Networking-Labs

Question 1

Write a Python function that receives   as one of its arguments and that returns a single random sample drawn from an exponential random distribution. Using this function, write a short piece of Python code to generate 1000 exponential random variables with =75. What is the mean and variance of the 1000 random variables you generated? Do they agree with the expected value and the variance of an exponential random variable with =75? Your values will not be exact. Why not? Note: if you do not know what you are expecting to see, you will need to look it up.

Statistic	**Computed Value**	**Expected Value**
Mean	0.0133048956165309	0.0133333333333333
Variance	0.0001716209800562	0.0001777777777778

They can't be exact because we are running a finite amount of experiments on a random variables. As we increase the number of experiments we can got close the theoretical expected and variance but will never be able to truly reach it.

Question 2

Build your simulator for an M/M/1. Explain in words how you compute the performance metrics E[n] and P_IDLE. Do not show code.

1. To computed E[N], the simulator needs to keep count of two variables, the total number of packets in the queue and the number of observers. Since the observer events happend much faster than the arrival/departure events we can use those times to accurately acheieve the average load on the queue. During each observer event the simulator adds the current packets in the queue to a total number of packets in the queue. Divide the sum by the number of times that it has been counted should be perfect to get the average size of the queue.

2. To compute P_IDLE, the simulator needs to keep track of its total time idling and divide it by the length of the simulation. During the simulation, when a packet arrival occurs, the simulator checks if the queue was previously empty. If it was empty it simply subtracts the current time - the last departure.

Question 3

*The packet length will follow an exponential distribution with an average of L = 2000 bits. Assume that C = 1Mbps. Use your simulator to obtain the following graphs. Plot them on separate figures.

1.*E[N], the average number of packets in the queue as a function of (for 0.25 < < 0.95, step size 0.1).*

2. *P_IDLE, the proportion of time the system is idle as a function of (for 0.25 < < 0.95, step size 0.1).*

Explain the trends in both graphs

1. As the queue utilization ( ) increases, the trend for the average exponentially increases. This is becuase even though the queue is infinite, there is a finite amount of time to process each packet and only one packet can be processed at a time. When a new packet arrives before the last packet has finished, that fills up our queue. The expected value of the service time can be represented as $\frac{E(\frac{1}{L})}{C} = \frac{L}{C}$ and the expected difference of the arrival time is $E\left(\frac{\rho \cdot C}{L}\right) = \frac{L}{C\rho}$

We can clearly see mathematically that on average the queue packets will arrive fasater they can be service by a facotr of $1/\rho$ which is why we get the exponential distribution appraching 1.

2. As the queue utilization ( ) increases, the trend for the amount of idle queue time decreases linereraly. This is because we are finiding the inverse so now it relative to ρ so as ρ increases, the idle time decreases.

Question 4

For the same parameters, simulate for =1.2. What do you observe? Explain.

Since $\rho > 1$ the queue will fill faster than it is able to empty itself. This means that as time increases, the queue will never empty and will get larager and larger.

Question 5

Build a simulator for an M/M/1/K queue and briefly explain how you implemented packet dropping. Explain in words how you computed P_LOSS. Do not show code.

Since I store the events in a priority queue and pop the most recents events repeadetly, all I need to do is increment the queue when a arrival event happens and decrement it when a deperature event occurs. Now that we know the queue length, all the simulator has to do, is during an arrival, if the queu length is larger the buffer size, drop the packet and incrementthe count. To then calculate P_LOSS all we need to do is diviede this drop countter by the number of arrivals.

Question 6

Let L=2000 bits and C=1 Mbps. Use your simulator to obtain the following graphs: - PIDLE as a function of (for 0.5 < < 1.5, step size 0.1), for K =

