Spring term 2008
TELCOM 2321 - CS 2520
Wide Area Networks

Lab assignment 2

Report due: Tuesday April 1, 2008

Groups of up to 3 students are allowed
One report per group is required

GSA: Chih-Kuang Lin
Office hours: Fri 12-2pm, 714 SIS
email: chl21+@pitt.edu

1 Introduction

The purpose of this lab assignment is to simulate TCP connections behavior with ns2 under different scenarios. ns2 provides an extensive and detailed implementation of TCP and is one of the most widely used simulation tools for both education and research purposes involving TCP.

The documentation related to ns2 TCP agents and sinks is available at section 35 of the ns2 manual (http://www.isi.edu/nsnam/ns/doc).

You are required to submit a written report by the due date where you must answer all the questions asked in the following sections. Add any figure, screenshot, script code that is required or that you consider relevant.

2 How to simulate a TCP connection

First thing to do is to understand how ns2 simulates TCP connections. Let's start with a classic Tahoe TCP. Download and check the OTcl script named `tahoe.tcl` from the course web page and answer the following questions.

Question 1 - Describe what the `tahoe.tcl` script does and which data it produces.

Question 2 - Run a simulation using the advertised window AW = 6. Plot a set of graphs showing all data produced by the script and add your comments.

3 Dealing with the bandwidth-delay product

With reference to the `tahoe.tcl` script, answer the following questions.

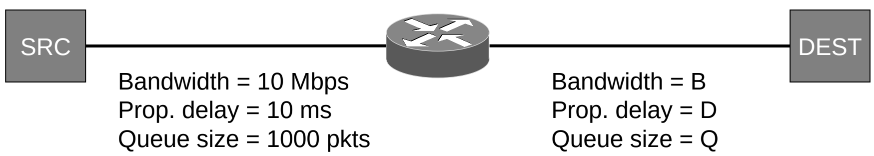
Question 3 - Find the bandwidth-delay product of the link between source and destination and run a simulation using the corresponding advertised window AW. Plot a set of graphs showing all data produced and add your comments.

Question 4 - Modify the `tahoe.tcl` script so that now it accepts four command line arguments, namely the advertised window AW in segments, the link bandwidth B in bps, the link propagation delay D in seconds and the link queue size Q in packets. Run a set of simulations with AW = 25, B = 10 Mbps, Q = 1000 and different values of D so that the bandwidth-delay product varies in the range from 10^3 to 10^7. Plot the normalized throughput computed at the end of each simulation as a function of the bandwidth-delay product, using a logarithmic scale for the x-axis. Choose your simulation points so that the curves are smooth, i.e. you should increase the density of your simulation points when the curve slope is high. Repeat the experiment setting AW = 250 and compare the two plots. Pay attention that, in order to have a reliable measure of the throughput when the delay D becomes large, the simulation should last proportionally. It is suggested to stop the simulation at time 500 D when this value is not less than 5.

4 Including a bottleneck

Modify the previous script so that now

- source and destination nodes are connected through an additional node, according to topology and parameters shown in the figure below
- the queue to be monitored is the one on the link connecting the intermediate node to the destination
- the command line parameters B, D and Q refer to the link connecting the intermediate node to the destination
- the simulation ends at time 20.0



Include the modified script in your report and answer the following questions.

Question 5 - Run a simulation using AW = 100, B = 8 Mbps, D = 10 ms, Q = 20. Plot a set of graphs showing all data produced and add your comments.

Question 6 - Run a set of simulations with AW = 100, B = 8 Mbps, D = 10 ms varying the queue size Q. Find the minimum queue size that avoids losses during the simulation time. Show graphs to justify your answer.

5 Comparing congestion control techniques

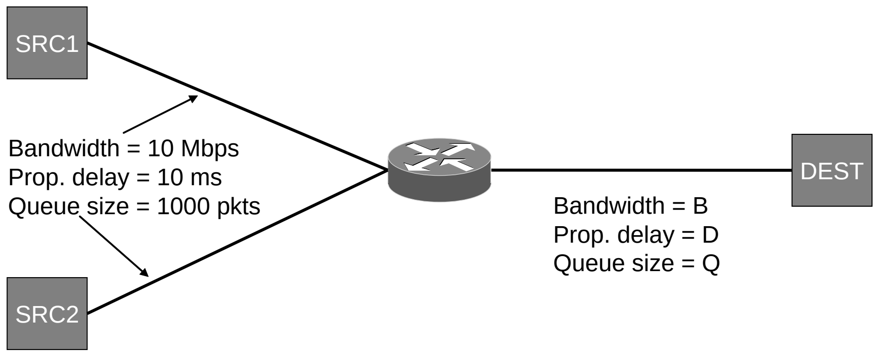
Different versions of TCP implementing different congestion control techniques are available in ns2. Defining a TCP agent as `[new Agent/TCP]` sets a Tahoe implementation. A Reno TCP can be defined as `[new Agent/TCP/Reno]`, while a TCP agent complying with RFC 793 not implementing Fast Retransmit and Fast Recovery mechanisms must be defined as `[new Agent/TCP/RFC793edu]` and then the agent's parameter `add793slowstart_` must be set to `true`.

Question 7 - Compare the graph of the sender window obtained in the answer to question 5 with the same graph obtained by simulating a TCP compliant with RFC 793 using the same parameters. Add your comments.

Question 8 - Compare the graph of the sender window obtained in the answer to question 5 with the same graph obtained by simulating a TCP Reno using the same parameters. Add your comments.

6 Sharing the bandwidth

Modify the script used so far to include an additional node connected to the intermediate node where an addtional TCP source generates segments directed to an additional TCPSink on the destination node, i.e. two TCPSinks must be defined on the same destination node. The advertised windows of the two senders AW1 and AW2 should be two differet parameters. Adopt the same parameters for both links connecting the sources to the intermediate node, as shown in the figure below.



Question 9 - Run a set of simulations using B = 10 Mbps, D = 10 ms, Q = 20 and the following advertised windows:

- AW1 = 20, AW2 = 30
- AW1 = AW2 = 50
- AW1 = 20, AW2 = 80

Plot the graphs of the sender window obtained and add your comments.

Question 10 - Repeat the same experiment using a SFQ queue instead of a DropTail on the link connecting the intermediate node to the destination. Plot the graphs of the sender window obtained and add your comments, comparing with the results of the previous question.

Question 11 - Repeat the same experiment using a RED queue instead of a DropTail on the link connecting the intermediate node to the destination. Plot the graphs of the sender window obtained and add your comments, comparing with the results of question 9.