

Bài tập Mạng máy tính

TS. Phạm Tuấn Minh
Khoa Công nghệ thông tin, Đại học Thủy lợi
minhpt@tlu.edu.vn
<http://netlab.tlu.edu.vn/~minhpt/>

1. Tổng quan, mô hình phân lớp

1.1. Bài 1

Bài tập này tính hai tham số quan trọng trong hiệu năng mạng máy tính là propagation delay và transmission delay. Cho hai nút mạng A và B kết nối bởi một liên kết có tốc độ là R bps. Giả sử hai nút mạng có khoảng cách là m mét và tốc độ lan truyền trên liên kết là s mét/giây. Nút mạng A gửi một gói tin kích thước L tới nút mạng B.

- Tính propagation delay, d_{prop} , theo m và s
- Tính thời gian truyền của gói tin, d_{trans} , theo L và R
- Bỏ qua processing delay và queuing delay, tính the end-to-end delay
- Giả sử nút mạng A bắt đầu truyền gói tin tại $t = 0$. Tại $t = d_{trans}$, bit cuối cùng của gói tin đang ở nút mạng A, trên đường truyền, hay đã tới nút mạng B?
- Giả sử d_{prop} lớn hơn d_{trans} . Tại $t = d_{trans}$, bit đầu tiên của gói tin đang ở nút mạng A, trên đường truyền, hay đã tới nút mạng B?
- Giả sử d_{prop} nhỏ hơn d_{trans} . Tại $t = d_{trans}$, bit đầu tiên của gói tin đang ở nút mạng A, trên đường truyền, hay đã tới nút mạng B?
- Giả sử $s = 2,5 \cdot 10^8$ m/s, $L = 120$ bit, $R = 56$ kbps. Tìm khoảng cách m để d_{prop} bằng d_{trans}

1.2. Bài 2

Cho gói tin có độ dài L gửi từ nút mạng A qua hai liên kết tới nút mạng đích B. Hai liên kết được nối với nhau bởi một switch.

Gọi d_i , s_i , và R_i là chiều dài, tốc độ lan truyền và tốc độ truyền của liên kết i với $i = 1, 2$. Packet switch processing delay của một gói tin là d_{proc} .

- Giả sử queuing delay bằng 0, tính end-to-end delay của gói tin theo d_i , s_i , R_i , ($i = 1, 2$), và L ?
- Giả sử kích thước gói tin là 1.000 byte, tốc độ lan truyền trên mỗi liên kết là $2,5 \cdot 10^8$ m/s, tốc độ truyền của hai liên kết là 1 Mbps, packet switch processing delay là 2 msec, chiều dài của liên kết thứ nhất là 6000 km, chiều dài của liên kết thứ hai là 3000 km. Tính end-to-end delay?

1.3. Bài 3

Tương tự bài 2, giả sử $R_1 = R_2 = R$ và $d_{proc} = 0$. Giả sử switch không thực hiện store-and-forward transmission mà truyền ngay lập tức mỗi khi nhận được một bit. Tính end-to-end delay?

1.4. Bài 4

Một packet switch nhận một gói tin và xác định liên kết ra cho gói tin. Khi gói tin này tới, một gói tin khác đã truyền được một nửa trên liên kết ra này và bốn gói tin khác đang đợi để được truyền đi. Các gói tin được truyền đi theo thứ tự đến.

Giả sử các gói tin có kích thước là 1.250 byte và tốc độ truyền của liên kết là 1 Mbps. Tính độ trễ hàng đợi (queuing delay) của gói tin đến?

Trường hợp tổng quát, khi gói tin có kích thước L , tốc độ truyền là R , x bit của gói tin đang truyền đã được truyền và n gói tin đang ở trong hàng đợi?

1.5. Bài 5

Giả sử N gói tin đến đồng thời một liên kết ở đó không có gói tin nào đang truyền hoặc đang chờ được truyền đi. Mỗi gói tin có chiều dài L và liên kết có tốc độ truyền R . Tính độ trễ hàng đợi trung bình của N gói tin.

