CS 536 (Park) **Assignment XI** **Due: Nov. 21 (Fri.), 1997**

PROBLEM 1

(a) Read Sections 8.1, 8.2, 8.3, 8.4 from P & D.

(b) Read "A taxonomy for congestion control algorithms in packet switching networks," C. Yang and A. Reddy. *IEEE Network*, pages 34–45, July/August 1995. Give a one-page critique (pros/cons) of their classification scheme.

PROBLEM 2

(a) Implement an adaptive congestion control algorithm in a sender-router-receiver set-up which utilizes the help of the router as follows. The router maintains a buffer of size B wherein incoming UDP packets are enqueued. It dequeues packets with exponential service rate μ. The router sends a control packet to the sender each time it awakes to service its queue indicating what the current queue length Q, $0 \leq Q \leq B$, is. Thus the control packet carries a field where the length information is enscribed.

The sender implements a rate-based congestion control algorithm whereby an interpacket spacing parameter Λ (msec) is used to separate successive packets in time. This is an infinite source model and the larger Λ the smaller the mean packet generation rate. The sender is to execute the following control law:

$$\text{new } \Lambda \leftarrow \text{old } \Lambda + \epsilon(Q - Q^*)$$

where $0 \leq Q^* \leq B$ is a target queue length that is to be maintained. Here $\epsilon > 0$ is an adjustment factor. Thus, if $Q > Q^*$, then Λ is increased which is tantamount to decreasing the sending rate, and vice versa.

Test the algorithm using $\mu = 20$ such that the mean time between successive servicing of packets is $1/\mu = 50$ msec; set $B = 40$ (packets of fixed MTU size of 1 kB), $Q^* = 20$, and set the initial value of Λ as $1/(2\mu)$. Run the algorithm five times using five different values for ϵ, from "very small" to "very large." For each run, record the size the of queue at each dequeueing instance into an array (do not use `printf` since this will slow the router down unnecessarily) and plot the trace at the end of the run. Each run should consist of the generation of 1000 UDP packets. Note that the receiver here is a purely dummy entity. Give an interpretation of your data as a function of varying the parameter ϵ. Is there an "optimal" value for ϵ?

(b) As a continuation of part (b), change the router such that its service rate μ varies as a function of the current queue length Q where $\mu = Q$ for $0 \leq Q \leq Q^*$ and $\mu = 40 - Q$ for $Q^* < Q \leq 40$. Repeat the experiments corresponding to part (b). What, if any, differences do you observe? Give an interpretation of your results.

PROBLEM 3

The file ~park/pub/cs536/fractal.dat on xinuserver contains a 10000 sec trace of throughput traffic at the bottleneck link of a router. Call the traffic trace $X(t)$. The first column contains the time stamp at 10 msec granularity starting at 0, and the second column contains the traffic volume in bytes. Compute the mean traffic rate A_s of this traffic trace. Generate a synthetic Poisson traffic trace of 10000 sec duration at granularity 10 msec such that the latter has mean traffic rate A_s as well. Call it $Y(t)$. Explain how this is done.

Let $X^{(100)}(t)$ be the aggregated time series (as defined in class) at the 100 sec level. That is,

$$X^{(100)}(0) = X(0) + X(0.01) + X(0.02) + \cdots + X(99.98) + X(99.99)$$

and similarly for $X^{(100)}(100), X^{(100)}(200), \ldots, X^{(100)}(9900)$. Let $X^{(10)}(t)$ be the corresponding aggregated time series at the 10 sec level, $X^{(1)}(t)$ at the 1 sec level, $X^{(0.1)}(t)$ at the 0.1 sec level, and $X^{(0.01)}(t) \equiv X(t)$ at the 0.01 sec level. Similarly for $Y^{(\cdot)}(t)$.

Using a graph plotting package (e.g., `gnuplot`), generate 5 plots, the first depicting the first 100 points of $X^{(100)}(t)$ (actually the entire series in the 100 sec case), the second depicting the first 100 points of $X^{(10)}(t)$, and so on with the fifth depicting the first 100 points of $X^{(0.01)}(t)$. Perform an analogous plotting for $Y(t)$ and its aggregated time series. Putting them side-by-side, what differences do you observe as you move up and down in scale (time)? Make sure to have the ordinate (i.e., vertical axis) displayed starting from 0 bytes upwards.

Perform a similar comparison for the *last* 100 points of the aggregated time series. Have your conclusions changed?

Compute the variances of $X^{(100)}(t), X^{(10)}(t), X^{(1)}(t), X^{(0.1)}(t), X^{(0.01)}(t)$ and compare them to the corresponding values for $Y^{(\cdot)}(t)$. What (if anything) do you observe? Give your interpretation.

What about the autocorrelation functions of $X(t)$ and $Y(t)$? Plot and compare them as a function of the (time) offset T. Can you reason about the performance impact of $X(t)$, $Y(t)$ on packet loss?