CHƯƠNG 7
BÀI TOÁN VỀ MẠNG

Bùi Tiến Lên

Đại học Khoa học Tự nhiên TPHCM

1/1/2021

Nội dung

1. MẠNG LUỒNG
2. LÁT CẮT TRÊN MẠNG LUỒNG
3. LÁT CẮT TRÊN ĐỒ THỊ

Giới Thiệu

- Bài toán luồng cực đại trên mạng là một trong những bài toán tối ưu trên đồ thị và có những ứng dụng rất rộng rãi trong lý thuyết và thực tế
- Bài toán được đưa ra vào đầu những năm 1950 và gắn liền với tên tuổi của hai nhà toán học **Ford** và **Fulkerson**

MẠNG LUỒNG

Các định nghĩa về mạng luồng

Định nghĩa 7.1

Mạng luồng (flow network) (*mạng chuẩn*) là một đồ thị có hướng $N = (V, E, C)$ với V là tập đỉnh thường được gọi là **nút** (node), E là tập cạnh thường được gọi là **cung** (arc) và C là hàm khả năng trong đó

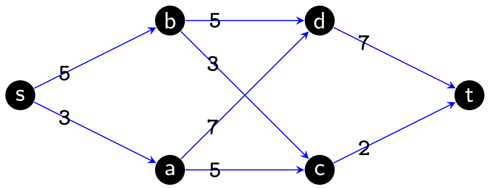
- Có duy nhất một nút s không có cung đi vào, được gọi là **nút phát** (source)
- Có duy nhất một nút t không có cung đi ra, được gọi là **nút thu** (sink)
- Mỗi cung $e = (u, v) \in E$ được gán một số thực không âm $c(u, v)$ được gọi là **khả năng thông qua** (capacity) của cung

Các định nghĩa về mạng luồng (cont.)

Định nghĩa 7.1

- Tập hợp $W^-(v)$ là tập các cung đi vào nút v
- Tập hợp $W^+(v)$ là tập các cung đi ra nút v

Các định nghĩa về mạng luồng (cont.)



Hình 7.1: Mạng gồm 6 nút, 8 cung, đỉnh phát s và đỉnh thu t

Các định nghĩa về luồng

Định nghĩa 7.2

Cho mạng $N = (V, E, C, s, t)$, **Luồng** (flow) f trong mạng là một ánh xạ

$$\begin{array}{cccc} f: & V^2 & \to & R^+ \\ & (u, v) & \mapsto & f(u, v) \end{array} \tag{7.1}$$

thỏa mãn được hai điều kiện sau

- **Điều kiện 1** (capacity constraint): luồng trên mỗi cung (u, v) không vượt quá khả năng thông qua của cung

$$0 \leq f(u, v) \leq c(u, v) \tag{7.2}$$

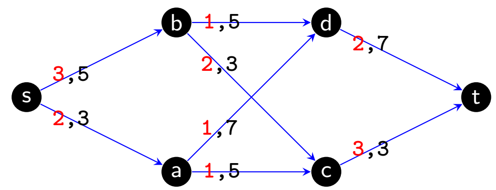
Các định nghĩa về luồng (cont.)

Định nghĩa 7.2

- ***Điều kiện 2*** *(**flow conversion**): điều kiện cân bằng luồng trên mỗi đỉnh v (trừ đỉnh thu và đỉnh phát) là tổng luồng trên các cung đi vào bằng tổng luồng trên các cung đi ra*

$$\sum_{e \in W^{+}(v)} f(e) = \sum_{e \in W^{-}(v)} f(e) \tag{7.3}$$

Các định nghĩa về luồng (cont.)



Hình 7.2: Một luồng f trên mạng, số thứ nhất trên cung là giá trị luồng, số thứ hai trên cung là khả năng thông qua

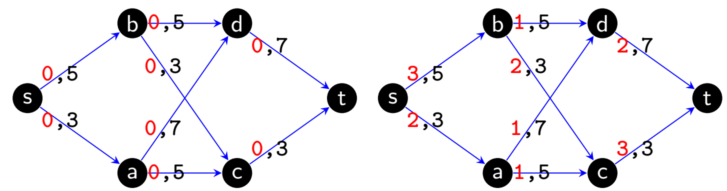
Các định nghĩa về luồng (cont.)

Định nghĩa 7.3

Cho một mạng $N = (V, E, C, s, t)$ và một luồng f trên mạng. Giá trị của một luồng trên mạng được định nghĩa là bằng tổng giá trị luồng các cung đi ra từ đỉnh phát hoặc tổng giá trị luồng các cung đi vào đỉnh thu

$$f(N) = \sum_{e \in W^{+}(s)} f(e) = \sum_{e \in W^{-}(t)} f(e) \tag{7.4}$$

Các định nghĩa về luồng (cont.)



(a) Luồng có giá trị 0 **(b)** Luồng có giá trị 5

Hình 7.3: Hai luồng khác nhau trên cùng một mạng

Bài toán luồng cực đại

Bài toán 7.1

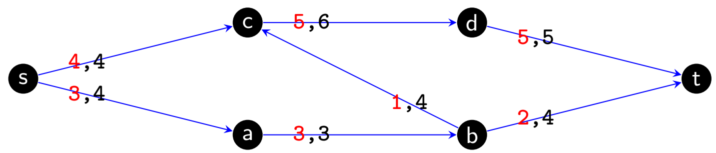
Cho một mạng $N = (V, E, C, s, t)$, gọi $\mathcal{F}$ là tập hợp các luồng trên mạng N. Bài toán tìm **luồng cực đại** (**max flow**) được phát biểu qua công thức sau

$$f = \arg\max_{f} \left(f\left(N\right), f \in \mathcal{F} \right) \tag{7.5}$$

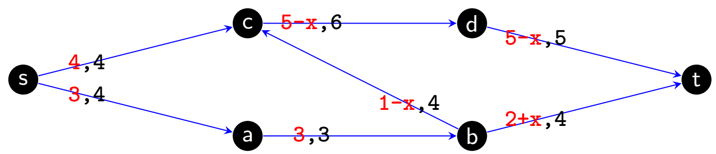
Lưu ý

Nói chung, tồn tại vô số luồng cực đại

Bài toán luồng cực đại (cont.)