1.6. Bài 6: Thực hành

Cài đặt phần mềm wireshark

2. Tầng ứng dụng: Web và HTTP, FTP, Email, DNS, lập trình socket

2.1. Bài 1

Cho chuỗi các kí tự mà phần mềm Wireshark đã ghi lại khi trình duyệt gửi bản tin HTTP GET. Kí tự `<cr><lf>` là kí tự về đầu dòng và xuống dòng

```
GET /cs453/index.html HTTP/1.1<cr><lf>Host: gaia.cs.umass.edu<cr><lf>User-Agent: Mozilla/5.0
(Windows;U; Windows NT 5.1; en-US; rv:1.7.2) Gecko/20040804 Netscape/7.2 (ax)
<cr><lf>Accept::ext/xml, application/xml, application/xhtml+xml, text/html;q=0.9,
text/plain;q=0.8,image/png,*/*;q=0.5<cr><lf>Accept-Language: en-us,en;q=0.5<cr><lf>Accept-Encoding:
zip,deflate<cr><lf>Accept-Charset: ISO-8859-1,utf-8;q=0.7,*;q=0.7<cr><lf>Keep-Alive:
300<cr><lf>Connection:keep-alive<cr><lf><cr><lf>
```

- Xác định URL của văn bản mà trình duyệt đã yêu cầu?
- Chỉ ra phiên bản của HTTP mà trình duyệt đang dùng?
- Trình duyệt yêu cầu kết nối kiểu non-persistent hay persistent?
- Xác định địa chỉ IP của nút mạng mà trình duyệt đang chạy ?

2.2. Bài 2

Đoạn văn bản dưới là phản hồi của server khi nhận được gói tin HTTP GET trong bài 1.

```
HTTP/1.1 200 OK<cr><lf>Date: Tue, 07 Mar 2008 12:39:45GMT<cr><lf>Server: Apache/2.0.52
(Fedora)<cr><lf>Last-Modified: Sat, 10 Dec2005 18:27:46 GMT<cr><lf>ETag: "526c3-f22-
a88a4c80"<cr><lf>Accept-Ranges: bytes<cr><lf>Content-Length: 3874<cr><lf>Keep-Alive:
timeout=max=100<cr><lf>Connection: Keep-Alive<cr><lf>Content-Type: text/html; charset=ISO-8859-
1<cr><lf><cr><lf><!doctype html public "-//w3c//dtd html 4.0 transitional//en"><lf><html><lf>
<head><lf> <meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1"><lf> <meta
name="GENERATOR" content="Mozilla/4.79 [en] (Windows NT 5.0; U) Netscape]"><lf> <title>CMPSCI
453 / 591 / NTU-ST550A Spring 2005 homepage</title><lf></head><lf> <bỏ qua các phần sau của gói tin
>
```

- Server tìm thấy văn bản hay không?
- Xác định thời gian mà văn bản trả lời được cung cấp?
- Xác định thời gian cuối cùng mà văn bản được sửa đổi?
- Xác định số byte trong văn bản trả lời?
- Xác định 5 byte đầu tiên trong văn bản trả lời?
- Có phải server chấp nhận kiểu kết nối persistent?

2.3. Bài 3

Truyền một tập tin $F = 10$ Gbit tới N nút. Server có tốc độ đường lên (upload rate) $u_s = 20$ Mbps, mỗi nút có tốc độ đường xuống (download rate) $d_i = 1$ Mbps và tốc độ đường lên u . Với $N = 10, 100, 1000$, $u = 200$ Kbps, 600 Kbps, 1 Mbps. Tính thời gian truyền cho mỗi cặp N và u trong trường hợp client-server và P2P. Viết kết quả dưới dạng bảng và chỉ ra trường hợp có thời gian truyền nhỏ nhất.

2.4. Bài 4

Truyền một tập tin F bit tới N nút trong kiến trúc client-server. Giả sử server có thể đồng thời truyền tới nhiều nút với tốc độ khác nhau, trong đó tổng tốc độ truyền không vượt quá u_s .

- a) Giả sử $u_s/N \leq d_{\min}$. Xác định cách truyền có thời gian truyền NF/u_s .
- b) Giả sử $u_s/N \geq d_{\min}$. Xác định cách truyền có thời gian truyền F/d_{\min} .

2.5. Bài 5

Truyền một tập tin F bit tới N nút trong kiến trúc P2P. Giả sử d_{\min} rất lớn sao cho băng thông tải xuống của các nút không bao giờ bị nghẽn.

