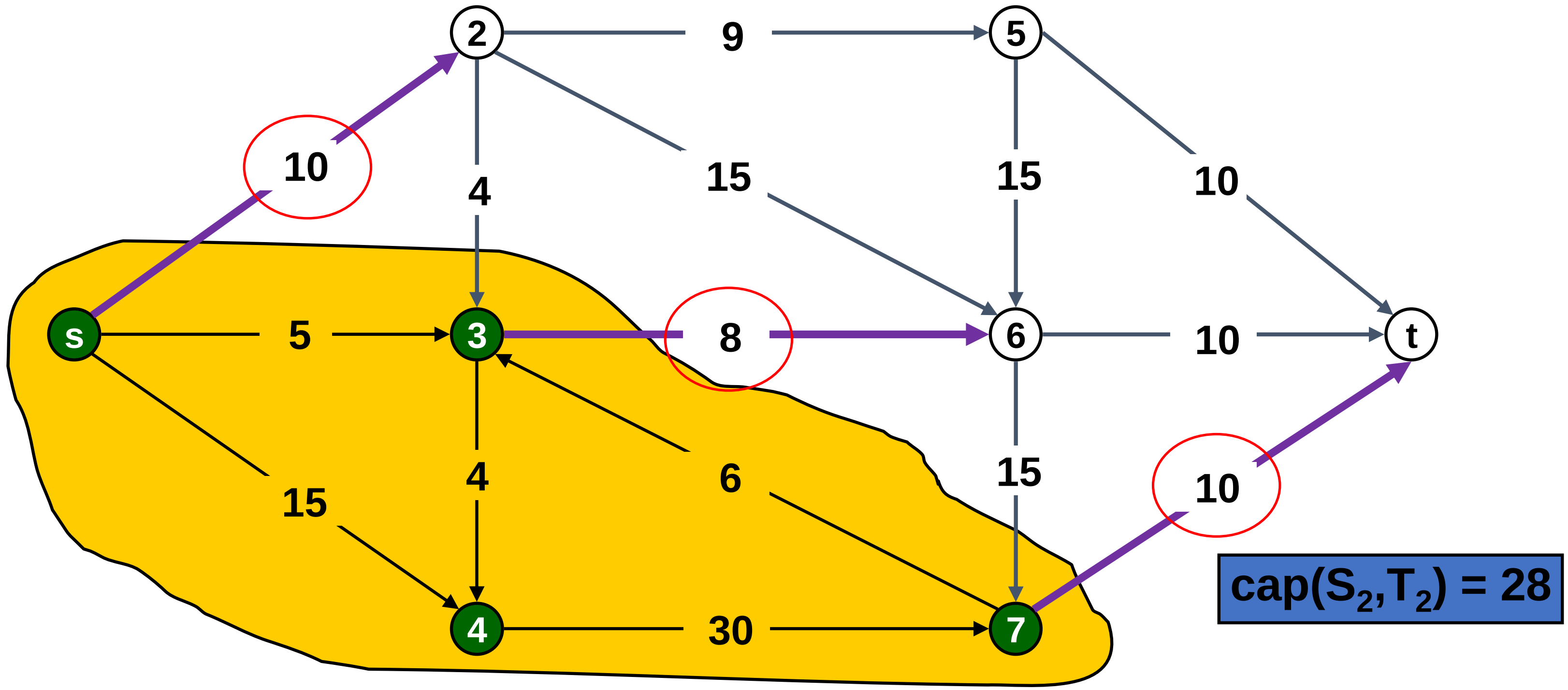
$S = \{s\}$

$T = \{t,2,3,4,5,6,7\}$

$Cap(S, T) = 10+5+15=30$

Lát cắt

Ví dụ 2 Lát cắt (S_2, T_2), S_2={s,3,4,7}, T_2={2,5,6,t} có khả năng thông qua 28



Luồng chảy qua lát cắt

Định nghĩa. Giả sử *f* là luồng trong mạng và (*S*,*T*) là lát cắt. Ta gọi giá trị luồng chảy qua lát cắt (*S*,*T*) là đại lượng

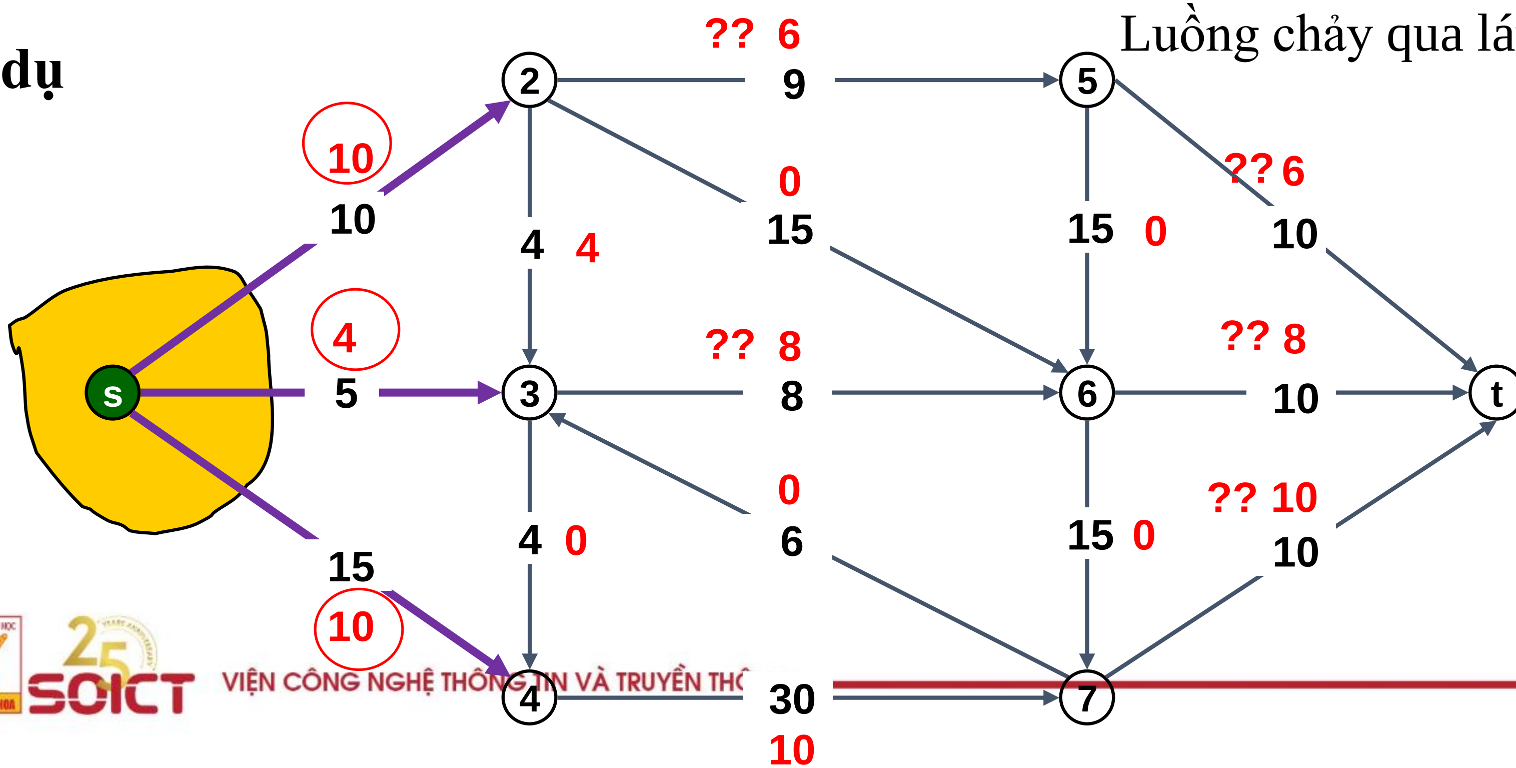
$$\boxed{\sum_{e\in S\to T} f(e)} - \boxed{\sum_{e\in T\to S} f(e)} \qquad 10+4+10 \quad 0$$

trong đó: $S \to T = \{(v,w)\in E: v\in S,\ w\in T\}$

$T\to S = \{(v,w)\in E: v\in T,\ w\in S\}$

$S = \{s\}$
$T = \{2, 3, 4, 5,6,7,t\}$
Luồng chảy qua lát cắt (*S*,*T*) = ?
24

Ví dụ



Luồng và lát cắt

Bổ đề 1. Giả sử *f* là luồng, và (*S*, *T*) là lát cắt. Khi đó giá trị luồng chảy qua lát cắt chính bằng giá trị của luồng:

$$\sum_{e \in S \to T} f(e) - \sum_{e \in T \to S} f(e) = \sum_{e \in E^+(s)} f(e) = \sum_{e \in E^-(t)} f(e) = val(f)$$

