CSC 591/ECE 592 IoT Analytics - Project 1
Task 3 - Statistical Estimation of the Response Time

Remember that any exchange of code with another student is forbidden as it constitutes cheating. Do not give and do not copy code! We will run Moss to verify that there has not been any code sharing. Offenders will receive zero for the entire project. Take this warning very seriously.

Objectives

The objective of this task is to use the simulation in order to calculate: the mean and $95-th$ percentile and their confidence intervals of the response time R_{RT} and R_{nonRT} of the RT and nonRT messages, respectively. We recall that R_{RT} is the time elapsed from the moment an RT message arrives at the RT queue to the moment it is fully processed and departs from the server, and R_{nonRT} is defined similarly.

Program Modification

For this, you have to modify your program as follows:

- Stop printing a line of output each time you handle an event as in Tasks 1 and 2. In other words, do not print the hand simulation table.
- Add a data structure to collect data on the response time.
  - For this, you will have to implement a queue structure for the RT and another one for the nonRT queues. Arrivals are added to the end of the queue, and messages are served from the top. Each data element represents a different message and it should contain the time of arrival.
  - When an RT message completes its service and departs from the server, calculate the elapsed time for the message by subtracting the current time from the time of its arrival. Save it to the array RT.
  - Same as above for non RT messages. Save the elapsed times to the array nonRT.
  - After you simulate the required number of messages, use the two arrays RT and nonRT to calculate the required statistical estimates. If space is a problem, you can use the RT and nonRT arrays to store the response times from one batch, and calculate the batch mean and batch $95-th$ percentile after you have simulated the total number in the batch. Save this information in another array, and reuse RT and nonRT arrayes for the next batch.
- Implement the batch means method to construct the confidence intervals.
- Calculate the mean, $95-th$ percentile and their confidence interval of R_{RT} and R_{nonRT}.

The length of the simulation is controlled by m, the number of batches and the number of observations b within a batch. Set $m = 51$, and $b = 1,000$. (We need a large batch size in order to estimate the percentile accurately.) In order to allow for the initial conditions, ignore the results from the first batch, and calculate all your performance metrics using the remaining 50 batches. Run your simulation until you have mb observations collected in each array RT and nonRT. Then apply the batch means method to RT and nonRT arrays separately. It is possible that one array may have more than mb observations. In this case use only the first mb observations.

The simulation code should prompt the user for the input values:

- mean inter-arrival time of RT messages, $MIAT_{RT}$.
- mean inter-arrival time of nonRT messages, $MIAT_{nonRT}$
- mean service time of an RT message, MST_{RT}.
- mean service time of a nonRT message MST_{nonRT}.
- Number of batches, m.
- Batch size, b.

Results:

Set $MST_{RT} = 2$, and $MST_{nonRT} = 4$

Set $MIAT_{RT} = 7$, and vary $MIAT_{nonRT}$ from 10 to 40 in increments of 5.

1. For each value of $MIAT_{nonRT}$, obtain the statistics: Mean and $95 - th$ percentile, and confidence intervals of R_{RT} and R_{nonRT}.
2. Graph your results on the mean response times of R_{RT} and R_{nonRT}, including the confidence intervals (drawn as vertical bars), as a function of $1/\lambda_{nonRT}$.
3. Same for the $95 - th$ percentile.
4. Comment on your results.

Deliverables on Task 3

What to submit: Submit your code, and a pdf file named "P1T3-Results-unityID.pdf" with all your results and comments.

Grading:

- 50 points if the code compiles and runs on eos correctly
- 30 points if the code gives correct results on test data designed by TA
- 10 points for summarizing and commenting your Task 3.1 results.
- 10 points for summarizing and commenting your Task 3.2 results

Failure to discuss your results will cause you a 10-point penalty per task.