Giả sử $u_s \leq (u_s + u_1 + \dots + u_N)/N$. Xác định cách truyền có thời gian truyền là F/u_s .

2.6. Bài 6: Thực hành

Thực hành giao thức HTTP: GET/reply, cấu trúc HTTP message, persistent connection và non-persistent connection. Thực hiện các bước sau:

Bước 1: Mở trình duyệt web

Bước 2: Mở Wireshark

Bước 3: Gõ "http" (chỉ có kí tự http, không có dấu nháy kép), trong filter

Bước 4: Đợi 1 phút, chạy Wireshark packet capture

Bước 5: Gõ vào trình duyệt <http://netlab.tlu.edu.vn/comnet/HTTP-wireshark-file1.html>

Bước 6: Dừng Wireshark packet capture

Quan sát kết quả các gói tin do Wireshark ghi lại và trả lời các câu hỏi sau:

- a) Trình duyệt xử lý phiên bản HTTP 1.0 hay 1.1?
- b) Cho biết phiên bản HTTP đang chạy trên server?
- c) Cho biết ngôn ngữ mà trình duyệt có thể chấp nhận (nếu có) ?
- d) Cho biết địa chỉ IP của máy tính chạy trình duyệt (nếu có) ?
- e) Cho biết địa chỉ IP của server netlab.tlu.edu.vn (nếu có) ?
- f) Cho biết mã trạng thái do server gửi cho trình duyệt ?
- g) Cho biết thời gian cập nhật gần nhất của tập tin HTML do server gửi cho trình duyệt ?
- h) Cho biết số byte phần nội dung do server gửi cho trình duyệt ?

3. Tầng giao vận: UDP, TCP

3.1. Bài 1

Client A khởi tạo một phiên Telnet tới Server S. Tại cùng một thời điểm, Client B cũng khởi tạo một phiên Telnet tới Server S. Chỉ ra các giá trị có thể của cổng nguồn và cổng đích của

- a) Segment gửi từ A tới S
- b) Segment gửi từ B tới S
- c) Segment gửi từ S tới A
- d) Segments gửi từ S tới B
- e) Nếu A và B là các host khác nhau, giá trị cổng nguồn trong segment gửi từ A tới S và giá trị cổng nguồn gửi từ B tới S có thể giống nhau được không?
- f) Tương tự câu hỏi e) trong trường hợp A và B là cùng một host

3.2. Bài 2

UDP và TCP sử dụng bù 1 để tính checksum. Giả sử có các ba byte sau: 01010101, 01110000, 01001100

- a) Tính giá trị bù 1 của tổng ba byte trên. Chú ý là UDP và TCP sử dụng các word 16-bit để tính checksum, nhưng bài tập này yêu cầu tính cho các byte 8-bit.
- b) Khi dùng bù 1, phía nhận phát hiện lỗi như thế nào? Tại sao UDP dùng bù 1 để tính tổng, tại sao không chỉ tính tổng?
- c) Lỗi 1 bit có thể không bị phát hiện không?
- d) Lỗi 2 bit có thể không bị phát hiện không?

3.3. Bài 3

Giả sử rằng với UDP, phía nhận tính Internet checksum của segment gửi tới và thấy rằng kết quả trùng với giá trị trong trường checksum. Phía nhận có chắc chắn rằng không có bit nào bị lỗi không? Giải thích.

3.4. Bài 4

Chỉ ra rằng với giao thức rdt2.1, bên gửi hoạt động theo sơ đồ trạng thái dưới có thể dẫn tới bên gửi và bên nhận rơi vào trạng thái deadlock (tại đó mỗi bên đợi một sự kiện mà sự kiện này không bao giờ xảy ra).