Luồng và lát cắt

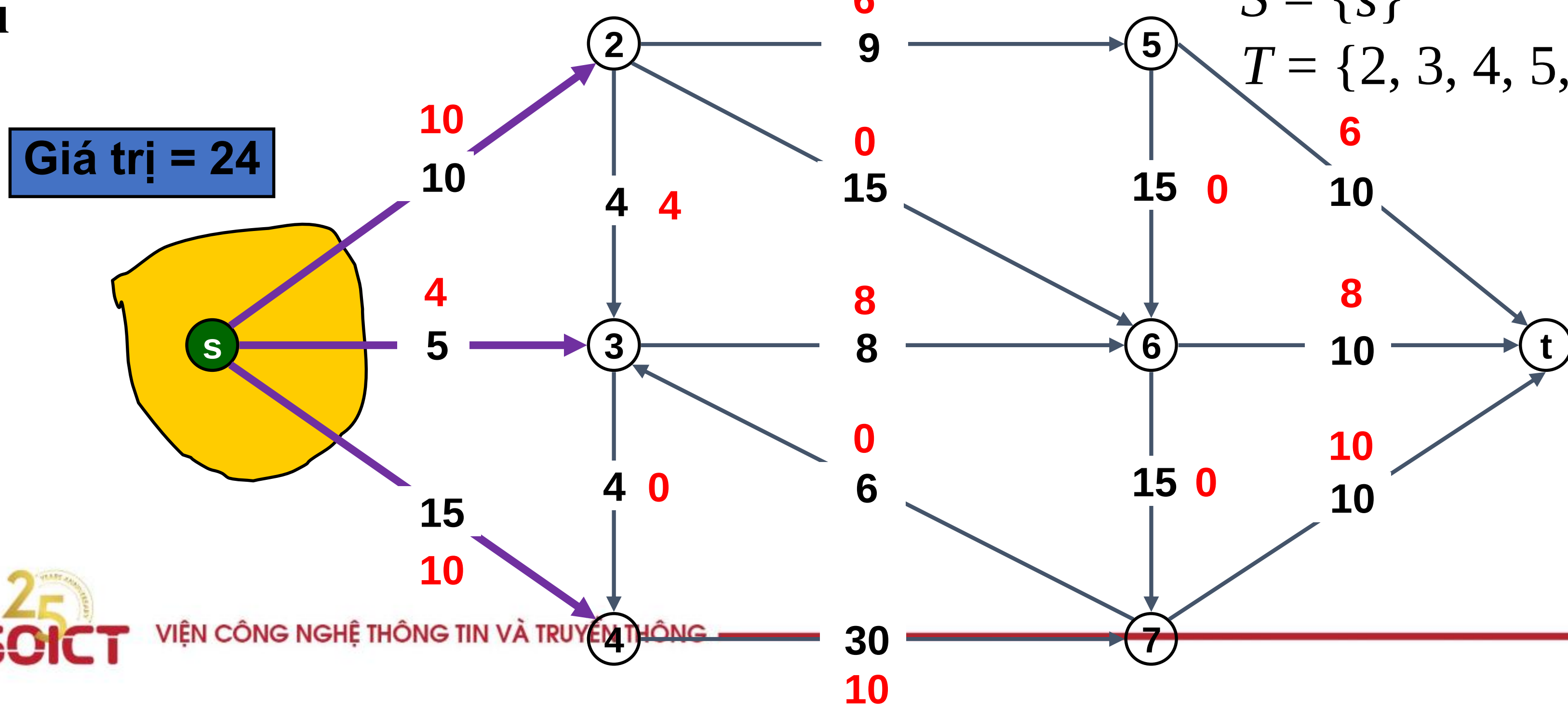
Bổ đề 1. Giả sử *f* là luồng, và (*S*, *T*) là lát cắt. Khi đó giá trị luồng chảy qua lát cắt chính bằng giá trị của luồng:

$$\boxed{\sum_{e \in S \to T} f(e) - \sum_{e \in T \to S} f(e)} = \boxed{\sum_{e \in E^+(s)} f(e) = \sum_{e \in E^-(t)} f(e) = val(f)}$$

Ví dụ

$S = \{s\}$

$T = \{2, 3, 4, 5, 6, t\}$



Luồng và lát cắt

Chứng minh bổ đề: Giả sử f là luồng còn (S, T) là lá cắt. Khi đó

$$\sum_{e \in S \to T} f(e) - \sum_{e \in T \to S} f(e) = \sum_{e \in E^+(s)} f(e) = val(f)$$

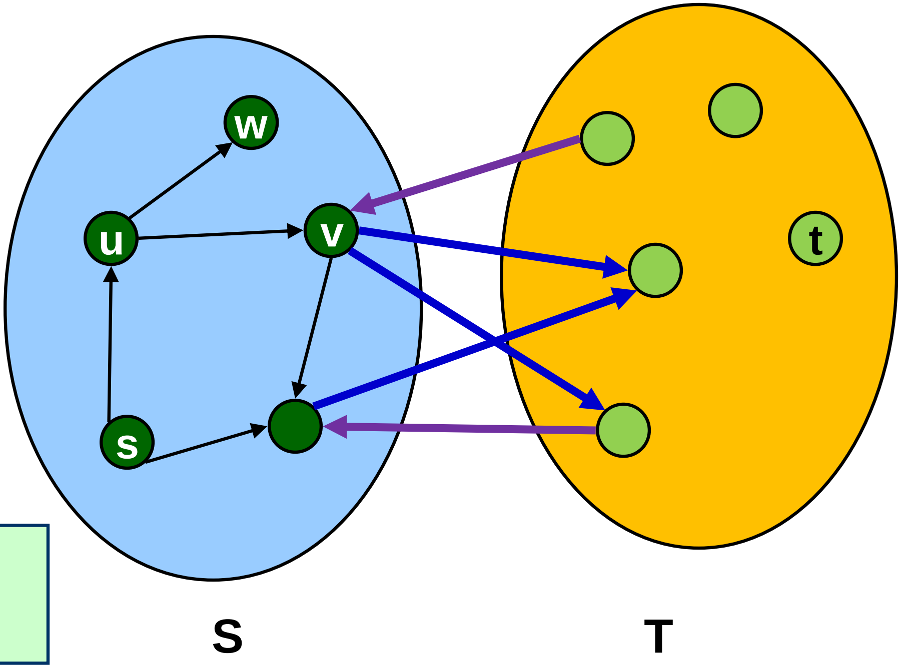
Chứng minh: Cộng tất cả các ràng buộc cân bằng luồng theo mọi v∈S, đơn giản biểu thức ta thu được:

$$0 = \sum_{v \in S} \left(\sum_{e \in E^+(v)} f(e) - \sum_{e \in E^-(v)} f(e) \right)$$

