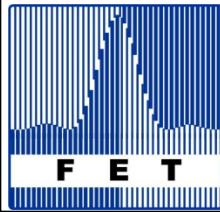
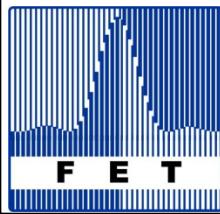


Thiết kế mạng Backbone



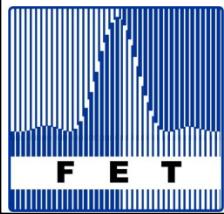
Nội dung

- Giới thiệu
- Giải thuật MENTOR
- Giải thuật MENTOR II



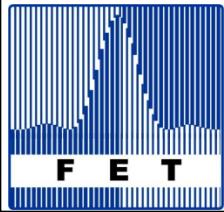
Giới thiệu

- Cây không phải là thiết kế được lựa chọn cho mạng xương sống. Lý do?
 - Muốn có nhiều đường riêng biệt từ nguồn đến đích
 - Muốn các thành phần được sử dụng tốt
 - Muốn sử dụng các liên kết dung lượng cao để kinh tế
- Thiết kế mắt lưới thường là thích hợp hơn
 - Thiết kế xương sống \cong thiết kế mắt lưới



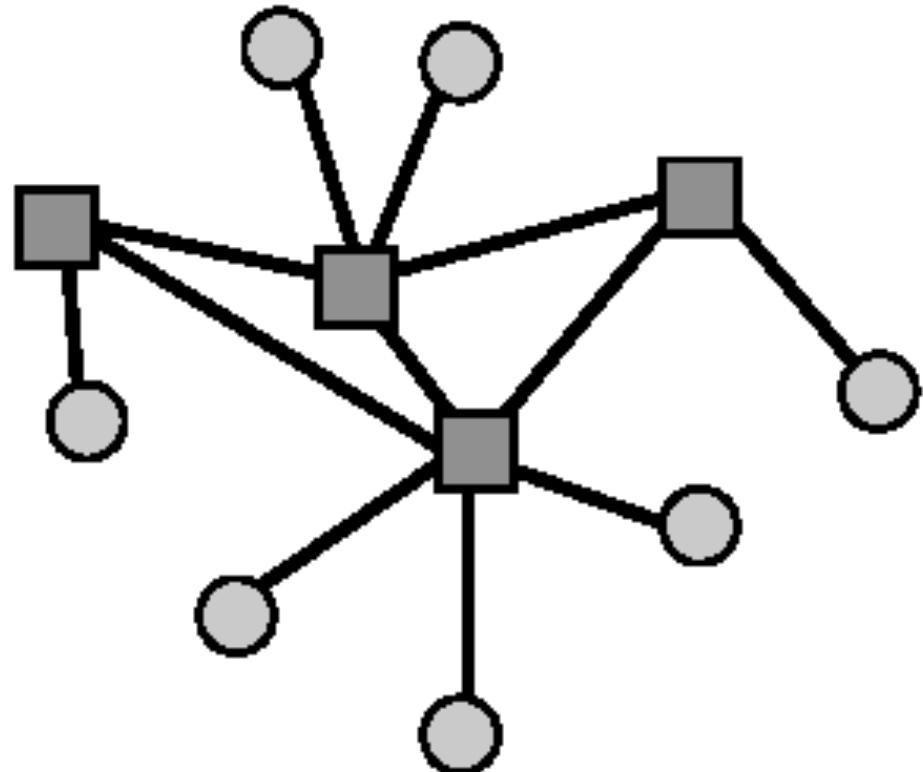
Thiết kế mắt lưới

- Mạng xương sống phải là mắt lưới, không phải là cây
 - Nhưng không phải mắt lưới đầy đủ (quá đắt tiền)
 - Những mục tiêu đề cập ở slide trước là trái ngược nhau, cần phải thực hiện thoả hiệp
 - Cấu hình mắt lưới đưa ra vấn đề định tuyến (cây chỉ có đường duy nhất)
 - Làm sao có thể đánh giá thiết kế mắt lưới ?



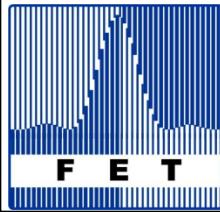
Ví dụ

- Xác định các nút trung tâm (xương sống)
- Nối các nút xương sống
- Nối các nút còn lại



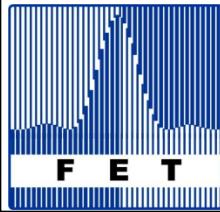
Thuật toán MENTOR

- Thuật toán có “chất lượng cao, độ phức tạp thấp”
- Gốc ban đầu là phát triển cho ghép kênh phân chia thời gian
- Nhưng vẫn có thể làm việc với các công nghệ khác



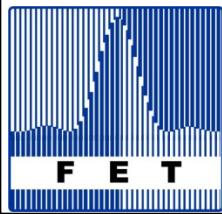
Thuật toán MENTOR (2)

- Giả sử ban đầu chỉ có một loại liên kết có dung lượng C
- Chia các nút thành nút **xương sống** và nút đầu cuối
 - Nút xương sống là những điểm tập hợp. Một vài thuật toán có thể thực hiện điều này
 - Hợp lại theo ngưỡng được sử dụng



Hợp lại theo ngưỡng

- Trọng số của một nút là tổng của tất cả các lưu lượng vào và ra nút
- Trọng số chuẩn hoá của nút i là
 $NW(i) = W(i)/C$
- Nút có $NW(i) > W$ được chọn làm nút xương sống
 - Trong đó W là tham số

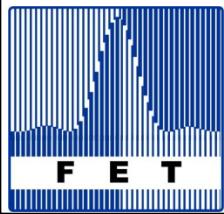


Hợp lại theo ngưỡng (2)

- Tất cả các nút không thoả mãn với tiêu chuẩn trọng số và gần nút xương sống sẽ được chọn làm nút đầu cuối
 - “Gần” được định nghĩa là khi giá liên kết từ nút đầu cuối e đến nút xương sống là nhỏ hơn một phần của giá liên kết lớn nhất

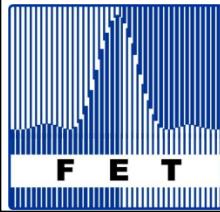
$$\text{MAXCOST} = \max_{i,j} \text{cost}(N_i, N_j):$$

$$\text{cost}(e, N_i) < \text{MAXCOST} * \text{RPARAM}$$



Hợp lại theo ngưỡng (3)

- Đối với những nút còn lại, xác định “thưởng” (merit) cho mỗi nút
 - Xác định toạ độ cho mỗi nút
 - Tính trung tâm của trọng lực của các nút



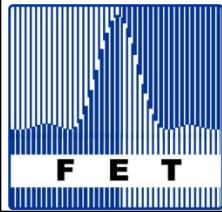
Trung tâm của trọng lực (CG)

- Định nghĩa (x_{ctr} , y_{ctr}) trong đó

$$x_{ctr} = \sum_n x_n W_n / \sum W_n$$

$$y_{ctr} = \sum_n y_n W_n / \sum W_n$$

Chú ý : toạ độ này không cần tương ứng với bất cứ nút thực tế



Khoảng cách tới CG

- **Định nghĩa**

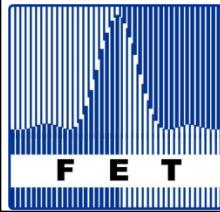
$$dc_n = [(x_n - x_{ctr})^2 + (y_n - y_{ctr})^2]^{0.5}$$

$$\text{maxdc} = \max(dc_n)$$

$$\text{maxW} = \max(W_n)$$

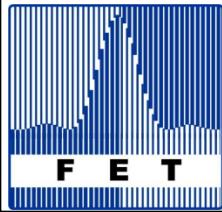
- Do đó

$$\text{merit}_n = 0.5(\text{maxdc} - dc_n)/\text{maxdc} + 0.5(W_n/\text{maxW})$$



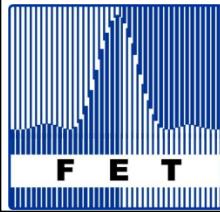
Phân loại các nút còn lại

- “Thưởng” đưa ra giá trị cân bằng giữa vị trí gần trung tâm và trọng số của nó
- Trong số những nút chưa được phân loại, chọn nút có thưởng cao nhất và chuyển nút này thành nút xương sống
- Phân loại các nút gần nút xương sống mới thành nút đầu cuối
- Cứ tiếp tục cho đến khi tất cả các nút đã được phân loại



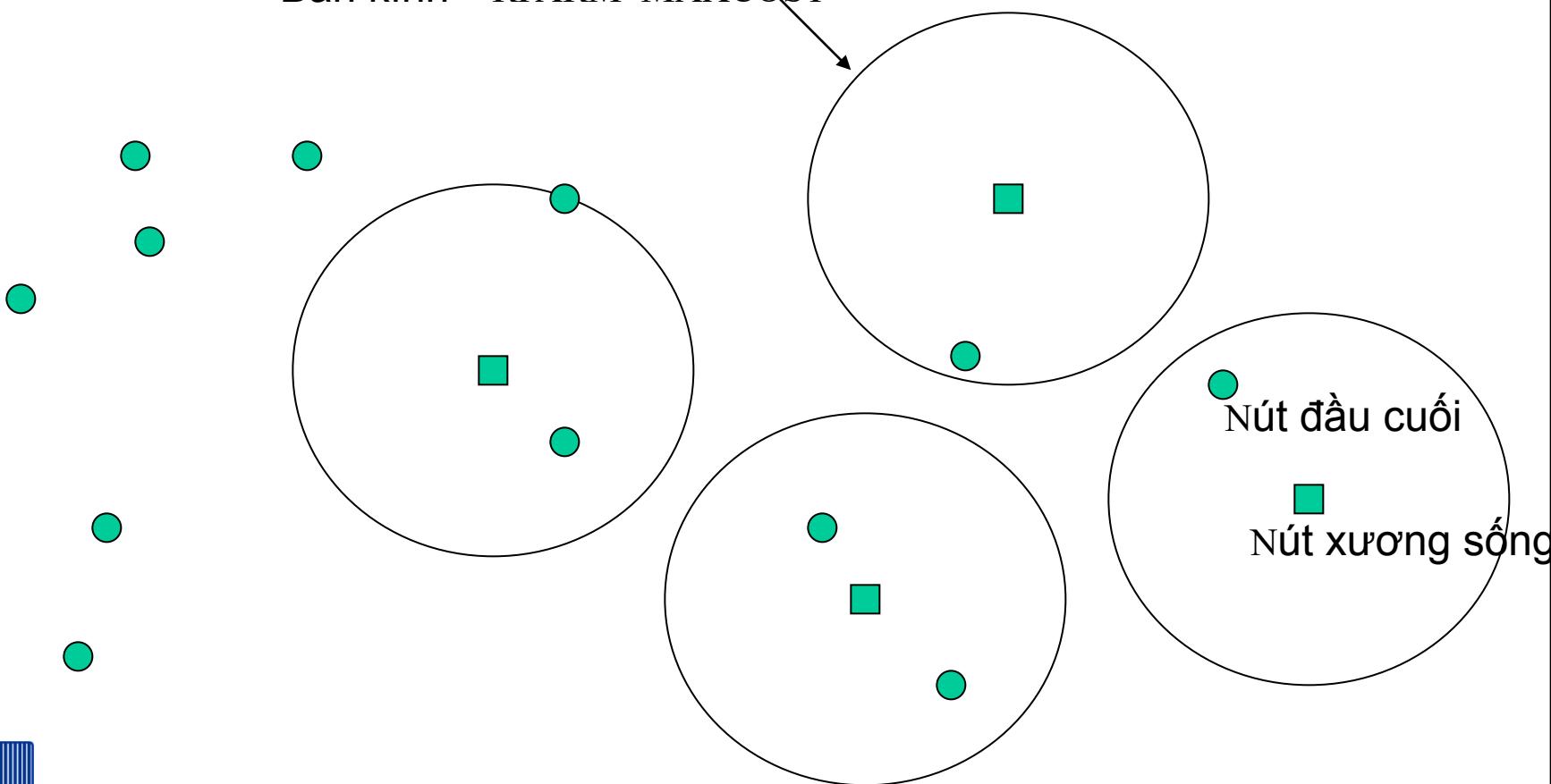
Thuật toán MENTOR (3)

- Lựa chọn nút xương sống trung tâm với giá trị moment nhỏ nhất đến trung tâm
$$\text{Moment}(n) = \sum \text{dist}(n, n^*) W_{n^*}$$
- Xây dựng cây Prim-Dijkstra, tham số α

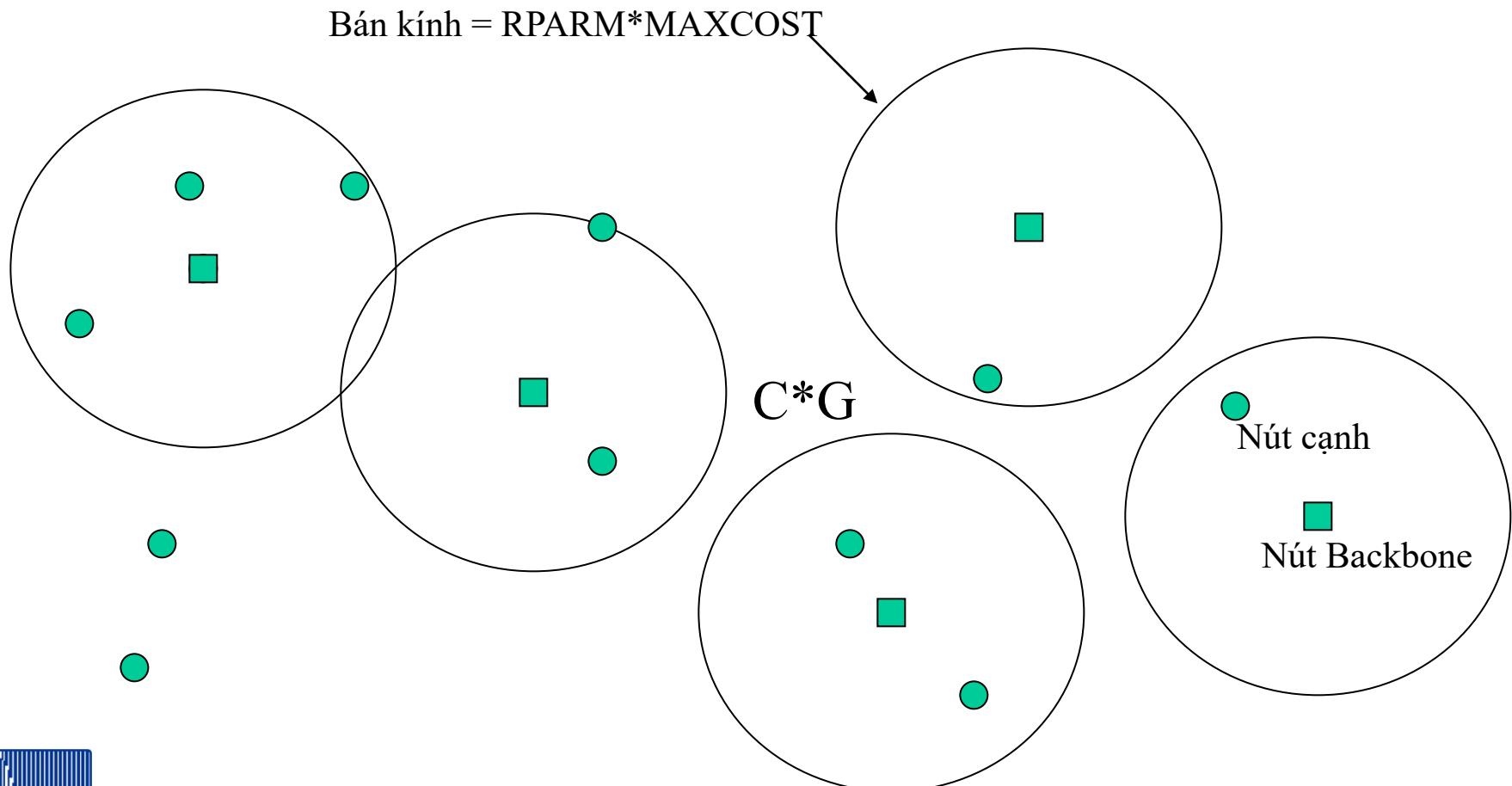


Ví dụ MENTOR

Bán kính = RPARM*MAXCOST

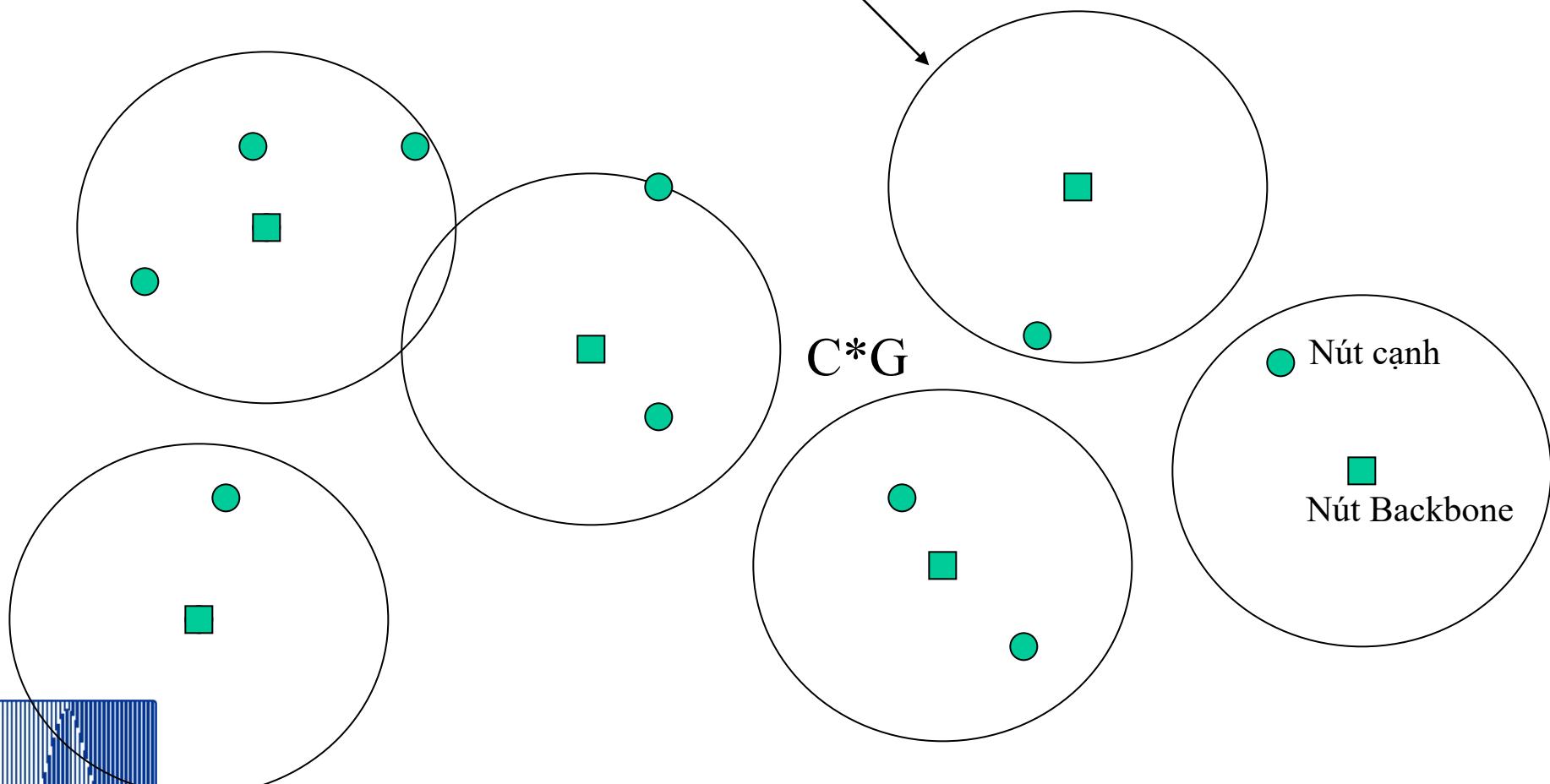


Ví dụ MENTOR (2)



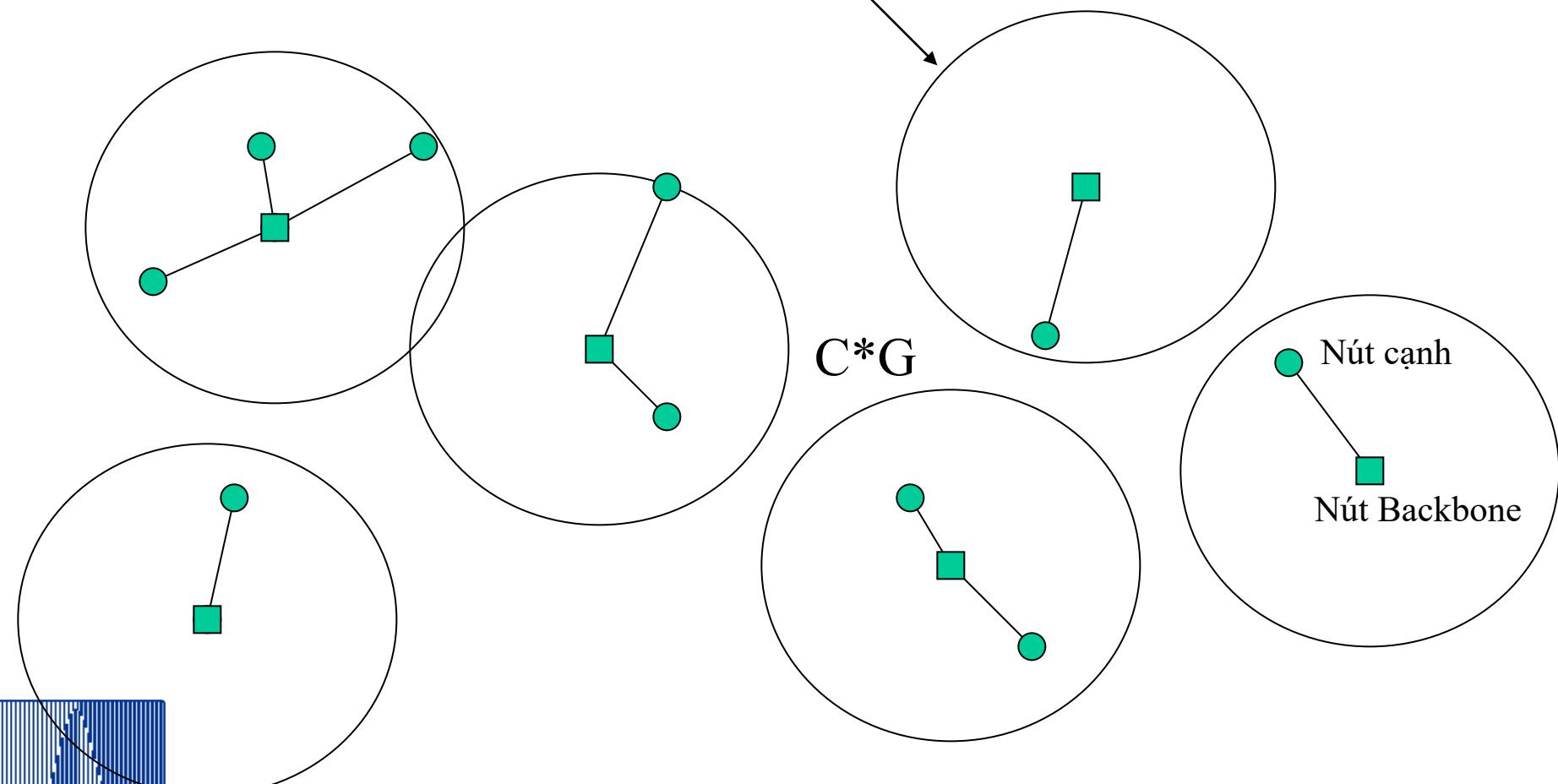
Ví dụ MENTOR (3)