Hình 7.4: Luồng cực đại



Hình 7.5: Luồng cực đại với $\forall x, 0 \leq x \leq 1$

Định lý về tính nguyên

Định lý 7.1

Nếu khả năng thông qua của các cung trên mạng đều là số nguyên thì luồng cực đại cũng sẽ có giá trị nguyên.

Chứng minh

Sinh viên đọc tài liệu [Trần and Dương, 2013] ■

Thuật toán Ford-Fulkerson tìm luồng cực đại

Cho mạng $N = (V, E, C, s, t)$, thuật toán gồm hai bước

Algorithm 1 Thuật toán Ford-Fulkerson

- **Bước khởi tạo**: Luồng f bắt đầu với giá trị trên các cung là 0
- **Bước lặp**:
  - Xây dựng đồ thị tăng luồng N_f
  - Tìm đường tăng luồng P trên đồ thị tăng luồng N_f
  - Dùng đường tăng luồng P để cập nhật giá trị luồng f
- Bước lặp kết thúc khi không tìm được đường tăng luồng và f cuối cùng chính là luồng cực đại

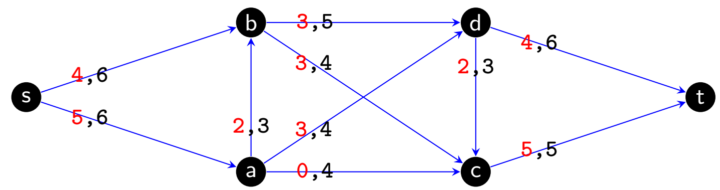
Đồ thị tăng luồng

Định nghĩa 7.4

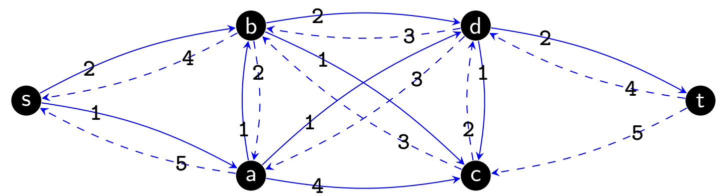
Cho mạng $N = (V, E, C, s, t)$ và f là luồng trên mạng. **Đồ thị tăng luồng** (residual network) là đồ thị có hướng có trọng số $N_f = (V, E_f)$ với các cung được xây dựng như sau:
Xét một cung $(u, v) \in E$

- Nếu $f(u, v) = 0$ thì $(u, v) \in E_f$ với trọng số $c(u, v)$
- Nếu $f(u, v) = c(u, v)$ thì $(v, u) \in E_f$ với trọng số $c(u, v)$
- Nếu $0 < f(u, v) < c(u, v)$ thì
  - $(u, v) \in E_f$ với trọng số $c(u, v) - f(u, v)$
  - $(v, u) \in E_f$ với trọng số $f(u, v)$
- Các cung thuộc N_f cũng thuộc N được gọi là **cung thuận**
- Các cung thuộc N_f không thuộc N được gọi là **cung nghịch**

Đồ thị tăng luồng (cont.)



Hình 7.6: Luồng f



Hình 7.7: Đồ thị tăng luồng N_f, các cung thuận vẽ liền, các cung nghịch vẽ gạch đứt

Đường tăng luồng

Định nghĩa 7.5

Gọi $P = (s = v_0, v_1, ..., v_k = t)$ là một đường đi từ s đến t trên đồ thị tăng luồng N_f được gọi là **đường tăng luồng** (augmenting path) cho luồng f.

Cập nhật đường tăng luồng

Gọi k là trọng số của cung nhỏ nhất. Cập nhật luồng f cho các cung trên đường đi P. Luồng cập nhật gọi là f^{new}

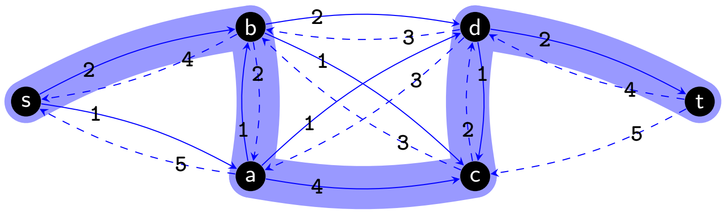
- $f^{new}(u, v) = f(u, v) + k$ nếu $(u, v) \in P$ là cung thuận
- $f^{new}(u, v) = f(u, v) - k$ nếu $(u, v) \in P$ là cung nghịch

Sau khi cập nhật ta có

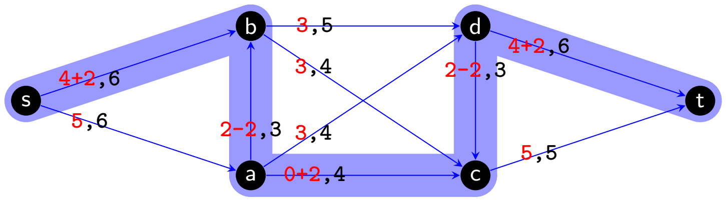
$$f^{new}(N) = f(N) + k$$

Nghĩa là luồng mới đã được tăng thêm k đơn vị so với luồng cũ

Cập nhật đường tăng luồng (cont.)



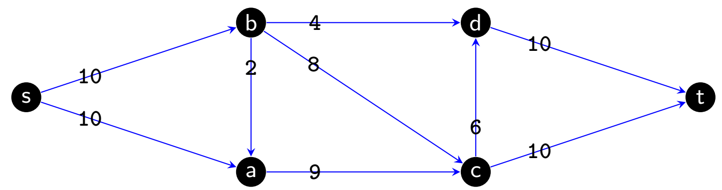
Hình 7.8: Đường đi từ đỉnh s đến t và có giá trị $k = 2$ trên N_f



Hình 7.9: Cập nhật các cung của luồng f nằm trên đường đi

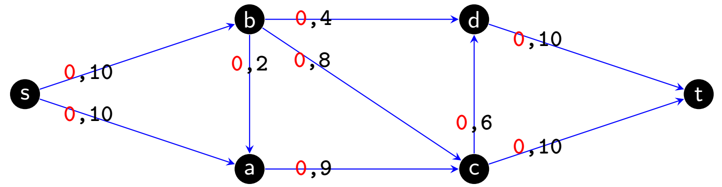
Minh họa thuật toán Ford-Fulkerson

- Cho mạng dưới, hãy tìm luồng cực đại

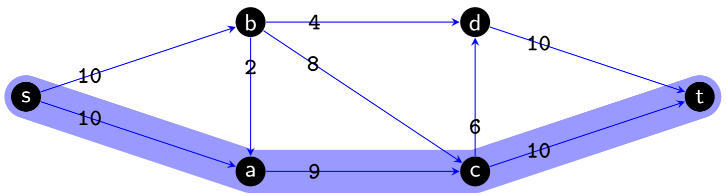


Hình 7.10: Mạng 6 nút 9 cung

Minh họa thuật toán Ford-Fulkerson (cont.)