$$= \sum_{e \in E^+(s)} f(e) - \left(\sum_{e \in S \to T} f(e) - \sum_{e \in T \to S} f(e) \right)$$

tổng theo các cung xanh

tổng theo các cung tím



từ đó suy ra đẳng thức cần chứng minh

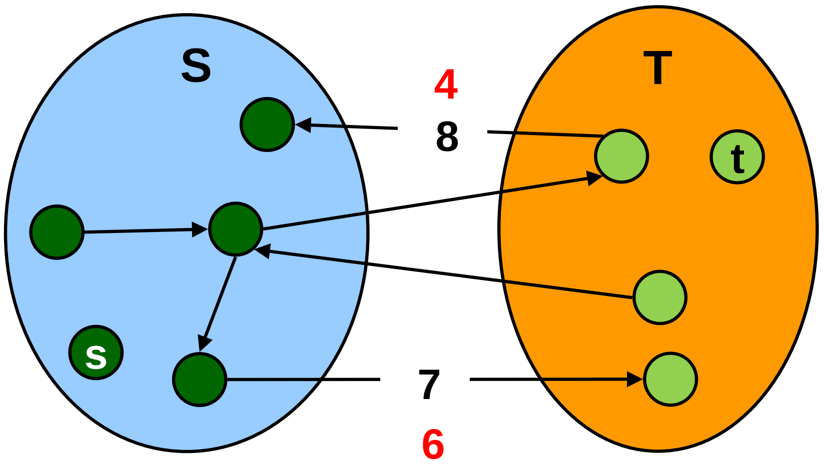
Luồng và lát cắt

Bổ đề 2. Giả sử f là luồng, còn (S, T) là lát cắt. Khi đó

$$\text{val}(f) \le \text{cap}(S, T).$$

Chứng minh

$$\begin{aligned} val(f) &= \sum_{e \in S \to T} f(e) - \sum_{e \in T \to S} f(e) \\ &\le \sum_{e \in S \to T} f(e) \\ &\le \sum_{e \in S \to T} c(e) \\ &= \text{cap}(S, T) \end{aligned}$$



Luồng cực đại và lát cắt nhỏ nhất(Max Flow and Min Cut)

Hệ quả. Giả sử f là luồng, còn (S, T) là lát cắt. Nếu $\boxed{\text{val}(f) = \text{cap}(S, T)}$, thì f là **luồng cực đại** còn (S, T) là **lát cắt hẹp nhất**

Luồng trong mạng G được gọi là luồng cực đại nếu trong số tất cả các luồng trong mạng G nó là luồng có giá trị lớn nhất

Lát cắt nhỏ nhất (hẹp nhất): lát cắt với kntq nhỏ nhất

Chứng minh: Xét f ' là luồng bất kỳ và (S',T') là lát cắt bất kỳ.
Theo bổ đề 2 ta có:

Theo giả thiết **Theo bổ đề 2**

$$\text{val}(f') \le \text{cap}(S,T) = \text{val}(f) \le \text{cap}(S',T')$$

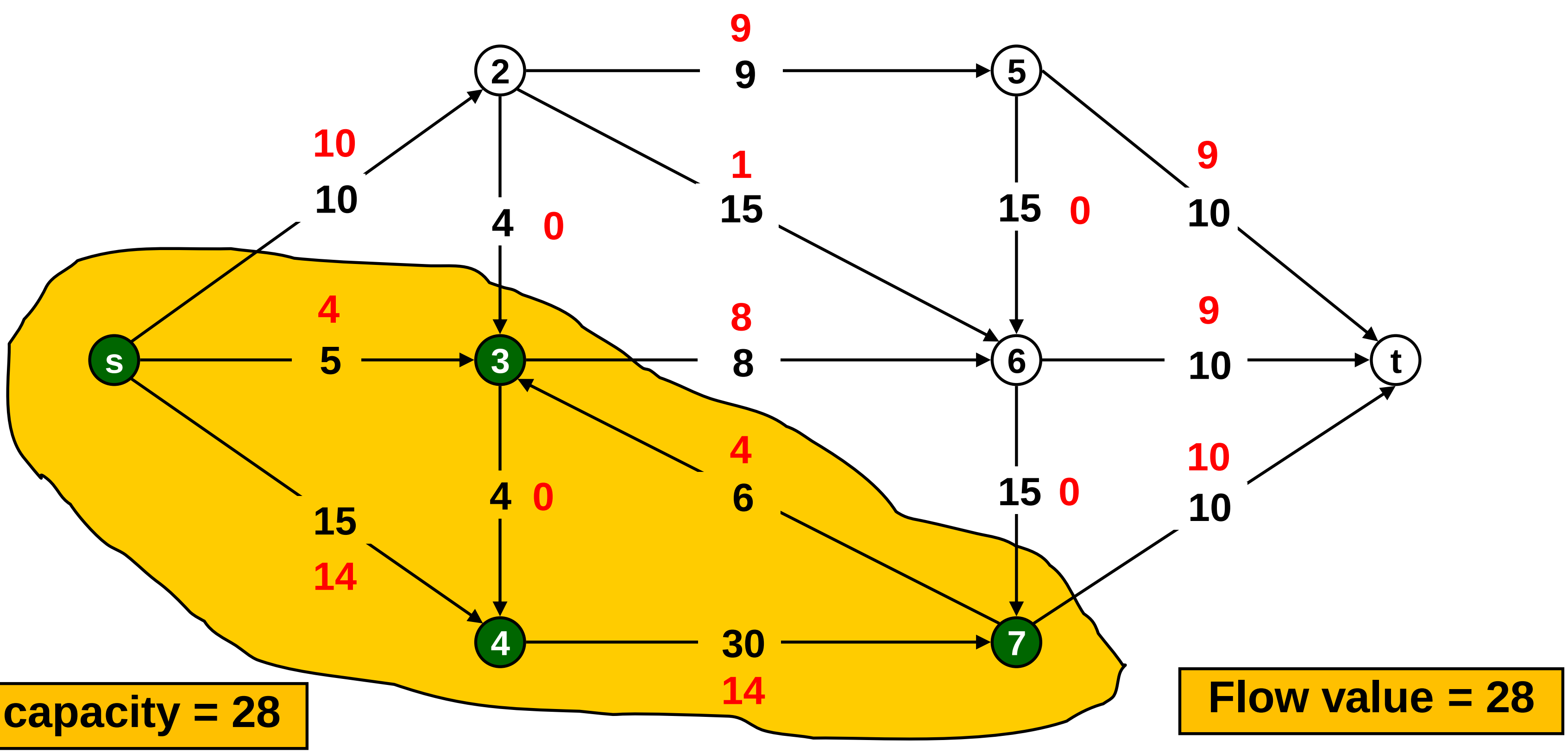
Bổ đề 2. Giả sử f là luồng, còn (S, T) là lát cắt. Khi đó
$\text{val}(f) \le \text{cap}(S, T)$

f' là luồng bất kỳ
$\text{val}(f') \le \text{val}(f)$ → f là luồng cực đại

(S', T') là lát cắt bất kỳ
$\text{cap}(S,T) \le \text{cap}(S',T')$ → (S,T) là lát cắt hẹp nhất

Định lý về luồng cực đại và lát cắt nhỏ nhất
Max-Flow Min-Cut Theorem

Đinh lý (Ford-Fulkerson, 1956): Trong mạng bất kỳ, giá trị của luồng cực đại luôn bằng khả năng thông qua của lát cắt nhỏ nhất.



