Carleton University

NET4001

Network Simulation

Marc St-Hilaire

[Term Project]

Student name: Taewoo Kim, Samuel Iwuno

Student number: 101022013, 100960021

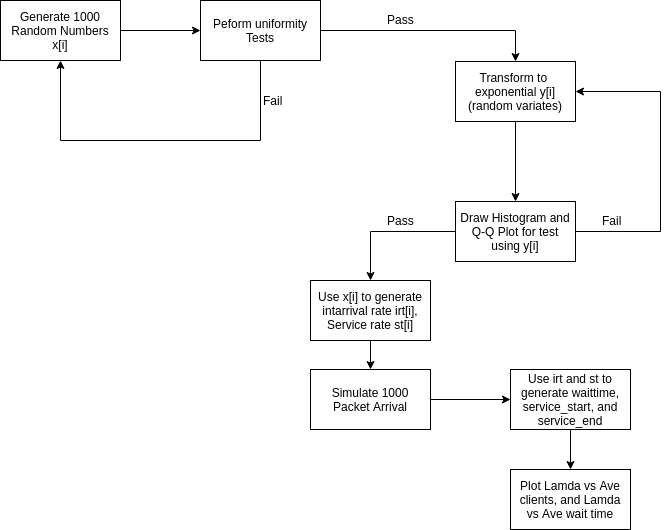
Date: Dec 7th, 2018

# 

# INTRODUCTION

The aim of this project is to practice the algorithms and techniques that have been seen during the semester. To be more specific, we will perform the simulation of an M/M/1 queueing system. The project is divided into 3 parts. The first part requires us to generate 1000 random variables between 0 and 1, and perform the Kolmogorov and Chi squared tests on the variables to test their uniformity. The second part is to then take that random generated range of numbers, transform them into exponential distribution and then draw the histogram and Q-Q plot to test its exponentiality. Finally, we use the results from part 2, to simulate a service receiving and processing a 1000 packets. Then testing whether those numbers correlate with thee mathematical analysis from the lectures

# ALGORITHMIC DETAILS



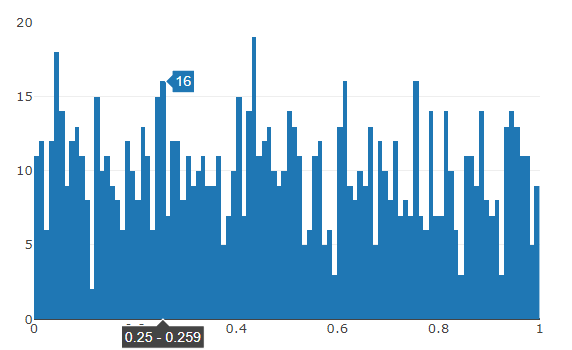
Language used is **JavaScript**. The buttons call appropriate functions

* clicking step1 calls **random\_generate()** which generates random numbers, draws histogram and performs uniformity test
* Step2 histogram calls **make\_histogram()** which then calls **exponential\_rand()** which generates the random variates and draws the histogram
* Step2 QQplot calls **make\_qq()** which in turn calls **exponential\_rand()** which generates the random variates, and draws the QQplot
* Step3 Average Clients calls **step3\_graph()** which then calls **single\_server\_queue()** with different values of lamda(λ) and then generates the average packets in system(L) vs Lamda(λ) graph
* Step3 Average delay calls **step3\_graph\_delay()** which then calls **single\_server\_queue()** with different values of lamda(λ) and then generates the average wait times(Wq) vs Lamda(λ) graph
* The **single\_server\_queue()** method generates the service times(st),inter-arrival times[irt], wait times[wt], service start[sts]], and service end times[ste]