3.5. Bài 5: Thực hành

Mở tập tin tcp-ethereal-trace-1 trong Wireshark. Quan sát và trả lời các câu hỏi sau:

- Cho biết sequence number của TCP segment chứa HTTP POST command?
- Giả sử TCP segment chứa HTTP POST là segment đầu tiên trong kết nối TCP. Xác định các sequence number của 6 segment đầu tiên trong kết nối TCP (bao gồm cả segment chứa HTTP POST)?
- Xác định thời gian mỗi TCP segment được gửi ?
- Khi nào nhận được ACK cho mỗi segment ?
- Xác định giá trị RTT cho mỗi segment ?
- Xác định chiều dài của mỗi segment ?
- Có segment nào được gửi lại không?
- Xác định sequence number của TCP SYN segment mà được dùng để khởi tạo kết nối TCP giữa client và server ? Dựa vào thông tin nào để xác định segment là SYN segment ?
- Xác định sequence number của SYNACK segment khi server trả lời SYN segment của client ? Giá trị của trường Acknowledgement trong SYNACK segment ? Dựa vào thông tin nào để xác định segment là SYNACK segment ?

3.6. Bài 6: Thực hành

Thực hành UDP theo các bước sau:

Bước 1: Chạy Wireshark và chạy Wireshark packet capture

Bước 2: Chạy phần mềm nào đó để gửi và nhận một số UDP packet (ví dụ: chạy Skype)

Bước 3: Dừng Wireshark packet capture

Bước 4: Gõ udp trong filter để chỉ hiện các UDP packet gửi và nhận tại máy thực hành

Chọn một UDP packet và trả lời câu hỏi sau:

- a) Xác định số trường trong UDP header, tên của mỗi trường
- b) Tính chiều dài của mỗi trường của UDP header
- c) Ý nghĩa của trường Length ? Kiểm chứng lại với UDP packet đang quan sát.
- d) Số byte lớn nhất có thể chứa trong UDP payload ?
- e) Giá trị lớn nhất của cổng nguồn ?
- f) Số giao thức xác định UDP ? (viết trả lời dạng hệ 16 và hệ 10)

Quan sát một cặp UDP packet trong đó máy thực hành gửi UDP packet đầu tiên và UDP packet thứ hai là trả lời cho UDP packet thứ nhất. Nhận xét quan hệ giá trị số cổng trong hai gói tin.

3.7. Bài 7: Thực hành

Thực hành TCP: Phân tích dữ liệu ghi lại các TCP sequence gửi và nhận khi truyền một tập tin 150KB (chứa nội dung của câu chuyện Alice's Adventures in Wonderland) từ một máy tính tới server.

- Quan sát TCP sử dụng sequence number và ack number để truyền dữ liệu tin cậy
- Quan sát thuật toán điều khiển tắc nghẽn (slow start và congestion avoidance)
- Quan sát cơ chế điều khiển luồng của bên nhận
- Quan sát thiết lập kết nối
- Quan sát hiệu năng (thông lượng và RTT) của kết nối TCP

4. Tầng mạng: Bảng định tuyến, cấu trúc IPv4

4.1. Bài 1

Giải thích sự khác biệt giữa định tuyến (routing) và chuyển tiếp (forwarding).

Giả sử Host A gửi tới Host B một TCP segment được đóng trong một IP datagram. Khi Host B nhận datagram, làm thế nào để tầng mạng tại Host B biết cần chuyển segment (trong payload của datagram) tới TCP chứ không phải là tới UDP ?

4.2. Bài 2

Giả sử một ứng dụng cứ mỗi 20 msec sinh ra chunk có độ dài 40 byte. Mỗi chunk được đóng gói trong một TCP segment, rồi trong một IP datagram. Xác định phần trăm dữ liệu ứng dụng trong IP datagram?

4.3. Bài 3

Gửi datagram kích thước 3000 byte qua liên kết có MTU là 500 byte. Giả sử datagram có identification number là 422. Xác định số fragment được sinh ra? Xác định các giá trị trong các trường của IP datagram dùng trong phân mảnh ?

4.4. Bài 4

Giả sử datagram giới hạn 1500 byte (gồm cả header) giữa nguồn Host A và đích Host B. Giả sử IP header 20 byte, xác định số datagram cần để gửi một tập tin MP3 có kích thước 4 triệu byte.

5. Tầng mạng: Đánh địa chỉ IPv4

5.1. Bài 1

Có phải router có địa chỉ IP ? Nếu có thì có bao nhiêu địa chỉ IP ?

Xác định giá trị 32 bit nhị phân tương ứng của địa chỉ IP 223.1.3.27 ?

Quan sát IP address, network mask, default router, và IP address của local DNS server trên máy ?