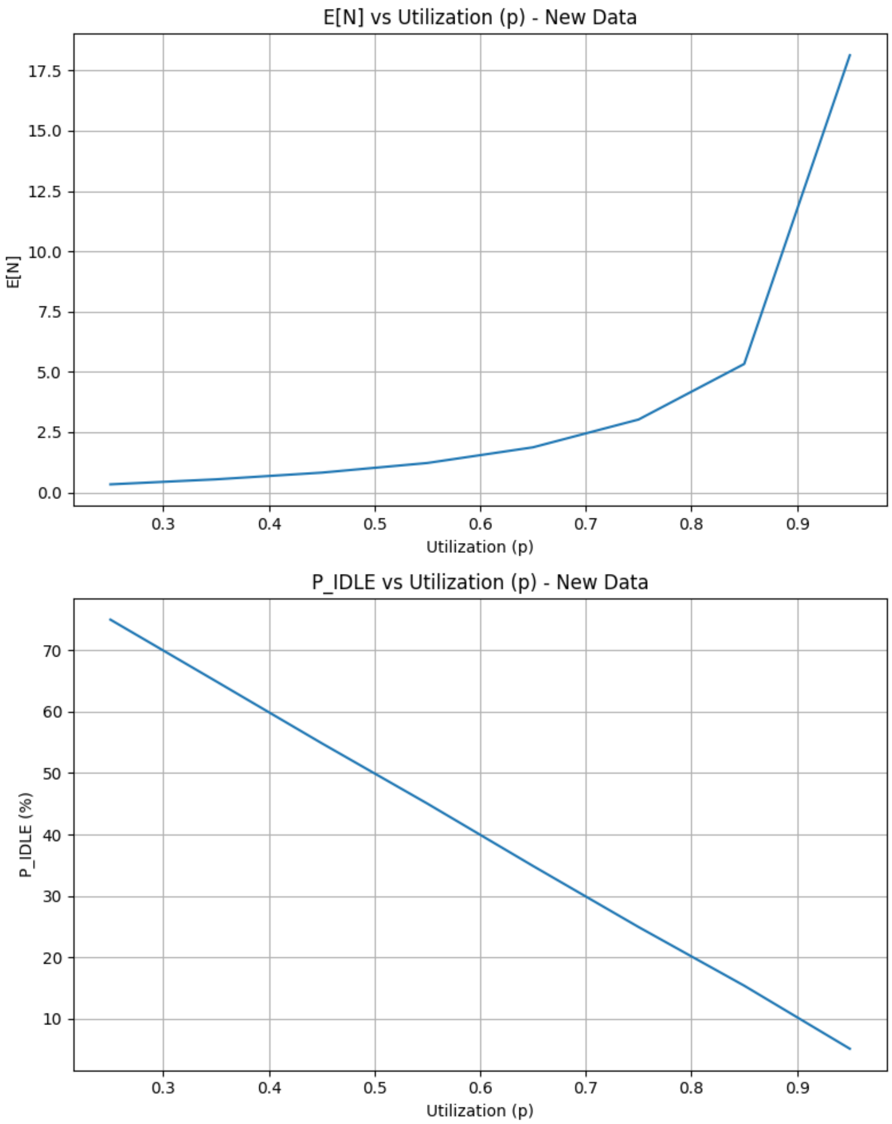
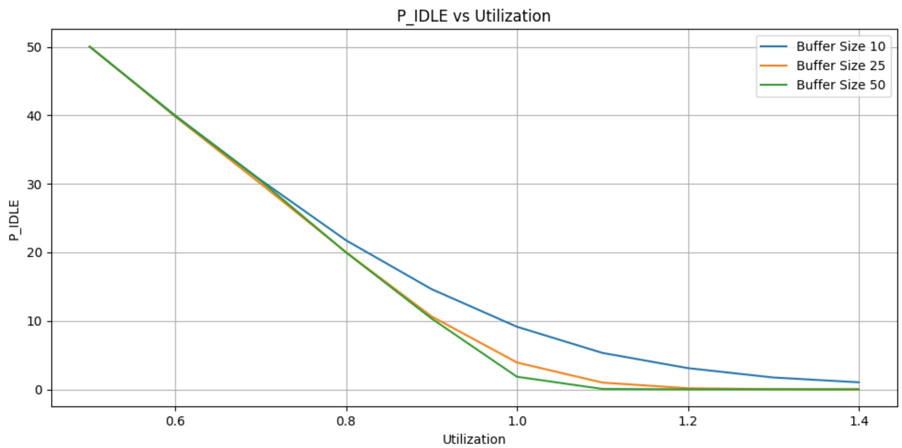
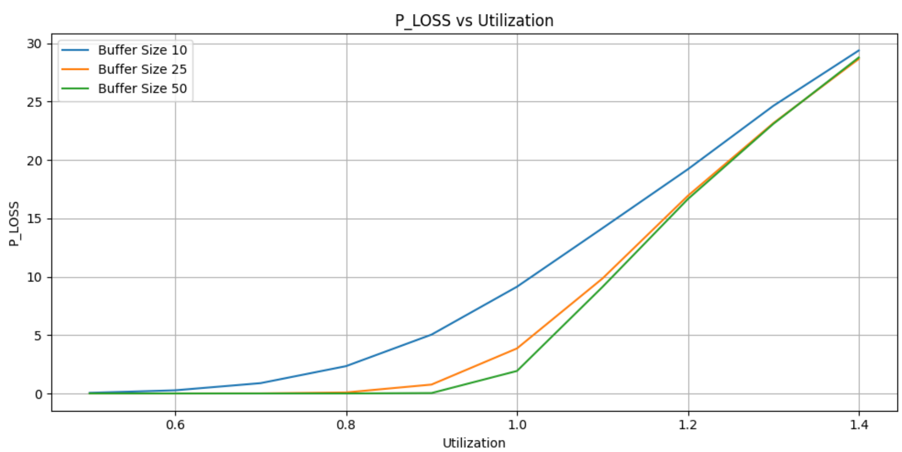


Figure 1: Graph for Question 3

10, 25, 50 packets. Show one curve for each value of K on the same graph. - PLOSS as a function of (for 0.5 < < 1.5) for K = 10, 25, 50 packets. Show one curve for eac value of K on the same graph. For which value of K does PIDLE reach zero soonest? Why is this the case?





P__IDLE reaches zero the soonest when the queue is the largest this. This is becuase buffer size is inversly proportional to the P__LOSS. So as the queue is filled the smaller queue will drop there packets, and when they have a large gap between arivals, will be able to empty there queue. For the larger queue, this is not the case as it has a larger buffer to empty before it can become empty.