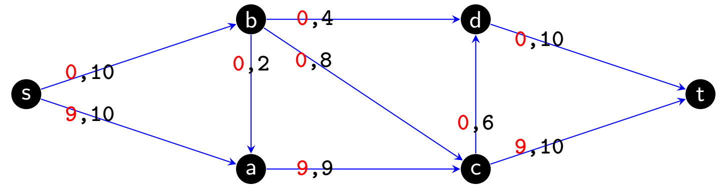


Hình 7.11: Luồng khởi tạo f

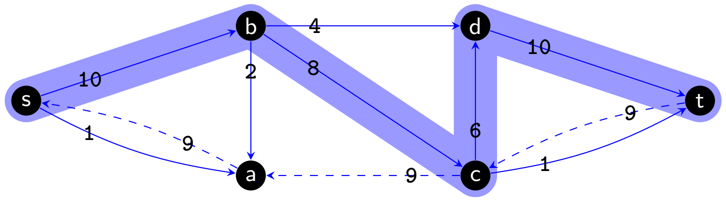


Hình 7.12: Đồ thị tăng luồng và đường đi tăng luồng

Minh họa thuật toán Ford-Fulkerson (cont.)

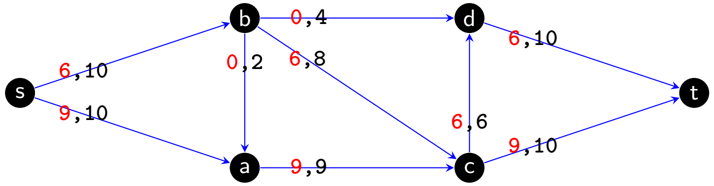


Hình 7.13: Luồng cập nhật f

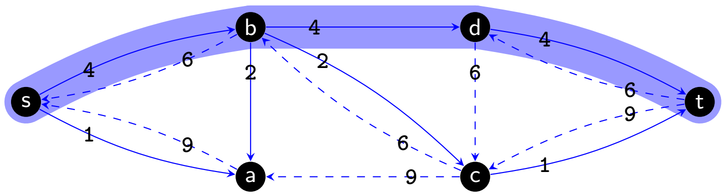


Hình 7.14: Đồ thị tăng luồng và đường đi tăng luồng

Minh họa thuật toán Ford-Fulkerson (cont.)

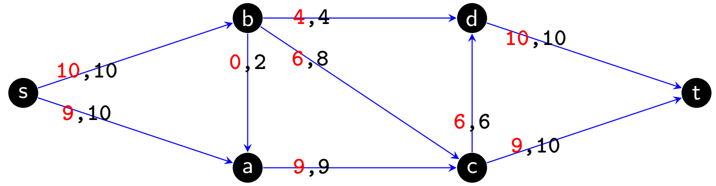


Hình 7.15: Luồng cập nhật f

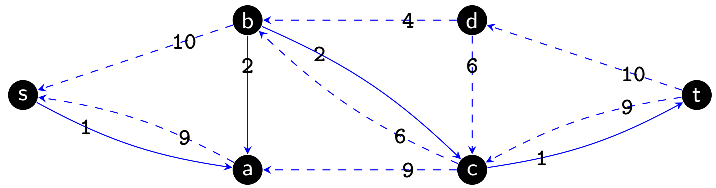


Hình 7.16: Đồ thị tăng luồng và đường đi tăng luồng

Minh họa thuật toán Ford-Fulkerson (cont.)

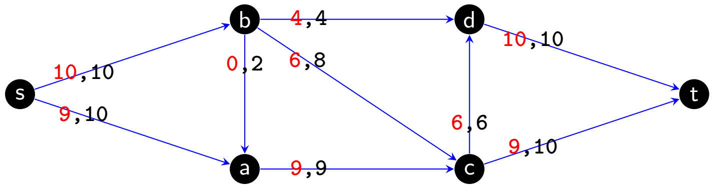


Hình 7.17: Luồng cập nhật f



Hình 7.18: Đồ thị tăng luồng và đường đi tăng luồng

Minh họa thuật toán Ford-Fulkerson (cont.)



Hình 7.19: Luồng cực đại f với giá trị cực đại là 19

Một số vấn đề

Khi cài đặt thuật toán Ford-Fulkerson ta lưu ý những điểm sau

- Có cần đồ thị tăng luồng không?
  - Xây dựng đồ thị tăng luồng
  - Không xây dựng đồ thị tăng luồng
- Tìm đường đi tăng luồng như thế nào?
  - Tìm đường đi bằng DFS
  - Tìm đường đi bằng BFS
  - Tìm đường đi "tối ưu"

Một số vấn đề (cont.)

Lưu ý

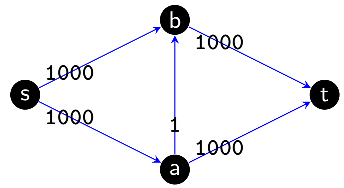
Phải chọn đường đi tăng luồng cẩn thận vì có những đường đi

- Làm cho thuật toán chạy lâu
- Làm cho thuật toán không bao giờ kết thúc

Một số vấn đề (cont.)