5.2. Bài 2

Một mạng chuyển mạch gói dùng 32 bit địa chỉ. Giả sử router có 4 link, đánh số từ 0 đến 3 và gói tin được chuyển tới link interface như sau:

Destination Address Range	Link Interface
11100000 00000000 00000000 00000000 through 11100000 00111111 11111111 11111111	0
11100000 01000000 00000000 00000000 through 11100000 01000000 11111111 11111111	1
11100000 01000001 00000000 00000000 through 11100001 01111111 11111111 11111111	2
otherwise	3

Xác định forwarding table có 5 dòng, dùng longest prefix matching, và chuyển gói tin tới đúng link interface.

Mô tả cách forwarding table xác định link interface cho các datagram có địa chỉ đích như sau:

```
11001000 10010001 01010001 01010101
11100001 01000000 11000011 00111100
11100001 10000000 00010001 01110111
```

5.3. Bài 3

Viết lại forwarding table trong bài 2 dưới dạng a.b.c.d/x thay vì viết dạng chuỗi nhị phân.

5.4. Bài 4

Xem xét một mạng chuyển mạch gói dùng 4 bit địa chỉ. Giả sử router sử dụng longest prefix matching và có forwarding table như sau:

Prefix match	Interface
--------------	-----------

00	0
01	1
10	2
11	3

Với mỗi interface, xác định dải địa chỉ đích và số địa chỉ trong dải.

5.5. Bài 5

Xem xét một router nối 3 subnet: subnet 1, subnet 2 và subnet 3. Giả sử tất cả các interface trong các subnet có prefix 223.1.17.0/24, Subnet 1 cần hỗ trợ ít nhất 60 interfaces, subnet 2 cần hỗ trợ ít nhất 90 interface, subnet 3 cần hỗ trợ ít nhất 12 interface. Xác định ba địa chỉ mạng (dưới dạng a.b.c.d/x) thỏa mãn các yêu cầu trên.

5.6. Bài 6

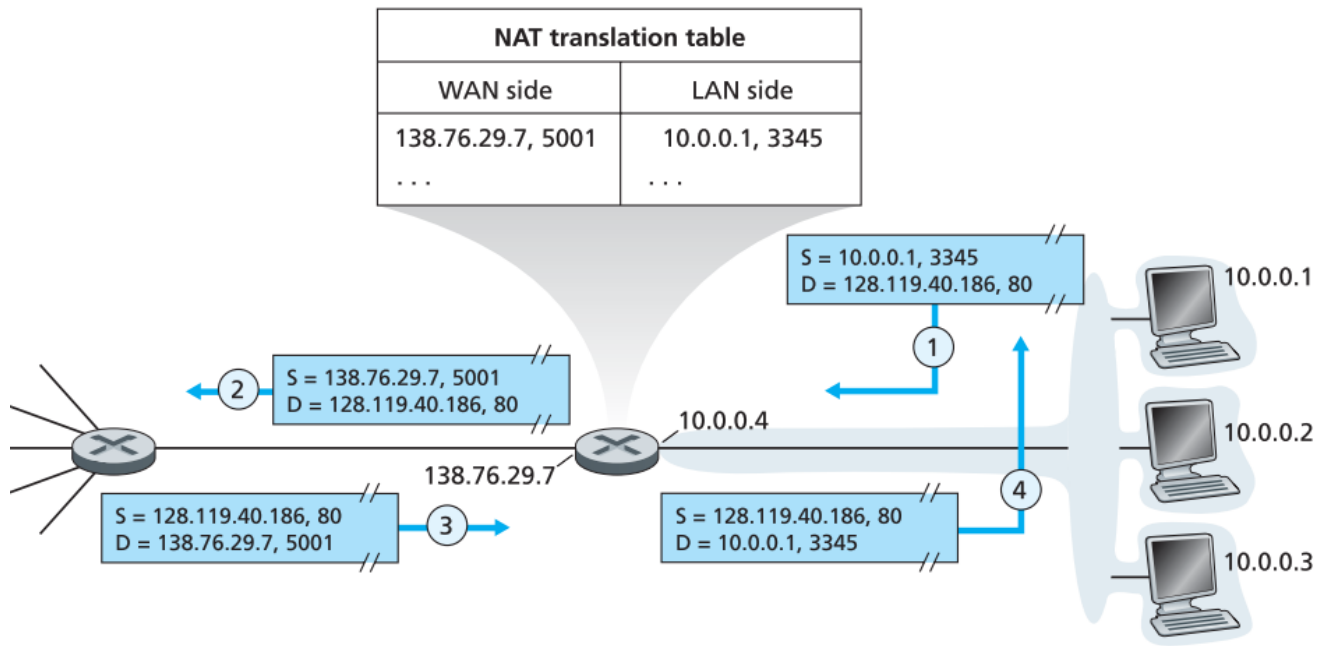
Cho một subnet có prefix 128.119.40.128/26. Cho ví dụ một địa chỉ IP (dạng xxx.xxx.xxx.xxx) có thể gán cho mạng này.