Bán kính = RPARM*MAXCOST



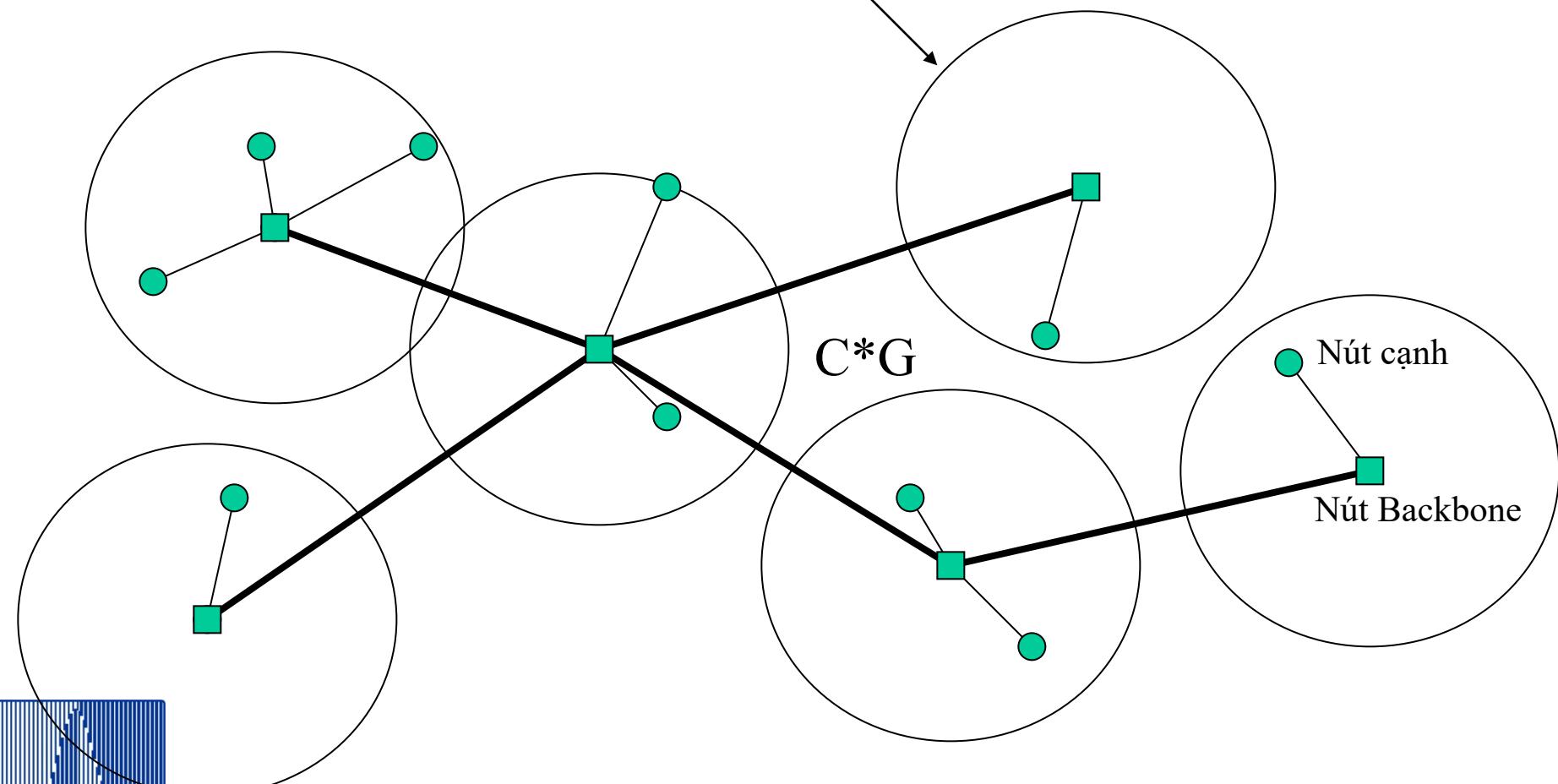
Ví dụ MENTOR (4)

Bán kính = RPARM*MAXCOST



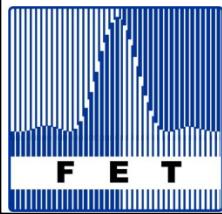
Ví dụ MENTOR (5)

Bán kính = RPARM*MAXCOST



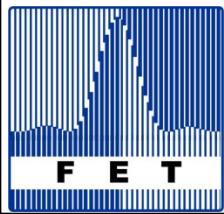
Cần phải cải tiến

- Như chúng ta đã biết, thiết kế cây có rất nhiều **nhược điểm** đặc biệt là đối với mạng lớn
- Thiếu độ dữ trữ tăng xác suất tắc nghẽn
 - **Mạng gần xích (α nhỏ)**
 - Kết hợp lưu lượng tại các liên kết “trung tâm” tăng tiền
 - Bước nhảy trung bình lớn trong mạng lớn
 - **Mạng gần sao (α lớn)**
 - Độ sử dụng liên kết thấp

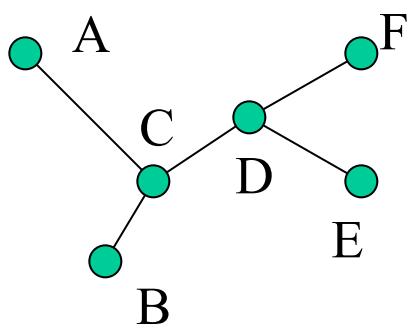


Tinh chỉnh thiết kế trong MENTOR

- Khái niệm dây và homing để thêm liên kết trực tiếp
- Dùng cây Prim-Dijkstra để xác định trình tự các nút
 - Dây được sắp theo từ bên ngoài vào trong
 - Cặp (N_1, N_2) sẽ có thứ tự sau tất cả các cặp (N_1^*, N_2^*) trong đó N_1 và N_2 nằm trên đường giữa N_1^* và N_2^*
 - Đường dài nhất sẽ được xếp trước



Ví dụ dây



Trình tự

AE

AF

BE

BF

3 bước nhảy

CE

CF

DA

DB

2 bước nhảy

AC

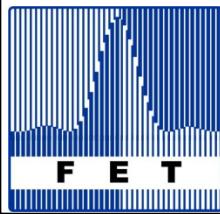
BC

CD

DE

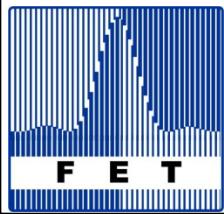
1 bước nhảy

DF



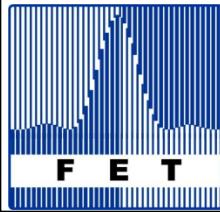
Nhận xét về dây

- Dây không duy nhất
- Dây khác nhau không ảnh hưởng nhiều việc thiết kế

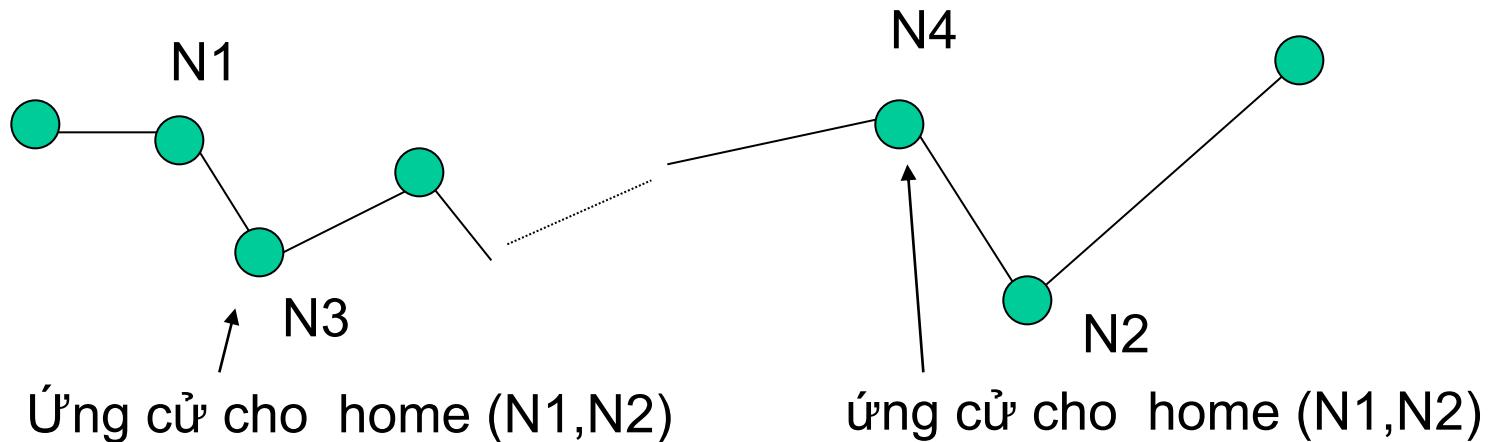


Homing

- Với mỗi cặp nút (N1, N2) mà không liền kề nhau, chúng ta lựa chọn home
 - Nếu 2 bước nhảy tách N1 và N2, home là nút ở giữa chúng
 - Nếu chúng có nhiều hơn 03 bước nhảy có rất nhiều ứng cử cho home của nó



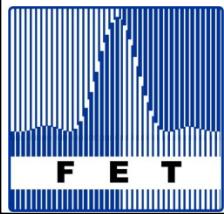
Homing (2)



Chọn nút N3 là home(N1,N2) nếu:

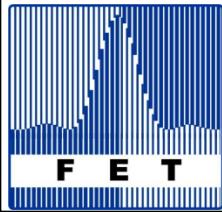
$$\text{Cost}(N1, N3) + \text{Cost}(N3, N2) < \text{Cost}(N1, N4) + \text{Cost}(N4, N2)$$

Nếu không chọn N4



Bước cuối cùng

- Xem xét mỗi cặp nút chỉ một lần, thêm liên kết nếu như nó sẽ mang đủ lưu lượng để bảo vệ bản thân
- Xem ma trận giá thành $T(Ni,Nj)$
 - Giả sử là đối xứng



Ví dụ 1 Phân loại nút

$W_5=8$, $W_6=9$, $W_{12}=7$, $W_{17}=5$, còn các nút khác có trọng số =1

Nút Backbones: 5, 6, 12, 17, 9, 10

- Lựa chọn nút xương sống trung tâm với giá trị moment nhỏ nhất đến trung tâm

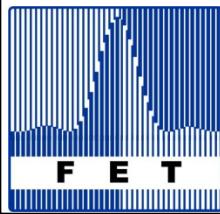
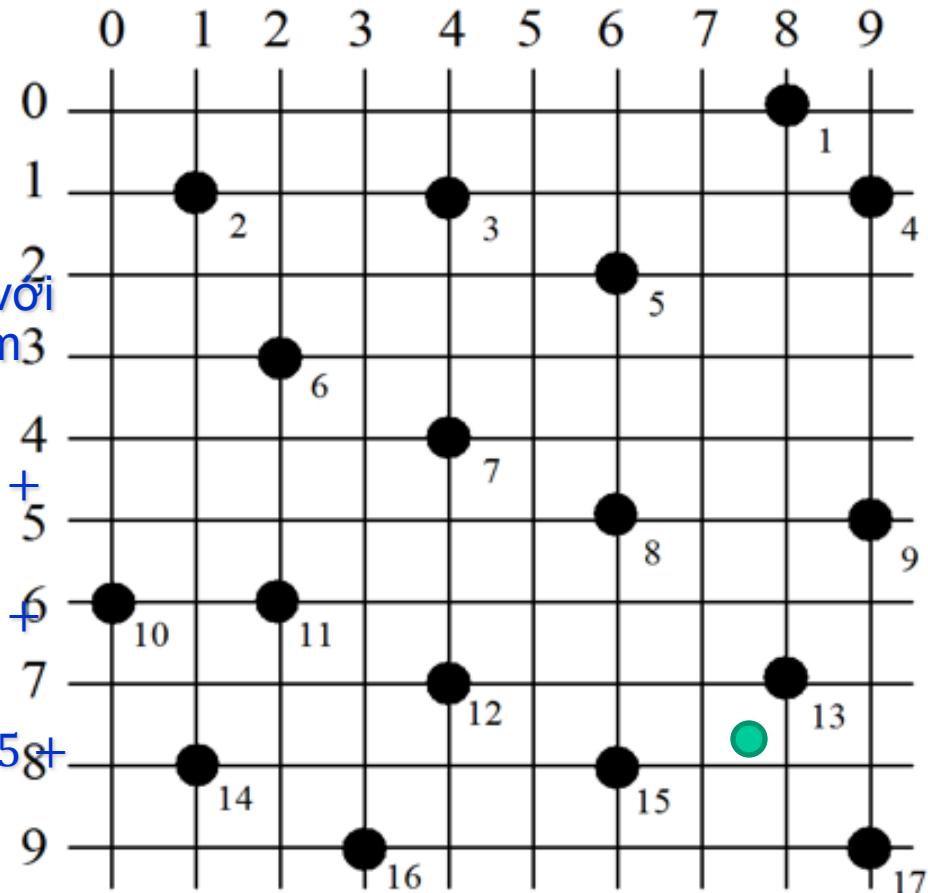
$$\text{Moment}(n) = \sum \text{dist}(n, n^*) W_{n^*}$$

$$\text{Moment}(5) = \sqrt{17} * 9 + \sqrt{29} * 7 + \sqrt{58} * 5 + \\ 3\sqrt{2} * 1 + 2\sqrt{13} * 1 = 124,3$$

$$\text{Moment}(6) = \sqrt{17} * 8 + 2\sqrt{5} * 7 + \sqrt{85} * 5 + \\ \sqrt{53} * 1 + \sqrt{13} * 1 = 121,9$$

$$\text{Moment}(12) = \sqrt{29} * 8 + 2\sqrt{5} * 9 + \sqrt{29} * 5 + \\ \sqrt{17} * 1 + \sqrt{29} * 1 = 119,7$$

Nút 12 là nút backbone trung tâm.



Ví dụ 1 Phân loại nút

$W_5=8$, $W_6=9$, $W_{12}=7$, $W_{17}=5$, còn các nút khác có trọng số = 1

Nút Backbones: 5, 6, 12, 17, 9, 10

$\text{Cost}(N1, N3) + \text{Cost}(N3, N2) < \text{Cost}(N1, N4) + \text{Cost}(N4, N2)$

Xét (9,10)

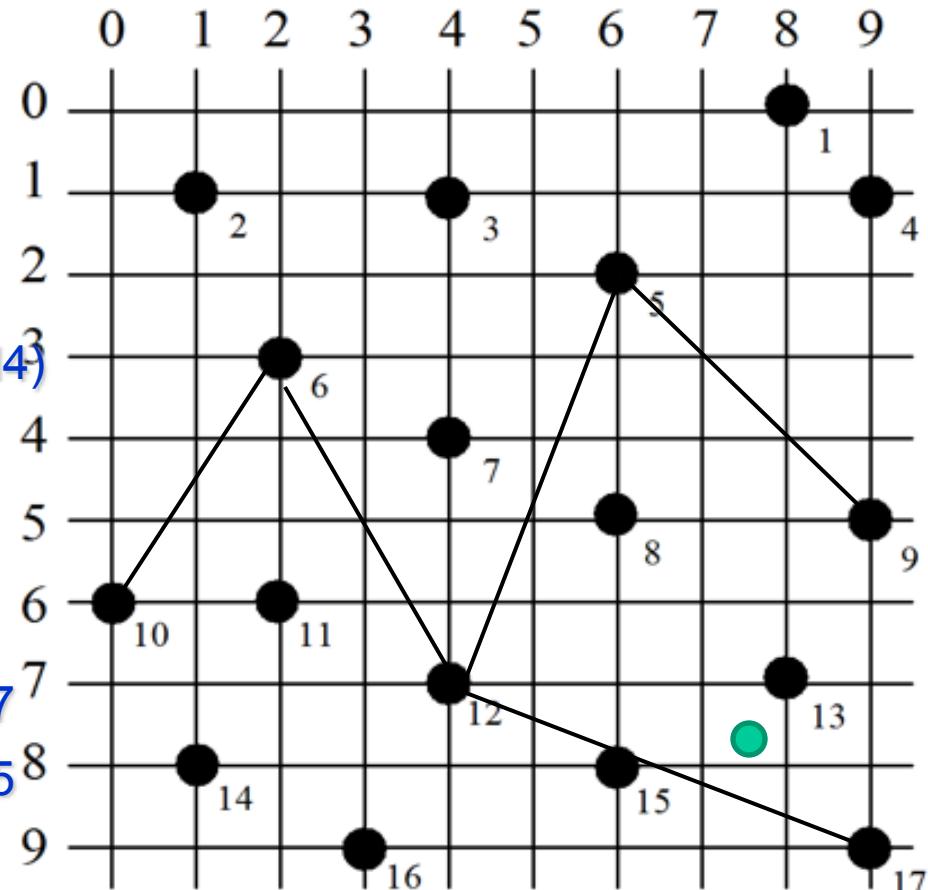
Ứng cử viên của nút Home là 5, 6, 12.

$\text{Cost}(10, 6) + \text{cost}(6, 9) = \sqrt{13} + \sqrt{53}$

$\text{Cost}(10, 5) + \text{cost}(5, 9) = 2\sqrt{13} + 3\sqrt{2} = 10,7$

$\text{Cost}(10, 12) + \text{cost}(12, 9) = \sqrt{17} + \sqrt{29} = 9,5$

Chọn nút 12 là nút home.



Bước cuối cùng (2)

Với mỗi cặp (N1,N2), thực hiện thuật toán sau:

1. Nếu dung lượng của liên kết là C, tính

$n = \text{ceil}[\text{lưu lượng } (N1, N2)/C]$ số lượng lưu lượng cần thiết

2. Tính độ sử dụng

$u = \text{lưu lượng } (N1, N2)/(n*C)$

3. Thêm liên kết giữa N1 và N2 nếu $u > u_{\min}$

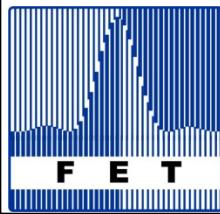
4. Nếu $u < u_{\min}$ chuyển lưu lượng 1 hop qua mạng

có nghĩa là, thêm lưu lượng (N1,N2) vào cả lưu lượng (N1,H) và
lưu lượng (H,N2)

Và làm tương tự với lưu lượng (N2,N1)

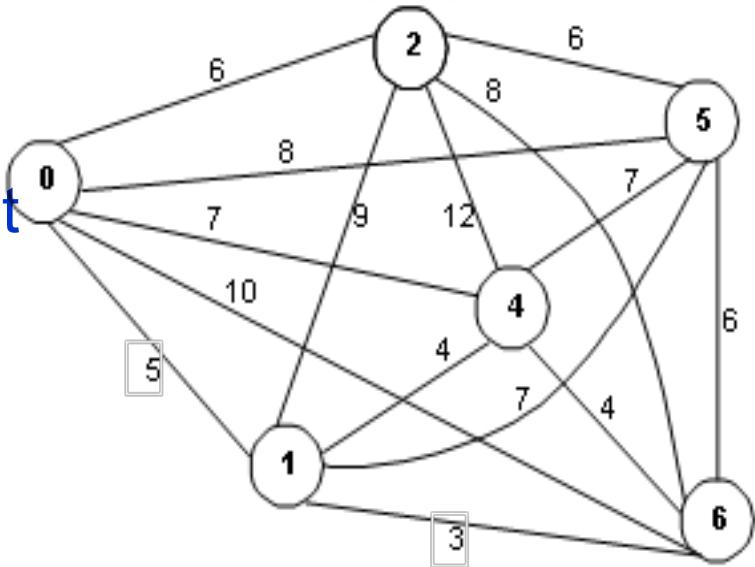
Chú ý - trường hợp đặc biệt nếu (N1,N2) thuộc cây gốc. Trong

trường hợp này chỉ thêm liên kết (N1,N2) vào thiết kế



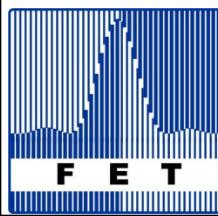
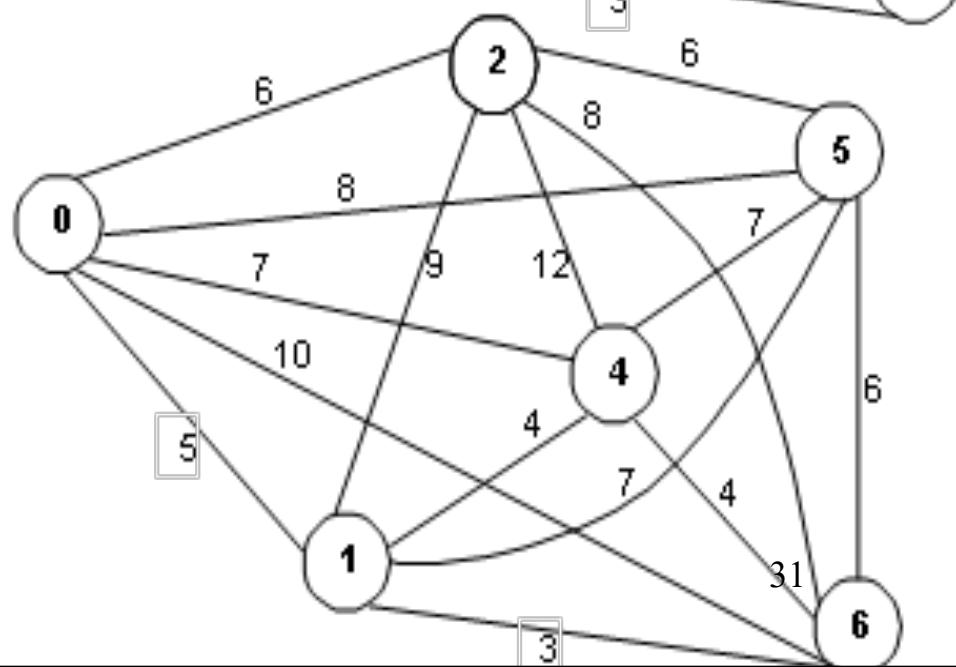
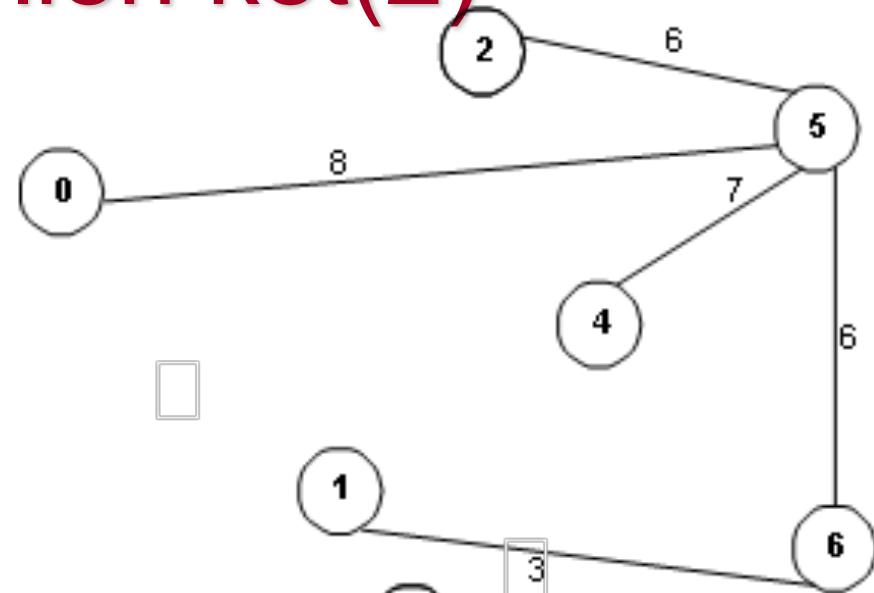
Ví dụ 1. thêm liên kết (1)

- tìm cây Prim-Dijkstra biết $\alpha=0.6$ biết nút trung tâm là nút 5
- Giả sử $u_{min} = 80\%$.
- Dung lượng các liên kết là 8.
- Lưu lượng giữa các nút như sau: $T(0,1)=11$, $T(4,1)=19$ và $T(2,1)=7$. Dùng giải thuật Mentor1 để thêm liên kết phù hợp với lưu lượng như trên.



