CS305 · Computer Network

Homework 1

Liu Leqi / 12011327

October 24, 2022

1 Q1

1.1 a

Since there is no nodal processing delay and queuing delay, we have the end-to-end delay as

$$\begin{aligned} d_{ete} &= K(d_{trans} + d_{prop}) \\ &= K(\frac{L}{R} + d) \end{aligned}$$

1.2 b

For using circuit switching network, the end-to-end delay is

$$d_{ete} = \tau + K(\frac{ML}{R} + d)$$

1.3 c

For a packet, we have

$$\begin{aligned} d_{proc} &= 25\mu s \\ d_{trans} &= \frac{L}{R} = 100\mu s \\ d_{prop} &= 40\mu s \end{aligned}$$

The time required to send both packets from source to destination is the total end-to-end delay. And the latter packet needs to queue.
Former packet arrives at the first router (no nodal process delay in source):

$$\begin{aligned} t_1 &= d_{trans} + d_{prop} \\ &= 100\mu s + 40\mu s = 140\mu s \end{aligned}$$

The latter packet still needs $100\mu s$ to arrive at the first router.
Former packet arrives at the destination:

$$\begin{aligned} t_2 &= d_{proc} + d_{trans} + d_{prop} \\ &= 25\mu s + 100\mu s + 40\mu s = 165\mu s \end{aligned}$$

During this time, the latter packet has arrived at the first router and left to destination. The latter packet still needs $100\mu s$ to arrive at the destination.
Therefore, the total time required to send both packets from source to destination is

$$t = t_1 + t_2 + 100\mu s = 405\mu s$$

2 Q2

2.1 a

Since the time slot is $\frac{L}{R}$ seconds, every packet does not need to queue if one packet arrival at the beginning of each time slot. Therefore, the average queuing delay is 0.

2.2 b

The transmission delay of N packets is NL/R. Since k is a nonnegative integer, at the beginning of each of the time slots $t = kNL/R$, the first packets will be no queue delay.
However, the packet needs to queue from the second packet. So the average queuing delay of these packets is

$$\overline{d_{queue}} = \frac{\sum_{n=1}^{N-1} nL/R}{N} = \frac{(N-1)L}{2R}$$

So when N approaches infinity, the average queuing delay will also approach infinity.

2.3 c

The traffic intensity of (a) and (b) respectively is

$$trafficIntensity_a = \frac{L/(L/R)}{R} = 1$$
$$trafficIntensity_b = \frac{NL/(NL/R)}{R} = 1$$

Their traffic intensity is the same, but their queuing delays are not the same. It means that traffic intensity may not completely indicate the queuing delay.

3 Q3

The 5-layer Internet protocol stack is

Application layer
Transport layer
Network layer
Link layer
physical layer

Table 1: The 5-layer Internet protocol stack

- Application layer supports network applications. Typical protocols are HTTP, IMAP, SMTP, DNS.
- Transport layer will do process-process date transfer. Typical protocols are TCP, UDP.
- Network layer routes datagrams from source to destination. Typical protocols are IP, ICMP.
- Link layer transfers date between neighbouring network elements. Typical protocols are PPP, Ethernet.
- Physical layer provides data path and transfers data. Typical protocols is DSL.

4 Q4

4.1 a

HTTP message runs on top of TCP. Because TCP provides reliable transport, flow control and congestion control, while UDP does not provide these so that the message sent by UDP might lose in data transfer and be out of order.

4.2 b

This is an HTTP request message.

4.3 c

This message corresponds to a persistent connection since the last line is *Connection:keep-alive*.

4.4 d

The header line *cookie:1150* should be included in the message.

4.5 e

The status line would be *HTTP/1.1 200 OK*$\backslash r\backslash n$. The content of the HTML file /cs453/index.html would be included in the entity body.