Giả sử ISP quản lý địa chỉ 128.119.40.64/26, cần tạo 4 subnet, trong đó mỗi subnet có cùng số địa chỉ IP. Xác định prefix (dạng a.b.c.d/x) cho 4 subnet?

5.7. Bài 7

Cho sơ đồ mạng như hình vẽ. Giả sử ISP gán địa chỉ router là 23.34.112.235 và địa chỉ mạng nội bộ là 192.168.1/24 (thay cho giá trị ghi trong hình vẽ).

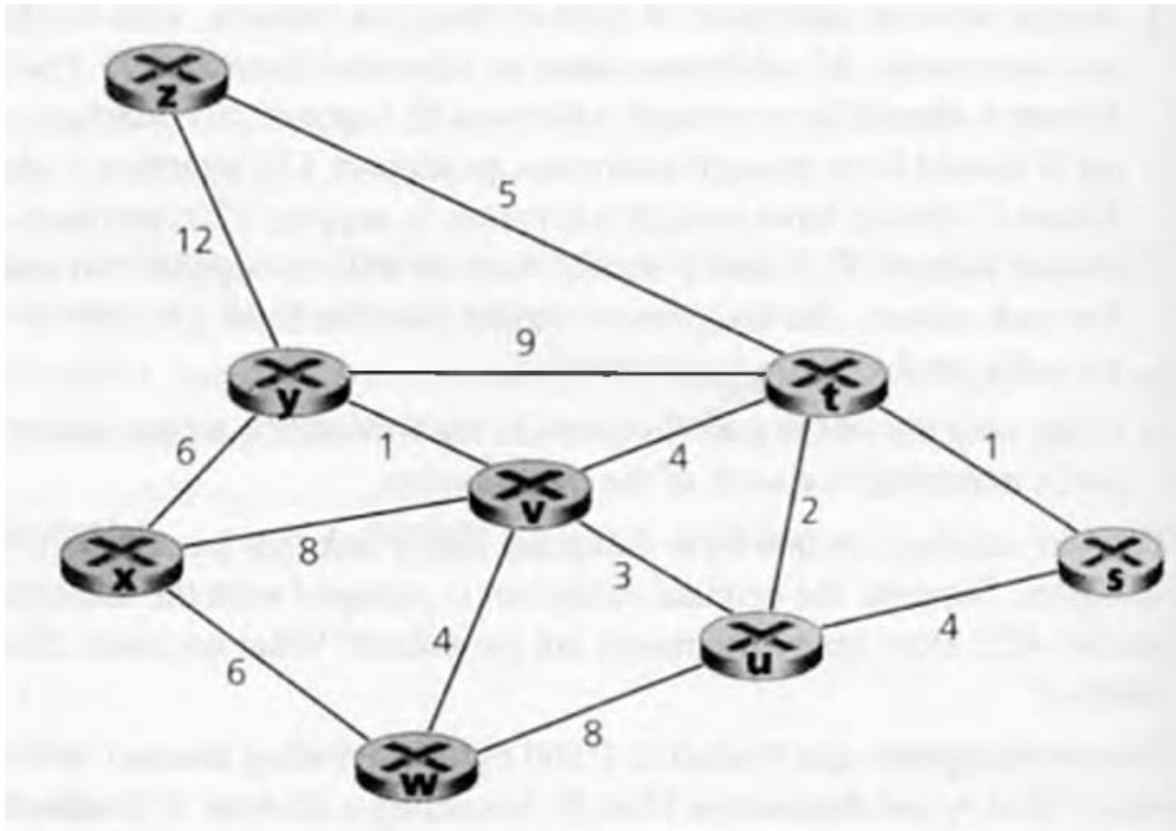
- Xác định địa chỉ cho các interface của mạng nội bộ
- Giả sử mỗi host có 2 kết nối TCP ra ngoài, tất cả tới cổng 80 của host 128.119.40.86. Xác định 6 dòng trong NAT translation table.
- Giải thích hoạt động của NAT dùng ví dụ trên