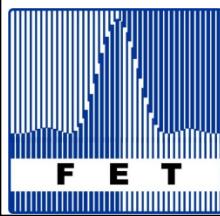
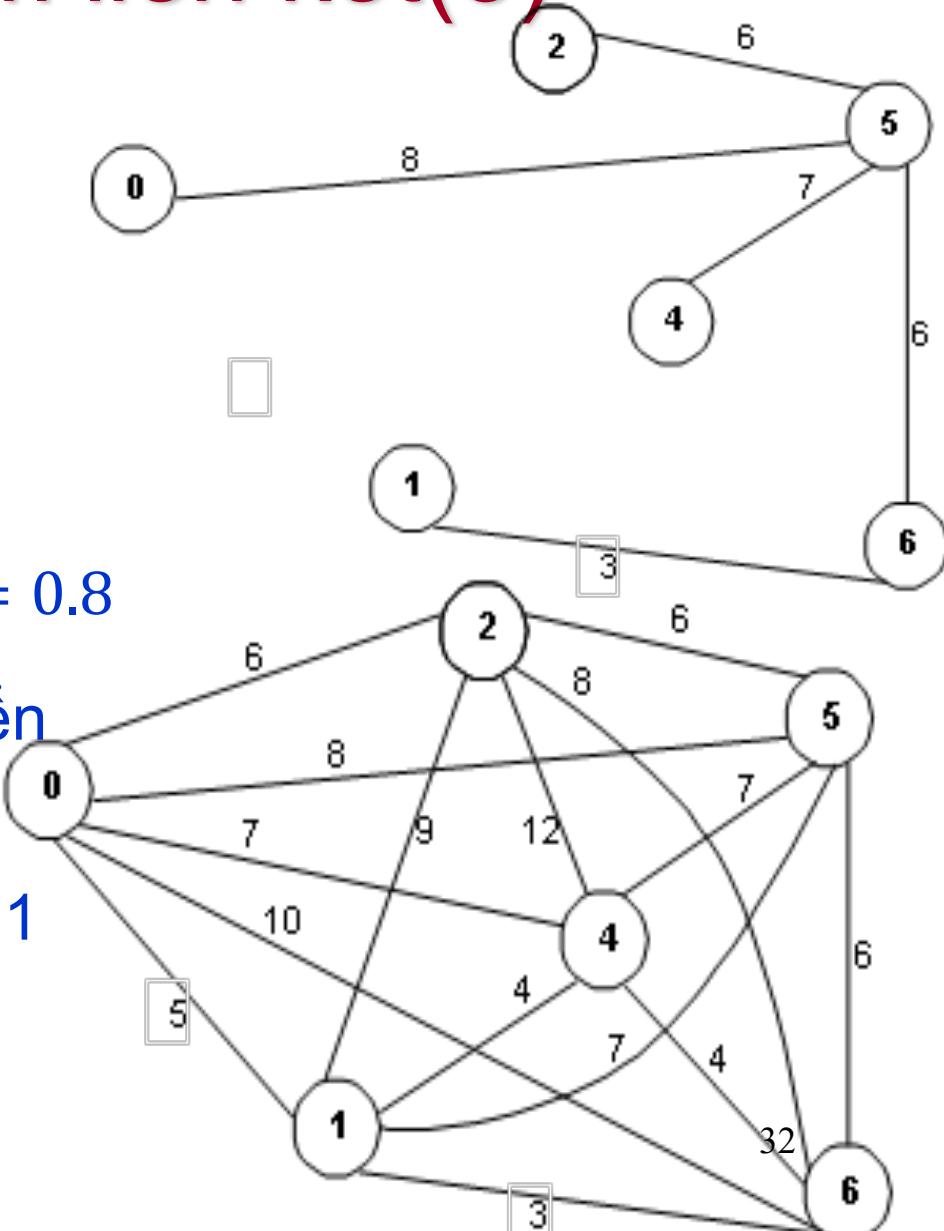
Ví dụ 1 thêm liên kết(2)

- $T(0,1)=11$, $T(4,1)=19$ và
 $T(2,1)=7$
 - $(0,1)$ 3 hops
 - $(4,1)$ 3 hops
 - $(2,1)$ 3 hops



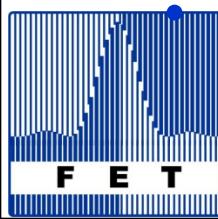
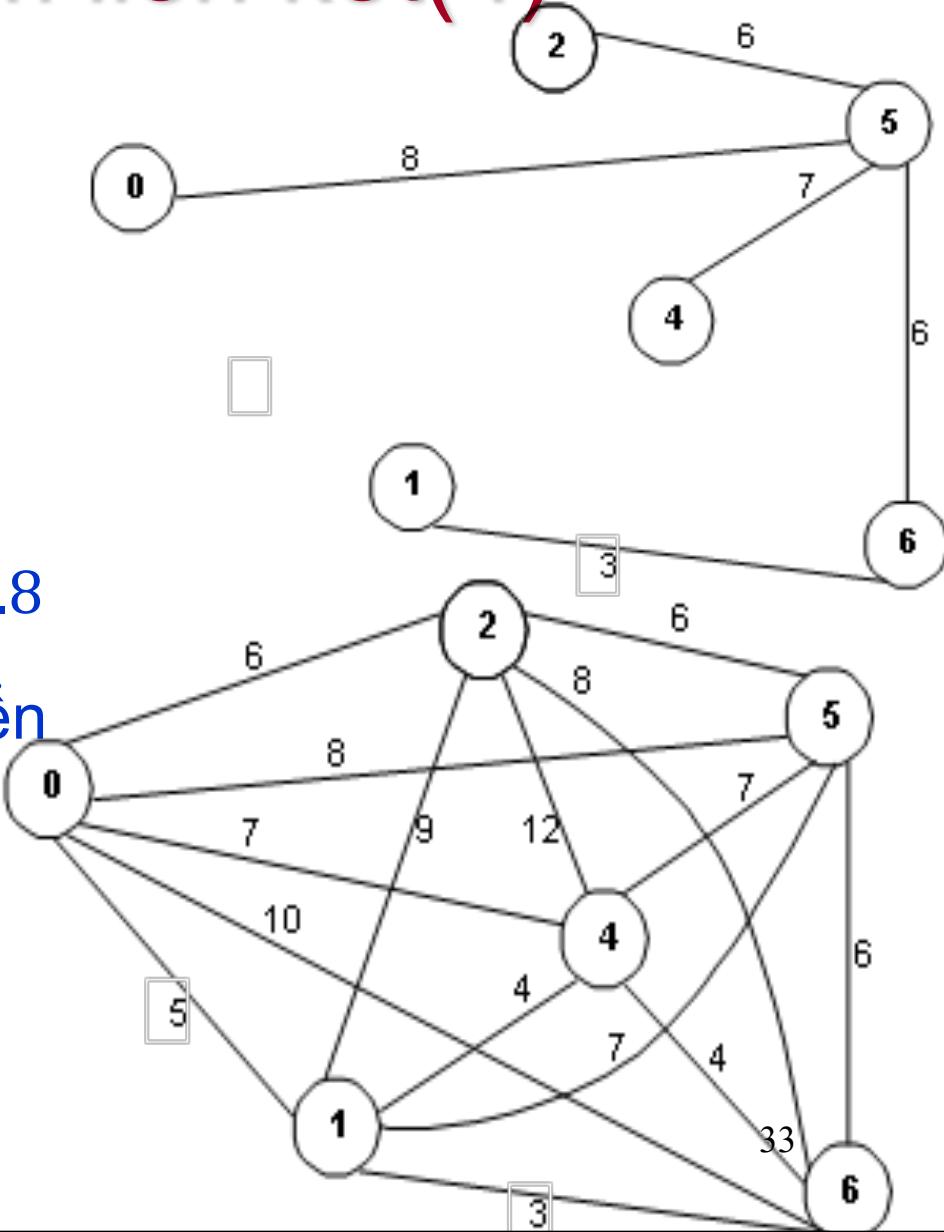
Ví dụ 1 thêm liên kết(3)

- Xét (0,1)
- $n = roundup \frac{T(0,1)}{C} = roundup \left(\frac{11}{8} \right) = 2$
- $u = \frac{11}{2.8} = 0.6875 < u_{min} = 0.8$
- Nút 6 là nút Home. Chuyển lưu lượng xuống
- $T(0,6)=T(6,1)=0+T(0,1)=11$



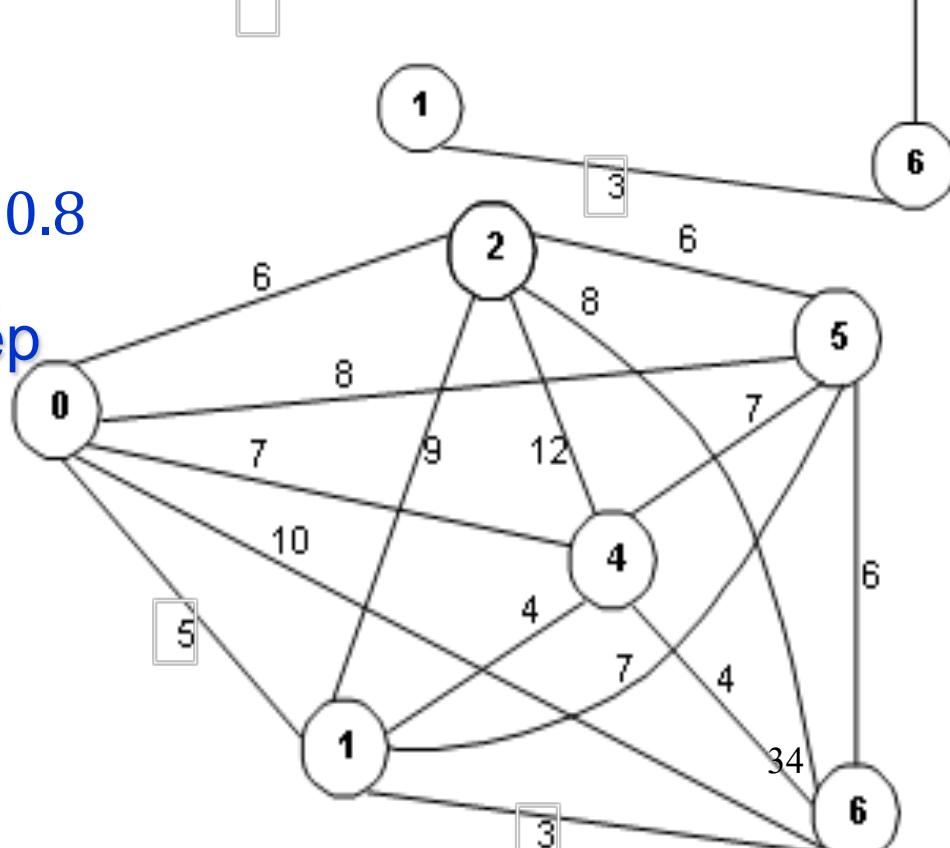
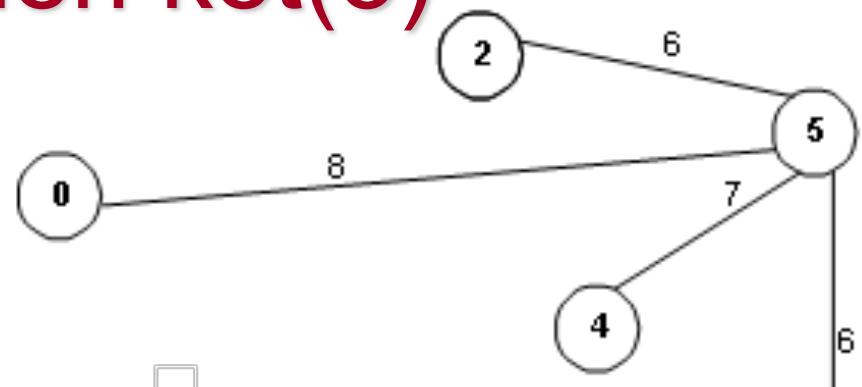
Ví dụ 1 thêm liên kết(4)

- Xét (4,1)
- $n = roundup \frac{T(4,1)}{C} = roundup \left(\frac{19}{8}\right) = 3$
- $u = \frac{19}{3.8} = 0.79 < u_{min} = 0.8$
- Nút 6 là nút Home. Chuyển lưu lượng xuống
- $T(4,6)=0+T(4,1)=19$
- $T(6,1)=T(6,1)_{cũ}+T(4,1)=11+T(4,1)=11+19=30$



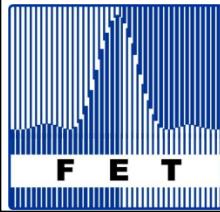
Ví dụ thêm liên kết(5)

- Xét (2,1)
- $n = roundup \frac{T(2,1)}{C} = roundup \left(\frac{7}{8}\right) = 1$
- $u = \frac{7}{1.8} = 0.875 > u_{min} = 0.8$
- → Thêm 1 liên kết trực tiếp giữa 2 và 1



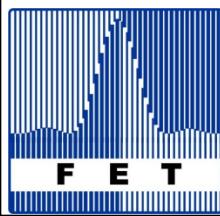
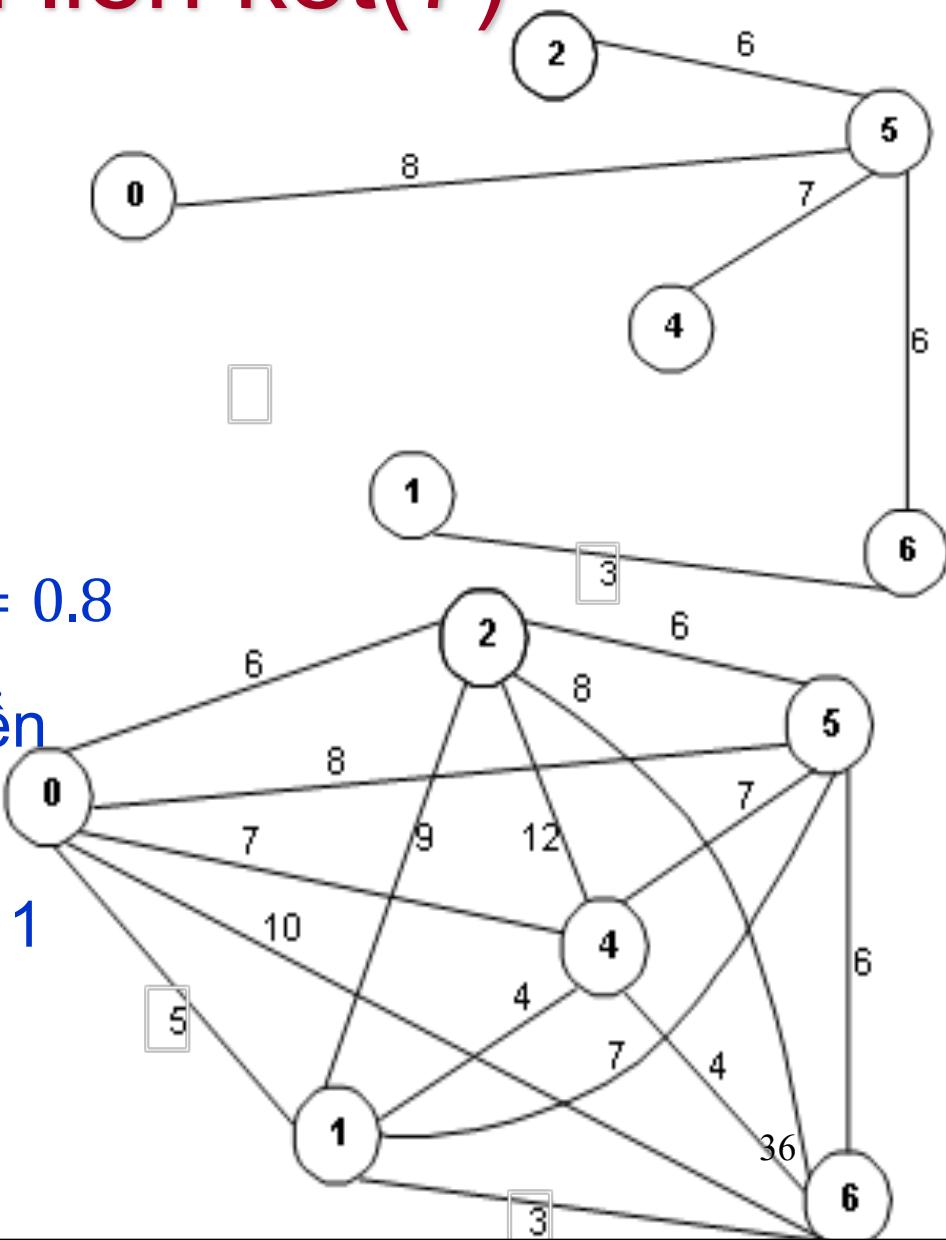
Ví dụ 1 thêm liên kết(6)

- (0,6) 2 hops dung lượng 11
- (4,6) 2 hops dung lượng 19
- (6,1) 1 hop dung lượng 30



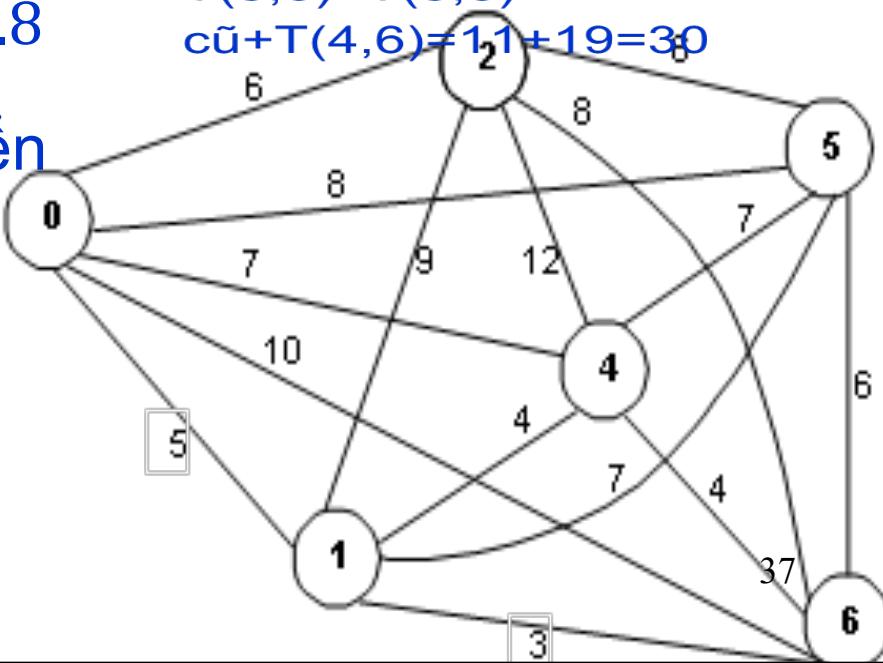
Ví dụ thêm liên kết(7)

- Xét (0,6) 2 hops
- $n = roundup \frac{T(0,6)}{C} = roundup \left(\frac{11}{8}\right) = 2$
- $u = \frac{11}{2.8} = 0.6875 < u_{min} = 0.8$
- Nút 5 là nút Home. Chuyển lưu lượng xuống
- $T(0,5)=T(5,6)=0+T(0,6)=11$



Ví dụ 1 thêm liên kết(8)

- Xét (4,6)
- $n = roundup \frac{T(4,6)}{c} = roundup \left(\frac{19}{8}\right) = 3$
- $u = \frac{19}{3.8} = 0.79 < u_{min} = 0.8$
- Nút 5 là nút Home. Chuyển lưu lượng xuống
- $T(4,5)=0+T(4,5)=19$
- $T(5,6)=T(5,6)$
 $c_{\text{c}}+T(4,6)=11+19=30$



Ví dụ 1 thêm liên kết(9)

- Bây giờ chỉ còn liên kết 1 hop:
 - (0,5) dung lượng 11
 - (4,5) dung lượng 19
 - (5,6) dung lượng 30
 - (6,1) dung lượng 30

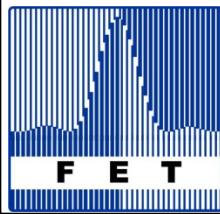
Ví dụ 1 thêm liên kết(9)

Xét liên kết 1 hop (0,5) dung lượng 11

$$- n = roundup \frac{T(0,5)}{c}$$

$$- = roundup \left(\frac{11}{8} \right) = 2$$

→Thêm 02 liên kết giữa (0,5)



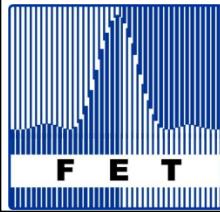
Ví dụ 1 thêm liên kết(9)

Xét liên kết 1 hop (4,5) dung lượng 19

$$- n = roundup \frac{T(4,5)}{c}$$

$$- = roundup \left(\frac{19}{8} \right) = 3$$

→Thêm 03 liên kết giữa (4,5)



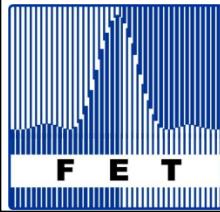
Ví dụ 1 thêm liên kết(9)

Xét liên kết 1 hop (5,6) dung lượng 30

$$- n = roundup \frac{T(5,6)}{c}$$

$$- = roundup \left(\frac{30}{8} \right) = 4$$

→Thêm 04 liên kết giữa (5,6)



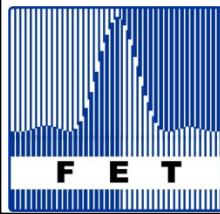
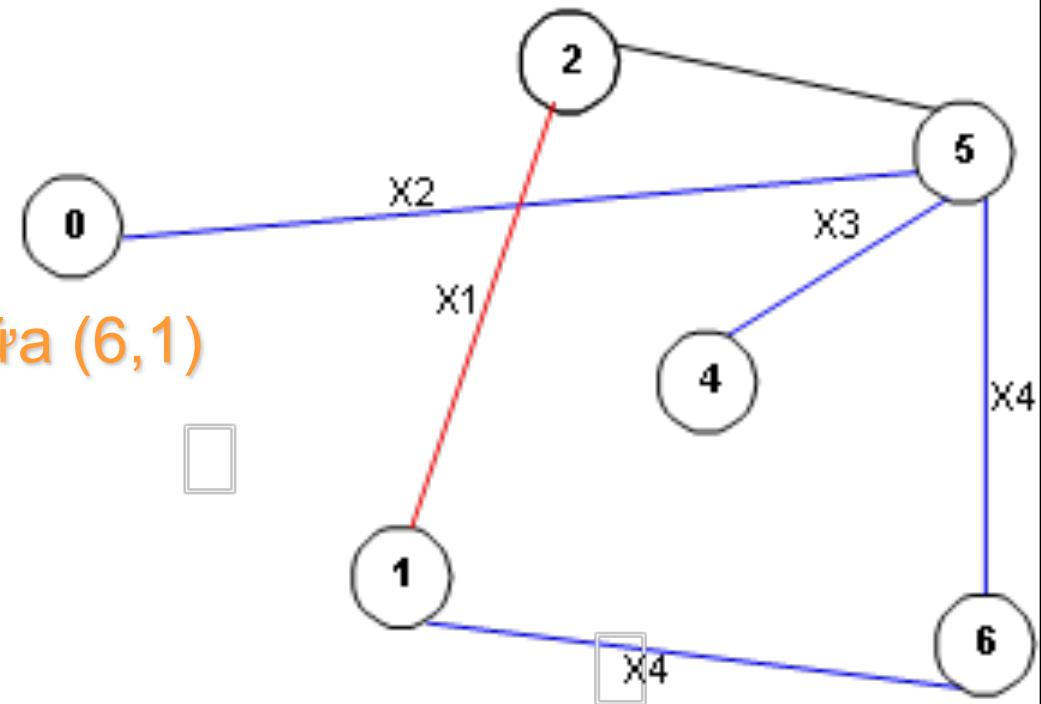
Ví dụ 1 thêm liên kết(9)