Nội dung chi tiết

6.1. Phát biểu bài toán và các ứng dụng

6.2. Lát cắt

6.3. Đồ thị tăng luồng và Đường tăng luồng

6.4. Thuật toán Ford-Fulkerson

6.5. Thuật toán Edmond-Karp

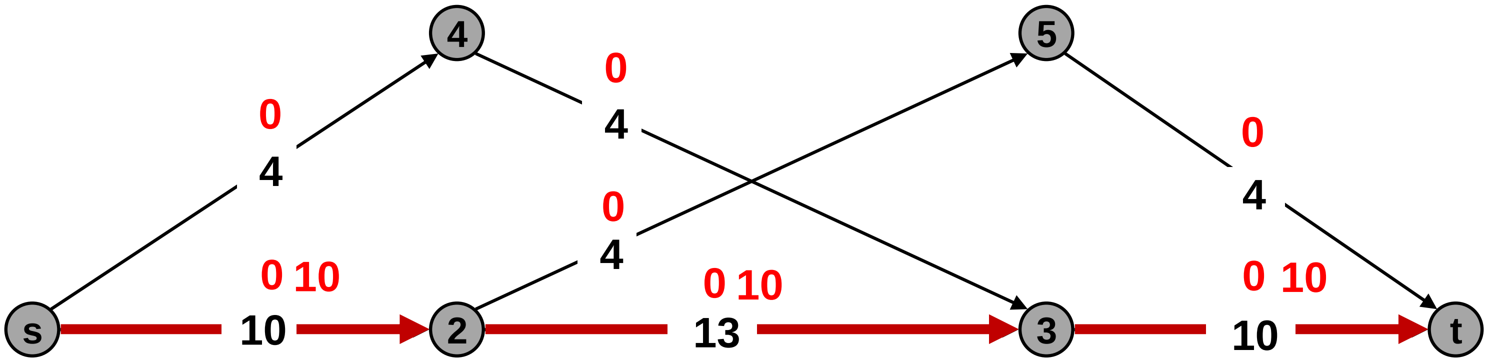
6.6. Một số ứng dụng

Thuật toán tham lam

- Bắt đầu từ luồng 0 (Luồng có giá trị = 0).
- Tìm đường đi P từ *s* đến *t* trong đó mỗi cung thoả mãn $f(e) < c(e)$.
- Tăng luồng dọc theo đường đi P.
- Lặp lại cho đến khi gặp bế tắc.

s->4->3->t ???

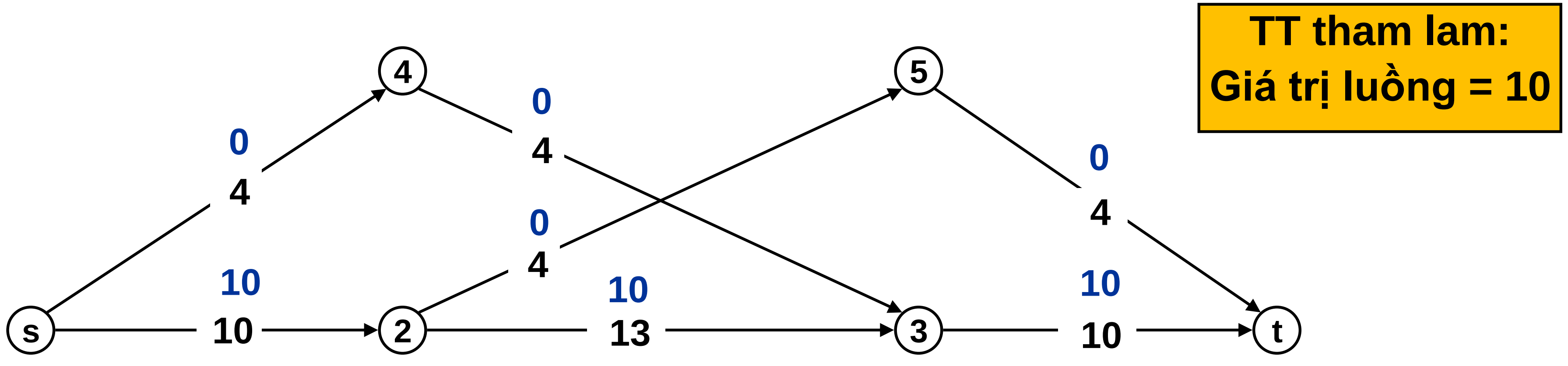
s->2->5->t ???

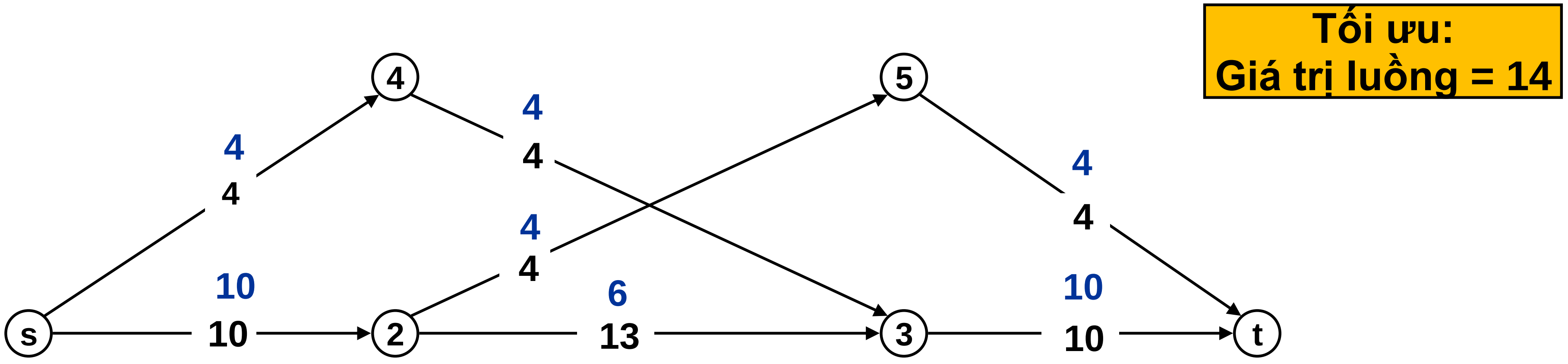


Luồng có giá trị = 10

Ý tưởng thuật toán

- Thuật toán tham lam không cho lời giải tối ưu.

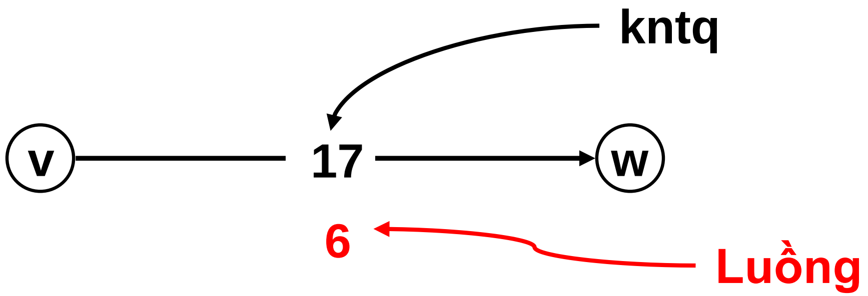




Đồ thị tăng luồng – Tập cung

Mạng đã cho G = (V, E).

- Cung e = (v, w) $\in$ E
- Luồng f (e)
- Khả năng thông qua c(e)

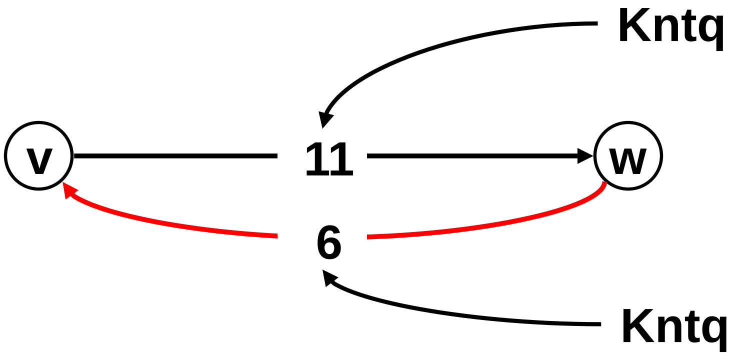


Đồ thị tăng luồng: $G_f = (V, E_f)$.