5 Q5

5.1 a

Since

$$\begin{aligned}
\Delta &= 650000bit/15Mbps = 0.043sec \\
\beta &= 20request/sec \\
AverageAccessDelay &= \frac{\Delta}{1-\Delta\beta} = \frac{0.043}{1-0.043\times 20} = 0.307sec
\end{aligned}$$

we have

$$\begin{aligned}
TotalAverageResponseTime &= AverageAccessDelay + AverageInternetDelay \\
&= 0.307 + 3 = 3.307sec
\end{aligned}$$

5.2 b

Since some requests are satisfied in cache, the request to Internet reduced. Then we have

$$\begin{aligned}
\Delta' &= 650000bit/15Mbps = 0.043sec \\
\beta' &= (1-0.4)\times 20request/sec = 12request/sec \\
AverageAccessDelay &= \frac{\Delta'}{1-\Delta'\beta'} = \frac{0.043}{1-0.043\times 12} = 0.089sec
\end{aligned}$$

If cache misses, we have

$$\begin{aligned}
TotalAverageResponseTime &= AverageAccessDelay + AverageInternetDelay \\
&= 0.089 + 3 = 3.089sec
\end{aligned}$$

If cache hits, we have the total average response time to be 0. Therefore, the total average response time is

$$3.089\times 0.6 + 0\times 0.4 = 1.85sec$$

6 Q6

6.1 a

$$\sum_{i=1}^{n} RTT_i + 2RTT_0 + 10\times 2RTT_0 = 22RTT_0 + \sum_{i=1}^{n} RTT_i$$

6.2 b

$$\sum_{i=1}^{n} RTT_i + 4 \times 2RTT_0 = 8RTT_0 + \sum_{i=1}^{n} RTT_i$$

6.3 c

$$\sum_{i=1}^{n} RTT_i + RTT_0 + RTT_0 + 10RTT_0 = 12RTT_0 + \sum_{i=1}^{n} RTT_i$$

7 Q7

7.1 a

In HTTP, the client pulls data from server and each object encapsulated in its own response message.
In SMTP, the client pushes data to server and multiple objects sent in multipart message.

7.2 b

No.
SMTP is the protocol for delivery and storage of e-mail message to receiver's server, it does not provide the actions for mail access.

7.3 c

Neither placing the receiver's mail server at the receiver's PC nor placing the sender's mail server at the sender's PC is a good idea.
Mail server runs both the client side and the server side of SMTP. If we place the receiver's mail server at the receiver's PC, the receiver's PC needs to remain on in order to receive new mail. If we place the sender's mail server at the sender's PC, the sender's PC needs to remain on in order to send mail repeatedly in case that the receiver's PC fails.

7.4 d

DNS runs on top of UDP.
UDP provides smaller data packets and as a consequence, it has fast speed. And DNS does not need consistent data to work and has a huge load on traffic.