Xét liên kết 1 hop (6,1) dung lượng 30

$$- n = \text{roundup} \frac{T(6,1)}{C}$$

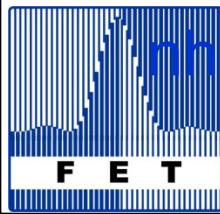
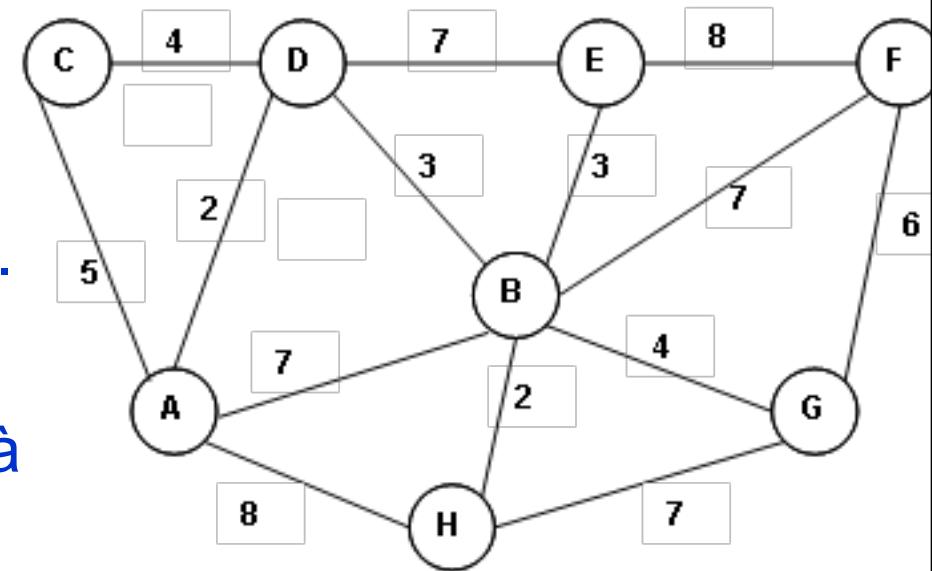
$$- = \text{roundup} \left(\frac{30}{8} \right) = 4$$

→Thêm 04 liên kết giữa (6,1)



Ví dụ 2 Thêm lưu lượng (1)

- tìm cây Prim-Dijkstra biết $\alpha=0.6$ biết nút trung tâm là nút A
- $umin = 75\%$.
- Dung lượng các liên kết là 10.
- Lưu lượng giữa các nút như sau: $T(D,G)=12$, $T(C,H)=17$ và $T(E,G)=9$. $T(C,B)=21$, Dùng giải thuật Mentor1 để thêm liên kết phù hợp với lưu lượng



nhiều trên

Ví dụ 2 Thêm lưu lượng (2)

Tìm cây Prim-Dijkstra

$$U=\{A\}, V=\{B,C,D,H\}$$

$$L(B)=\alpha \cdot d(A) + d_{AB} = 0 + 7 = 7$$

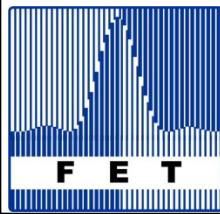
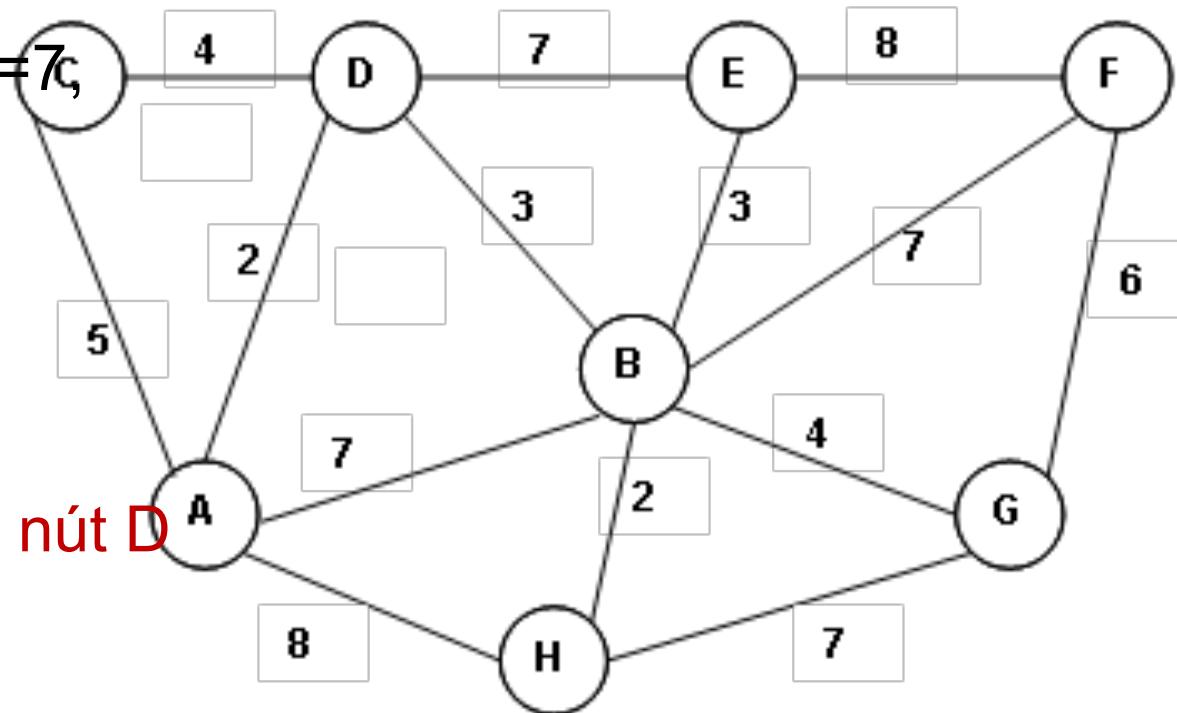
$$\pi(B)=A$$

$$L(D)=2, \pi(D)=A$$

$$L(C)=5, \pi(C)=A$$

$$L(H)=8, \pi(H)=A$$

→ $L(D)$ nhỏ nhất thêm nút D vào cây.



Ví dụ 2 Thêm lưu lượng (3)

$$U=\{A,D\}, V=\{B,C,E,H\}$$

$$L(E)=\alpha \cdot d(D) + d_{DE} = 0.6 \times 2 + 7 = 8.2,$$

$$\pi(E)=D$$

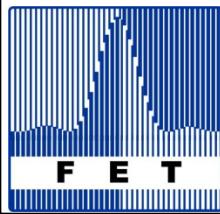
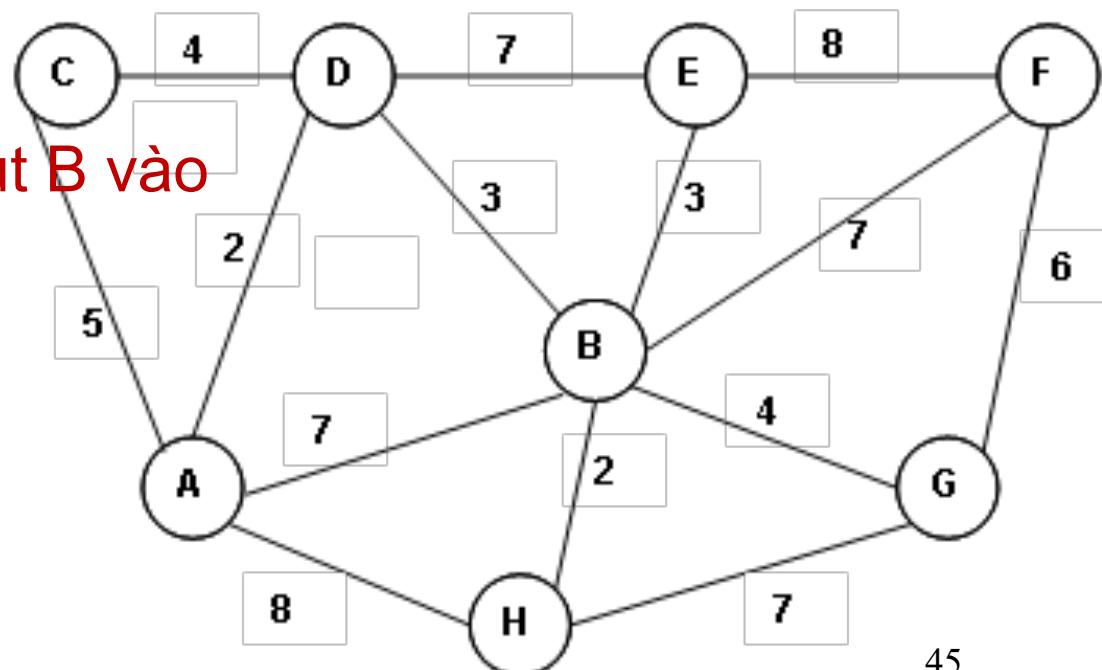
$$\text{Vì } \alpha \cdot d(D) + d_{DB} = 0.6 \times 2 + 3 = 4.2 < 7$$

$$\text{nên } L(B)=4.2, \pi(B)=D$$

$$L(C)=5, \pi(C)=A$$

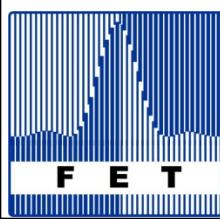
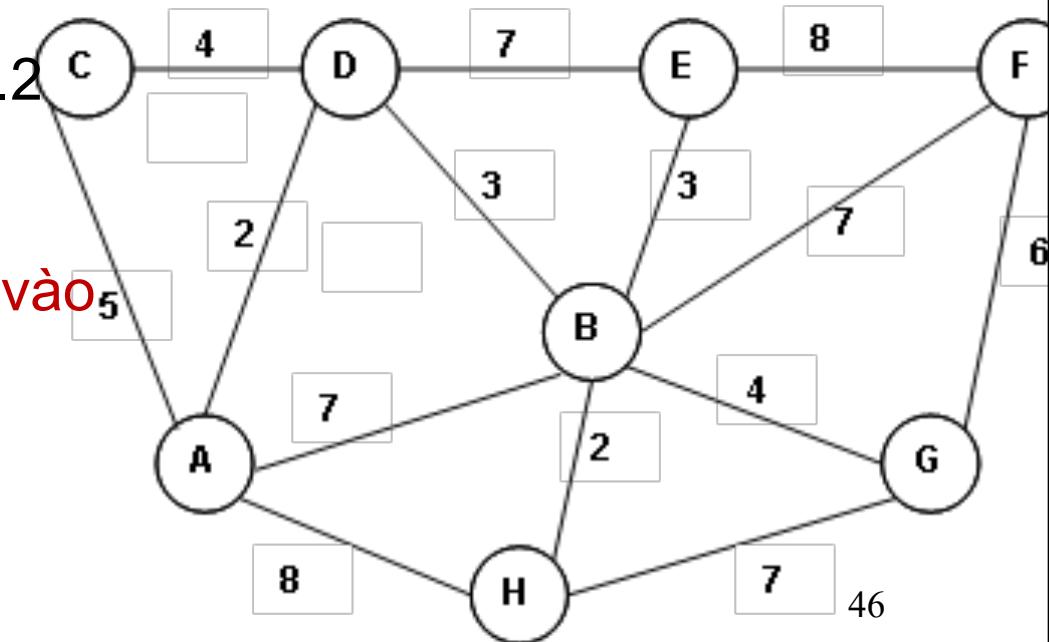
$$L(H)=8, \pi(H)=A$$

→ $L(B)$ nhỏ nhất thêm nút B vào cây.



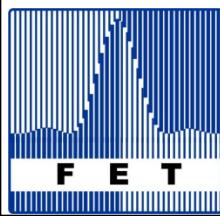
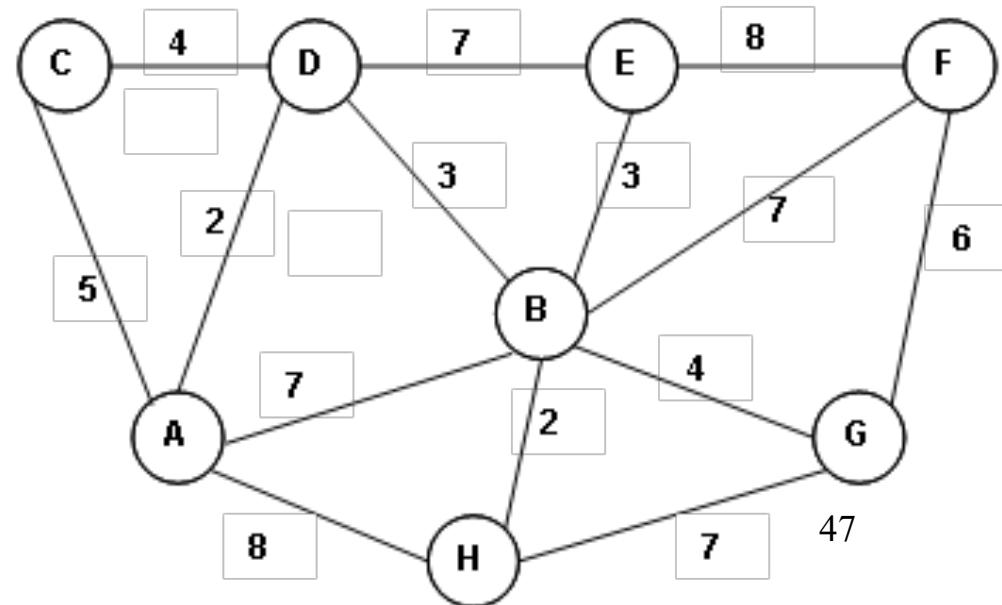
Ví dụ Thêm lưu lượng (4)

- $U=\{A, D, B\}$, $V=\{C, E, H, G, F\}$
- $L(G)=\alpha \cdot d(B) + d_{BG} = 0.6 \times 5 + 4 = 7$,
 $\pi(G)=B$
- $L(F)=\alpha \cdot d(B) + d_{BF} = 0.6 \times 5 + 7 = 10$,
 $\pi(F)=B$
- Vì $\alpha \cdot d(B) + d_{BH} = 0.6 \times 5 + 2 = 5 < 8$
nên $L(H)=5$, $\pi(H)=B$
- Vì $\alpha \cdot d(B) + d_{BE} = 0.6 \times 5 + 3 = 6 < 8.2$
nên $L(E)=6$, $\pi(E)=B$
- $L(C)=5$, $\pi(C)=A$
- → $L(H)$ nhỏ nhất thêm nút H vào cây.



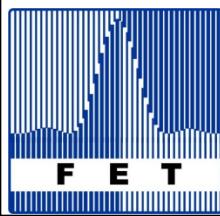
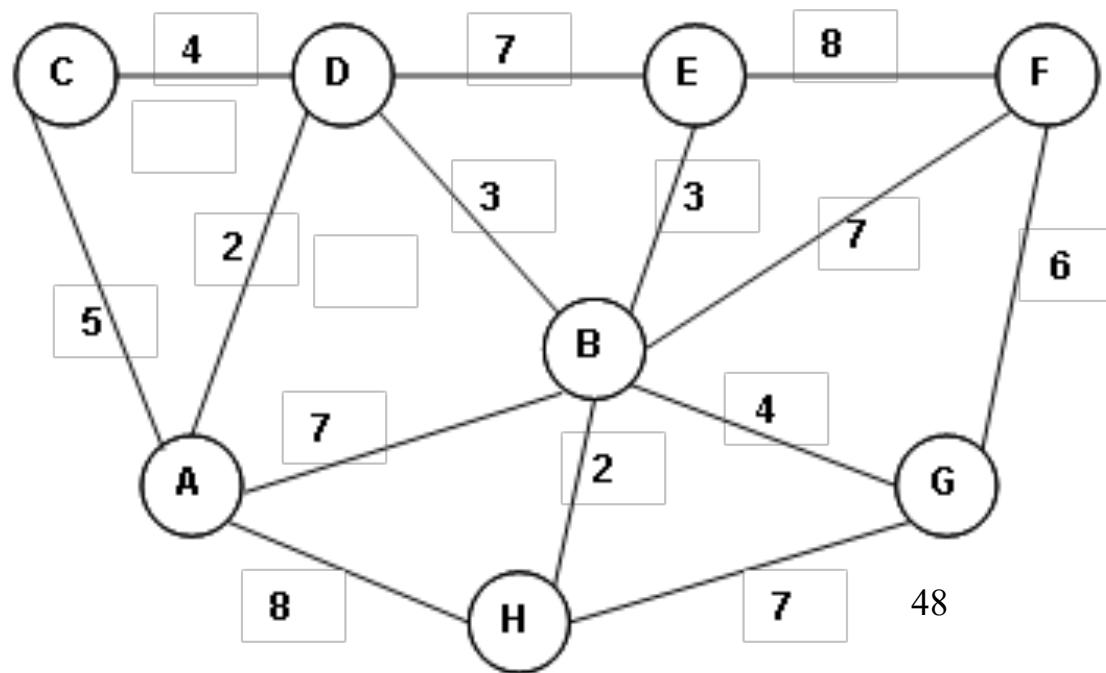
Ví dụ 2 Thêm lưu lượng (5)

- $U=\{A, D, B, H\}$, $V=\{C, E, G, F\}$
- $L(G)=7$, $\pi(G)=B$
- $L(F)=10$, $\pi(F)=B$
- $L(E)=6$, $\pi(E)=B$
- $L(C)=5$, $\pi(C)=A$
- → $L(C)$ nhỏ nhất thêm nút C vào cây.



Ví dụ 2 Thêm lưu lượng (6)

- $U=\{A,D,B,H,C\}$, $V=\{E,G,F\}$
- $L(G)=7$, $\pi(G)=B$
- $L(F)=10$, $\pi(F)=B$
- $L(E)=6$, $\pi(E)=B$
- → $L(E)$ nhỏ nhất thêm nút E vào cây.



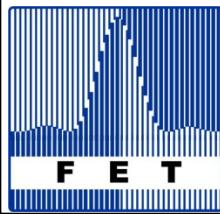
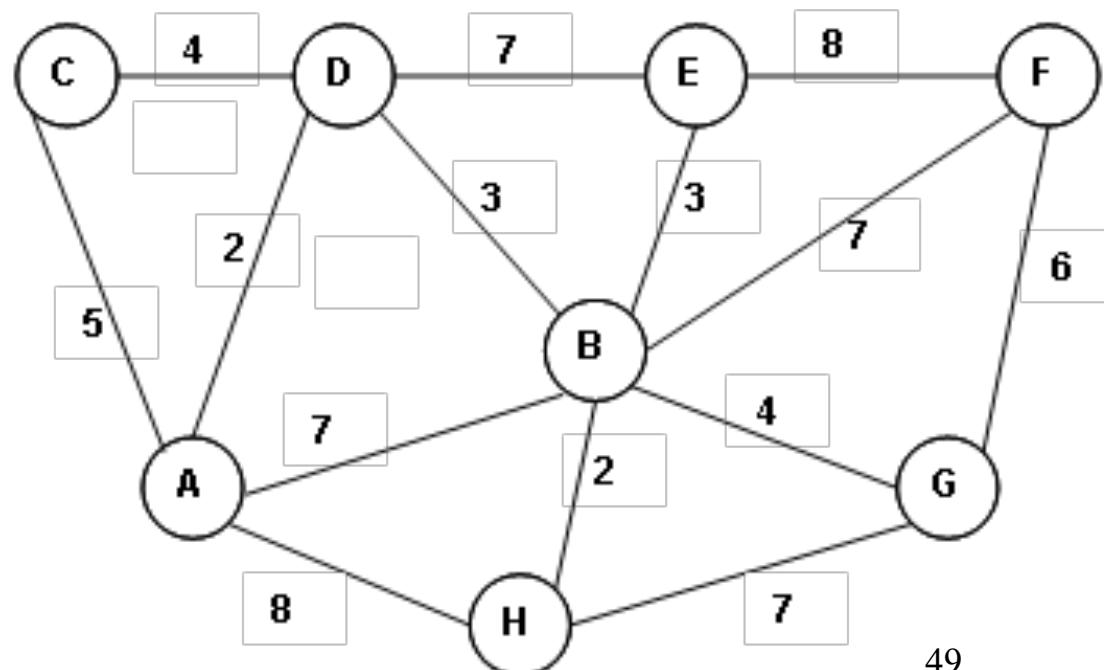
Ví dụ 2 Thêm lưu lượng (7)

$U=\{A,D,B,H,C,E\}$, $V=\{G,F\}$

$L(G)=7$, $\pi(G)=B$

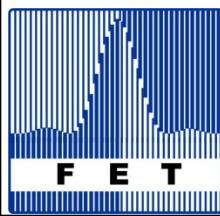
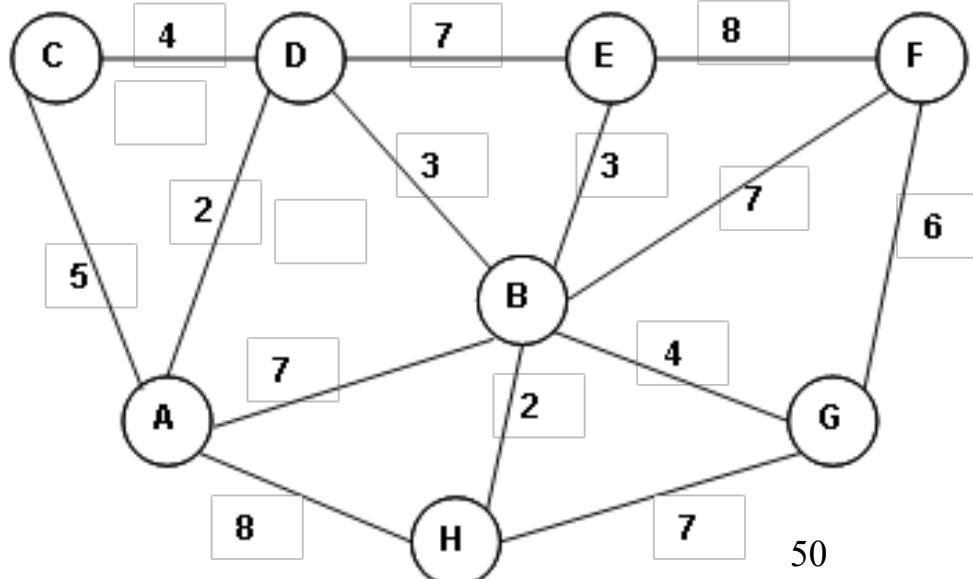
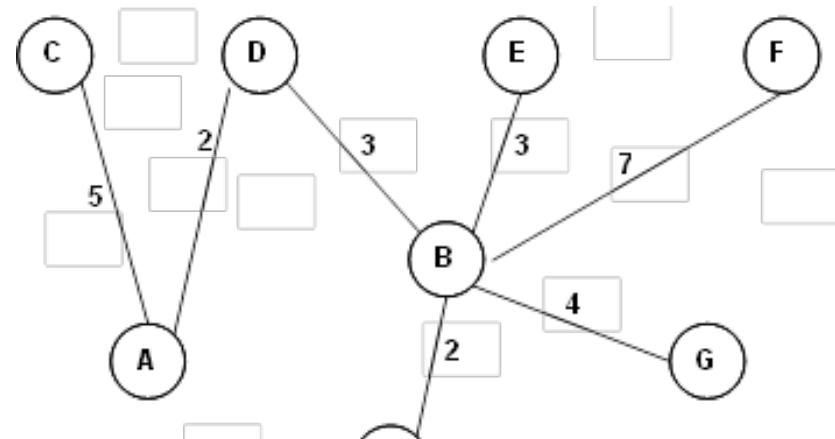
$L(F)=10$, $\pi(F)=B$

→ $L(G)$ nhỏ nhất thêm nút G vào cây.