- $E_f = \boxed{\{e: f(e) < c(e)\}} \cup \boxed{\{e^R : f(e) > 0\}}$
- Khả năng thông qua của các cung

$$c_f(e) = \begin{cases} \boxed{c(e) - f(e) \quad \text{nếu} \quad e \in E} \\ \boxed{f(e) \qquad\qquad \text{nếu} \quad e^R \in E} \end{cases}$$

$e = (u,v) \Rightarrow e^R = (v,u)$

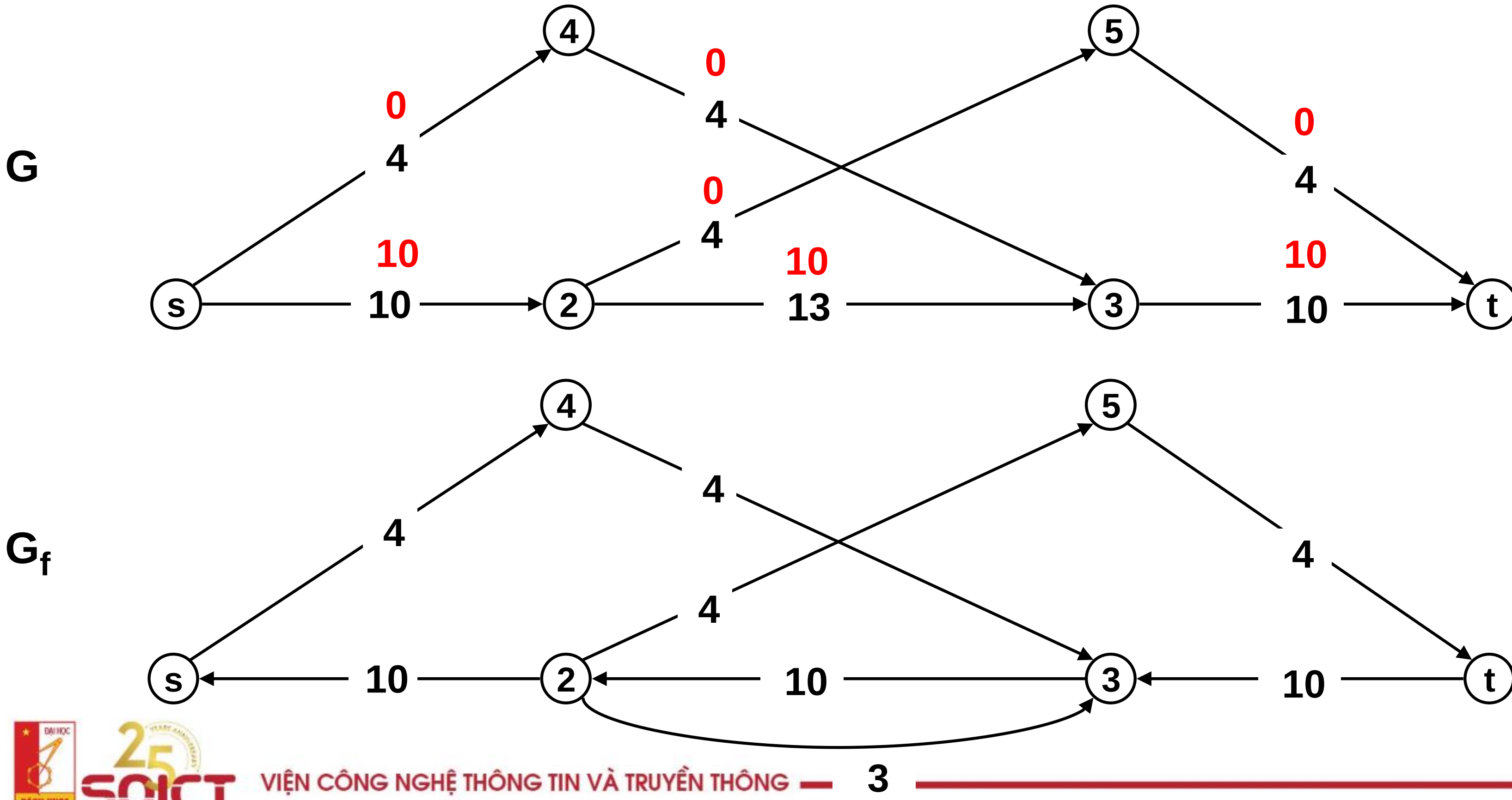


Đồ thị tăng luồng - Ví dụ

Đồ thị tăng luồng: $G_f = (V, E_f)$.

$$c_f(e) = \begin{cases} c(e) - f(e) & \text{nếu} \quad e \in E \\ f(e) & \text{nếu} \quad e^R \in E \end{cases}$$

- $E_f = \{e : f(e) < c(e)\} \cup \{e^R : f(e) > 0\}$.
- $c_f(e)$ cho biết lượng lớn nhất có thể tăng luồng trên cung e.
- $c_f(e^R)$ cho biết lượng lớn nhất có thể giảm luồng trên cung e.



(s,4)
(s,2)
(2,3)
(4,3)
(2,5)
(3,t)
(5,t)

Đồ thị tăng luồng - Ví dụ

Đồ thị tăng luồng: $G_f = (V, E_f)$.

- $E_f = \{e : f(e) < c(e)\} \cup \{e^R : f(e) > 0\}$.
- $\boxed{c_f(e) \text{ cho biết lượng lớn nhất có thể tăng luồng trên cung e.}}$
- $c_f(e^R)$ cho biết lượng lớn nhất có thể giảm luồng trên cung e.

$$c_f(e) = \begin{cases} c(e) - f(e) & \text{nếu } \ e \in E \\ f(e) & \text{nếu } \ e^R \in E \end{cases}$$

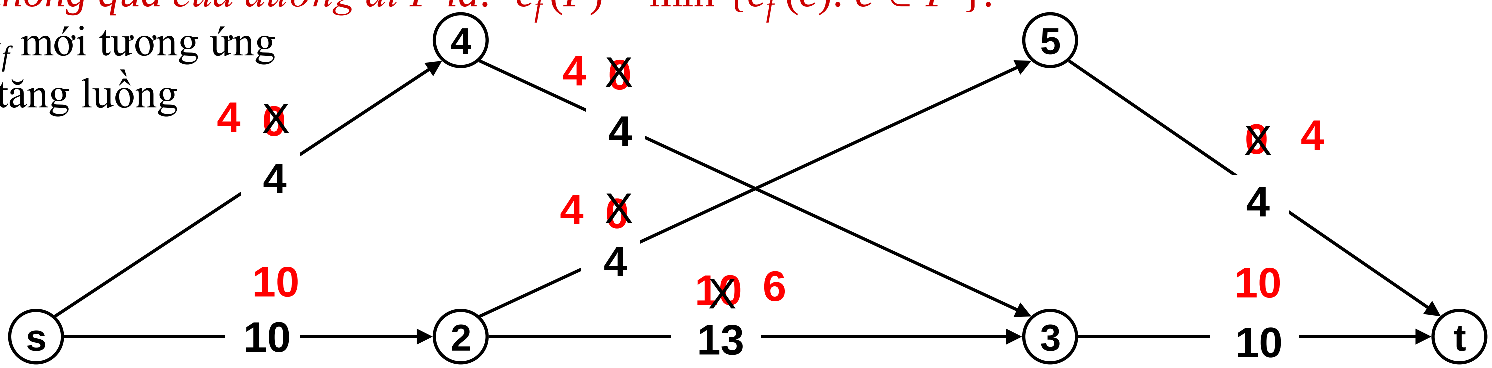
Đường tăng luồng = đường đi từ *s* đến *t* trên đồ thị tăng luồng G_f.

