Computer Networks Fall 2015: Project

Date Assigned:10/18/2015

Due Date for Part 1: 11/13/2015

Due Date for Part 2:12/7/2015

1 Project Overview

This project consists of two parts. The first part is the simulation of a simple queueing system with a finite buffer to study the packet loss probability as a function of the buffer size and the traffic intensity. The second part is the simulation of the Ethernet exponential backoff algorithm. We will implement both project in Python using SimPy. You will find the required documentation at the following site `https://simpy.readthedocs.org/en/latest/contents.html#`. The site contains instructions to install SimPy and many examples. The following two sections (Section 2 and Section 3) describe the two parts.

2 Part 1

Figure 1 shows a simple queueing system model of output port of a router.

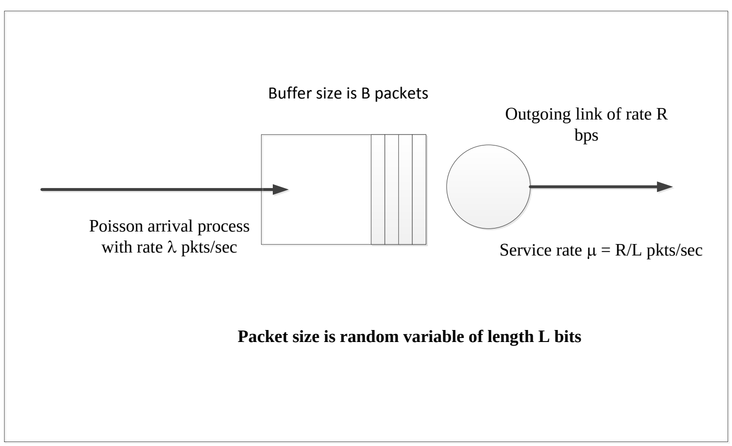

Figure 1: A finite buffer queue

Packets arrive following a Poisson process with rate λ. Packets are of variable length and transmitted on the link of rate R bps. We will assume that the effective service time of a packet is negative exponentially distributed with rate parameter μ packets per second. To keep things simple we will assume that $\mu = 1$ pkts/sec. The buffer size if B packets.

You are required to write a Python code using SimPy to simulate the above system and plot (or tabulate) the following results

1. For $\lambda = 0.2, 0.4, 0.6, 0.8, 0.9, 0.99$ and $B = 10, 50$ determine the packet loss probability P_d.
2. Compare the above results using the theoretical formula derived following the discussion in class.

2.1 Deliverables for Part 1

1. Group size: 2. One submission per group.
2. Due date for Part 1: 11/13/2015, by 5 PM.
3. Report should include 1) the derivation of the theoretical formula to calculate P_d, 2) the simulation code, and 3) tables (or plots) of the results.
4. Submit a hard copy in the homework box in Kemper 2131.

3 Part 2

In this part you will simulate and analyze the binary exponential backoff algorithm of the IEEE 802.3 Ethernet protocol. Before you get started you should read Section 5.3.2 (Random Access Protocols) of the text. We will cover it in class but you should read ahead.

Figure 2 shows the simulation model.

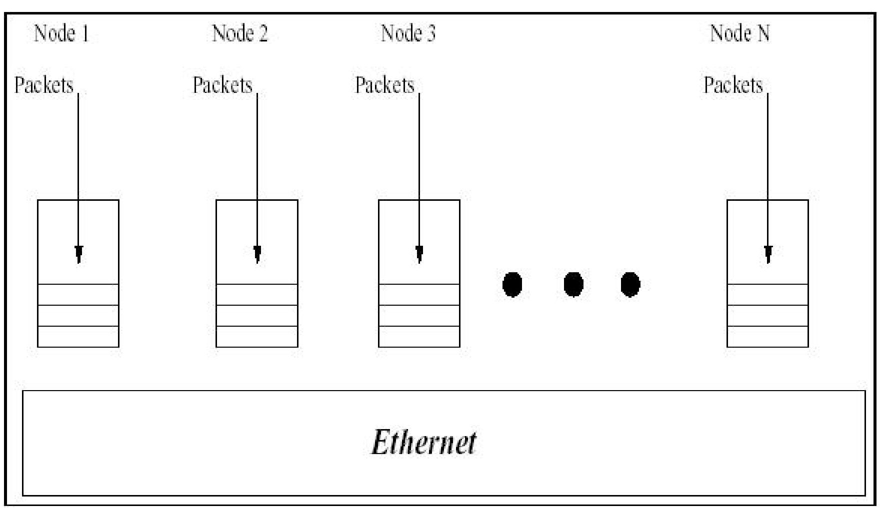

Figure 2: Ethernet simulation model.