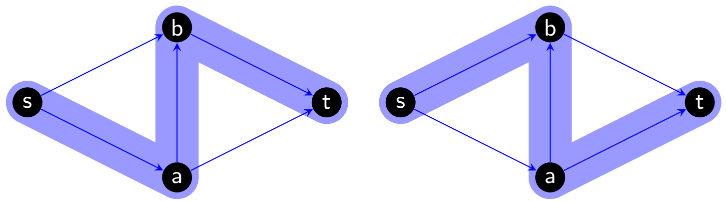
Ví dụ 7.1

Cho mạng như hình dưới hãy lần lượt hiện việc tăng luồng trên mạng theo hai đường tăng luồng P_1 và P_2 như sau: $P_1, P_2, P_1, P_2, \ldots$



Hình 7.20: Mạng 4 nút và 5 cung

Một số vấn đề (cont.)



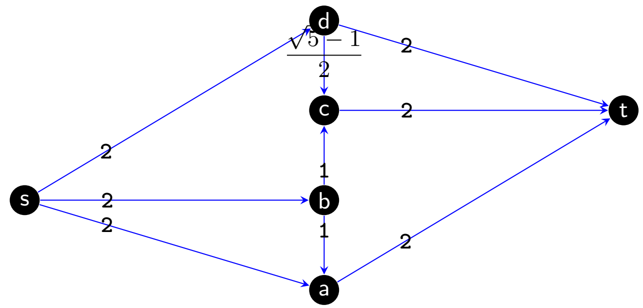
(a) Đường tăng luồng P_1 s-a-b-t **(b)** Đường tăng luồng P_2 s-b-a-t

Hình 7.21: Hai đường tăng luồng

Một số vấn đề (cont.)

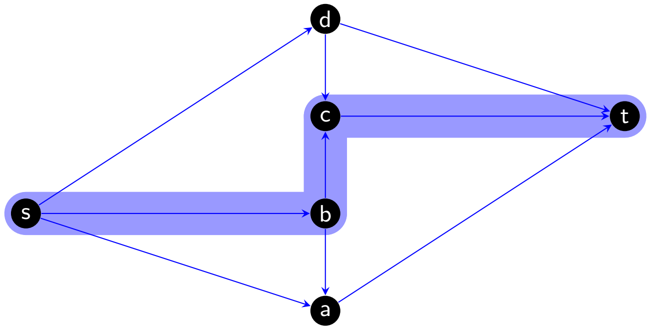
Ví dụ 7.2

Cho mạng hình dưới hãy lần lượt thực hiện việc tăng luồng theo các đường tăng luồng sau: $P_0, P_1, P_2, P_1, P_3, P_1, P_2, P_1, P_3, \ldots$



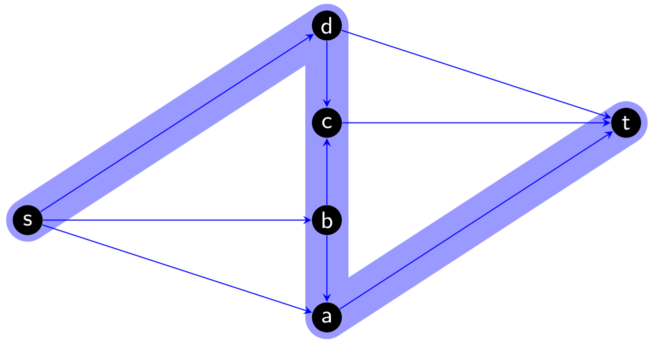
Hình 7.22: Mạng 6 nút

Một số vấn đề (cont.)



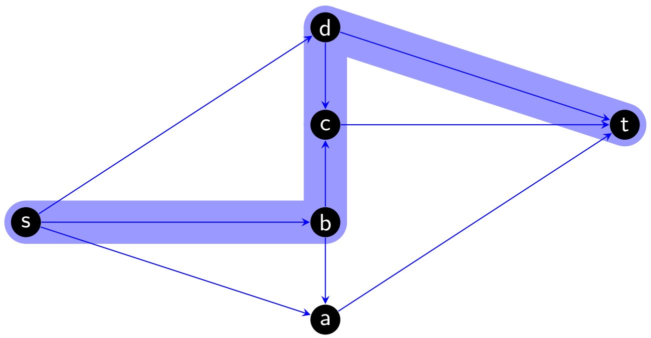
Hình 7.23: Đường tăng luồng P_0

Một số vấn đề (cont.)



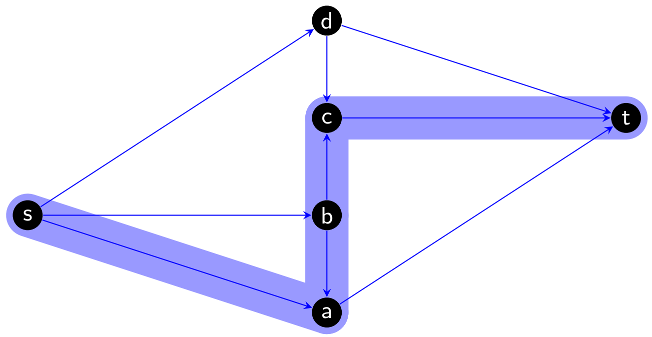
Hình 7.24: Đường tăng luồng P_1

Một số vấn đề (cont.)



Hình 7.25: Đường tăng luồng P_2

Một số vấn đề (cont.)



Hình 7.26: Đường tăng luồng P_3

Cải tiến thuật toán Ford-Fulkerson

Một trong những cải tiến là chỉ tính toán trên đồ thị tăng luồng mở rộng. Cho mạng $N = (V, E, C, s, t)$

Algorithm 2 Thuật toán cải tiến

- **Bước khởi tạo**: khởi tạo đồ thị tăng luồng mở rộng N_f
- **Bước lặp**:
  - Tìm đường tăng luồng P trên đồ thị tăng luồng mở rộng N_f
  - Dùng đường tăng luồng P để cập nhật trọng số đồ thị tăng luồng mở rộng N_f
- Bước lặp kết thúc khi không tìm được đường tăng luồng và xác định luồng cực đại f từ đồ thị tăng luồng mở rộng cuối cùng N_f

Cải tiến thuật toán Ford-Fulkerson (cont.)