Khả năng thông qua của đường đi P là: $c_f(P) = \min \{c_f(e): e \in P\}$.

Xây dựng G_f mới tương ứng

Tìm đường tăng luồng trên G_f mới

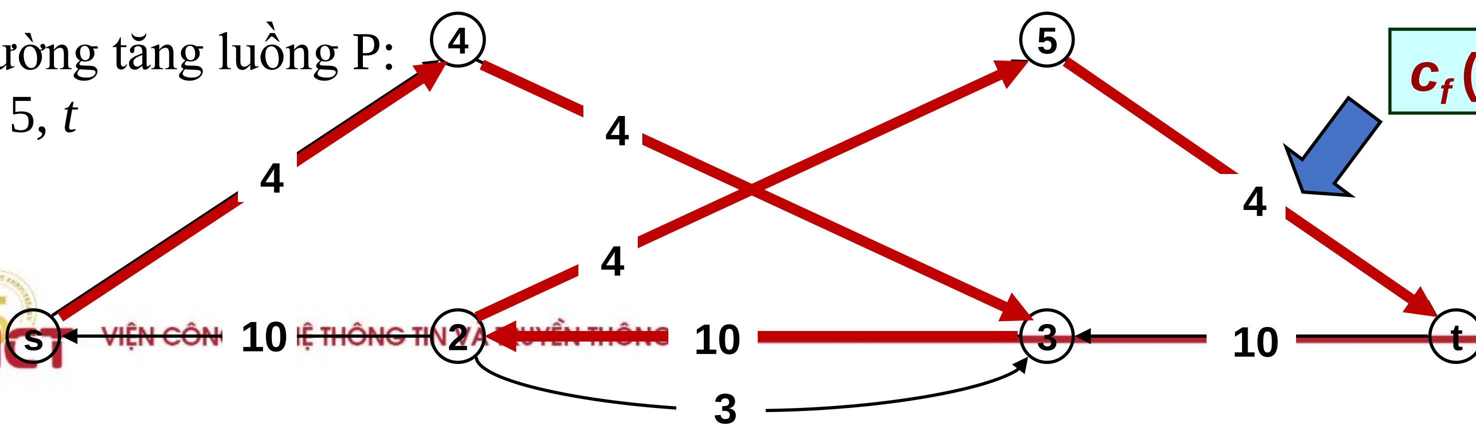
G



(s,4)
(s,2)
(2,3)
(4,3)
(2,5)
(3,t)
(5,t)

Ví dụ: Đường tăng luồng P:
s, 4, 3, 2, 5, t

G_f



$c_f(P) = 4$

Đường tăng luồng

$$c_f(e) = \begin{cases} c(e) - f(e) & \text{nếu } \; e \in E \\ f(e) & \text{nếu } \; e^R \in E \end{cases}$$

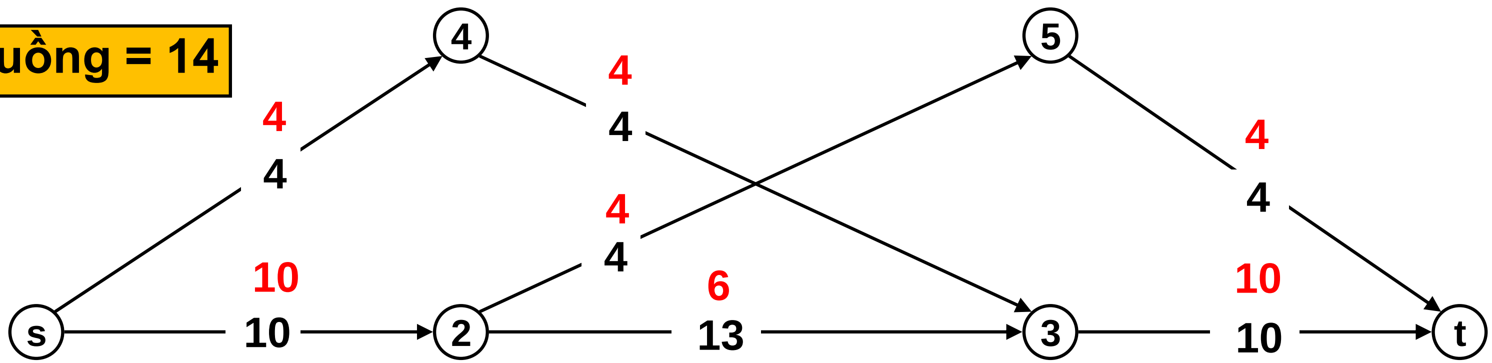
Đường tăng luồng = đường đi từ *s* đến *t* trên đồ thị tăng luồng.

- Luồng là cực đại $\Leftrightarrow$ không tìm được đường tăng luồng???

(*s*,4)
(*s*,2)
(2,3)
(4,3)
(2,5)
(3,*t*)
(5,*t*)

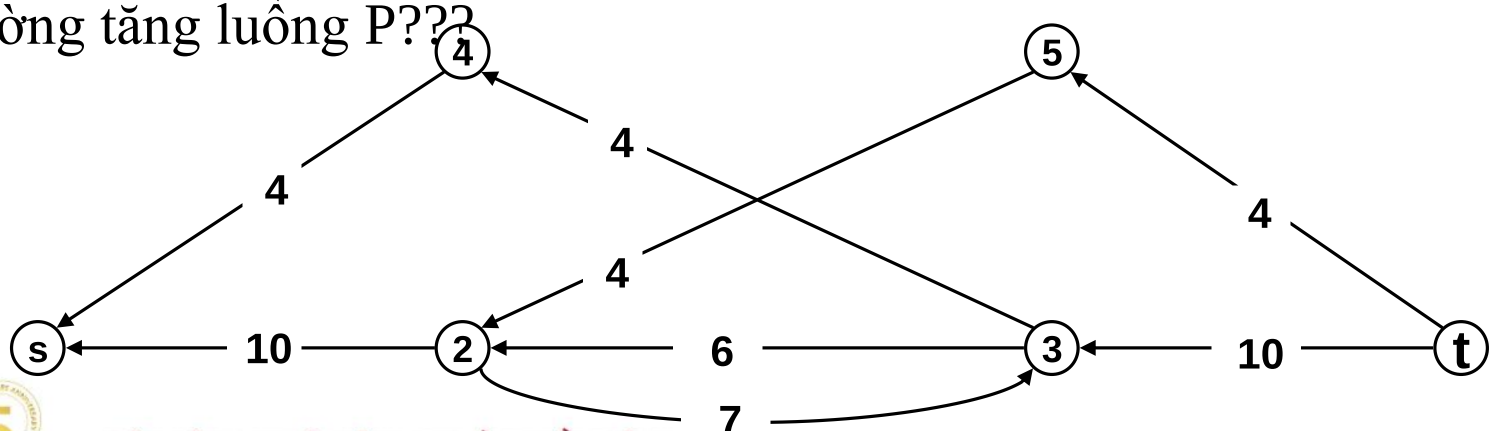
Giá trị luồng = 14

G



Tìm đường tăng luồng P???

G_f



Định lý về luồng cực đại và lát cắt nhỏ nhất

Định lý đường tăng luồng (Ford-Fulkerson, 1956): Luồng là cực đại khi và chỉ khi không tìm được đường tăng luồng.

Định lý về luồng cực đại và lát cắt nhỏ nhất (Ford-Fulkerson, 1956): Giá trị của luồng cực đại bằng khả năng thông qua của lát cắt nhỏ nhất.

Ta sẽ chứng minh định lý tổng hợp sau:

Định lý. Giả sử *f* là luồng trong mạng. Ba mệnh đề sau là tương đương

(i) Tìm được lát cắt (S, T) sao cho $\text{val}(f) = \text{cap}(S, T)$.

