Lab 2

Objectives:

To explore how Stop and Wait protocol works (via CNET). Investigate the differences in which different type of TOPOLOGIES effect the throughput of a network. Learn how to implement/alter Stop and Wait protocol to handle multiple nodes (in this case 2 Hosts and 3 Routers)

Part 1:

1. CNETa

2 hosts, 0 routers, 0 mobiles, 0 accesspoints

Simulation time	: 300000000
Events raised	: 141
Messages generated	: 47
Messages delivered	: 46
Message bandwidth	: 5736
Average delivery time	: 2789142
Frames transmitted	: 93
Frames received	: 92
Efficiency (bytes AL) / (bytes PL)	: 95.60
EV_REBOOT	: 2
EV_PHYSICALREADY	: 92
EV_APPLICATIONREADY	: 47

1. CNETb

2 hosts, 0 routers, 0 mobiles, 0 accesspoints

Simulation time	: 300000000
Events raised	: 108
Messages generated	: 36
Messages delivered	: 35
Message bandwidth	: 27547
Average delivery time	: 4927264
Frames transmitted	: 71
Frames received	: 70
Efficiency (bytes AL) / (bytes PL)	: 96.53
EV_REBOOT	: 2
EV_PHYSICALREADY	: 70
EV_APPLICATIONREADY	: 36

1.CNETc

2 hosts, 0 routers, 0 mobiles, 0 accesspoints

Simulation time	: 300000000

Events raised	: 112
Messages generated	: 26
Messages delivered	: 25
Message bandwidth	: 1887
Average delivery time	: 8475454
Frames transmitted	: 68
Frames received	: 67
Frames corrupted	: 17
Efficiency (bytes AL) / (bytes PL)	: 57.06
EV_REBOOT	: 2
EV_PHYSICALREADY	: 67
EV_APPLICATIONREADY	: 26
EV_TIMER1	: 17

1.CNETd

2 hosts, 0 routers, 0 mobiles, 0 accesspoints

Simulation time	: 300000000
Events raised	: 92
Messages generated	: 8
Messages delivered	: 7
Message bandwidth	: 873
Average delivery time	: 18318986
Frames transmitted	: 52
Frames received	: 52
Frames corrupted	: 31
Efficiency (bytes AL) / (bytes PL)	: 18.12
EV_REBOOT	: 2
EV_PHYSICALREADY	: 52
EV_APPLICATIONREADY	: 8
EV_TIMER1	: 30

2. Equation for avg throughput

$$\overline{THR} = \frac{Num\ of\ Orginal\ pkt\ sent\ during\ period\ T_{simulation}}{T_{simulation}}$$

$$\overline{T_{Error_{free}}} = \frac{1}{T_{pkt} + RTT}$$

$$T_{pkt} = \frac{L}{R} \qquad RTT = 2 * Propagation\ Delay$$

Therefore average throughput with errors:

$$\overline{THR} = \frac{THR_{Error_{free}}}{Avg\ number\ of\ time\ a\ pkt\ is\ transmitted\ by\ a\ sender = \overline{N}_r}$$

$$\overline{THR} = \frac{1 \Big/ \frac{L}{R} + RTT}{N_r}$$

2.CNETa

$$\overline{THR} = \frac{1 \Big/ \frac{L}{R} + RTT}{N_r} \quad = \quad \overline{THR} = \frac{1 \Big/ \frac{2000\ Bytes}{7000\ Bytes/sec} + 2*2.5\ sec}{1\ pkt} \quad = \quad \overline{THR} = \frac{0.189189189}{1\ pkt} \quad =$$

$\overline{THR} = 0.189189189\ Message/sec$

$0.189189189\ pkt/sec * 2000\ Byte/Message = 0.378378378\ KByte/sec$

2.CNETb

$$\overline{THR} = \frac{1 \Big/ \frac{L}{R} + RTT}{N_r} \quad = \quad \overline{THR} = \frac{1 \Big/ \frac{2000\ Bytes + 32768\ Bytes/2}{7000\ Bytes/sec} + 2*2.5\ sec}{1\ pkt} \quad = \quad \overline{THR} = \frac{0.133628588}{1\ pkt} \quad =$$

$\overline{THR} = 0.133628588\ Message/sec$

$0.133628588\ pkt/sec * ((2000 + 32768)/2)\ Byte/Message = 2.32299937379\ KByte/sec$