In order to develop the simulation model, we will make the following assumptions:

1. We will assume that time is slotted into equal length of time slots. In the subsequent discussion, the length of the time slot will be denoted by T_s.
2. We will let N denote the number of hosts. As before, λ will denote the mean arrival rate of packets. We will assume the hosts are identical, and each hosts receiving packets with a mean arrival rate of λ packets/second. The arrival process follows a Poisson process with rate parameter λ pkts/sec (same as Part 1 of the project).
3. Hosts can transmit only at slot boundaries.
4. If at a particular slot boundary there are more than one host ready to transmit, there will be a collision. When hosts collide, they will schedule their retransmission using the following binary exponential backoff algorithm. The number of slots to delay after the n^{th} retransmission attempt is chosen as a uniformly distributed integer in the range $0 \leq r \leq 2^K$, where $K = \min(n, 10)$.
5. In this phase, we will be interested in plotting the throughput where throughput is defined as the number of successful transmission per time unit. In the simulation, you can count the number of slots in which there is successful transmission and divided that by the total number of slots that you simulate.

3.1 A Sample Execution

In order to understand how the backoff algorithm works it will help you to work though this problem. We consider 3 hosts operating following the same slotted system described above. Each host maintains 3 variables

L: The number of packets in the queue.

N: The number of times the packet at the head of the queue has been *retransmitted*. When a new packet comes to the head of the queue n is reset to 0.

S: The slot number when the next transmission attempt will be made for the packet at the head of the queue.

Figure 3 shows the state of the variables for the three nodes.

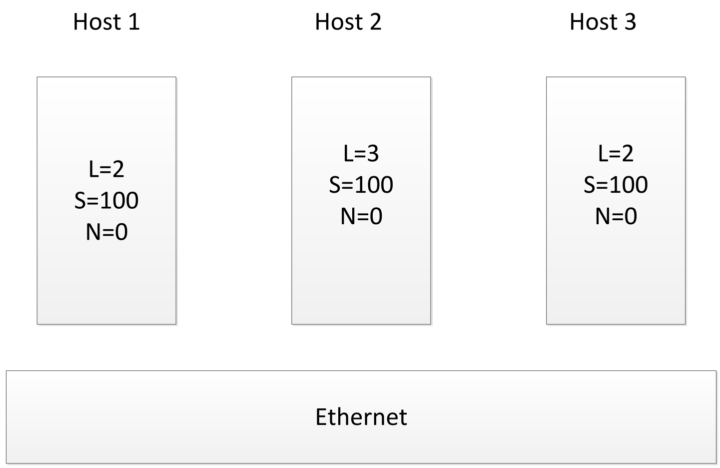

Figure 3: 3 Node Ethernet simulation model

Assuming that all the random numbers are drawn from the following sequence, determine the number of slots required to transmit all the packets from the three nodes.

```
0.6621015449 0.0598288625 0.5979568986 0.0940302045 0.9731501949
0.0662842158 0.5671123588 0.2960073405 0.0836140907 0.1982544372
0.3973937067 0.1825762354 0.7300244789 0.9750743033 0.4226574674
0.7283845034 0.9393420117 0.3463454130 0.0346137777 0.5296165026
0.7221479421 0.4893986084 0.7332225929 0.3619475896 0.3159643647
0.7164982632 0.8700478233 0.4864938182 0.2679576606 0.1540246727
0.5281929232 0.6834302917 0.4536605629 0.7499270914 0.0581091905
0.0344902470 0.6920460807 0.2915925588 0.8664808415 0.7130973549
0.1030703052 0.6871530802 0.7494474363 0.6699161683 0.2188189046
0.5913891383 0.9112765496 0.0002334074 0.5658022752 0.0542878532
```

3.2 Simulation Analysis

Develop a simulation model to simulate a system with $N = 10$ hosts. To develop the simulation model you can build on the single server queue model that you were given for Part 1. You will now have $N = 10$ queues one for each host and there will be one server that will implement the binary exponential backoff algorithm.

Based on the simulation model, obtain the following results:

1. Plot or tabulate the throughput as a function of λ with the binary exponential backoff algorithm as described above. Slot time $T_s = 1$ and number of hosts $N = 10$. Obtain the throughput for the values of $\lambda = 0.01, 0.02, 0.03, 0.04, 0.05, 0.06, 0.07, 0.08, 0.09$.

2. Do the same as above with the following modification to the binary exponential backoff algorithm: the number of slots to delay after the n^{th} retransmission attempt is chosen as a uniformly distributed integer in the range

range $0 \leq r \leq K$, where $K = \min(n, 1024)$. This is linear backoff. Again obtain the throughput for the values of $\lambda = 0.01, 0.02, 0.03, 0.04, 0.05, 0.06, 0.07, 0.08, 0.09$.

3.3 Deliverables and Other Details for Part 2

The following are the submission guidelines.

1. Group size: 2. One submission per group.
2. Due date for Part 2: 12/7/2015, by 5 PM.
3. Commented Code: Your code should be in ŝimulation2.py. Please submit using handin on the CSIF machines. The command to submit is: ĥandin cs152a hw2 simulation2.py
4. Report+code: Submit a hard copy of the report and the code in the homework box in Kemper 2131. The report should contain plots obtained from the simulations and a short (1-2 paragraphs) analysis of what each result means.