COMP 6320 Design and Analysis of Computer Networks

ns-2 (Research) Project
(Due Date on Syllabus)

Objective

The objective of this research project is to initiate students to the research process. Through this project, students will discover a networking simulation tool ns-2 widely used in Academia and Industry. Students are encouraged to find/discuss related literature. Students will have to provide at the end of this research project their own opinion based on their experiments and findings.

The Problem

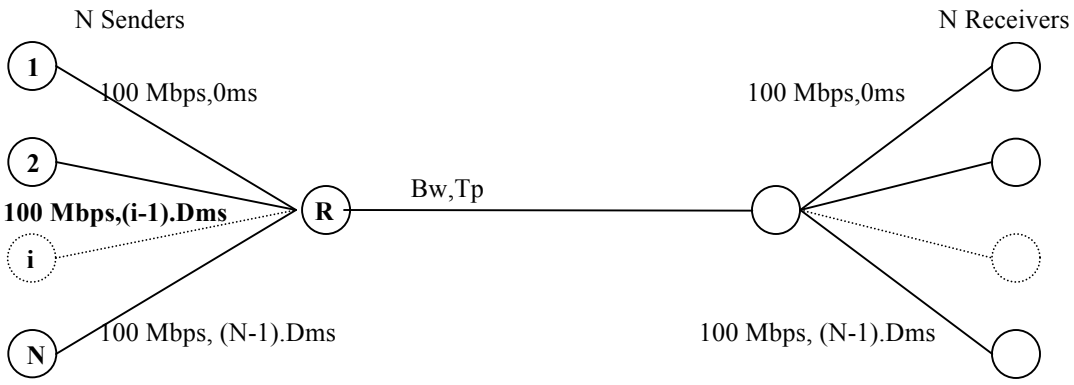
TCP is a reliable transport protocol that currently carries more than 95% of all Internet traffic. The congestion control algorithms (Van Jacobson) of TCP are the basis of the current stability and henceforth success of the Internet. Through some milestone implementations "hacks", TCP was successfully tuned for most applications and environments. However, when the available bandwidth becomes very high, TCP as tuned today does not perform well (See RFC 3649 Sally Floyd HighSpeed TCP (HSTCP)).

In this project, we plan to evaluate HSTCP.

In this project, we want to answer two questions:
First, we want to know if HSTCP is well adapted to very high speeds and performs much better than TCP Newreno.
Second, we want to identify environments where it really performs well and whether it handles well contention.

Part A: HSTCP assessment (**85 points**)
You will simulate multiple experiments and collect results that you will plot. Then, you will have to analyze your results to draw conclusions.

Here is the topology that you will have to simulate:



This topology varies depending on:

1) The bandwidth of the bottleneck **Bw** that will take the following values that you deem of interest
2) The propagation time **Tp** that will take: 50ms, 250ms, 500ms, and 1s.
3) The delay **D** that will take two values : 0 and 100
4) The value N takes values 2, 8, 16, and 32

Each set of values (Bw, Tp, N, D) defines ONE experiment setting. For one experiment setting, you will have to run about 30 simulations to get statistically valid results. For each simulation run, you will vary the starting times of the TCP connections.
C Code (driver.c) and Tcl template (essais.tcl) are provided to show you how to drive and collect results conveniently for multiple simulation runs. Here are the metrics you will collect:

a) the **average throughput** of all senders. The throughput of the network in the amount of data that traversed the network successfully per unit time. Use the Kbits (Kilobit) as a unit. For each run, you have to measure the throughput for EACH TCP connection and take the average.

b) the **fairness index.** The fairness index FI measures whether each TCP connection gets a fair share. FI is computed as follows: let T1,…Ti…and TN the throughput achieved by N TCP connections. FI = $(\Sigma Ti)^2/(N.\Sigma\ Ti^2)$. FI varies from 0 to 1. Closer is FI to 1, higher is fairness.

Answering the questions

During this research project, the students will have to implement/design a set of experiments. These experiments will be completed using the ns-2 simulation tool.
To work on this project, you must use your own machines or engineering Unix machines (I advise the second option). Students are strongly advised to discuss findings, problems, success, and misery with the instructor. Definitely, this is not a project you can complete the LAST week. Here are some milestones you might find helpful.

1) Read Van Jacobson and complete simple scripts from Marc Greiss tutorial. By Oct. 8.
2) Read RFC 3649. Write some simple Tcl scripts using HSTCP and execute them. By Oct. 15.
3) Write simple scripts driven with C. By October 22.
4) Write the C code to drive your experiments and collect data. By Nov. 5.
5) Conduct experiments, collect results, and report findings. By November 20.
6) Write report for this part and start your website. By November 27.
7) Polish the report by November 30.

What to turn in? One submission per group

1) **Hard copy** of your final report
2) Electronic copy of your report and code (C and Tcl) on Canvas.
   Failing to submit **BOTH** (electronic and hard copy) copies result in a delayed submission with the penalties described in the syllabus.