(ii) *f* là luồng cực đại.

(iii) Không tìm được đường tăng luồng *f*.

Nội dung chi tiết

6.1. Phát biểu bài toán và các ứng dụng

6.2. Lát cắt

6.3. Đồ thị tăng luồng và Đường tăng luồng

6.4. Thuật toán Ford-Fulkerson

6.5. Thuật toán Edmond-Karp

6.6. Một số ứng dụng

Thuật toán Ford – Fulkerson

Tăng luồng *f* dọc theo đường tăng P

```
float Augment(f,P)
{
   b ← cf(P)
   FOR e ∈ P
     IF (e ∈ E) // cạnh thuận
        f(e) ← f(e) + b
      ELSE       // cạnh nghịch
        f(eR) ← f(e) - b
   RETURN f
}
```

Thuật toán Ford-Fulkerson

```
float Ford_Fulkerson(G,c,s,t)
{
   FOR e ∈ E    // Khởi tạo luồng 0
      f(e) ← 0
   Gf ← đồ thị tăng luồng f

   WHILE (tìm được đường tăng luồng P)
   {  f ← augment(f, P)
      Cập nhật lại Gf
   }
   RETURN f
}
```

Thời gian tính

Câu hỏi: Thuật toán Ford-Fulekerson có phải là thuật toán đa thức? (thuật toán với thời gian tính bị chặn bởi đa thức bậc cố định của độ dài dữ liệu vào)

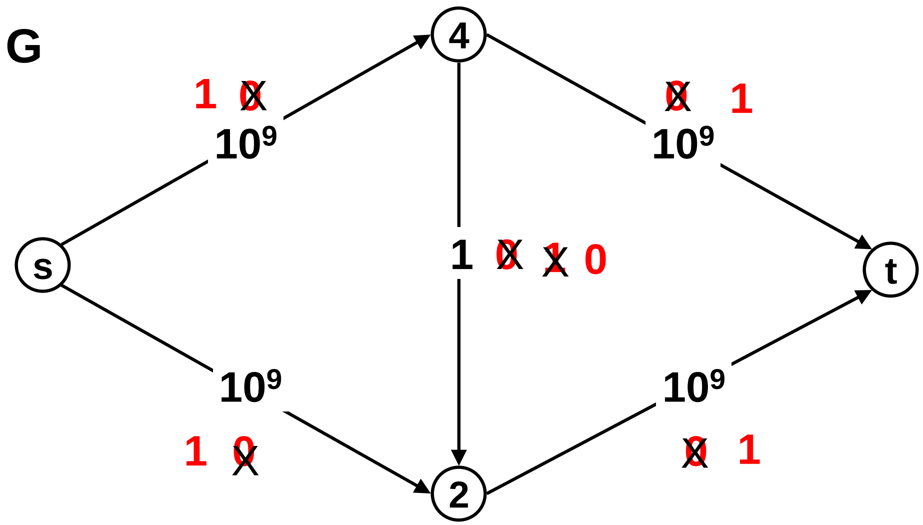
Trả lời: Không phải. Nếu khả năng thông qua lớn nhất là C thì thuật toán có thể phải thực hiện cỡ C bước lặp.

Ví dụ sau đây cho thấy thuật toán có thể phải thực hiện rất nhiều bước lặp

Thuật toán F-F không là thuật toán đa thức

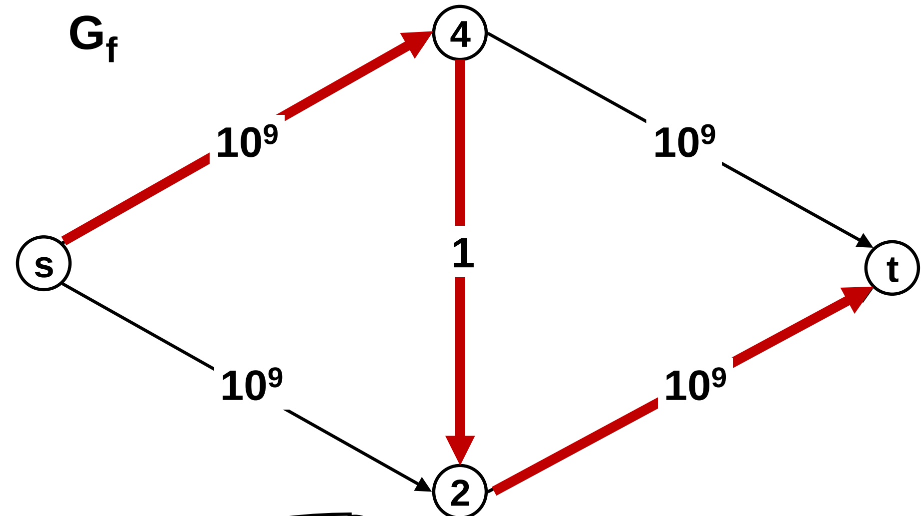
Đường tăng luồng P: *s*, 4, 2, *t*

Tăng luồng dọc theo P



Đường tăng luồng P’: *s*, 2, 4, *t*

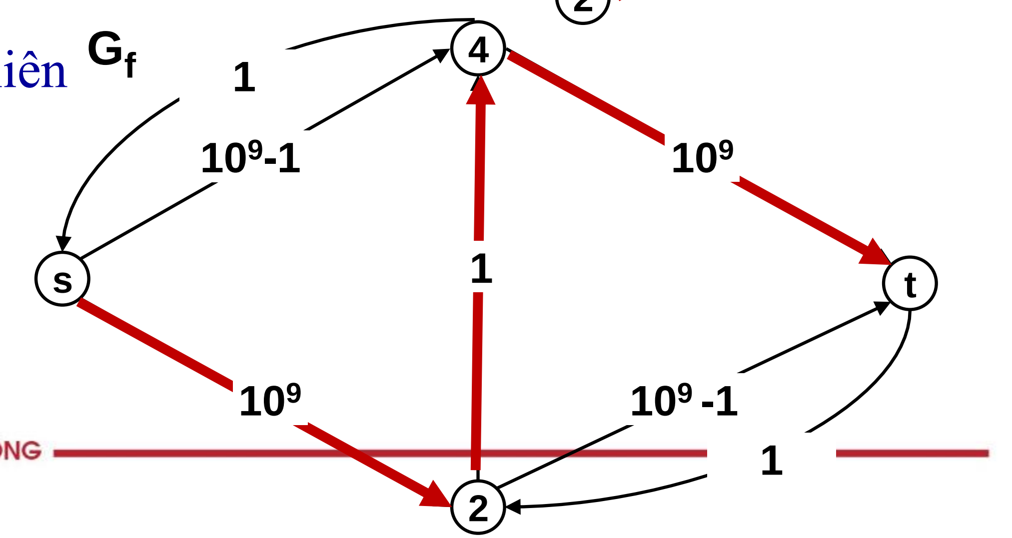
Tăng luồng dọc theo P’



Thuật toán thực hiện tằng luồng luân phiên dọc theo hai đường tăng luồng P và P’

2×10^9 lần lặp

(s, 4)
(s, 2)
(4, 2)
(2, t)
(4, t)



Thời gian tính

Câu hỏi: Thuật toán Ford-Fulekerson có phải là thuật toán đa thức? (thuật toán với thời gian tính bị chặn bởi đa thức bậc cố định của độ dài dữ liệu vào)

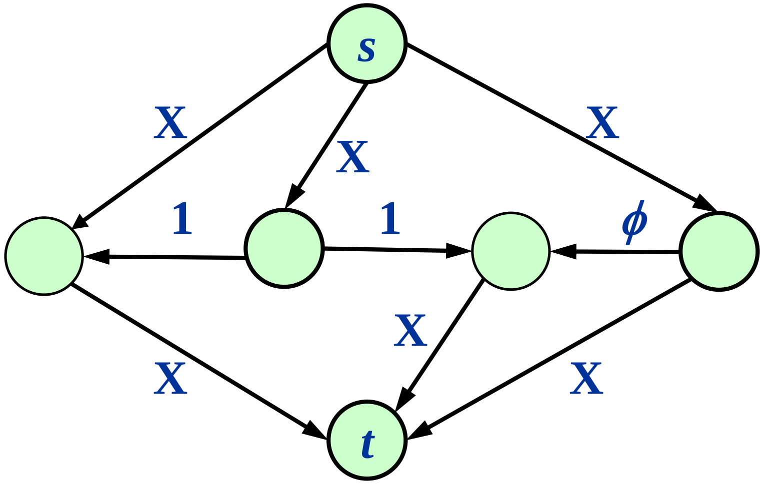
Trả lời: Không phải. Nếu khả năng thông qua lớn nhất là C thì thuật toán có thể phải thực hiện cỡ C bước lặp.

