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

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?

PROBLEM 3

This is an extension of Problem 4 in Assignment VIII. To induce controlled dropping of packets (emulating a noisy channel), implement a router, call it `my_router`, that accepts UDP packets with the (payload) format

destination-IP-address payload

on port ABC (choose as you wish), examines the destination IP-address, then forwards the payload to the destination host. The router, before forwarding, however, is to drop the current packet with probability p which is a fixed parameter.

The structure of the router consists of a SIGPOLL handler which dequeues packets whenever they arrive, examines the header, then forwarding or dropping the packet.

Repeat the experiments carried out in Problem 4 of Assignment VIII, now, with `my_router` running on a third host through which all traffic (data and ACK) is channeled. Use two values for p, 0.2 and 0.1, and compare their performance. Use the runs from Problem 4 in Assignment VIII as an approximate sample point corresponding to $p = 0$. Give an interpretation of the results.

PROBLEM 4

This is a variation of Problem 2 using a sender-router-receiver set-up. Rather than invoke the help of the router for congestion control purposes, however, we will implement a purely end-to-end control mechanism. The traffic is still to be routed through the router as before which maintains a buffer of size $B = 30$ (packets).

The goal of the congestion control algorithm is to maximize throughput when sending $X = 1000$ packets (fixed size 1 kB) where *throughput* ν is defined to be

$$\nu = \frac{X - A}{T}$$

where $0 \leq A \leq X$ is the number of dropped packets (you can detect this at the receiver using sequence numbers) and T is the transfer time at the sender of all X packets. That is, T is the difference of the time that the last packet was sent from that of the first packet. Reliability is not an issue. Thus v *may* be maximized if no packets are dropped ($A = 0$) and T is minimal. Why is this not necessarily so? Explain.

The control law which dynamically adjusts the interpacket spacing parameter Λ is now as follows. Let `RTT` be the round-trip time of the most recent successfully received packet. The round-trip time is computed by having the sender put a time stamp of the send time on a packet (in addition to its sequence number) and having the receiver acknowledge the receipt of every k'th ($k \geq 1$) successfully received packet by sending an ACK packet containing the sequence number and sender's time stamp. Upon receive of an ACK packet, the sender takes another time stamp and by taking the time stamps' difference, computes the round-trip time. Let `RTT'` be the newly computed round-trip time. Then,

$$\text{new } \Lambda \leftarrow \text{old } \Lambda + \epsilon \cdot \text{sgn}(\texttt{RTT'} - \texttt{RTT})$$

if $|\texttt{RTT'} - \texttt{RTT}| > \theta$, and new $\Lambda \leftarrow$ old Λ (i.e., Λ stays unchanged) otherwise. Here $\theta > 0$ is a fixed threshold parameter and $\text{sgn}(\cdot)$ is the sign function; i.e., $\text{sgn}(s) = 1$ if $s \geq 0$, and $\text{sgn}(s) = -1$ if $s < 0$. What is this control law trying to do? Explain.

Test your implementation by running your system and measuring throughput ν for $\theta = 1$ ms, $\epsilon = 2$ ms, initial $\Lambda = 25$ ms, and $k = 5$. As part of the control law, enforce 5 ms $\leq \Lambda \leq$ 60 ms. The router (unlike in Assignment XI) services packets at a fixed (deterministic) interval of 30 ms. Give a time trace of Λ, `RTT`, and queue length, again, being careful not to use `printf`. Interpret your results.

Rerun your experiments for $\epsilon = 4, 6, 8, 10$ ms and show ν as a function of ϵ. What do you observe? (You may need to run each experiment 3–4 times to get a usable average performance value.)

Do the same for $\theta = 2, 3, 4, 5$ ms (with $\epsilon = 2$ ms). What do you observe?

What is the throughput when no control law is active (i.e., Λ is unchanged) and Λ is fixed at its initial value of 25 ms? Measure the throughput values for initial Λ of 10, 15, 20, 30, 35, 40 ms and plot ν as a function of initial Λ. In your opinion, how does the dynamic control law compare against the performance of static control?