CN | HW2

Setu Gupta (2018190)

1a.

Packet name	Start time	End time
ap1	0	30
aq1	30	130
aq2	130	140
ap2	140	180

1b.

Time	Round	Bits of P transferred	Bits of Q transferred	Event
0	0	0	0	ap1, aq1, aq2 arrive
60	30	30	30	ap1 finished
100	70	30	70	ap2 arrives
160	100	60	100	aq1 finished
180	110	70	110	ap2, aq2 finished

Packet name	Start time	End time
ap1	0	60
aq1	0	160
aq2	160	180
ap2	100	180

1c.

$S_{p1} = 0$
$F_{p1} = S_{p1} + P_{p1} = 0 + 30 = 30$

$S_{q1} = 0$
$F_{q1} = S_{q1} + P_{q1} = 0 + 100 = 100$

$S_{q2} = \max\{F_{q1}, A_{q2}\} = \max\{100, 0\} = 100$
$F_{q2} = S_{q2} + P_{q2} = 100 + 10 = 110$

$S_{p2} = \max\{F_{p1}, A_{p2}\} = \max\{30, 100\} = 100$
$F_{p2} = S_{p2} + P_{p2} = 100 + 40 = 140$

Packet name	Start time	End time
ap1	0	30
aq1	30	130
aq2	130	140
ap2	140	180

1d. (Assuming order in terms of finish time)
ap1, aq1, aq2 order remains the same as long as $F_{p2} > F_{q2}$.

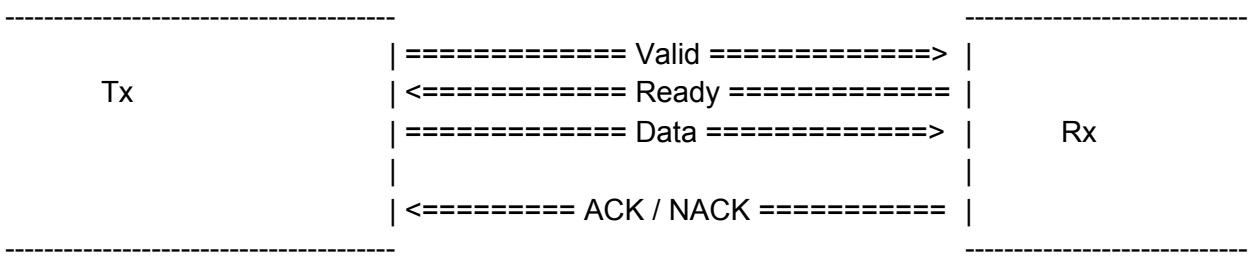
$F_{p2} > F_{q2}$
=> $F_{p2} > 110$
=> $S_{p2} + P_{p2} > 110$
=> $S_{p2} + 40 > 110$
=> $S_{p2} > 70$
=> $\max\{F_{p1}, A_{p2}\} > 70$
=> $\max\{30, A_{p2}\} > 70$
=> $A_{p2} > 70$

Moreover, ap2 must arrive at least as soon as aq2 is finished i.e. $A_{p2} <= 140$
Therefore $70 < A_{p2} <= 140$

2. L2 layer retransmission mechanism can be used if the erroneous packets transferred over a link can be retransmitted within using L2 layer protocols only. One such solution is enforcing handshaking and ACK/NACK between two L2 nodes.
For this we can use two signals: Ready and Valid. Ready is set/reset by the receiver and Valid is set/reset by the sender.

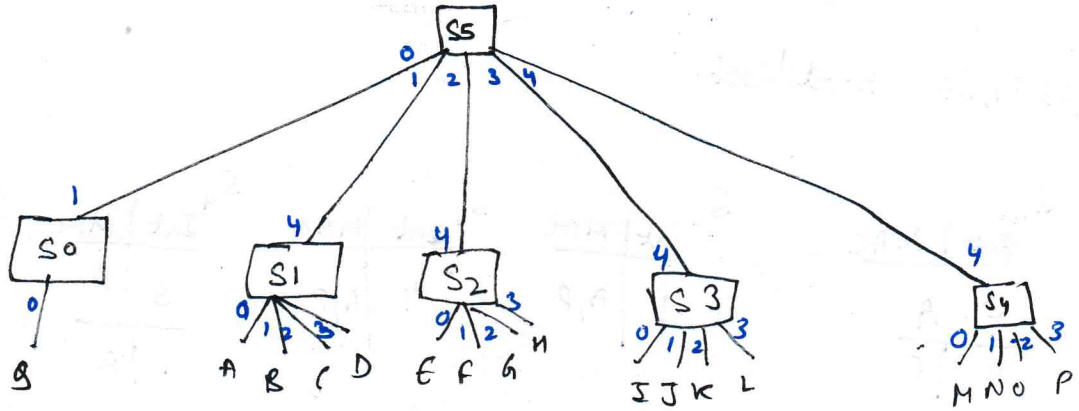
Ready	Valid	Description	Transfer data?
0	0	Neither Tx nor Rx can allow transmission.	No
0	1	Tx is ready to send the data but Rx is busy.	No
1	0	Rx is ready to send the data but Tx is busy.	No
1	1	Rx and Tx both are ready.	Yes

After successful transmission (Ready && Valid), the receiver must send back ACK/NACK based on whether the received packet passes the CRC check or not. If the Tx receives an ACK, the session is terminated, else Tx will again send the packet. If Tx receives a false NACK (i.e Rx wanted to send ACK but due to errors, the bit was flipped to NACK), Rx will simply ignore the incoming retransmitted packet and again send an ACK.
Here is a diagram of the proposed system. The heads of the arrows show who reads the signal and tails show who generates the signal.



4a. eBGP
4b. iBGP
4c. eBGP
4d. I1 as 3b got to know about x from 1c over a eBGP packet. As of now, it doesn't know any other path to x except the one via I1. Hence the entry is (x, I1)
4e. I2. After the new link is constructed, 3b will receive paths to x via 3a and 3c via iBGP. The cost from I2 is 1+3 = 4 and the cost from I3 is 5. Hence 3b choses I2. Then entry will be (x, I2). Moreover, this is a better choice as compared to I1 as it only needs two inter AS hops with I2 as compared to 3 with I1

Q-3)



Only showing filled entries

a) A → L

S1 broadcasts & stores MAC of A at interface 0

S5 broadcasts " " " " " " " 1

S0 " " " " " " " 1

S2 " " " " " " " 4

S3 " " " " " " " 4

S4 " " " " " " " 4

S_0:

Interface	MAC
1	A

S_1

Interface	MAC
0	A

S_2

Interface	MAC
4	A

S_3

Interface	MAC
4	A

S_4

Interface	MAC
4	A

S_5

Interface	MAC
1	A

b.) P → Q

S0, S1, S2, S3, S4, S5 broadcast.

S0

Int	MAC
1	A,P

S1

Int	MAC
0	A
4	P

S2

Int	MAC
4	A,P

S3

Int	MAC
4	A,P

S4

Int	MAC
3	P
4	A

S5

Int	MAC
1	A
4	P

c.) Q → P

S0 sends via port 1

S5 sends via port 4

S4 sends via port 3

S0

Int	MAC
0	Q
1	A,P

S1

Int	MAC
0	A
4	P

S2

Int	MAC
4	A,P

S3

Int	MAC
4	A,P

S4

Int	MAC
3	P
4	A,Q

S5

Int	MAC
0	Q
1	A
4	P