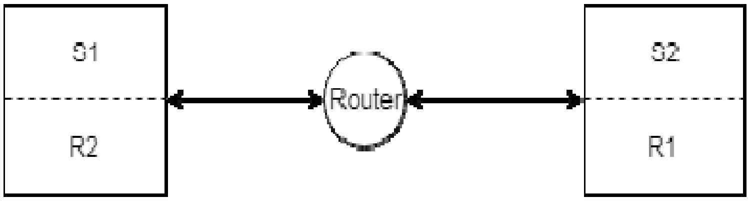
Lab Assignment 2

COL 334/672

Due date: September 26 11.55 pm
Team Size : Max 2 people per team

Note: Solve all problems on your own. Approach the instructor/TA for clarifications. Your solutions must be submitted as a pdf report. Upload the report along with your programming code to Moodle according to the instructions given at the end of the assignment. Note that you are expected to use a tool called Mininet.

You are required to understand the Go-Back N protocol, and implement and test the same. You may use MiniNet emulation to emulate a topology that consists of two Data-Link entities that send frames to each other over a direct link between them. The Data Link entities encapsulate "packets" it receives from its corresponding Network entity and sends them out as data frames. And they do so in both directions. The code you write for the Data Link entities ensures that packets, encapsulated as data frames, are sent to its peer Data Link entity are delivered to the corresponding Network entity without loss, without duplication and in-sequence.



You will need to emulate what happens in real life. That is,

1. There are two types of frames, viz. Data Frames and ACK frames.
2. Data frames are of length L, where L is a uniformly distributed random no. with a minimum of say 512 bits, and a maximum of 2024 bits (inclusive of 256 bit header, but does not include the 32 bit error-check bits).
3. ACK frames are of fixed size, viz. 256 bits (inclusive of 256 bit header, but not the 32 bit error-check bits).
4. Channel capacity is 1 Mbps each way, and full-duplex.
5. Error-probability of a data frame is 0.1, independent of frame size,
6. Error-probability of an ACK frame is 0.05.
7. One-way propagation delay (inclusive of interrupt delay if any) is a random delay between 0 to 5 ms.
8. Transmit window size is 7, and receive window size is 1.
9. The Network entity at each end generates packets 400 packets/sec.

To emulate the above, you will have to write code that runs as "network entity", and similarly as data-link and physical layer entities. And on both ends. Check if MiniNet emulator allow you to emulate propagation delay and loss of frames. If not you will need to add a device (a layer 2 switch) which drops frames with appropriate probability, as also propagation delay. And in both directions.

Having emulated the above, you need to observe the performance of the system in both directions. The parameters you need to track are delay, throughput, drop rate. Questions that you need to answer before you start monitoring the performance are:

1. What defines end-to-end delay?
2. What defines throughput?
3. What is meant by drop rate?
4. How does the system performance vary when
    a. Error- probability is 0.01, 0.05, 0.1
    b. Window size 5, 7 and 9
    c. Propagation delay is 0ms,3ms and 5ms

Submit your results as 3 files:

1. The code running at each end (and the code running on switch if implemented),
2. Observed values of performance parameters,
3. A brief report in the form of a PPT that brings out what you have done, and how?

Comment your programming code well. Upload a single zip file with name1_ENTRYNUMBER1_name2_ENTRYNUMBER2_LA2.zip (e.g.: ABC_2014CS10100_DEF_2014CS100101_LA2.zip) to moodle containing (i) the pdf report. (ii) mininet project code