SIMULTATION PROJECT

Communication Networks
January – March 2021

Objective – To get hands-on training and experience in developing and running simulation models in Omnet++ and able to visualize and analyze system performance. Bonus tasks have bonus points. Each task result output should be shown as a graph supported by a short analysis report of not more than 5–10 lines.

(Submission Deadline: 26.02.2021 by 23:59 via email) – Project Exam on 27.02.2021

NOTE: On 20.02, I may randomly check the progress on your project.

Project Description

The objective of this project is to model, simulate and analyze the **Go-Back-N** protocol. As explained in the lecture slides, Go-Back-N protocol is based on sliding window flow control method.

The project can be implemented by extending the simple tic-toc model. *The following features should be demonstrated.*

Station A (e.g., TIC) sending packets at specified rate towards Station B (e.g., TOC)

The following variables should be configurable via the omnetpp.ini file,

1. The **data rate**. That is, the rate at which TIC sends packets to TOC
2. The **window size (W)** for TOC. That is, the number of packets TIC can send towards TOC without receiving ACKs from TOC. Or this indicates the maximum buffer (or queue) size in the TOC.
3. The **size of N** (where N < W). That is the number of packets that the receiver receives before sending an acknowledgment.
4. The **packet loss/error rate**. This should be random. HINT: use exponential function.
5. Keep the size of sequence number field in the packets to 8 bits, to allow maximum number of 256 sequence numbers before the sequence number wraps around (The sequence number should be able to wrap around).

At the start of the simulation, the TIC should send a control message to TOC querying its window size (i.e., W), which is configured in the TOC at the time of initialization via the omnetpp.ini file. The TOC then replies to the query from TIC indicating the window size (W). i.e., the receive (TOC) buffer size. This implies that you have to implement a buffer in the TOC (i.e., receiver side). HINT: Use the omnet class cQueue for implementing the buffer.

The query_reply message from TOC (indicating the Window size) should be used as a trigger by TIC to start sending the frames/packets.

The TOC should then send a Receive Ready (i.e., RR message) or Receive Not Ready (RNR) message in case it receives or not receives frames transmitted from TIC.

Preferably, the TIC should keep the copies of all unacknowledged messages (in a queue) in case of retransmissions. Or you can simply maintain a table (i.e., array of integers) of unacknowledged sequence numbers. In other words, you also have to maintain the Send Window counter in the TIC and the Receive Window counter in TOC.

Your simulation model should be able to **correctly demonstrate** the operation of the **Go Back N** flow control mechanism in case of

1. No packet loss.
2. In case of packet loss or errors. HINT: generate packet losses using the error parameter randomly

The **report** should present

1. message sequence chart to explain your logic,
2. a flow diagram (i.e., flow chart) to explain your implementation
3. tables indicating your test parameters.
4. graphs to demonstrate the performance of your model for different values of transmission rates, window sizes etc.

Please refer to the lecture slides and the reference books for details on the Go Back N flow control protocol.

Hints:

1. Check the DYNA Client model for message modeling. See how using a single message file they can model different type of messages.
2. Check the DYNA Switch model to learn how to implement and manage a queue in your TOC module.