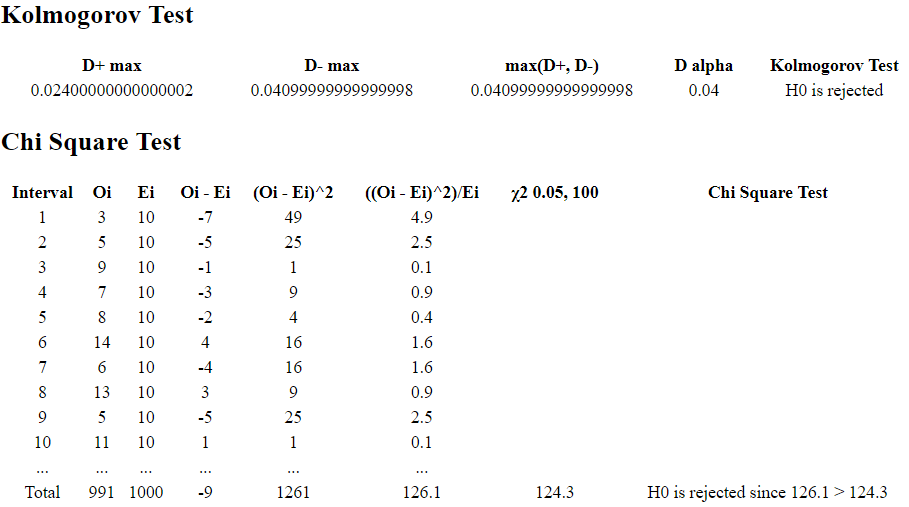
# 

# RESULTS AND ANALYSIS

Step1)



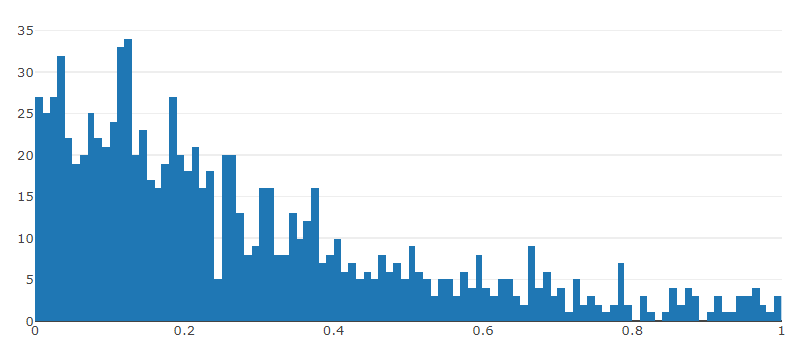
Randomly generated numbers shows uniform distribution between [0, 1]. Using 100 classes for intervals and tested accept/reject based on *Kolmogorov Test* and *Chi Squre Test*



Tested random numbers dataset based on Kolmogorov Test and Chi Square Test

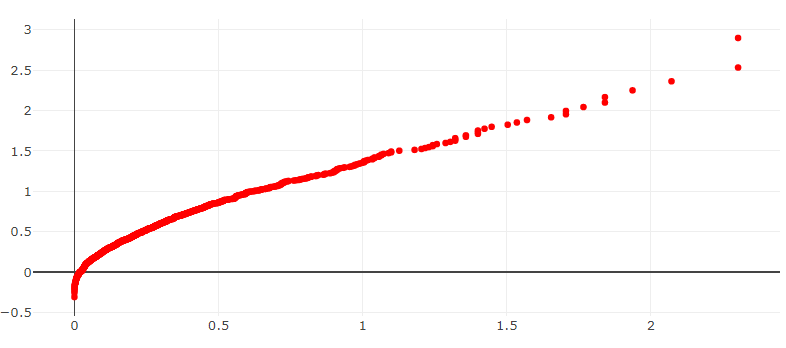
# Step2)

1. Exponential distribution random variates



# Histogram of random variates based on Random Generated Numbers in Step1), observing the graph, we see that it is aligned right, which is a characteristics of an exponential distribution