Định nghĩa 7.6

Cho mạng $N = (V, E, C, s, t)$ và f là luồng trên mạng. **Đồ thị tăng luồng mở rộng** là đồ thị có hướng có trọng số $N_f = (V, E_f)$ với các cung được xây dựng như sau: Nếu $(u, v) \in E$ thì

- $(u, v) \in E_f$ với trọng số $c(u, v)$
- $(v, u) \in E_f$ với trọng số 0

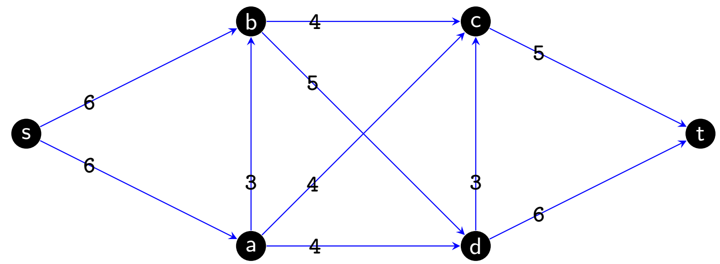
Cải tiến thuật toán Ford-Fulkerson (cont.)

Gọi P là một đường đi tăng luồng từ s đến t trên đồ thị tăng luồng mở rộng N_f (*Lưu ý đường đi không đi qua cung có trọng số bằng 0*). Gọi k là trọng số của cung nhỏ nhất. Cập nhật trọng số cho đồ thị tăng luồng mở rộng. Gọi (u, v) là cung thuộc P

- $N_f(u, v) = N_f(u, v) - k$
- $N_f(v, u) = N_f(v, u) + k$

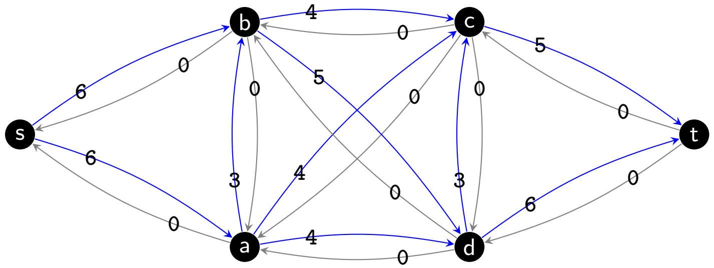
Minh họa thuật toán

- Tìm luồng cực đại cho mang sau



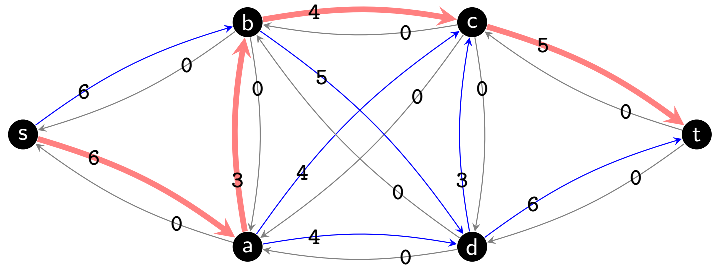
Hình 7.27: Mạng 6 nút và 10 cung

Minh họa thuật toán (cont.)



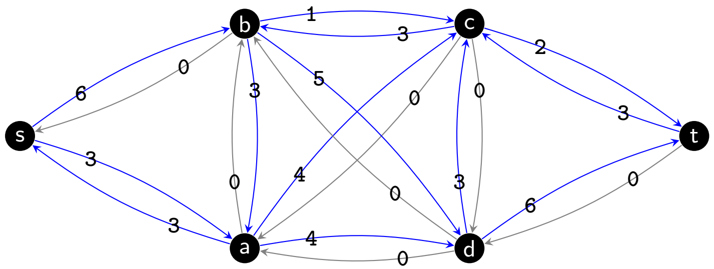
Hình 7.28: Khởi tạo đồ thị tăng luồng

Minh họa thuật toán (cont.)



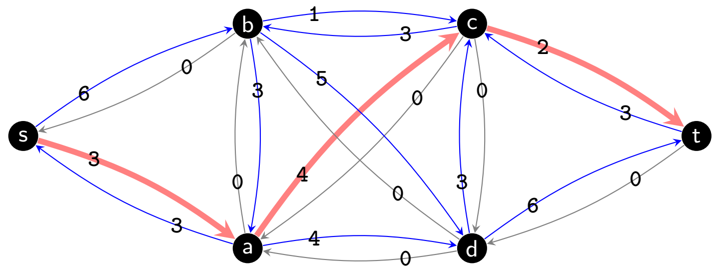
Hình 7.29: Đường đi tăng luồng

Minh họa thuật toán (cont.)



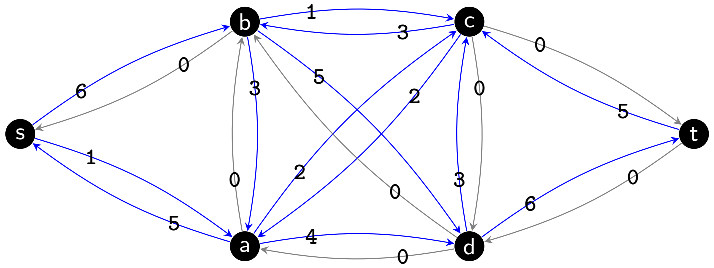
Hình 7.30: Cập nhật đồ thị tăng luồng

Minh họa thuật toán (cont.)



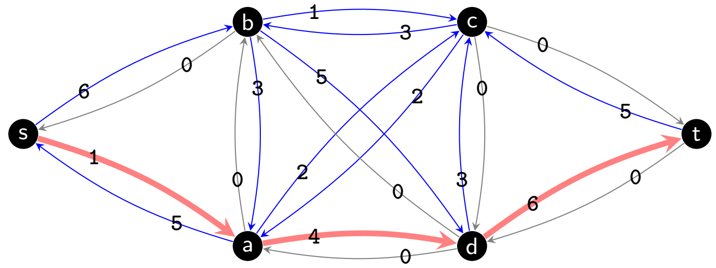
Hình 7.31: Đường đi tăng luồng

Minh họa thuật toán (cont.)