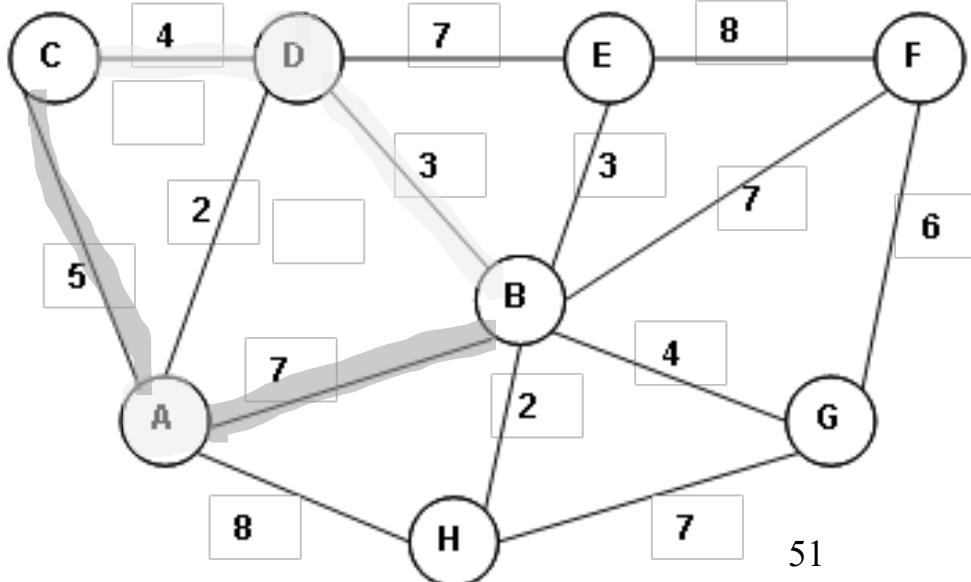
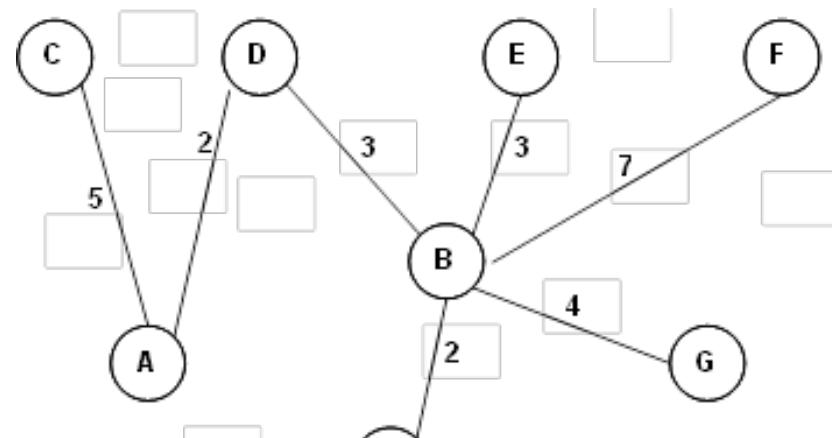
Ví dụ 2 Thêm lưu lượng (9)

- $u_{min} = 75\%$.
- $T(D,G)=12$, $T(C,H)=17$ và
 $T(E,G)=9$. $T(C,B)=21$
 - (D,G) 2 hops
 - (C,H) 4 hops
 - (E,G) 2 hops
 - (C,B) 3 hops



Ví dụ 2 Thêm lưu lượng (10)

- $T(D,G)=12$, $T(C,H)=17$ và
 $T(E,G)=9$. $T(C,B)=21$, $C=10$
 - (C,H) 4 hops
 - (C,B) 3 hops
 - (D,G) 2 hops
 - (E,G) 2 hops

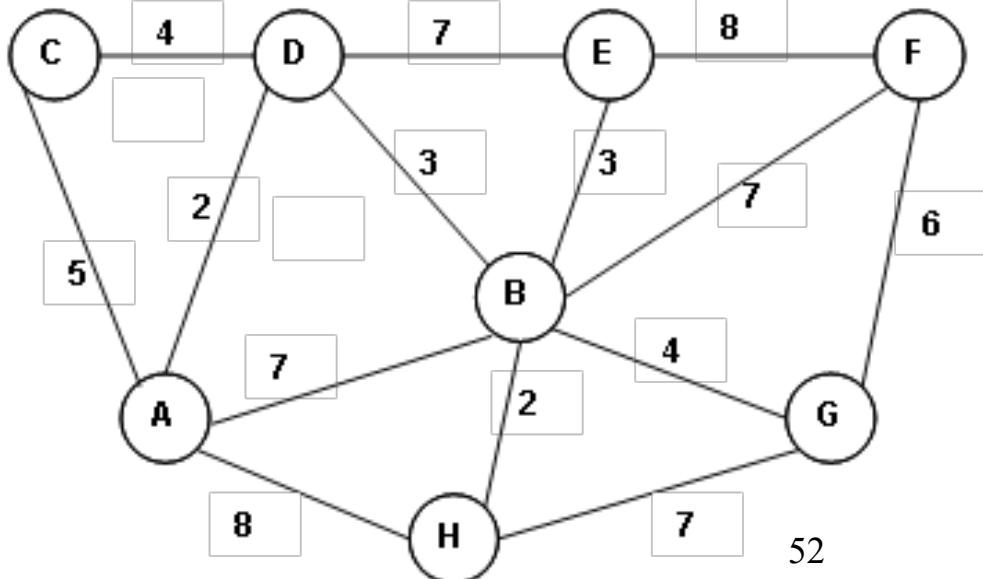
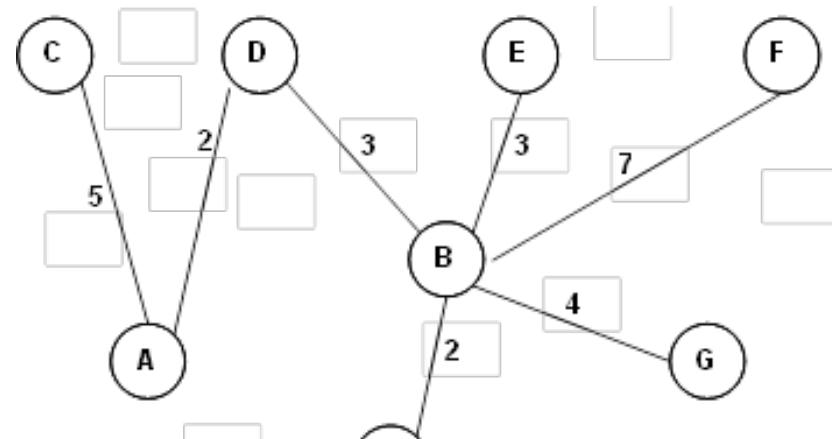


Ví dụ 2 Thêm lưu lượng (11)

- Xét (C,H) 4hops
- $n = roundup \frac{T(C,H)}{C} = roundup \left(\frac{17}{10} \right) = 2$

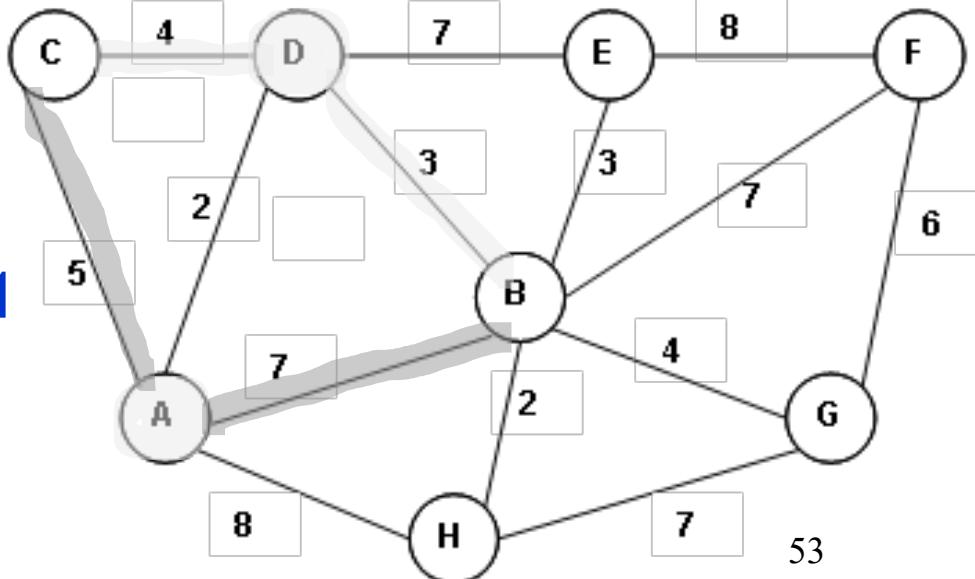
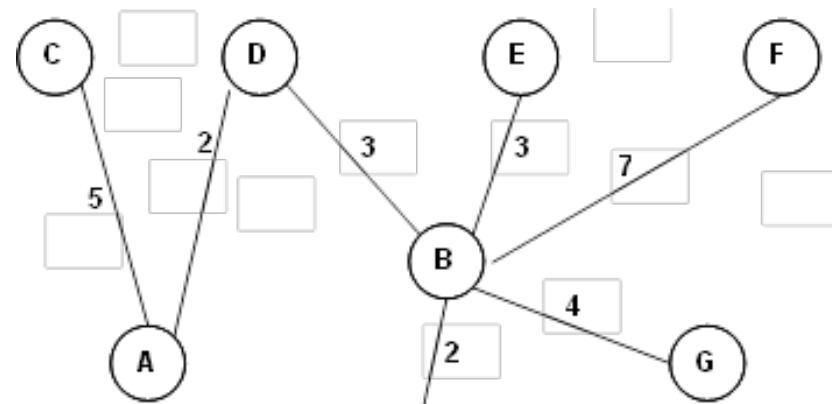
$$u = \frac{17}{2 \times 10} = 0.85 > u_{min} = 0.75$$

- Thêm 02 liên kết trực tiếp giữa C và H



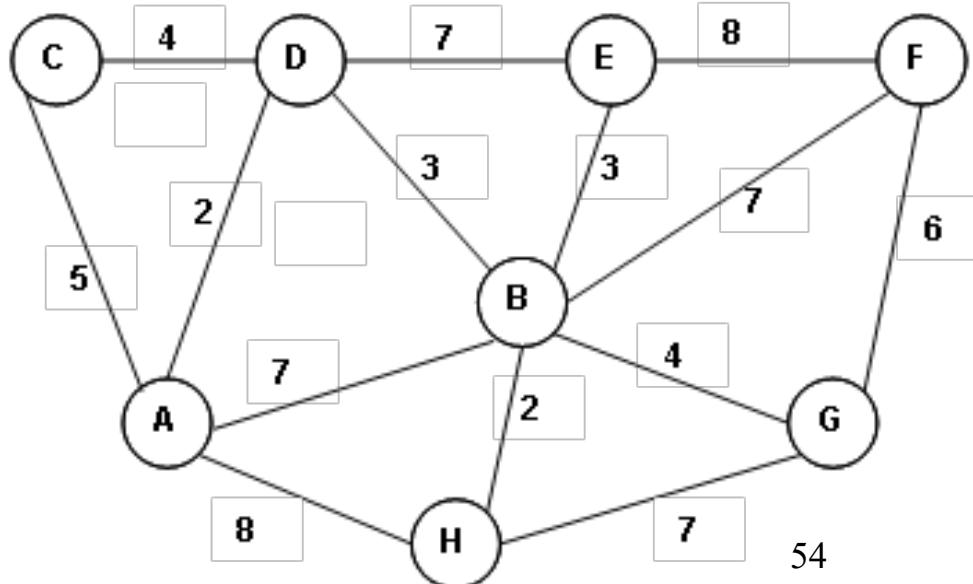
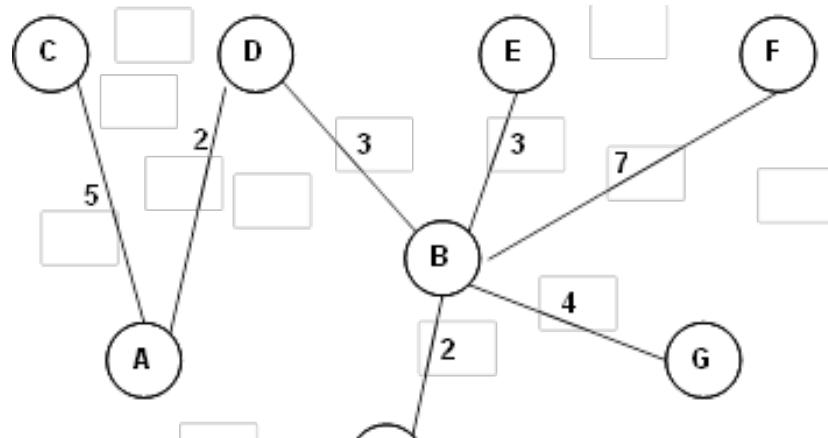
Ví dụ 2 Thêm lưu lượng (12)

- Xét (C,B) 3hops
- $n = \text{roundup} \frac{T(C,B)}{C} = \text{roundup} \left(\frac{21}{10} \right) = 3$
- $u = \frac{21}{3 \times 10} = 0.7 < u_{min} = 0.75$
- Nút D là nút Home. Chuyển lưu lượng xuống
- $T(C,D)=T(D,B)=0+T(C,B)=21$

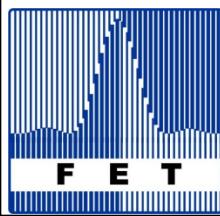


Ví dụ 2 Thêm lưu lượng (12)

- (D,G) 2 hops
- (E,G) 2 hops
- (C,D) 2 hops
- (D,B) 1 hop dung lượng 21

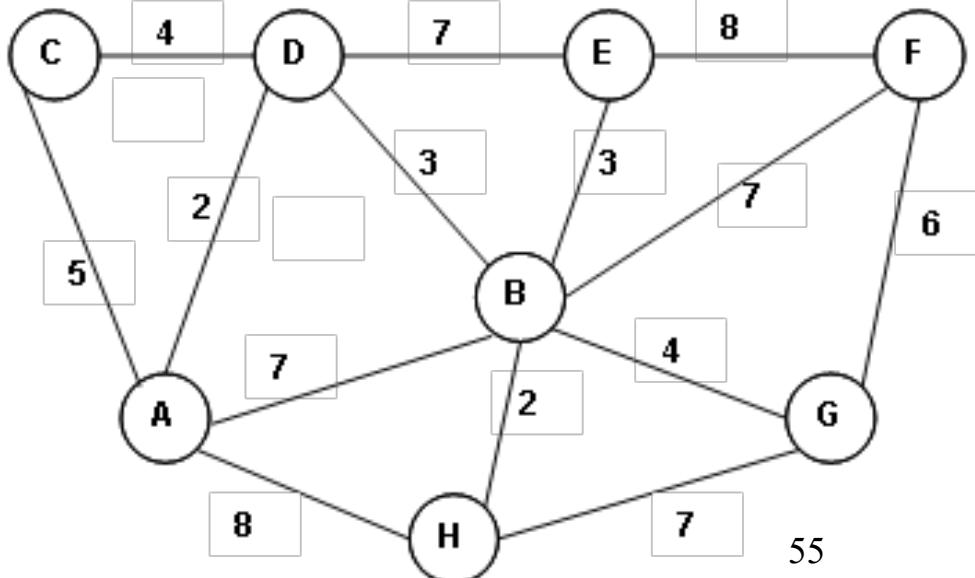
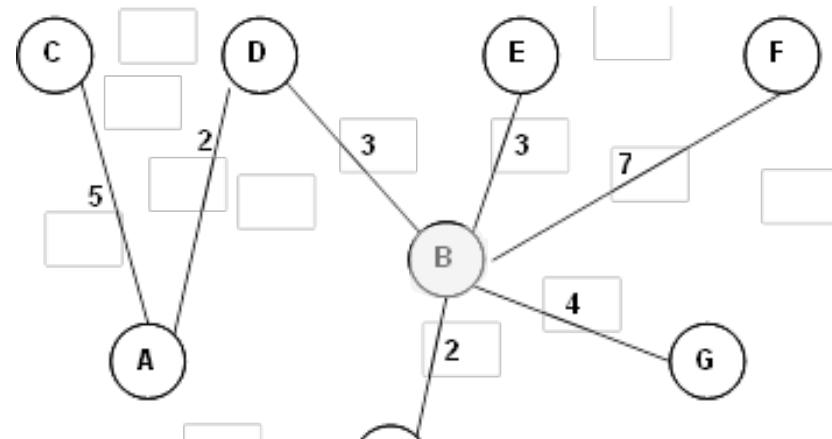


54



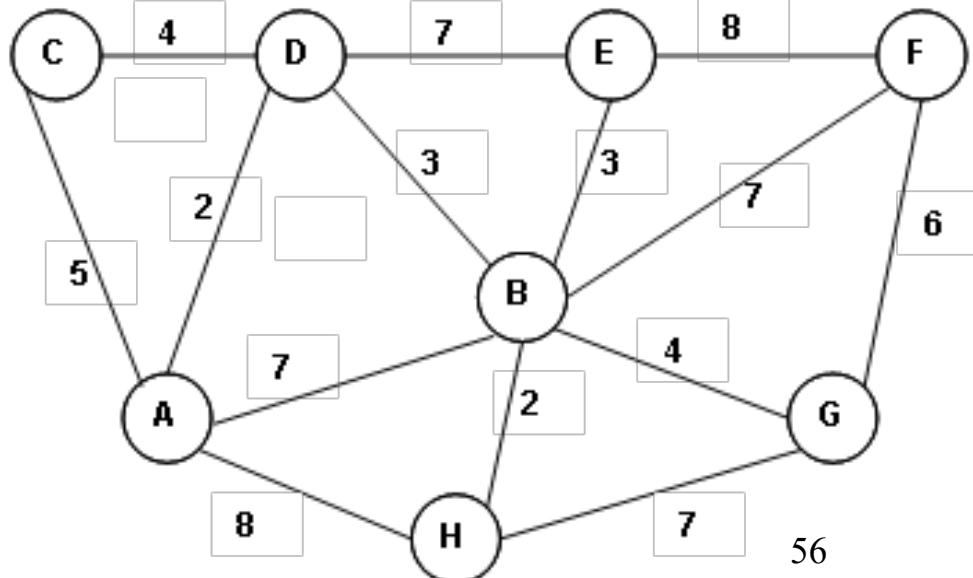
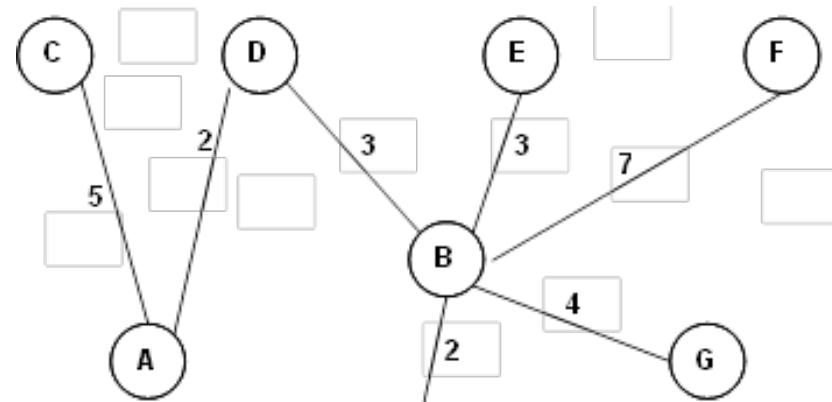
Ví dụ 2 Thêm lưu lượng (12)

- Xét (D,G) 2 hops
- $n = roundup \frac{T(D,G)}{C} = roundup \left(\frac{12}{10} \right) = 2$
- $u = \frac{12}{2 \times 10} = 0.6 < u_{min} = 0.75$
- Nút B là nút Home. Chuyển lưu lượng xuống
- $T(D,B)=T(D,B) cũ + T(D,G)=21+12=33$
- $T(B,G)=0+T(D,G)=12$



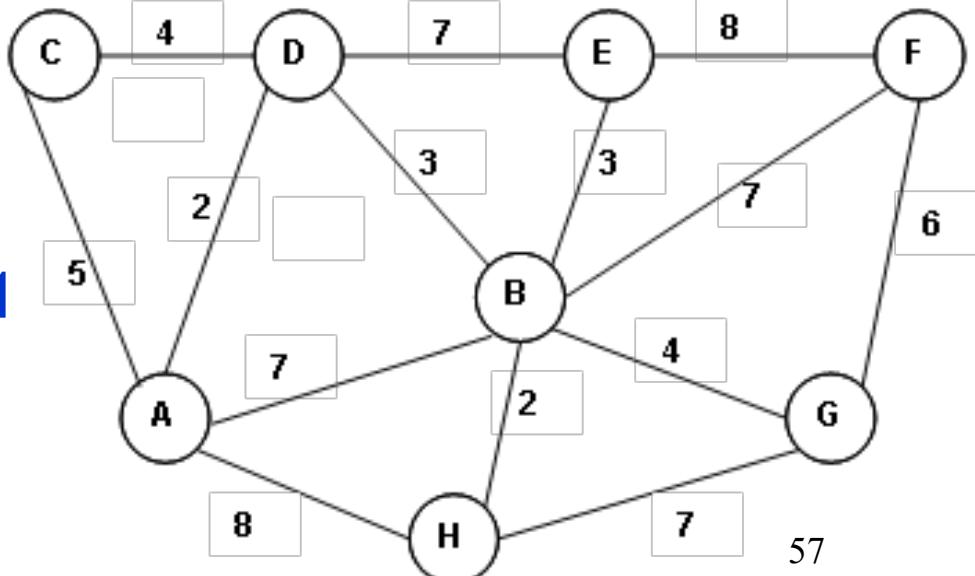
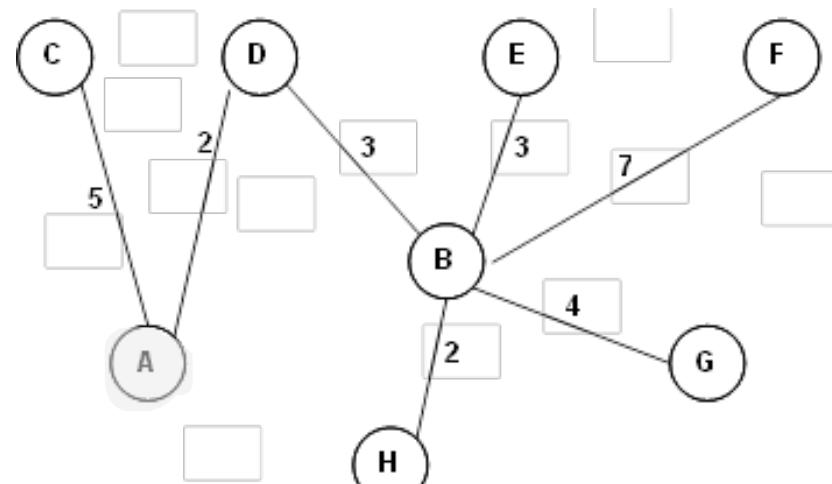
Ví dụ 2 Thêm lưu lượng (13)

- Xét (E, G) 2hops
- $n = roundup \frac{T(E,G)}{C} = roundup \left(\frac{9}{10} \right) = 1$
- $u = \frac{9}{10} = 0.9 > u_{min} = 0.75$
- Thêm 01 liên kết trực tiếp giữa E và G