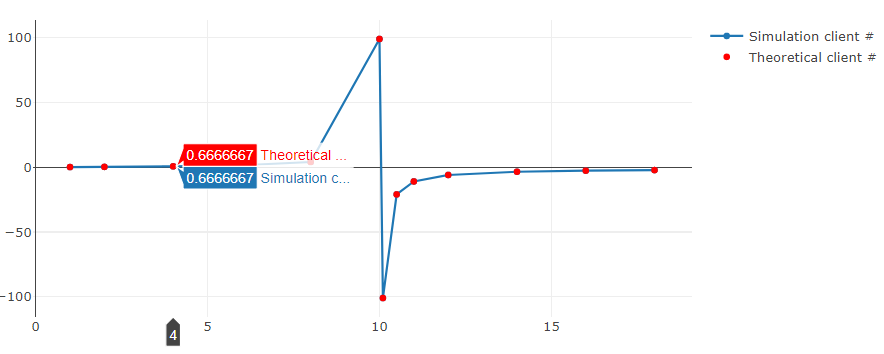
b) QQ plot



The QQ shows the random variates(sorted) vs F-1((j-1/2)/n) . since this approximates a straight line, we can conclude that it is indeed an exponential distribution

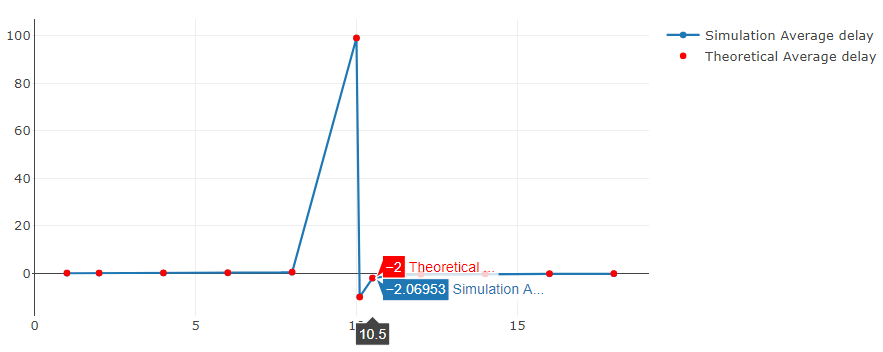
# Step3)

a) Average clients



x-axis as lambda, and y-axis as number of clients. As x(=lambda) is getting close to 10, number of clients are infinitely increasing. Simulation and Theory results were showing infinity, but for representing graph, infinite number is replaced by 100, -100/.

b) Average Delay



Similarly shown in Step3-a), as x(=lambda) is getting close to 10, delay is infinitely increasing. Infinite number is replaced by 100 for representing graph.

# PROBLEMS ENCOUNTERED

1. Graph library support did not meet some of the simulation needs

For Step2 QQplot, we were hoping to get best-fit single line graph for 1000 of random number dataset. But it scatter dataset or connecting line was only available.

b) Showing infinity number was not supported in the graph, resulting infinity value at lambda = 10 was missing, instead we used 100, -100 to represent infinity number.

c) Graph does not show smooth lines to show exponential growth, graph rather show steep graphs.

d) Using more x-axis dataset(lambda values) would make graphs more visible since between lambda = 10 and other values show dramatic increase or decrease. We added more values for drawing better graphs, but more lambda value dataset would make it smooth lines in the graphs.

# CONCLUSION

With randomly generated numbers, the research was able to make random variates and verified those numbers are exponentially distributed. Also with different methods of testing such as Kolmogorov test and Chi Square test, reject/accept testing was applied to confirm the randomness of generated numbers.

In Step 3, average clients and average delay were calculated with using equations based on randomly generated number dataset. This result was compared with *Little Theorem* to verify whether there is a discrepancy between simulation dataset including service time and inter-arrival time and given values for service rate and arrival rate (μ = 10, λ = 1, 2, 4, 6, 8, 10, 12). Both simulation graph and predictive graph show very close results in values.

# CONTRIBUTION

**Taewoo Kim** - Mainly responsible for Step 1 random generated numbers and Step 3 graphs, writing Results and Analysis, Conclusion Sections. Responsible for coding conventions in Javascript and HTML files as well as variable names. Kolmogorov and Chi Square test, Simulated and calculated results values of Average clients, Average Delay.

**Samuel Iwuno** - Responsible for Generating Random Variates and generating Histograms and QQplot int Step 2. And generating service times, interarrival times and arrival times in step 3, responsible for the introduction and algorithmic details