2.CNETc

$$\overline{THR} = \frac{1 \Big/ \frac{L}{R} + RTT}{N_r} \quad = \quad \overline{THR} = \frac{1 \Big/ \frac{2000\ Bytes}{7000\ Bytes/sec} + 2*2.5\ sec}{2\ pkt} \quad = \quad \overline{THR} = \frac{0.189189189}{2\ pkt} \quad =$$

$\overline{THR} = 0.094594945\ Message/sec$

$0.094594945\ pkt/sec * 2000\ Byte/Message = 0.189189189\ KByte/sec$

2.CNETd

$$\overline{THR} = \frac{1 \Big/ \frac{L}{R} + RTT}{N_r} \quad = \quad \overline{THR} = \frac{1 \Big/ \frac{2000\ Bytes}{7000\ Bytes/sec} + 2*2.5\ sec}{4\ pkt} \quad = \quad \overline{THR} = \frac{0.189189189}{4\ pkt} \quad =$$

$\overline{THR} = 0.04729729725\ Message/sec$

$0.04729729725\ pkt/sec * 2000\ Byte/Message = 0.0945945945\ KByte/sec$

3. CNETa

$\overline{THR} = 46\ Messages/300\ seconds = 0.153333333\ Message/second$

$0.153333333\ pkt/sec * 2000\ Byte/Message = 0.306666666\ KByte/sec$

3.CNETb

$\overline{THR}=35\,Messages/300\,seconds=0.116666666\,Message/second$

$0.116666666\,pkt/sec*((2000+32768)/2)\,Byte/Message=2.028133333\,KByte/sec$

3.CNETc

$\overline{THR}=25\,Messages/300\,seconds=0.083333333\,Message/second$

$0.083333333\,pkt/sec*(2000)\,Byte/Message=0.166666666\,KByte/sec$

3.CNETd

$\overline{THR}=7\,Messages/300\,seconds=0.023333333\,Message/second$

$0.023333333\,pkt/sec*(2000)\,Byte/Message=0.046666666\,KByte/sec$

4. If you look at the theoretical value obtained from part 2 and compare it with the theoretical value obtained from part 3. You can observe the biggest difference in value with ~ 0.3 difference (12.7% difference). While the other CNETa suffers from difference of ~0.07, (or 18.9% difference) CNETc with ~0.02 (11.9%) and finally CNETd with a difference of ~0.048 (50.6%).

The reason for differences between theoretical and the actual result could be to due:

1.small sample size (which may lead to misleading results)

2.In our theoretical calculation we assume there is a message ready to be sent after the first. But in our simulations messages are created at a set message rate

3. For probability of packet corrupt/lost are probability therefore it is possible in our test simulation the amount of corrupt/lost packets are above or below the average probability of packet corrupt/lost.

4. For CNETb one of the main factor on why the difference is much greater than the other is due to it having the probability of sending a packet with the min size of 2000 and max size of 32768. Therefore it then fall under the same reasoning from #3.

Part 2 Design:

1. The starting host sends a packet to Router 1
Router one buffers it and sends an ACK back to Host 1 and the data to Router 2
2. If Router one receives more data during this time (the ACK got back to Host 1 and it then sends the next packet to Router 1) and did not receive an ACK for the data it sent to Router 2, ignore the data from Host 1.
3. If Router 1 does not receive an ACK for the data it sent to Router 2, resend the packet in buffer.
4. If corruption packet is received, the node will drop it and therefore the previous node would then time out and resend the packet. The same event occurs if the packet is lost, whereby the previous node would then resend the packet.
5. This process is then repeated for the remaining routers and Host 2.

Difficulties of Part 1:

Not much difficulty experience in Part 1 other than ensuring the use of correct values in equation calculation.

Difficulties of Part 2:

The ability to identify why at times my router would transmit length = 0 instead of the actual length of the frame which cause an CNET_application error. Currently I believe it has something to do with how I am saving the value of the length of the buffer.

Another difficulty that I was not able to address was how to handle corrupt or lost frames / ACKs, this is because I was unable to get my protocol working with no errors.

Acknowledgments:

TA (**Mohamed Elmorsy**)

CNET documentation:

http://www.csse.uwa.edu.au/cnet/faq.html#debug

More Practice Lab #2