6. Tầng mạng: Giải thuật định tuyến, Giao thức định tuyến

6.1. Bài 1

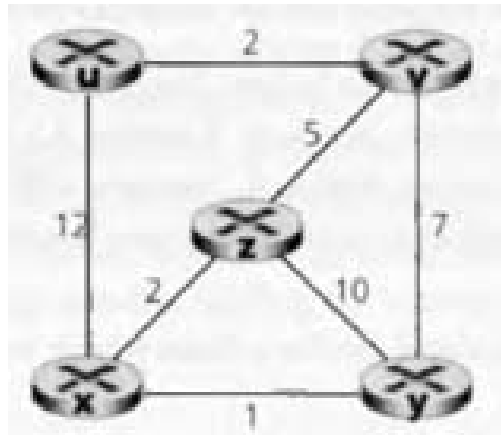
Dùng giải thuật Dijkstra trong định tuyến link state để xác định đường đi ngắn nhất từ s tới mọi nút. Vẽ bảng kết quả như ví dụ.



step	N'	$D(v), p(v)$	$D(w), p(w)$	$D(x), p(x)$	$D(y), p(y)$	$D(z), p(z)$
0	u	2,u	5,u	1,u	∞	∞
1	ux	2,u	4,x		2,x	∞
2	uxy	2,u	3,y			4,y
3	uxyv		3,y			4,y
4	uxyvw					4,y
5	uxyvwz					

6.2. Bài 2

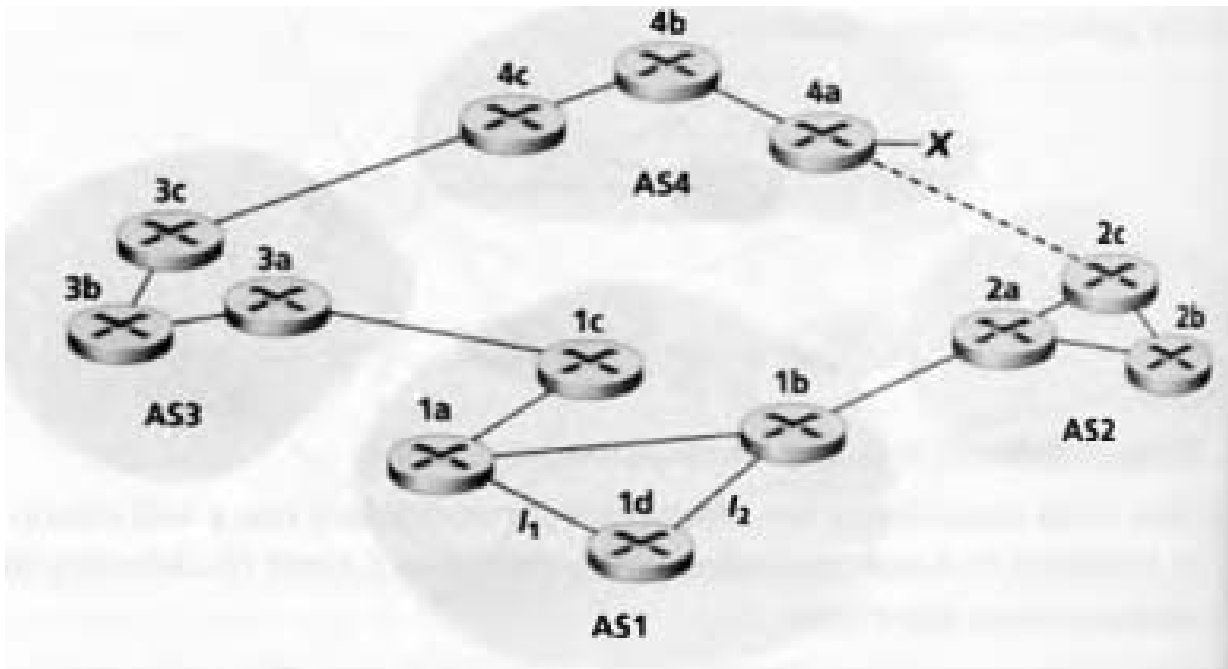
Giả sử mỗi nút ban đầu biết cost tới các nút láng giềng. Dùng giải thuật distance-vector tính bảng khoảng cách tại nút z.