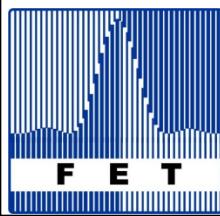
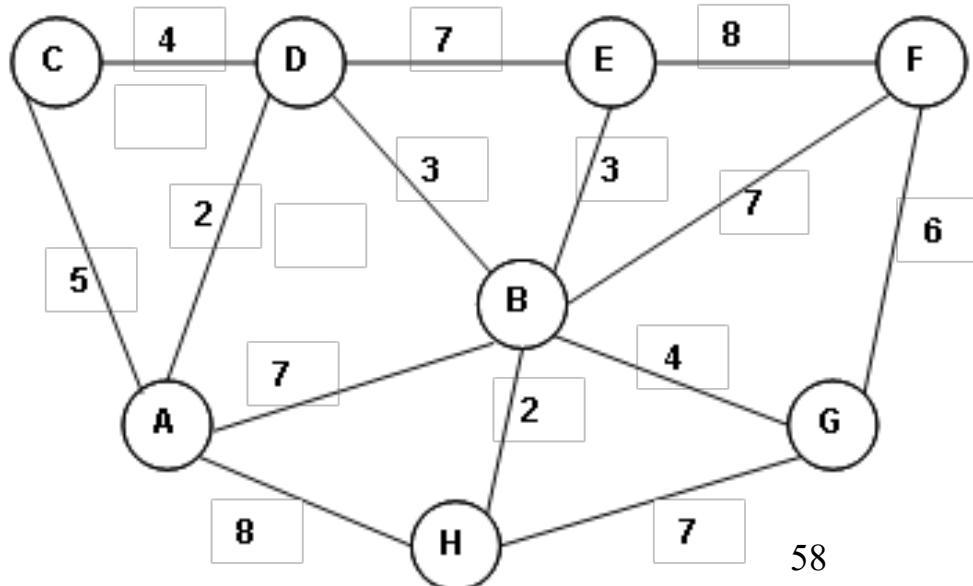
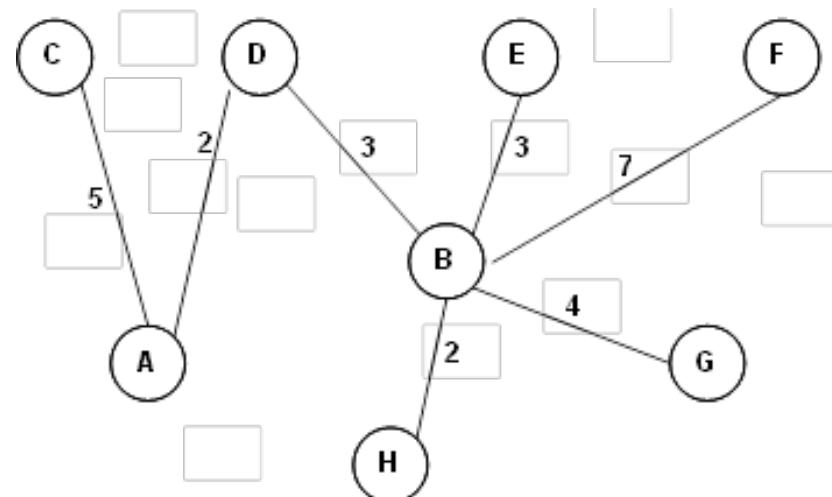
Ví dụ 2 Thêm lưu lượng (13)

- Xét (C,D) 2 hops
- $n = roundup \frac{T(C,D)}{c} = roundup \left(\frac{21}{10} \right) = 3$
- $u = \frac{21}{3 \times 10} = 0.7 < u_{min} = 0.75$
- Nút A là nút Home. Chuyển lưu lượng xuống
- $T(C,A)=T(A,D)=0+T(C,D)=21$



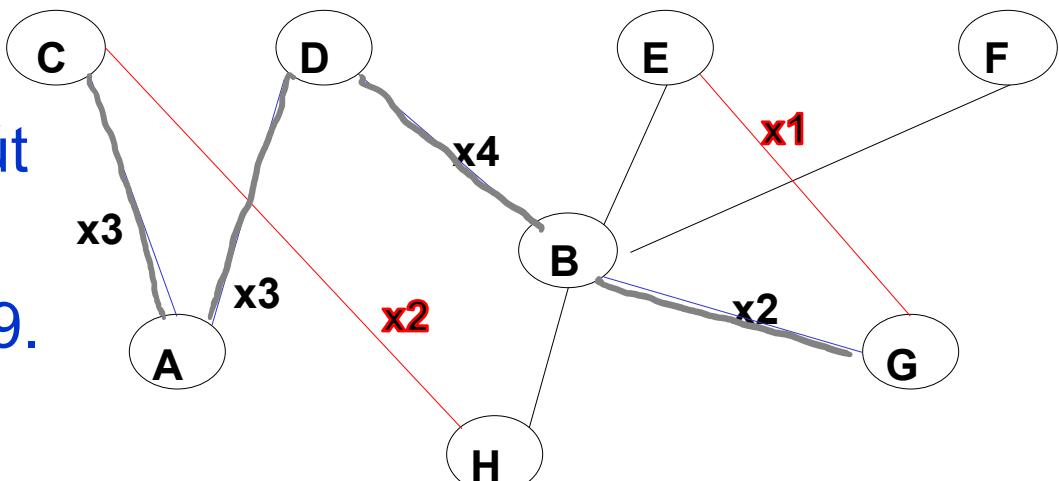
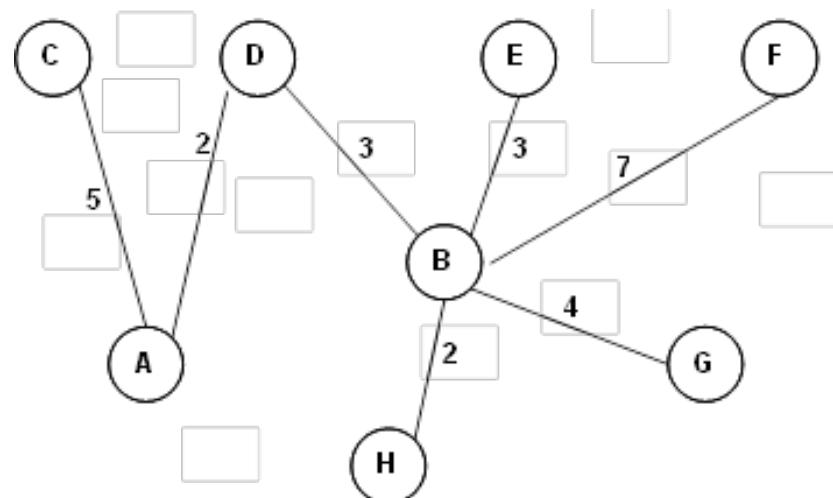
Ví dụ 2 Thêm lưu lượng (14)

- Lưu lượng 1 hop
- $T(C,A)=T(A,D)=21 \rightarrow n = 3$
- $T(D,B)=33 \rightarrow n=4$
- $T(B,G)=12 \rightarrow n=2$



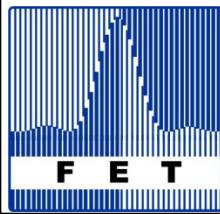
Ví dụ 2 Thêm lưu lượng (8)

- Lưu lượng 1 hop
- $T(C,A)=T(A,D)=21$ n =3
- $T(D,B)=33$ n=4
- $T(B,G)=12$ n=2
- Dung lượng các liên kết là 10.
- Lưu lượng giữa các nút như sau: $T(D,G)=12$, $T(C,H)=17$ và $T(E,G)=9$.
 $T(C,B)=21$



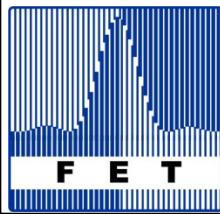
Nhận xét

- Thuật toán thêm liên kết kết hợp lưu lượng để chữa các liên kết giữa các nút mà có nhiều bước nhảy
- Nếu lưu lượng giữa N1 và N2 không thể đi qua đường liên kết trực tiếp thì nó sẽ được định tuyến thông qua nút home H
- Cuối cùng, trong mạng lớn, đủ lưu lượng để kết hợp trên đường liên kết trực tiếp



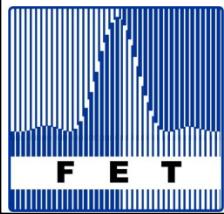
Nhận xét (2)

- Chất lượng hoạt động của MENTOR được giảm sát bởi thông số độ sử dụng và tham số xây dựng cây Prim-Dijkstra α
- Dễ dàng thêm liên kết mới, điều khiển bằng $umin$
- Dạng cây ban đầu được điều khiển bằng α



Chất lượng hoạt động của MENTOR

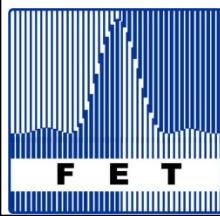
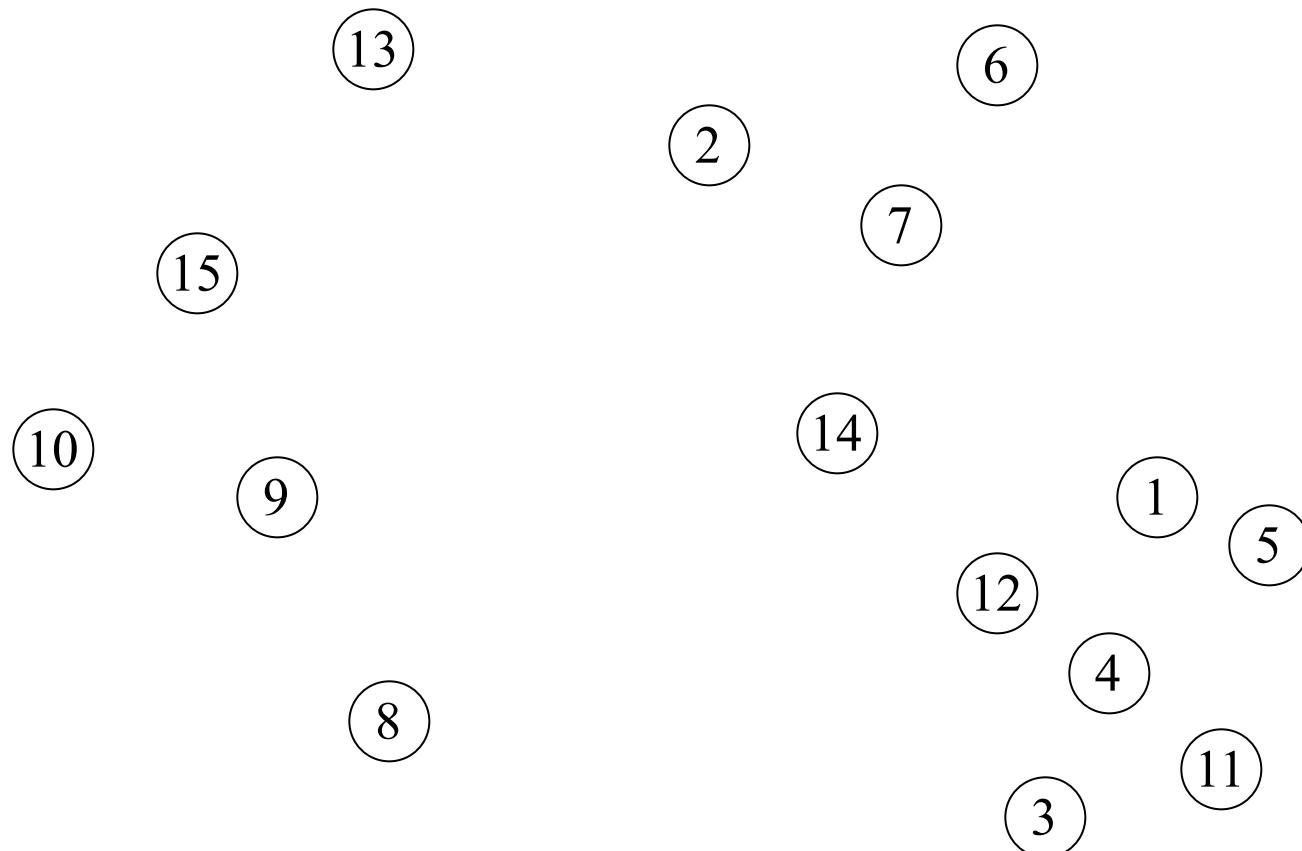
- Thuật toán giá thành thấp
- 03 bước chính
- Lựa chọn xương sống
- Xây dựng cây
- Thêm liên kết
- Tất cả đều $O(n^2)$
 - Có thể phải làm nhiều lần, thay đổi thông số



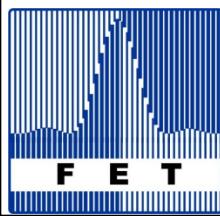
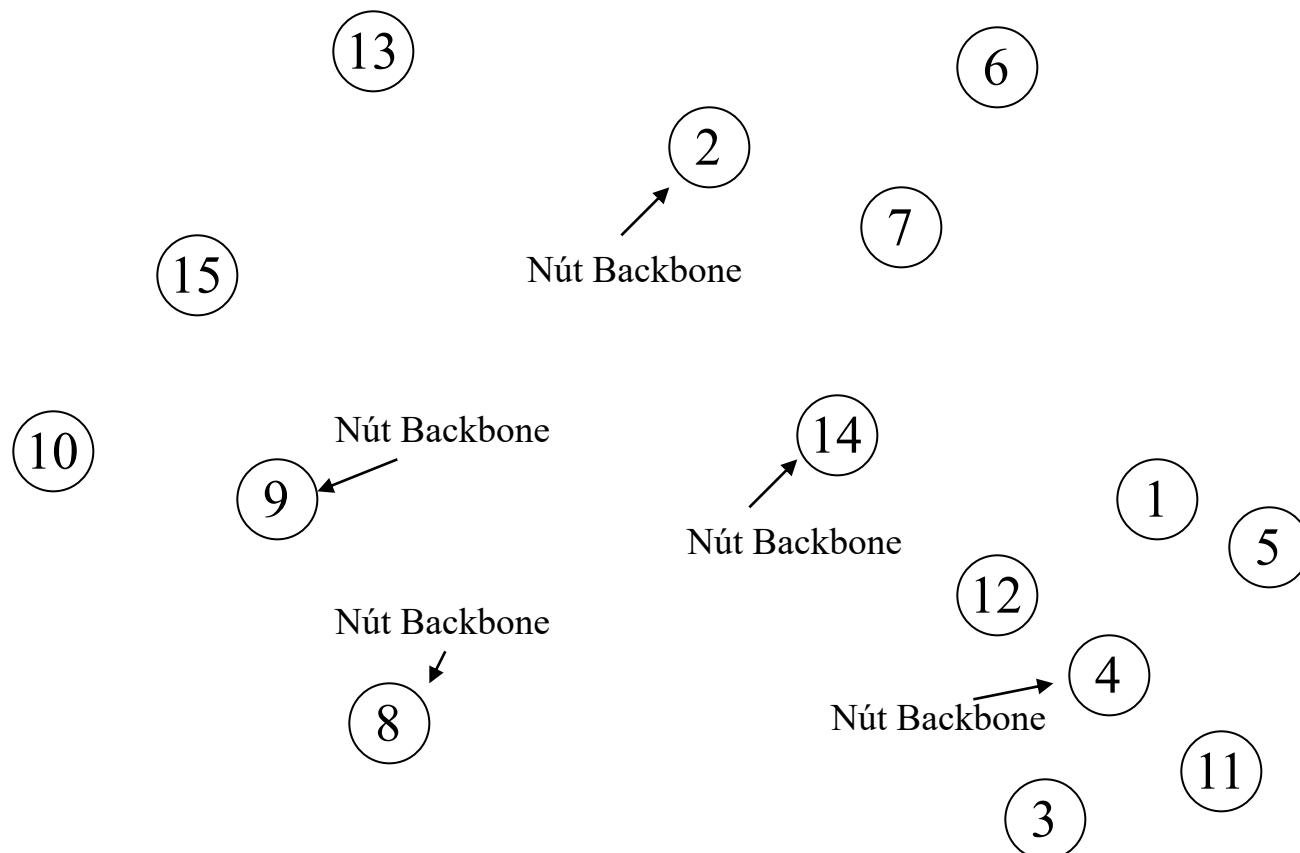
Ví dụ MENTOR

FTP site

15 nút , tốc độ 60 256 kbps circuits



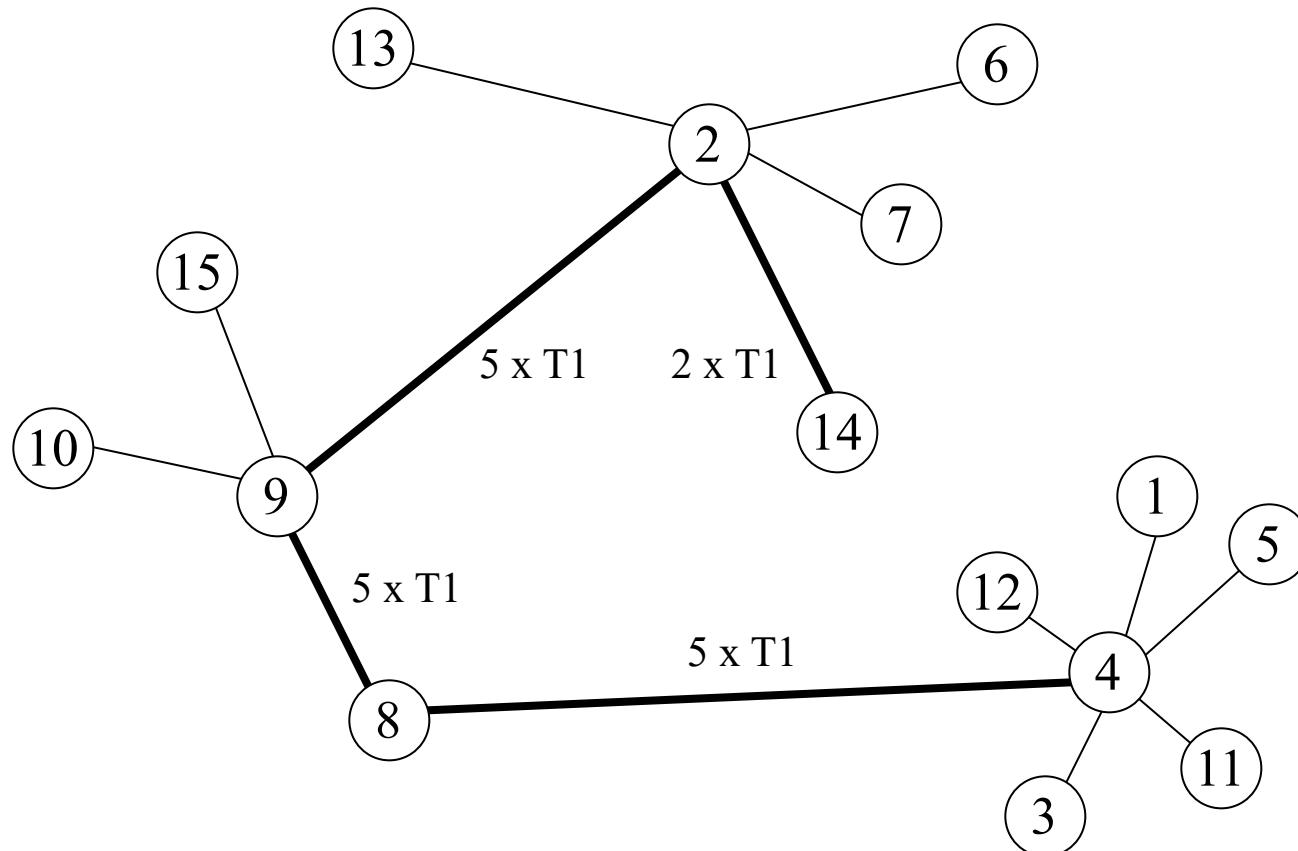
Lựa chọn nút xương sống



Thiết kế ban đầu

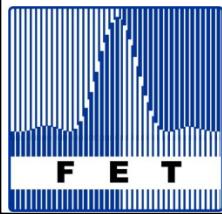
$\alpha = 0$

Giá = \$269,785/tháng



Duyệt lại thiết kế ban đầu

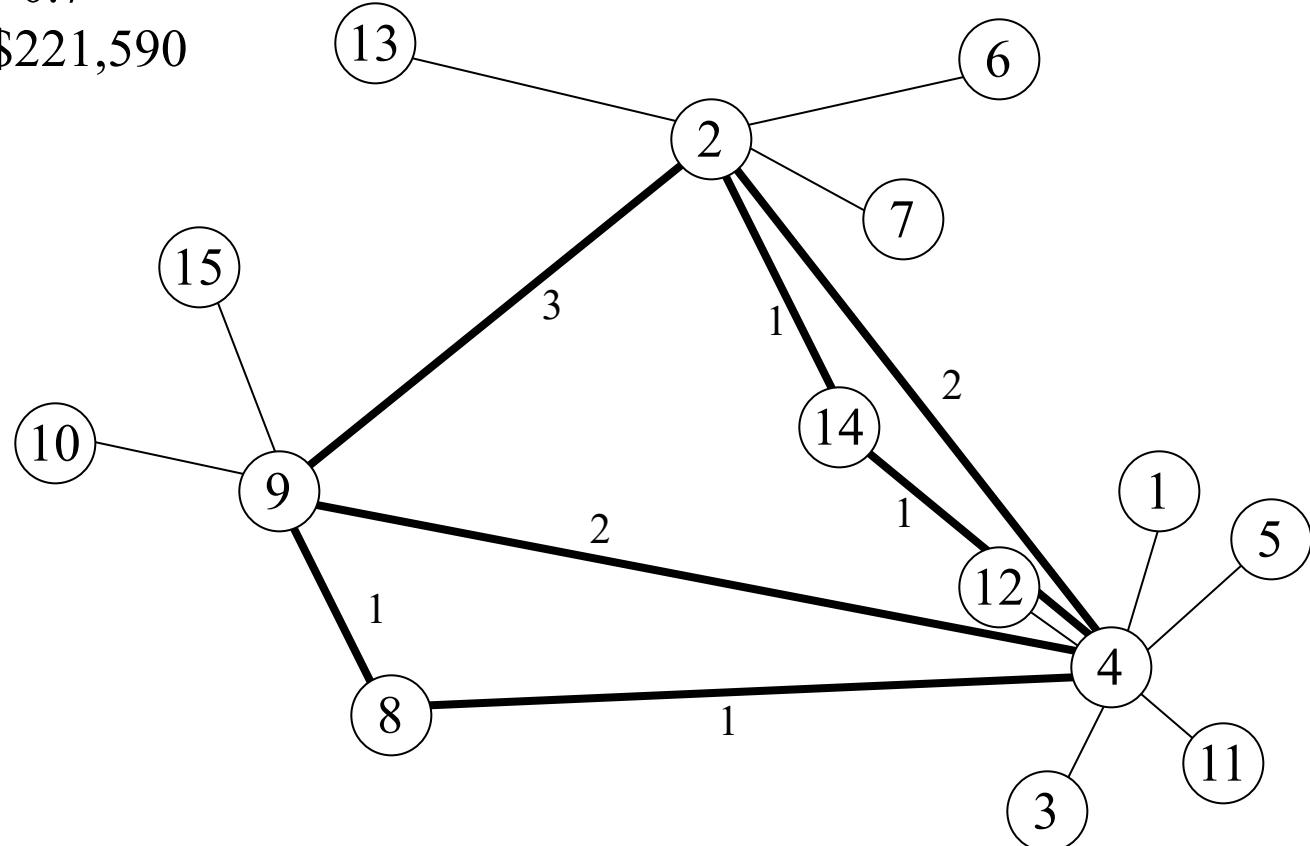
- Các liên kết xương sống có nhiều (5) liên kết T1 →
Có thể không phải là tốt
- Nguyên tắc thiết kế :
 - Nếu thiết kế có nhiều đường liên kết tốc độ cao song song, thường thì thiết kế mảng lưới sẽ tốt hơn
 - Giá rẻ hơn, độ đa dạng lớn hơn
 - Chú ý điều này không thể chứng minh toán học