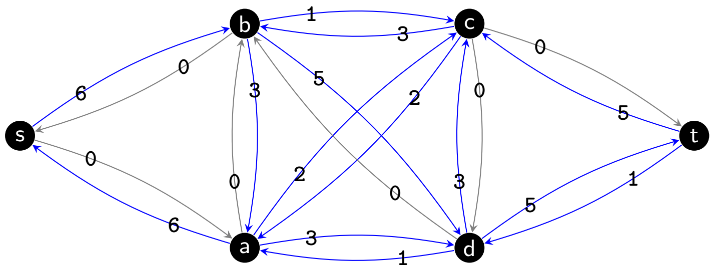
Hình 7.32: Cập nhật đồ thị tăng luồng

Minh họa thuật toán (cont.)



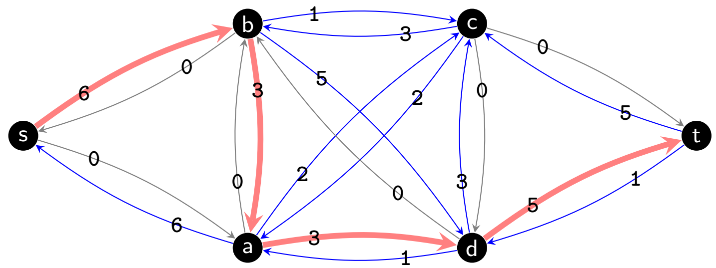
Hình 7.33: Đường đi tăng luồng

Minh họa thuật toán (cont.)



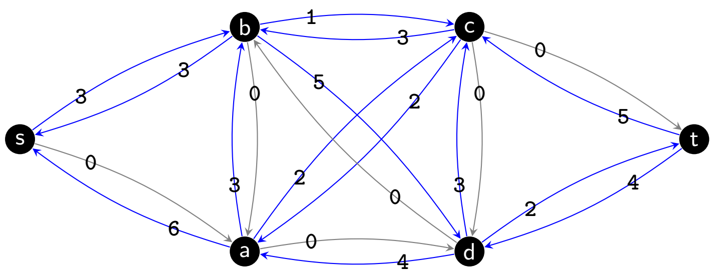
Hình 7.34: Cập nhật đồ thị tăng luồng

Minh họa thuật toán (cont.)



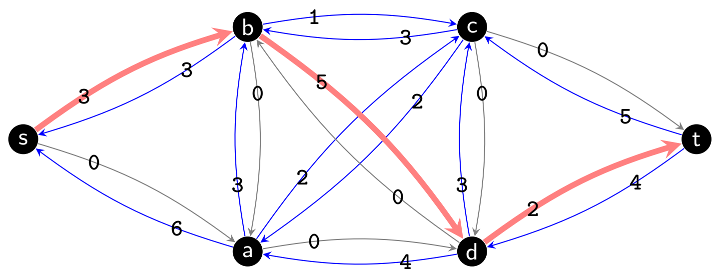
Hình 7.35: Đường đi tăng luồng

Minh họa thuật toán (cont.)



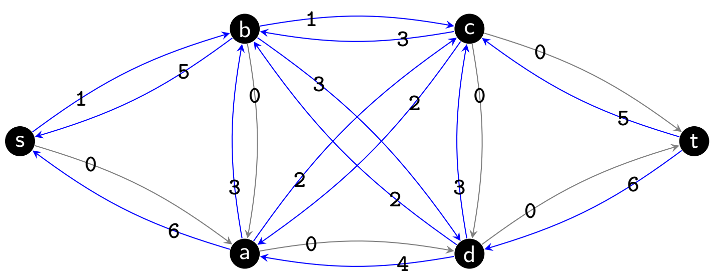
Hình 7.36: Cập nhật đồ thị tăng luồng

Minh họa thuật toán (cont.)



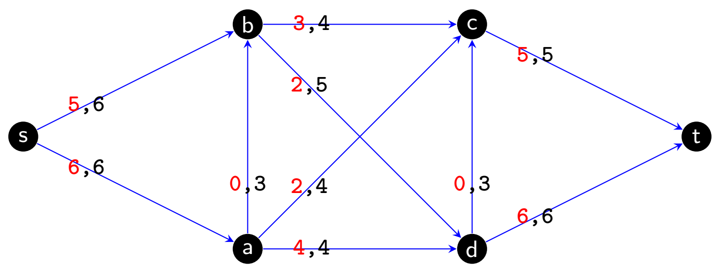
Hình 7.37: Đường đi tăng luồng

Minh họa thuật toán (cont.)



Hình 7.38: Cập nhật đồ thị tăng luồng

Minh họa thuật toán (cont.)



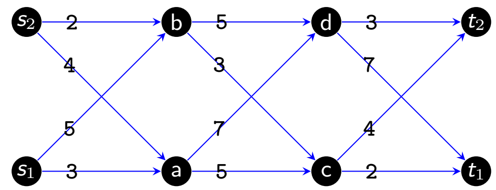
Hình 7.39: Luồng cực đại

Các dạng mở rộng

Mạng chuẩn có thể mở rộng thành

- Mạng với **nhiều điểm phát hoặc nhiều điểm thu** (multiple sources or sinks)
- Mạng với **khả năng thông qua nút** (vertex capacity)
- Mạng với khả năng thông qua cung bị chặn hai phía

Mạng với nhiều điểm phát hoặc nhiều điểm thu



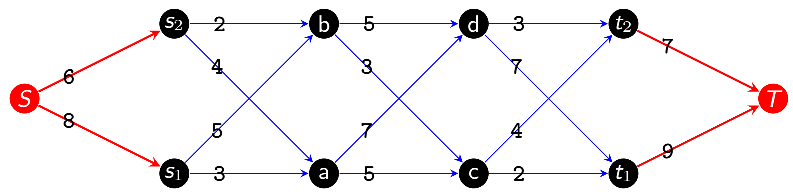
Hình 7.40: Mạng gồm 2 nguồn phát và 2 nguồn thu

Mạng với nhiều điểm phát hoặc nhiều điểm thu (cont.)