6.3. Bài 3

Giả sử AS3 và AS2 đang sử dụng giao thức định tuyến intra-AS OSPF. AS1 và AS4 đang sử dụng giao thức định tuyến intra-AS RIP. Giả sử eBGP và iBGP dùng làm giao thức định tuyến inter-AS. Ban đầu không có liên kết vật lý giữa AS2 và AS4.

- Router 3c học prefix x từ giao thức định tuyến nào: OSPF, RIP, eBGP, iBGP
- Router 3a học x từ giao thức định tuyến nào
- Router 1c học x từ giao thức định tuyến nào
- Router 1d học x từ giao thức định tuyến nào



7. Tầng liên kết dữ liệu: Phát hiện và sửa lỗi, điều khiển truy cập

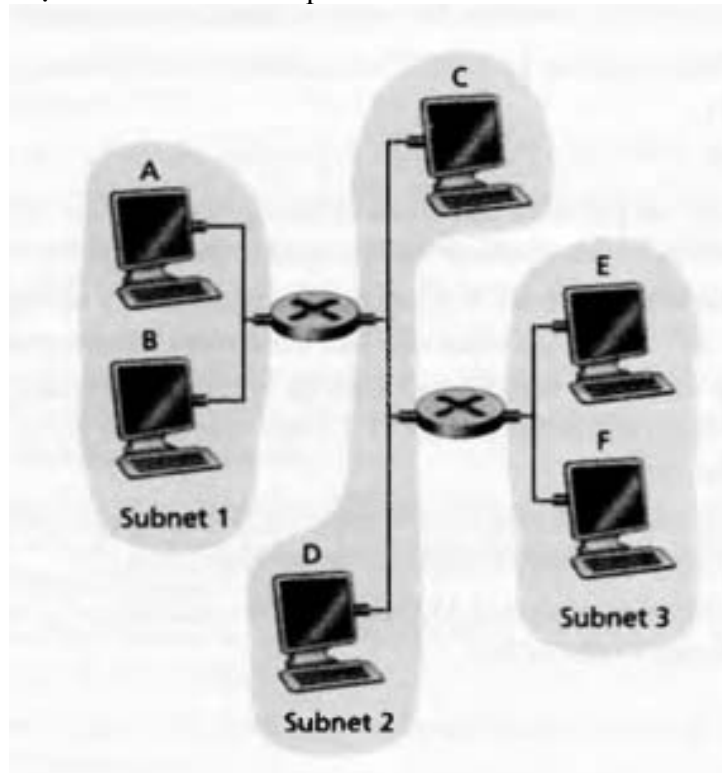
7.1. Bài 1

Giả sử trong mạng quảng bá LAN có các nút A, B và C. Nếu A gửi một nghìn IP datagram tới B với địa chỉ MAC đích là địa chỉ MAC của B.

- C có xử lý các frame này không ?
- Nếu có, C có chuyển các frame này tới tầng mạng của C không ?
- Nếu A gửi với địa chỉ MAC là địa chỉ broadcast MAC thì câu trả lời thế nào?

7.2. Bài 2

Cho 3 mạng LAN được kết nối với nhau qua 2 router như hình vẽ.



- Gán địa chỉ IP cho các interface, trong đó subnet 1 sử dụng địa chỉ 111.111.111.xxx, subnet 2 sử dụng địa chỉ 122.122.122.xxx, subnet 3 sử dụng địa chỉ 133.133.133.xxx
- Gán địa chỉ MAC cho các adapter
- Mô tả quá trình gửi một IP datagram từ Host A tới Host F, giả sử các ARP table đã được cập nhật.
- Mô tả quá trình gửi một IP datagram từ Host A tới Host, giả sử ARP table tại host A là rỗng, các ARP table khác đã được cập nhật.