Thiết kế hiệu chỉnh

$u_{min} = 0.7$

Giá = \$221,590

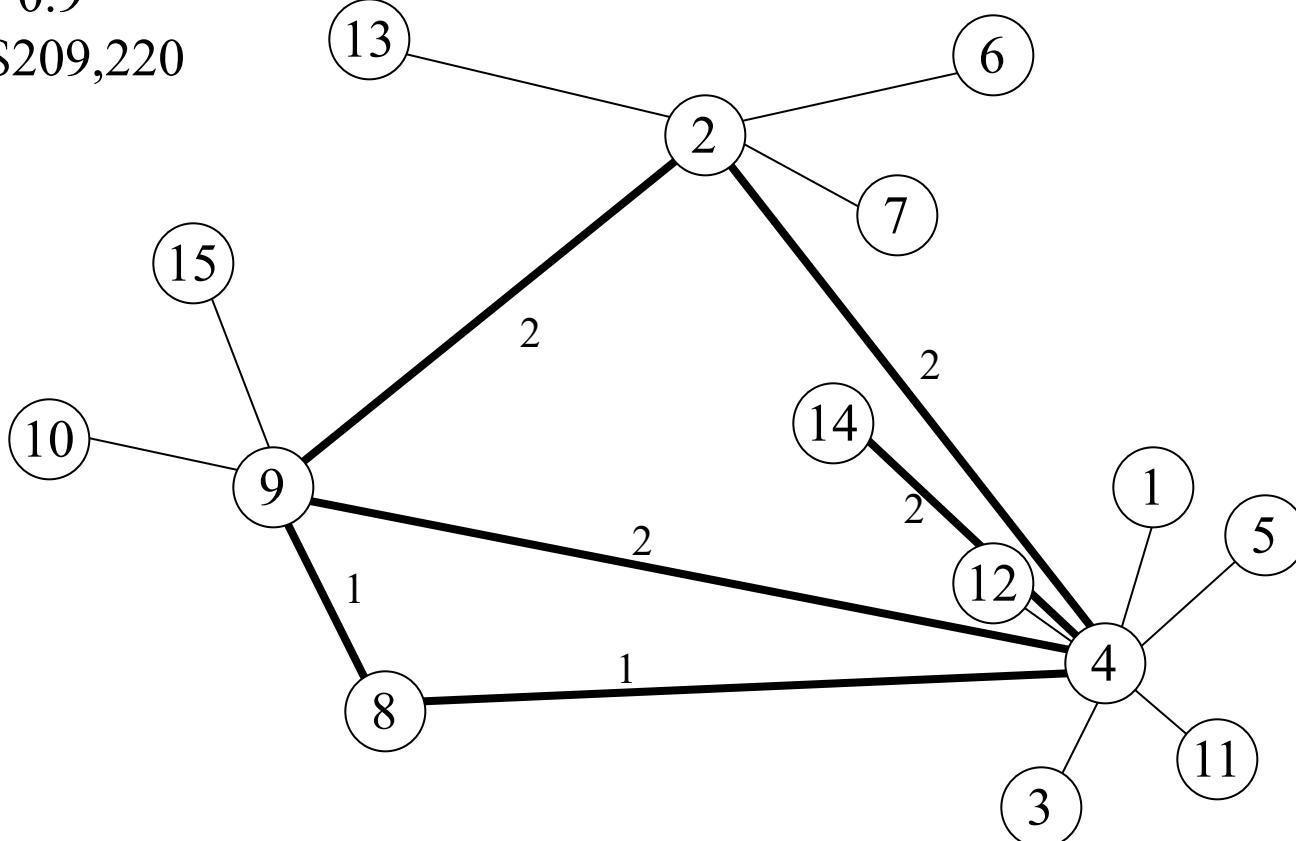


Thiết kế 5-node xương sống “tốt nhất”

$\alpha = 0.1$

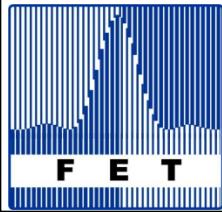
umin = 0.9

Giá = \$209,220



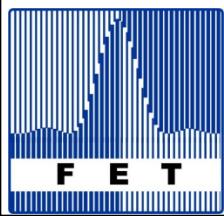
Nhận xét

- Chú ý rằng chúng ta tạo ra nhiều thiết kế bằng cách thay đổi một số thông số và lựa chọn thiết kế tốt nhất
- Tuy nhiên, không có gì đảm bảo là thiết kế này thực tế là “tốt nhất”
- Thực tế, thay đổi số nút xương sống có thể đưa ra thiết kế tốt hơn
 - 13-nút xương sống cho thiết kế giá \$191,395
 - 12-nút xương sống giá \$198,975

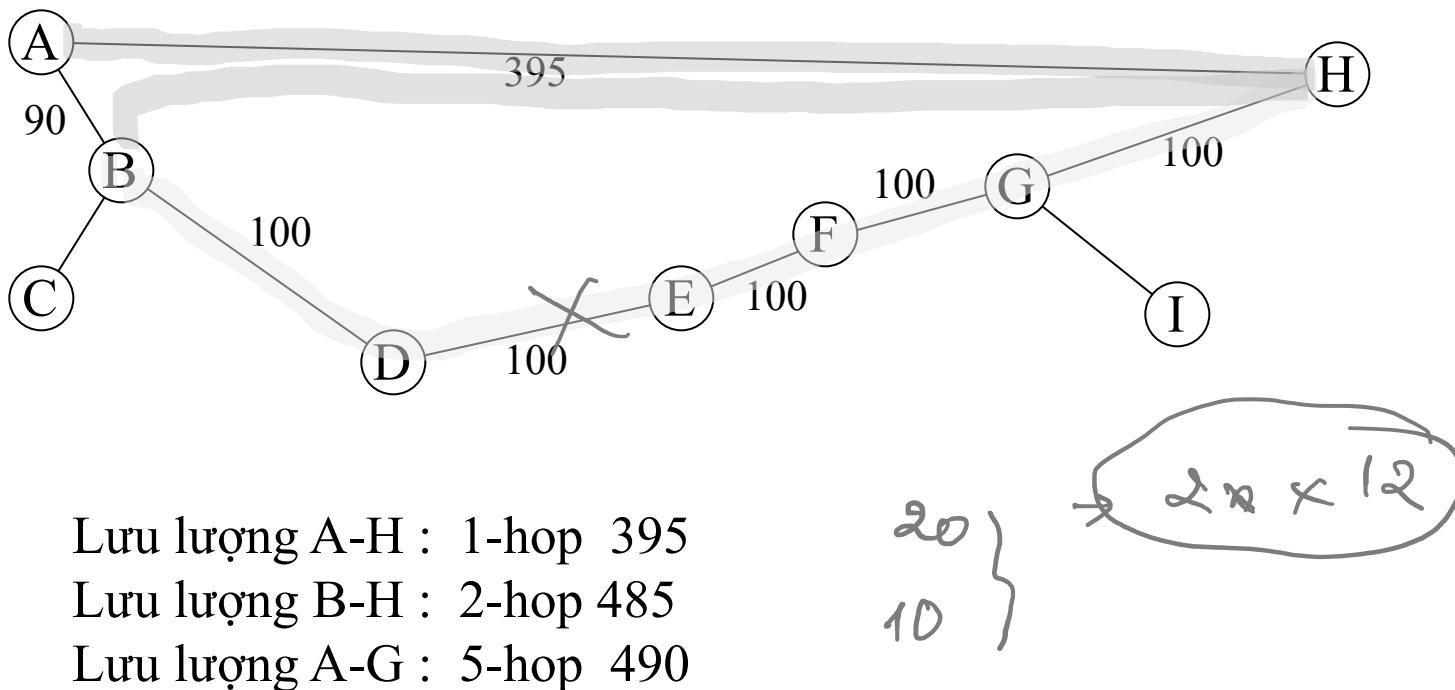


Định tuyến

- Sau khi chúng ta thiết kế mạng xong, chúng ta xem xét lưu lượng sẽ đi qua đâu
- Đưa ra một vấn đề mà trung tâm chính là **chất lượng hoạt động** của thuật toán định tuyến
- Với mỗi cặp nút N_0 and N_1 , định nghĩa tuyến bằng (N_0, N_1, h, n)
 - Trong đó $n = 0$, nếu h là liền kề với N_0 và $n = 1$ nếu h liền kề với N_1
- Nếu N_0 và N_1 là liền kề, chúng ta có đường trực tiếp
- Nếu không đường là liên kết (N_0, h) và đường ($N_1 - n, h, h^*, n^*$)
 - Cứ tiếp tục như vậy cho đến khi toàn bộ đường được thiết lập

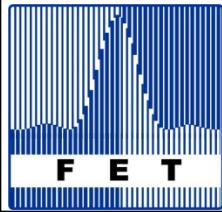


MENTOR với OSPF



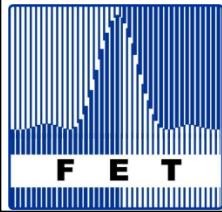
MENTOR-II

- Định tuyến có một số giới hạn
 - Nó có thể không tìm thấy đường khả thi cho thiết kế mạng của chúng ta
- MENTOR-II cải thiện giải thuật thiết kế có tính đến giới hạn của thuật toán định tuyến
 - Như vậy sẽ tăng độ phức tạp của quá trình thiết kế



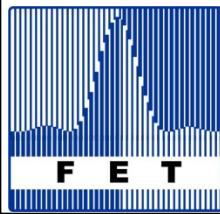
Incremental Shortest Path (ISP)

- Mục đích của thuật toán ISP là xác định tất cả những cặp có thể sử dụng liên kết trực tiếp thay cho đường hiện thời
 - Ban đầu tất cả các đường đều đi qua cây
 - Sau khi thêm liên kết trực tiếp, tình thế trở nên phức tạp hơn



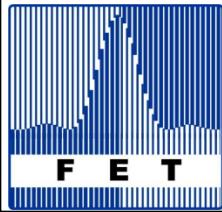
ISP (2)

- Có 02 ma trận nxn
 - Khoảng cách đường ngắn nhất Shortest-path distances (sp_dist)_{nxn}
 - Ma trận con trả nút trên đường ngắn nhất (sp_pred)_{nxn}
 - Lưu tất cả các đường ngắn nhất trong mạng đồng thời
 - Cập nhật ma trận sau mỗi lần thêm liên kết

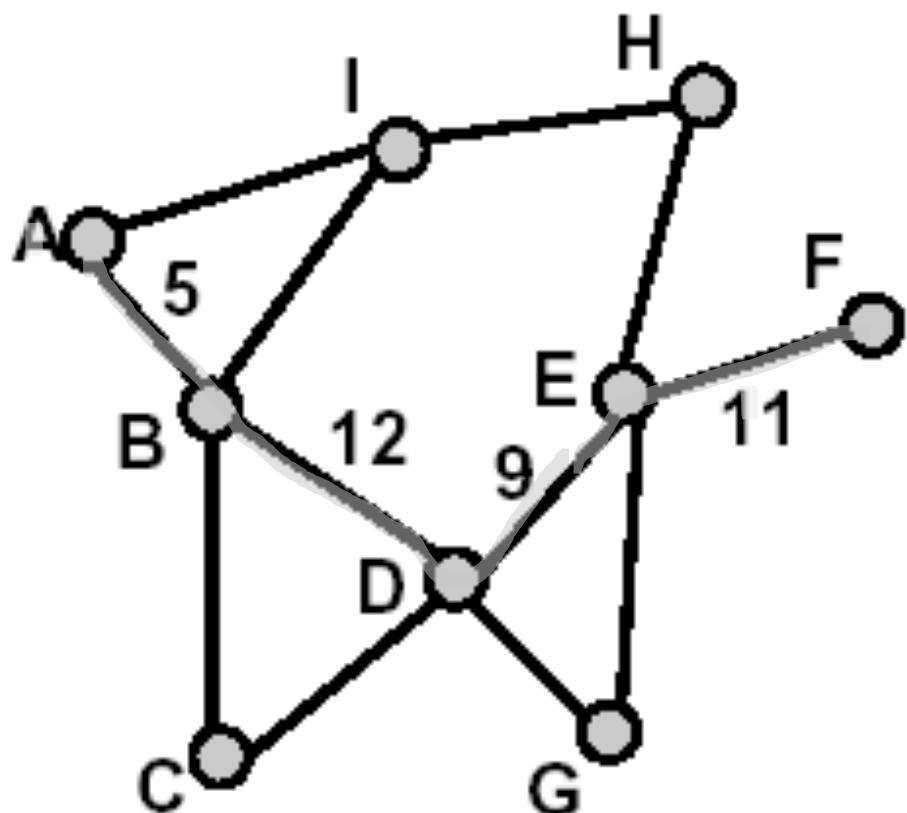


sp_pred

- $\text{sp_pred}(i,j) = m$ chứa nút tiếp theo trên đường ngắn nhất từ i đến j
- Có thể đi ngược cây từ i đến j bằng việc xem tại $\text{sp_pred}(i,m)$, v.v



Ví dụ : ma trận



For pair (A,F):

$\text{sp_dist}[A,F] = 37$

$\text{sp_pred}[A,F] = E$

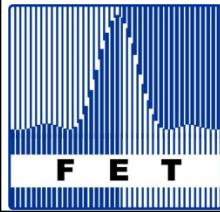
$\text{sp_pred}[A,E] = D$

$\text{sp_pred}[A,D] = B$

$\text{sp_pred}[A,B] = A$

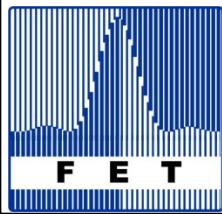
ISP trong MENTOR II

- Mỗi liên kết phải được ấn định độ dài lớn hơn giá của nó
 - Do không muốn cho liên kết có độ dài ngắn phi lý

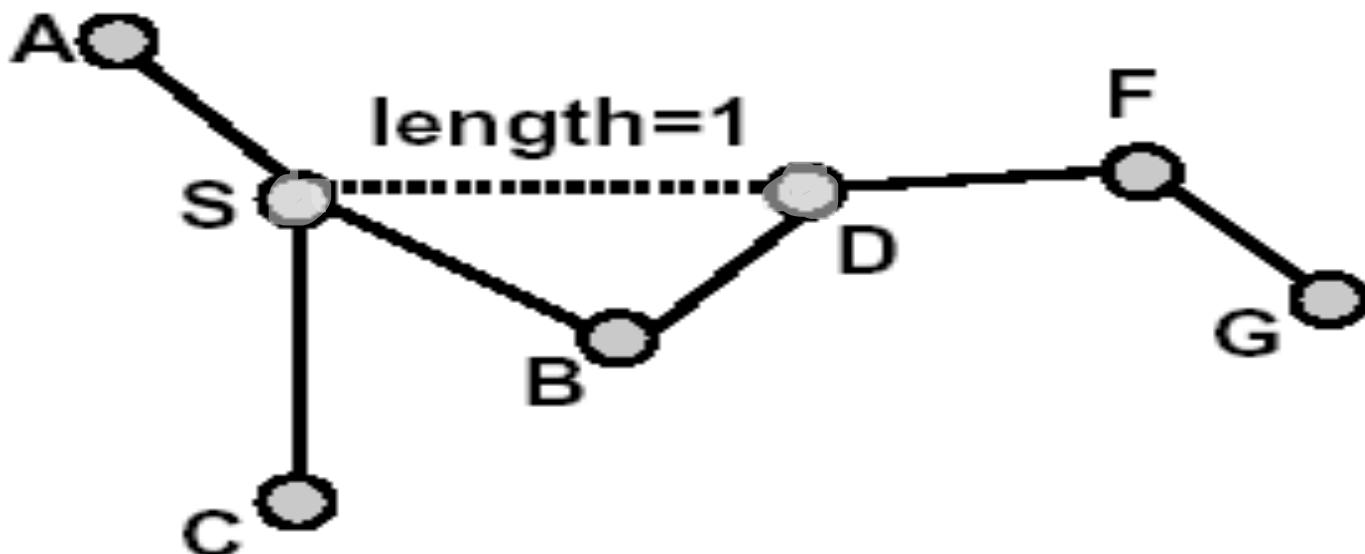


ISP trong MENTOR II (2)

- Khi xem xét liệu có thêm một liên kết trực tiếp giữa nguồn S và đích D, xây dựng s_list và d_list
- Liên kết trực tiếp có độ dài L
- Thêm nút vào s_list nếu
 $sp_dist[\text{node}, s] + L < sp_dist[\text{node}, d]$
- Thêm nút vào d_list nếu
 $sp_dist[\text{node}, d] + L < sp_dist[\text{node}, s]$
Có thể có nút sẽ có đường ngắn hơn nhờ vào liên kết mới



Ví dụ : s_list và d_list



s_list = { A, C, S }

d_list = { F, G, D }

$\textcircled{B}:$ $B \rightarrow S = 1$. $1 + 1 > 1$
 $B \rightarrow D = 1$

ISP trong MENTOR II (3)

- Xem tất cả mọi cặp (ni,nj) trong đó ni thuộc s_list và nj thuộc d_list
- Nếu $sp_dist[ni,s] + L + sp_dist[d,nj] < sp_dist[ni,nj]$
Khi đó lưu lượng (ni,nj) sẽ chuyển sang liên kết dự định
Độ dài lớn nhất cho lưu lượng (ni,nj) để chuyển đi
 $maxL = sp_dist[ni,nj] - sp_dist[d,nj] - sp_dist[ni,s]$



ISP in MENTOR II (4)

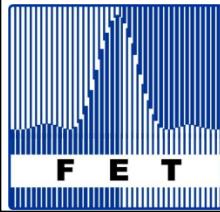
- Chúng ta có sắp xếp các cặp theo MaxL và có thứ tự - ví dụ

$$\text{maxL(P1)} = 2000$$

$$\text{maxL(P2)} = 1800$$

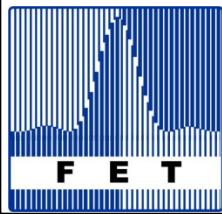
$$\text{maxL(P3)} = 1800$$

$$\text{maxL(P4)} = 1700$$



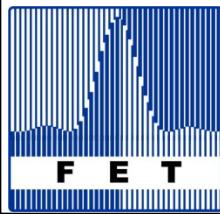
MENTOR II – Tóm lại(1)

1. Chia các nút thành nút xương sống và nút đầu cuối
2. Lựa chọn nút trung tâm
3. Xây dựng cây Prim-Dijkstra, ấn định mỗi liên kết một trọng số tương ứng với giá tiền
4. Tính toán khoảng cách trên cây Prim với mọi cặp nút: ma trận $sp_dist[][]$ và $sp_pred[][]$
5. Thêm liên kết 1- luồng (*Liên kết giữa 2 nút đầu cuối hoặc giữa 01 nút đầu cuối và nút backbone*)
 - 1 luồng là lưu lượng giữa một cặp nút nguồn – đích
 - Liên kết 1 luồng là liên kết chỉ mang lưu lượng giữa hai điểm.



MENTOR II – Tóm lại (2)

6. Chuyển các yêu cầu nút sang xương sống
 - E1 và e2 là nút đầu cuối với b_i là nút backbone
 - Nếu $b_1 = b_2$, thì lưu lượng sẽ không được đưa vào mạng xương sống .
 - Còn nếu không thì chuyển lưu lượng giữa e1 và e2 vào b1 và b2
7. Sắp xếp các cặp xương sống theo thứ tự giảm dần của độ dài khoảng cách

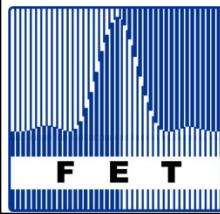


MENTOR II – Tóm lại

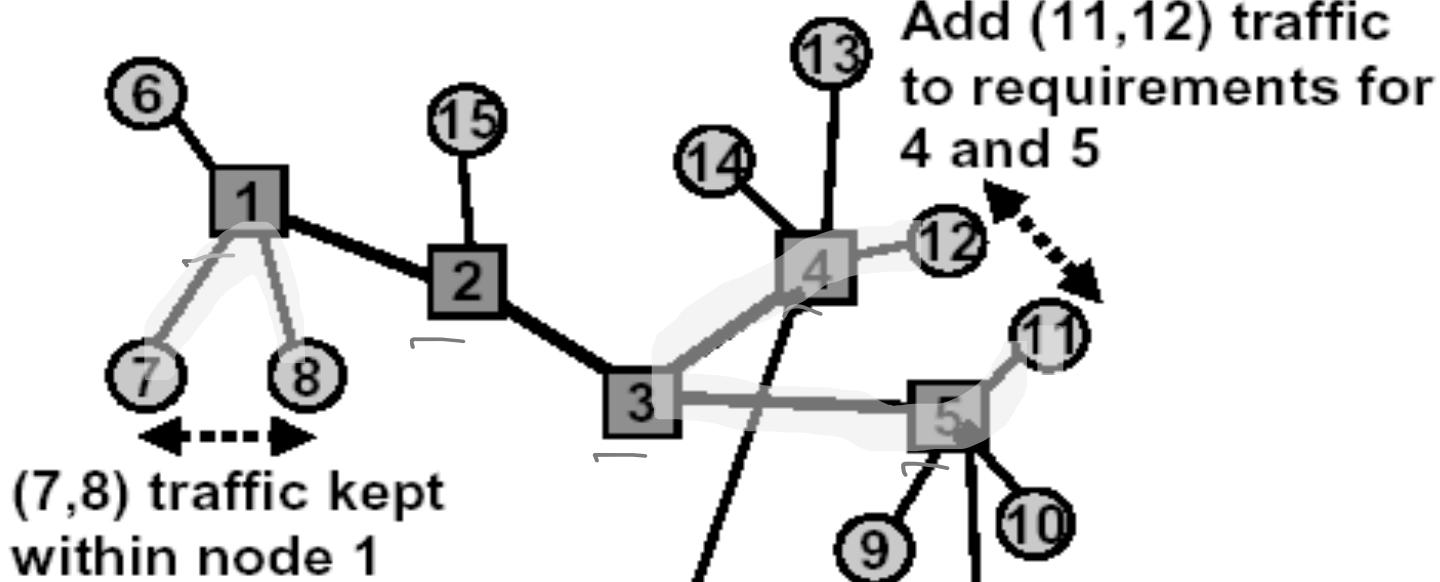
8. Xem xét mỗi cặp P sử dụng thuật toán ISP. Với mỗi cặp, giải thuật tính tập các cặp xương sống (P_1, P_2, \dots, P_k) có thể sử dụng liên kết trực tiếp sao cho
 - $\text{req_len}_1 > \text{req_len}_2 > \dots > \text{req_len}_k$

Thêm liên kết với đội dài d, nếu với giá trị này độ sử dụng của liên kết là đủ

9. Thiết lập độ dài liên kết 01 luồng để chỉ mang lưu lượng giữa hai điểm cuối.



Chuyển yêu cầu

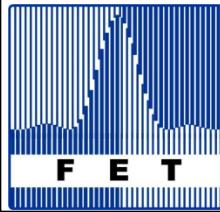


$$w(4) := w(4) + F_{11,12} + F_{12,11}$$

$$w(5) := w(5) + F_{11,12} + F_{12,11}$$

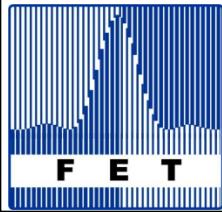
Liên kết 01 luồng

- MENTOR-II có điểm yếu khi thuật toán nhóm vùng chọn quá ít nút xương sống
 - Lựa chọn nút xương sống xác định bởi tham số WPARM và RPARM