8 Q8

8.1 a

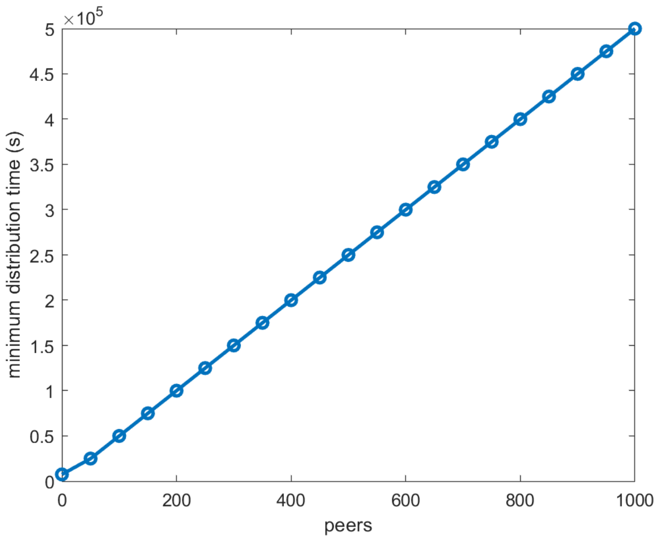

Figure 1: Client-server distribution

8.2 b

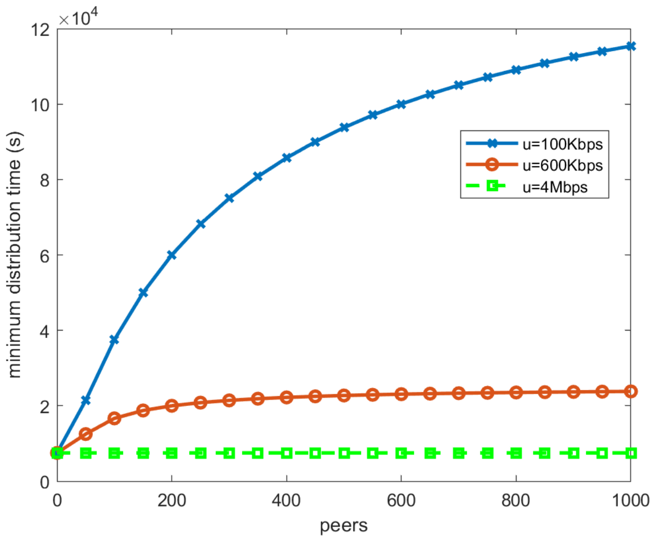

Figure 2: P2P distribution

The MATLAB script for picture

```
N = 0:50:1000;
F = 15*1000^3;
us = 30*1000^2;
d = 2*1000^2;
u = [100*1000 600*1000 4*1000^2];

t1 = N*F/us;
t2 = F/d;
cs_dis_time = t1.*(t1>=t2)+t2.*(t1<t2);
figure
plot(N, cs_dis_time, '-o', 'LineWidth', 2)
ylabel('minimum distribution time (s)')
xlabel('peers')

t3 = F/us;
t4 = F/d;
t51 = N*F./(us+N*u(1));
t52 = N*F./(us+N*u(2));
t53 = N*F./(us+N*u(3));

p2p_dis_time1 = t3.*(t3>=t4 & t3>=t51)+t4.*(t4>t3 & t4>t51)+t51.*(t51>t3 & t51>t4);
p2p_dis_time2 = t3.*(t3>=t4 & t3>=t52)+t4.*(t4>t3 & t4>t52)+t52.*(t52>t3 & t52>t4);
p2p_dis_time3 = t3.*(t3>=t4 & t3>=t53)+t4.*(t4>t3 & t4>t53)+t53.*(t53>t3 & t53>t4);
figure;
plot(N, p2p_dis_time1, '-x', 'LineWidth', 2);
hold on
plot(N, p2p_dis_time2, '-o', 'LineWidth', 2);
hold on
plot(N, p2p_dis_time3, '--gs', 'LineWidth', 2);
hold off
ylabel('minimum distribution time (s)')
xlabel('peers')
legend('u=100Kbps', 'u=600Kbps', 'u=4Mbps', 'Location', 'best');
```

9 Q9

9.1 a

Suppose the parallel transmitting rate is u_s/N to N peers, which satisfies that $N \cdot u_s/N \leq u_s$. Since $u_s/N \leq d_{min}$, the download rate is no less than u_s/N. Therefore, the distribution time is $F/(u_s/N) = NF/u_s$. Since the N distributions are parallel, the total distribution time is also NF/u_s.

9.2 b

Suppose the parallel transmitting rate is d_{min} to N peers, which satisfies that $N \cdot d_{min} \leq u_s$ as $u_s/N \geq d_{min}$. Assume all the peers download the file with rate d_{min}, the distribution time is F/d_{min}. Since the N distributions are parallel, the total distribution time is also F/d_{min}.

9.3 c

From (a) and (b) we can know that $D_{c-s} \geq \max\{NF/u_s, F/d_{min}\}$ since $D_{c-s} \geq NF/u_s$ in $u_s/N \leq d_{min}$ and $D_{c-s} \geq F/d_{min}$ in $u_s/N \geq d_{min}$.