Mạng với nhiều nguồn phát $s_1, s_2, ..., s_m$ và nhiều nguồn thu $t_1, t_2, ..., t_n$ sẽ được biến đổi thành mạng chuẩn như sau

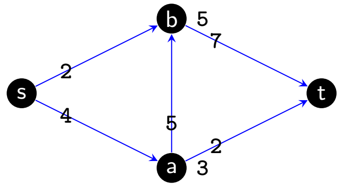
- Thêm vào một **nguồn phát giả** (dummy source) S nối với tất cả các nguồn phát s_i
  - Khả năng thông qua của cung giữa S và s_i là tổng khả năng phát của s_i
- Thêm vào một **nguồn thu giả** (dummy sink) T nối với tất cả các nguồn thu t_j
  - Khả năng thông qua của cung giữa t_i và T là tổng khả năng thu của t_j

Mạng với nhiều điểm phát hoặc nhiều điểm thu (cont.)



Hình 7.41: Biến đổi mạng gồm 2 nguồn phát và 2 nguồn thu thành mạng chuẩn gồm 1 nguồn phát và 1 nguồn thu

Mạng với khả năng thông qua nút



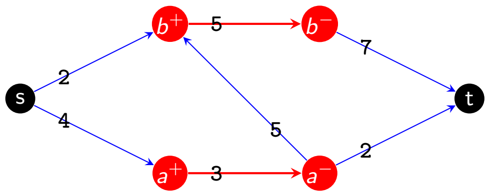
Hình 7.42: Mạng có nút a và b bị hạn chế khả năng thông qua

Mạng với khả năng thông qua nút (cont.)

Mạng với nút bị hạn chế khả năng thông qua sẽ được biến đổi như sau

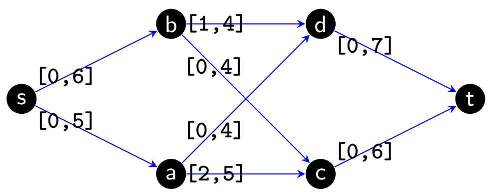
- Đối với nút u bị hạn chế khả năng thông qua sẽ được tách thành hai đỉnh u^+ và u^-
- Các cung đi vào u sẽ đi vào u^+
- Các cung đi ra u sẽ đi ra u^-
- Một cung nối u^+ và u^- sẽ có khả năng thông qua là khả năng thông qua của đỉnh u

Mạng với khả năng thông qua nút (cont.)



Hình 7.43: Biến đổi mạng có nút a và b bị hạn chế khả năng thông qua thành mạng chuẩn

Mạng với khả năng thông qua cung bị chặn hai phía



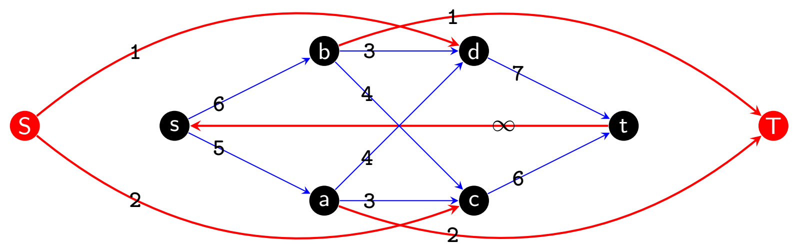
Hình 7.44: Mạng với cung bị chặn hai phía, mỗi cung sẽ có hai giá trị: giá trị thứ nhất là khả năng thông qua tối thiểu, giá trị thứ hai là khả năng thông qua tối đa

Mạng với khả năng thông qua cung bị chặn hai phía (cont.)

Mạng với khả năng thông qua của cung (u, v) bị giới hạn bởi $[c_{min}, c_{max}]$ sẽ được biến đổi như sau

- Thêm cung (t, s) với khả năng thông qua của cung là ∞
- Thêm vào nguồn phát S và nguồn thu T
- Với cung (u, v) có $c_{min}(u, v) > 0$ thì
  - Thêm cung (S, v) với khả năng thông qua là $c_{min}(u, v)$
  - Thêm cung (u, T) với khả năng thông qua là $c_{min}(u, v)$
  - Cập nhật lại khả năng thông qua cho cung (u, v) là $c(u, v) = c_{max}(u, v) - c_{min}(u, v)$

Mạng với khả năng thông qua cung bị chặn hai phía (cont.)



Hình 7.45: Mạng với cung bị chặn hai phía được biến đổi thành mạng chuẩn

Mạng với khả năng thông qua cung bị chặn hai phía (cont.)

Định lý 7.2

- *Nếu giá trị luồng cực đại trong mạng N' bằng $\sum_{e \in E} c_{min}(e)$ thì tồn tại luồng cực đại tương ứng trong N*
- *Nếu giá trị luồng cực đại trong mạng N' không bằng $\sum_{e \in E} c_{min}(e)$ thì không tồn tại luồng cực đại tương ứng trong N*

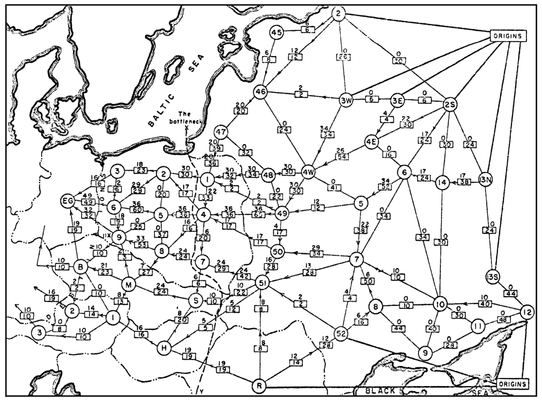
Chứng minh

Sinh viên tự chứng minh ■

Các ứng dụng của bài toán luồng cực đại

Bài toán luồng cực đại có nhiều ứng dụng thực tế. Có thể áp dụng nó để giải quyết

- Bài toán ghép cặp
- Bài toán chuyển tải nguyên liệu: điện, nước hay dầu...