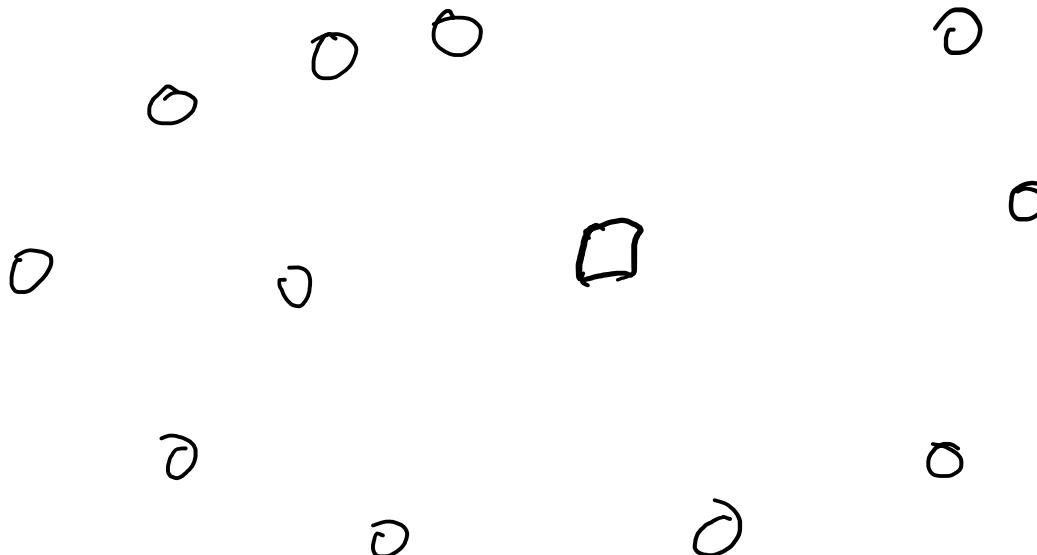
Lựa chọn thông số sai

- Lựa chọn thông số sai có thể dẫn đến nhiều vấn đề
- Ví dụ, WPARM = 100, RPARM = 1
 - Thường là, chỉ có một nút sẽ được chọn là nút xương sống vì WPARM khá lớn
 - *Vậy tính toán thường có cho thêm nút xương sống không?*



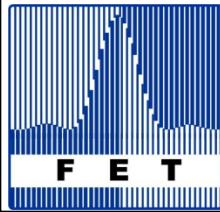
Lựa chọn thông số sai (2)

- Sau đó không có nút nào được chọn là nút xương sống do RPARAM
- Với chỉ có một nút xương sống, không có liên kết trực tiếp nào sẽ được nối
- Kết quả cuối cùng là mạng cây



Giải quyết vấn đề

- Thêm liên kết gọi là liên kết 1 luồng giữa các điểm cuối
- Chỉ mang lưu lượng giữa điểm cuối được liên kết
- Thêm bước mới vào thuật toán MENTOR để thêm liên kết 1 luồng khi thích hợp



Thêm liên kết

Với mỗi cặp ($N1, N2$), thực hiện thuật toán sau:

1. Nếu dung lượng của liên kết là C , tính

$n = \text{ceil}[\text{lưu lượng } (N1, N2)/C]$ số lượng lưu lượng cần thiết

2. Tính độ sử dụng

$u = \text{lưu lượng } (N1, N2)/(n*C)$

3. Thêm liên kết giữa $N1$ và $N2$ nếu $u > umin$

4. Nếu $u < umin$ chuyển lưu lượng 1 hop qua mạng

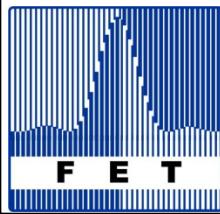
có nghĩa là, thêm lưu lượng $(N1, N2)$ vào cả lưu lượng $(N1, H)$ và
lưu lượng $(H, N2)$

Và làm tương tự với lưu lượng $(N2, N1)$

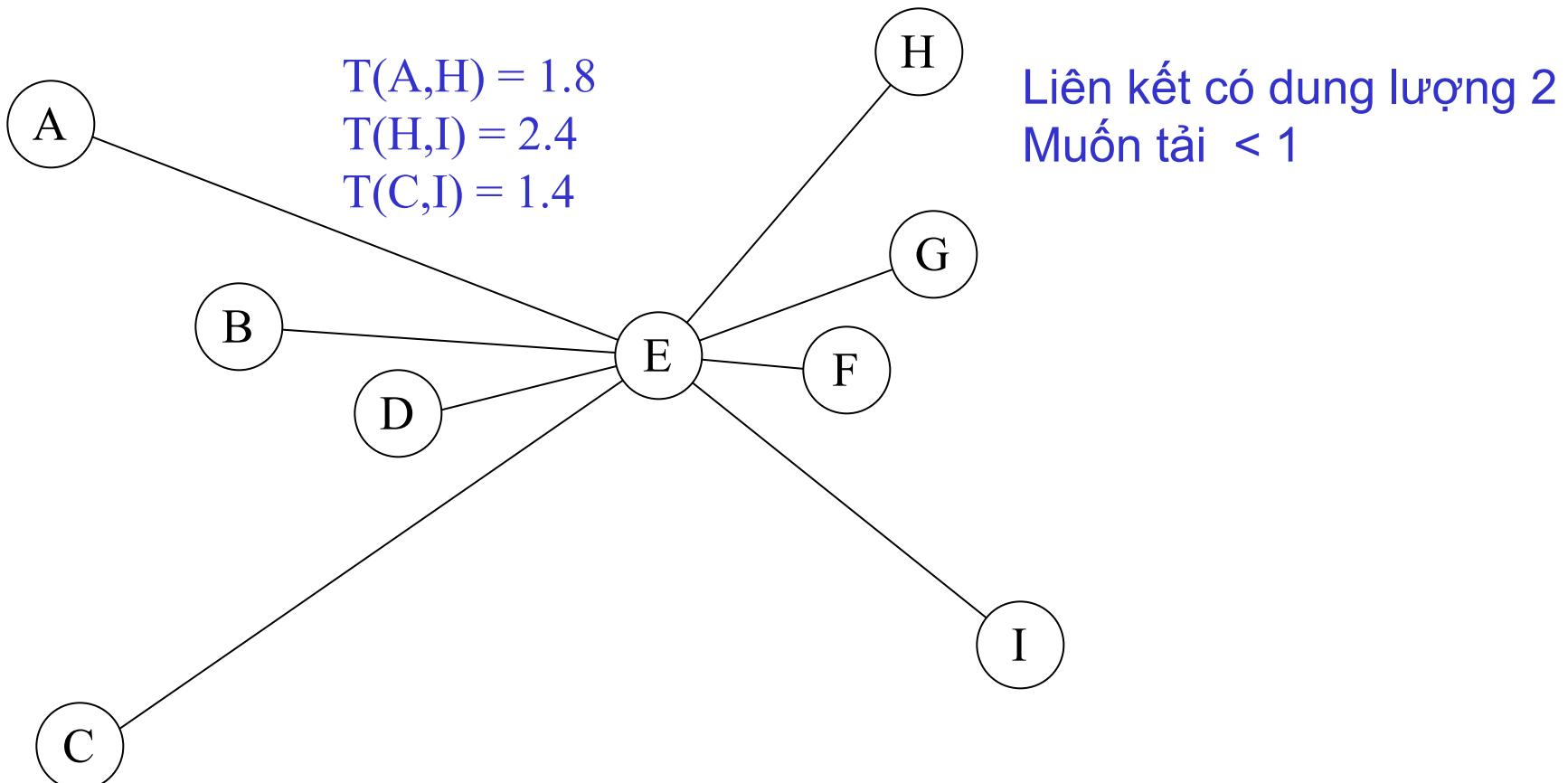
Chú ý - trường hợp đặc biệt nếu $(N1, N2)$ thuộc cây gốc. Trong

trường hợp này chỉ thêm liên kết $(N1, N2)$ vào thiết kế

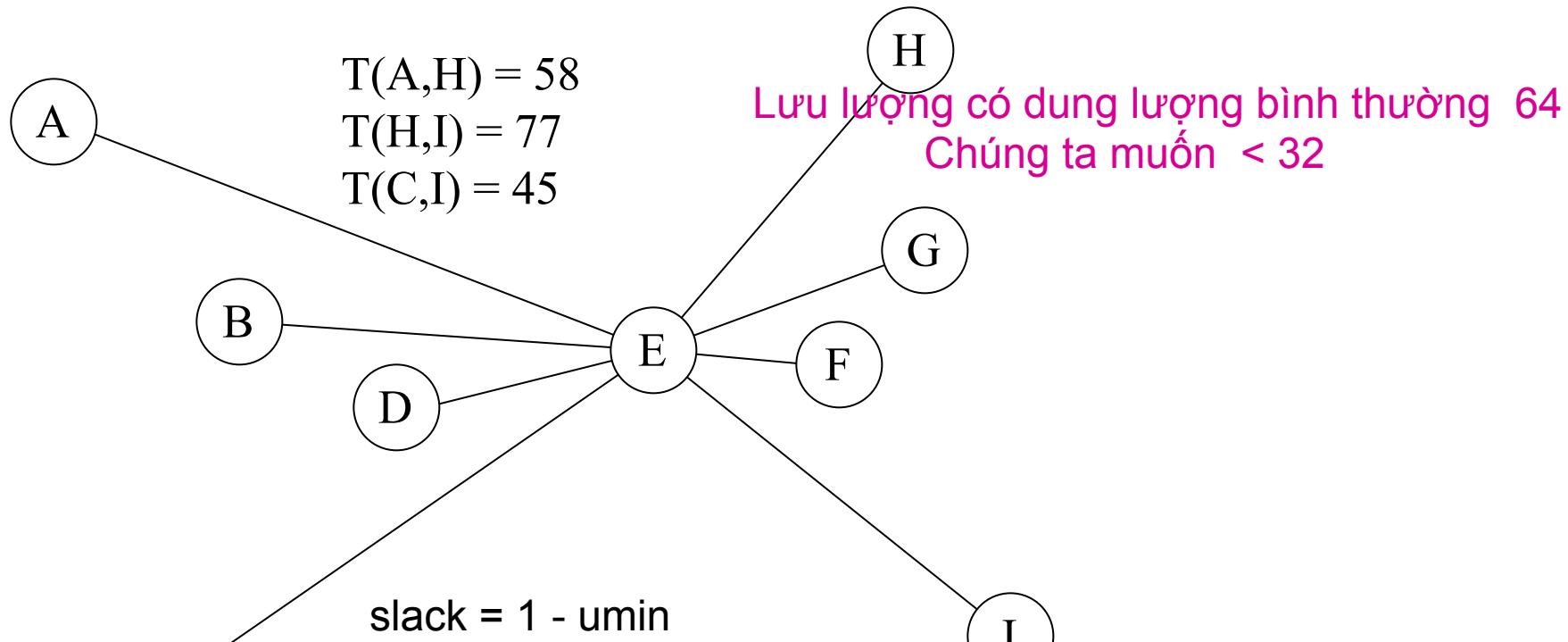
slack=1-umin



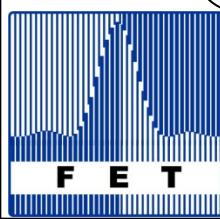
Ví dụ liên kết 1 luồng



Ví dụ liên kết 1 luồng (2)

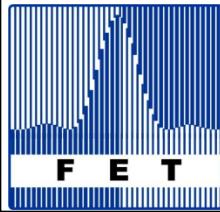


Nếu slack = 0.1, chúng ta thêm 2 đường song song A và H
Nếu slack = 0.2, thêm cả 3 liên kết H và I
Nếu slack = 0.3, thêm cả 2 liên kết C và I

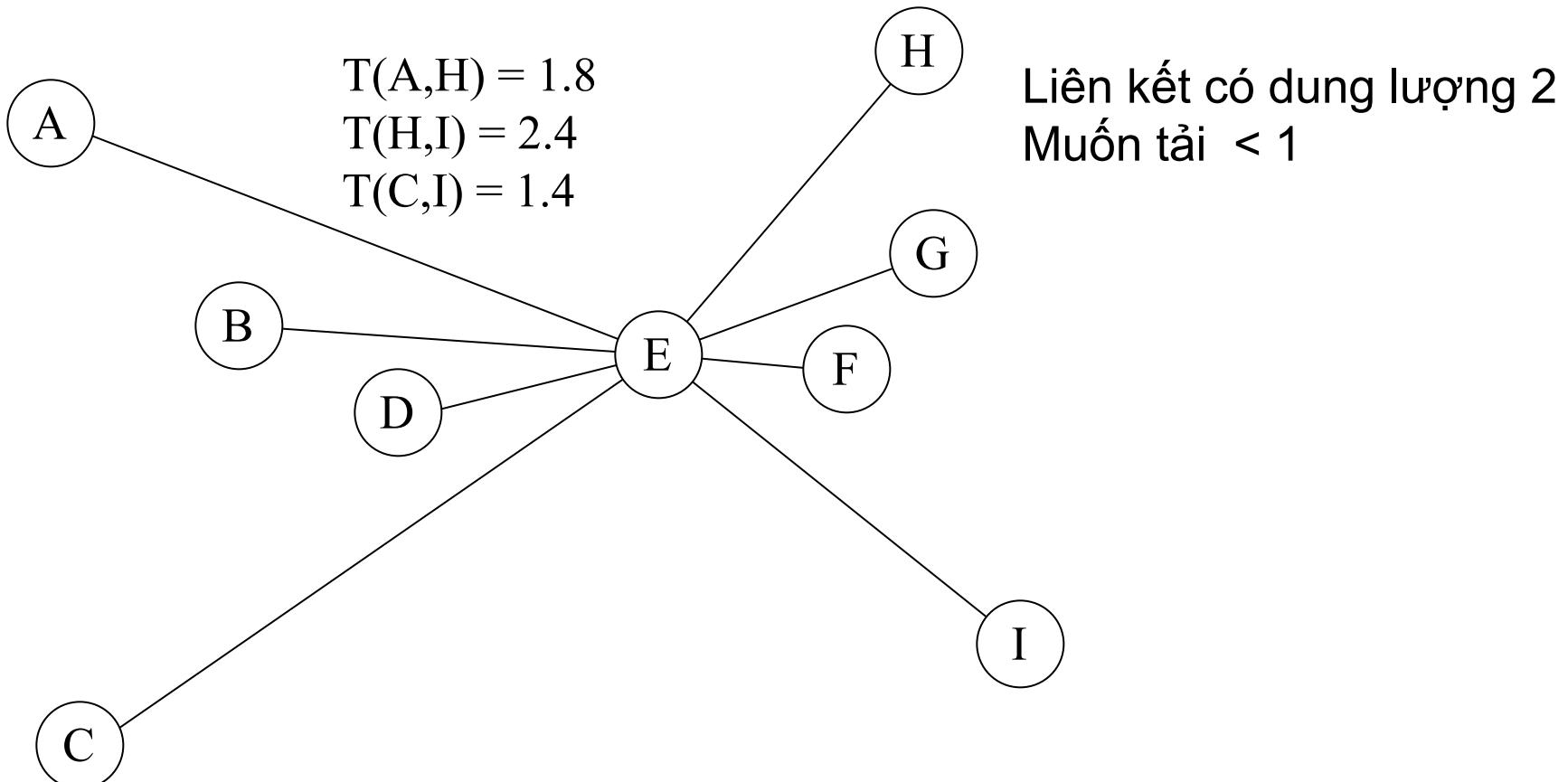


Ví dụ liên kết 1 luồng(3)

- Thêm liên kết A-đến-H, v..v có thể hoặc không là ý tưởng tốt hoàn toàn phụ thuộc vào phần còn lại của ma trận lưu lượng
- Không nên chuyển những yêu cầu này sang xương sống vì chúng được mang vào mạng riêng rẽ

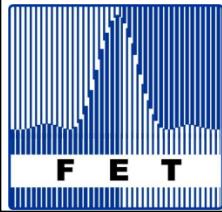


Ví dụ liên kết 1 luồng



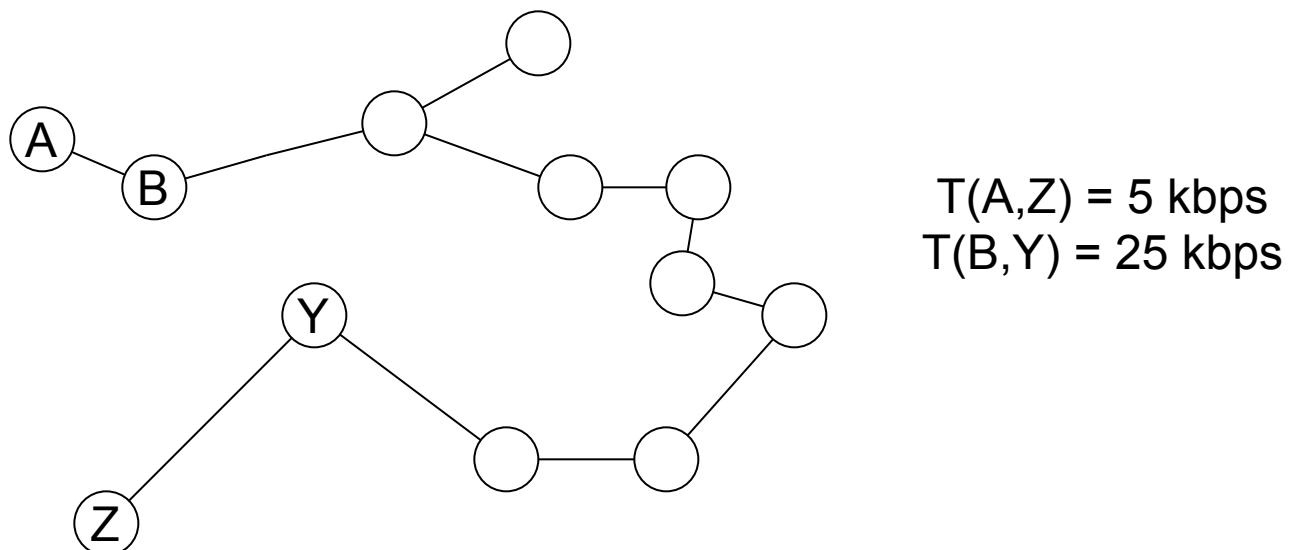
Thiết lập độ dài liên kết xương sống 1 luồng

- Để cho nó chỉ mang lưu lượng giữa 02 điểm cuối
- Giả sử tất cả các liên kết có độ dài > 1
- Với định tuyến đường ngắn nhất, độ dài đó sẽ được đặt bằng $sp_dist_{tree} - 1$



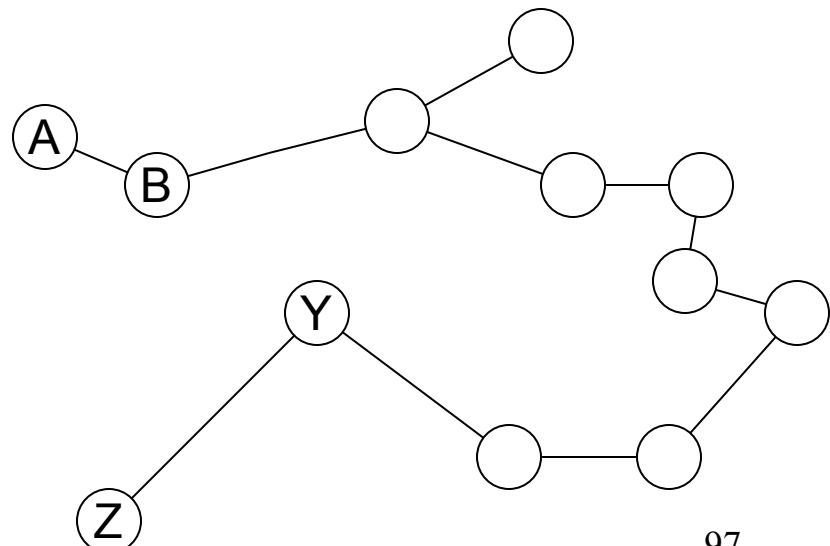
Sắp xếp lại (1)

- Giả sử liên kết dung lượng 64 kbps ,
- Thực hiện thêm liên kết trực tiếp nếu như liên kết này sẽ thu hút được lưu lượng từ 25 and 32 kbps



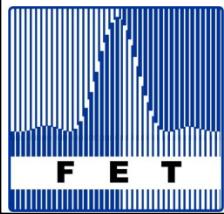
Sắp xếp lại(2)

- Khi xem xét từng cặp, (A, Z) sẽ được xem xét trước (B, Y), vì $sp_dist(A, Z) > sp_dist(B, Y)$
- Thuật toán sẽ thêm một liên kết giữa A và Z và xóa lưu lượng giữa B và Y.
→ Chuyển lưu lượng giữa các nút lớn lại đi thông nút nhỏ. → *Thiết kế tồi*
- . → Giải pháp là sắp xếp lại



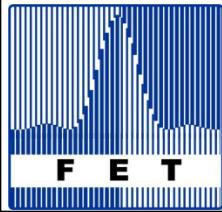
Sắp xếp lại(3)

- Việc sắp xếp lại được thực hiện như sau:
- Bắt đầu với danh sách của các cặp nút(P_1, P_2, \dots, P_k)
- Kiểm tra xem có cặp P_i nào trong danh sách có sau hai đặc điểm sau:
 - P_i chưa được thực hiện bằng cách thêm liên kết trực tiếp.
 - Có lưu lượng nhiều hơn gấp đôi so với lưu lượng giữa 2 điểm cuối của cặp P



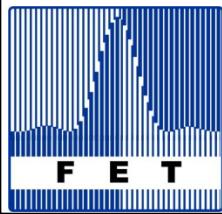
Sắp xếp lại(4)

- Nếu có cả 2 đặc điểm trên thì sự sắp xếp lại sẽ được thực hiện.
 - P_i sẽ được xếp trước P
- Cẩn thận khi lập trình giải thuật sắp xếp lại các cặp nút để tránh tạo ra vòng lặp vô hạn.



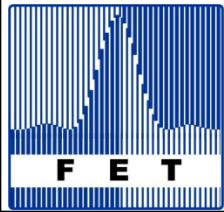
MENTOR đa tốc độ(1)

- MENTOR chỉ dùng liên kết một tốc độ
- Chuyển sang cho mạng liên kết nhiều tốc độ



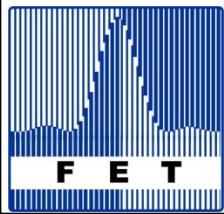
MENTOR đa tốc độ(2)

- **Ví dụ:** Thiết kế mạng với các liên kết 56 Kbps
- Cần thêm 4 đường song song giữa A và B (Dung lượng 224 Kbps)
Nếu giá của đường T1 giá
 - Gấp 3 lần đường 56 Kbps?
 - Gấp 4 đến 5 lần đường 56 Kbps?
 - Gấp 8 lần đường 56 Kbps?



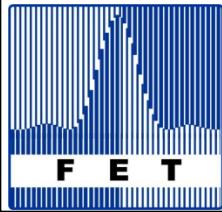
MENTOR đa tốc độ(3)

- Khi xem xét thay thế các liên kết tốc độ cao bằng một liên kết có tốc độ cao hơn, có 3 trường hợp:
 - Giá của nhiều liên kết tốc độ cao > giá của liên kết tốc độ cao hơn.
 - Giá của nhiều liên kết tốc độ cao = giá của liên kết tốc độ cao hơn.
 - Giá của nhiều liên kết tốc độ cao < giá của liên kết tốc độ cao hơn.



MENTOR đa tốc độ(3)

- Trong trường hợp 1, rõ ràng là lợi hơn khi sử dụng đường tốc độ cao hơn.
- Trong trường hợp 2, vẫn còn lợi- vẫn cải thiện chất lượng hoạt động và cho phép để phát triển



MENTOR đa tốc độ(4)

- Trong trường hợp 3, bây giờ phải trả thêm tiền nhưng chất lượng hoạt động được cải thiện và chi phí tiết kiệm được khi lưu lượng tăng lên (tương lai).
 - Tránh được phí bỏ đường cũ, phí thiết lập, tốc độ cao trong tương lai.
 - Quyết định hoàn toàn phụ thuộc tốc độ dự đoán