Khả năng thông qua của các cung là số thực thì tồn tại ví dụ cho thấy thuật toán Ford-Fulkerson không dừng.

Zwick xây dựng ví dụ cho thấy thuật toán có thể không dừng, nếu như khả năng thông qua là số vô tỷ

Ví dụ: Thuật toán không dừng

Zwick xây dựng ví dụ sau đây cho thấy thuật toán Ford-Fulkerson có thể không dừng, nếu như khả năng thông qua là số vô tỷ



Có 6 cung với khả năng thông qua X, 2 cung khả năng thông qua 1 và một cung khả năng thông qua

$$\phi = (\sqrt{5}-1)/2 \approx 0.618034...$$

Thời gian tính

Giả thiết: Tất cả các khả năng thông qua là các số nguyên trong khoảng từ 0 đến C.

Bất biến: Mỗi giá trị luồng $f(e)$ và mỗi khả năng thông qua $c_f(e)$ luôn luôn là số nguyên trong quá trình thực hiện thuật toán.

Định lý: Thuật toán dừng sau không quá val(f *) $\leq nC$ lần lặp.

Chứng minh.

Sau mỗi lần tăng luồng, giá trị của luồng tăng thêm ít nhất 1.

Hệ quả. Thời gian tính của thuật toán Ford-Fulkerson là $O(m.n.C)$

Hệ quả. Nếu $C = 1$, thì thuật toán đòi hỏi thời gian $O(mn)$.

Nội dung chi tiết

6.1. Phát biểu bài toán và các ứng dụng

6.2. Lát cắt

6.3. Đồ thị tăng luồng và Đường tăng luồng

6.4. Thuật toán Ford-Fulkerson

6.5. Thuật toán Edmond-Karp

6.6. Một số ứng dụng

Chọn đường tăng luồng như thế nào?

Cần hết sức cẩn thận khi lựa chọn đường tăng, bởi vì

- Một số cách chọn dẫn đến thuật toán hàm mũ.
- Cách chọn khôn khéo dẫn đến thuật toán đa thức.
- Nếu kntq là các số vô tỷ, thuật toán có thể không dừng

Mục đích: chọn đường tăng sao cho:

- Có thể tìm đường tăng một cách hiệu quả.
- Thuật toán đòi hỏi thực hiện càng ít bước lặp càng tốt.

Chọn đường tăng với

- khả năng thông qua lớn nhất. (đường béo - fat path)
- khả năng thông qua đủ lớn. (thang độ hoá kntq – capacity scaling)
- số cạnh trên đường đi là ít nhất. (đường ngắn nhất - shortest path)

Thuật toán Edmond-Karp **Dùng thuật toán BFS**

Thuật toán Edmonds – Karp

Edmonds and Karp, *JACM* 1972

- Nếu đường tăng được chọn là đường ngắn nhất từ s đến t, thì thời gian tính của thuật toán sẽ là $O(|E|^2\,|V|)$.

Thuật toán Ford-Fulkerson

```
float Ford_Fulkerson(G,c,s,t)
{
    FOR e ∈ E    // Khởi tạo luồng 0
        f(e) ← 0
    Gf ← đồ thị tăng luồng f

    WHILE (tìm được đường tăng luồng P)
    {   f ← augment(f, P)
        Cập nhật lại Gf
    }
    RETURN f
}
```

Tìm đường tăng luồng nhờ thực hiện BFS.

- Dễ thực hiện.
- Đường tăng có ít cạnh nhất.

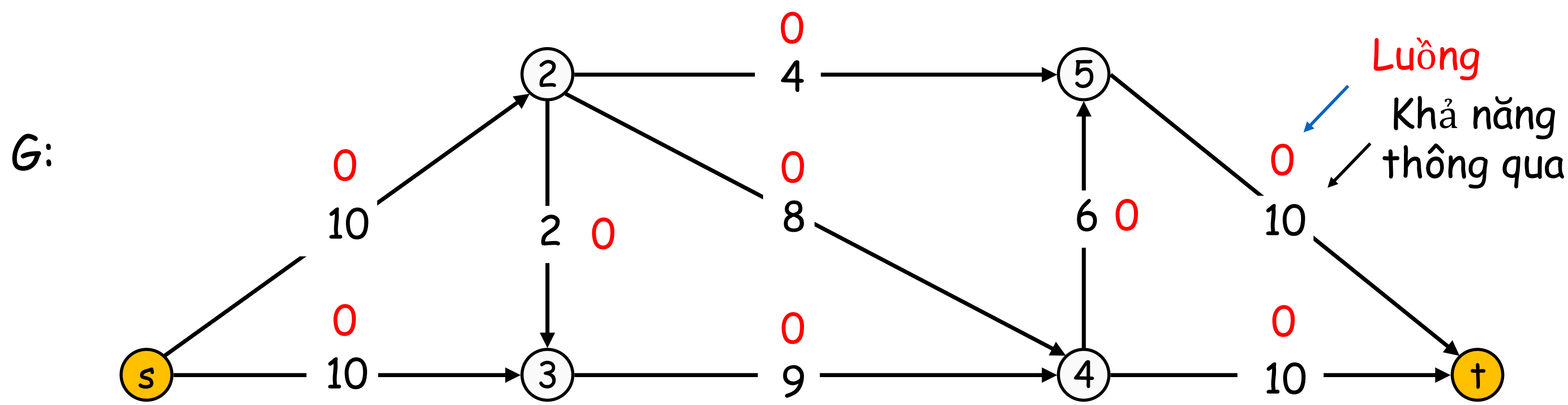
$$O(|E|^2 |V|)$$

Thuật toán Edmonds – Karp

```
FOR e ∈ E
    f(e) ← 0
Gf ← đồ thị tăng luồng (residual graph)

WHILE (tồn tại đường tăng luồng)
{
    tìm đường tăng luồng P bởi BFS
    f ← augment(f, P)
    hiệu chỉnh Gf
}
RETURN f
```

Ví dụ: Áp dụng thuật toán Edmonds-Karp, tìm luồng cực đại cho mạng sau



Giá trị luồng = 0

Xây dựng đồ thị tăng luồng G_f

Nội dung chi tiết

6.1. Phát biểu bài toán và các ứng dụng

6.2. Lát cắt

6.3. Đồ thị tăng luồng và Đường tăng luồng

6.4. Thuật toán Ford-Fulkerson

6.5. Thuật toán Edmond-Karp

6.6. Một số ứng dụng

Một số ứng dụng

- Một số bài toán luồng tổng quát
    - Bài toán với nhiều điểm phát và điểm thu
    - Bài toán với hạn chế thông qua ở nút
- Một số ứng dụng trong tổ hợp
    - Bài toán cặp ghép cực đại trong đồ thị hai phía
    - Độ tin cậy của mạng