LÁT CẮT TRÊN MẠNG LUỒNG

Các định nghĩa về lát cắt

Định nghĩa 7.7

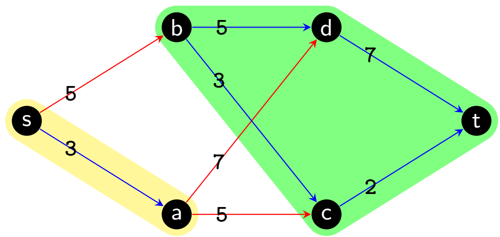
Cho một mạng $N = (V, E, C, s, t)$

- Một **lát cắt** (cut) (S, T) là một phân hoạch tập đỉnh V sao cho $s \in S$ và $t \in T$
- Một **tập cắt** (cut-set) là một tập các cung $\{(u, v) \in E | u \in S, v \in T\}$
- Khả năng thông qua của lát cắt (S, T) được định nghĩa là

$$c(S, T) = \sum_{u \in S, v \in T} c(u, v) \tag{7.6}$$

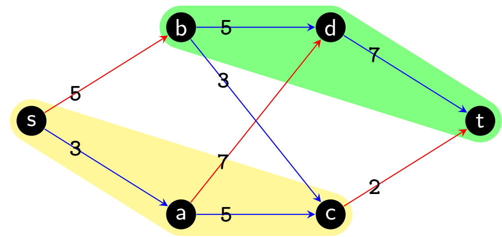
- **Lát cắt nhỏ nhất** (min cut) là lát cắt có khả năng thông qua nhỏ nhất

Các định nghĩa về lát cắt (cont.)



Hình 7.46: Mạng có lát cắt (S={s,a},T={b,c,d,t}), tập cắt là {(s,b), (a,c), (a,d)}, khả năng thông qua của lát cắt là 17

Các định nghĩa về lát cắt (cont.)



Hình 7.47: Mạng có lát cắt (S={s,a,c},T={b,d,t}), tập cắt là {(s,b), (a,d), (c,t)}, khả năng thông qua của lát cắt là 14

Bài toán tìm lát cắt nhỏ nhất

Bài toán 7.2

Cho một mạng $N = (V, E, C, s, t)$. Hãy tìm lát cắt nhỏ nhất

Tự học

Sinh viên hãy tự nghiên cứu ■

Mối quan hệ giữa luồng và lát cắt

Cho một mạng $N = (V, E, C, s, t)$. Ta có những nhận xét sau

- Giá trị của một luồng f trong mạng bất kỳ luôn nhỏ hơn giá trị khả năng thông qua của một lát cắt (S, T) bất kỳ trong mạng
$$f(N) \leq c(S, T) \tag{7.7}$$
- Giá trị luồng cực đại trong mạng không vượt quá khả năng thông qua của lát cắt nhỏ nhất trong mạng

Mối quan hệ giữa luồng và lát cắt (cont.)

Định lý 7.3

Giá trị luồng cực đại trong mạng bằng giá trị khả năng thông qua của lát cắt nhỏ nhất

Chứng minh

Sinh viên tự chứng minh ■

Mối quan hệ giữa luồng và lát cắt (cont.)

Định lý 7.4

Các mệnh đề dưới đây là tương đương:

1. *f là luồng cực đại*
2. *Không tìm được đường tăng luồng P cho f*
3. *Tồn tại một lát cắt (S, T) sao cho $c(S, T) = f(N)$*

Chứng minh

Sinh viên đọc tài liệu [Trần and Dương, 2013] ■

LÁT CẮT TRÊN ĐỒ THỊ

Các định nghĩa

Định nghĩa 7.8

Cho một đồ thị $G = (V, E)$

- Một **lát cắt** (cut) (S, T) là một phân hoạch tập đỉnh V
- Một **tập cắt** (cut-set) là một tập các cạnh $\{(u, v) \in E | u \in S, v \in T\}$
- **Lát cắt nhỏ nhất** (min cut) là lát cắt có tổng số cạnh ít nhất (đối với đồ thị không có trọng số) hoặc tổng trọng số nhỏ nhất (đối với đồ thị có trọng số).

Thuật toán Karger

- **Bài toán**: Cho một đồ thị $G = (V, E)$

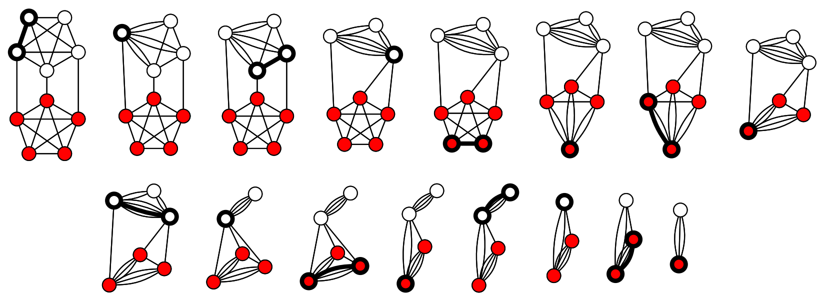
Algorithm 3 Thuật toán ngẫu nhiên Karger

```
function CONTRACT(G(V, E))
    while |V| > 2 do
        chọn ngẫu nhiên một cạnh e ∈ E
        sử dụng biến đổi co cạnh e: G ← G/e
    return đồ thị G với 2 đỉnh
```

- **Lưu ý**: thuật toán chỉ đảm bảo lời giải gần tốt nhất. Do đây là thuật toán ngẫu nhiên do đó sẽ được chạy nhiều lần để tăng khả năng có lời giải tốt nhất.

Minh họa thuật toán

- Cho đồ thị 10 đỉnh, kết quả chạy được đồ thị 2 đỉnh có mincut $= 3$ (đây cũng là kết quả tốt nhất)



Tài liệu tham khảo

- Diestel, R. (2005).
  Graph theory. 2005.
  Springer-Verlag.
- Moore, E. F. (1959).
  The shortest path through a maze.
  Bell Telephone System.
- Rosen, K. H. and Krithivasan, K. (2012).
  Discrete mathematics and its applications.
  McGraw-Hill New York.
- Tarjan, R. (1972).
  Depth-first search and linear graph algorithms.
  SIAM journal on computing, 1(2):146–160.

Tài liệu tham khảo (cont.)

- Trần, T. and Dương, D. (2013).
  Giáo trình lý thuyết đồ thị. 2013.
  NXB Đại Học Quốc Gia TPHCM.
- West, D. B. et al. (2001).
  Introduction to graph theory.
  Prentice hall Englewood Cliffs.