DSCI-3411 Fundamentals of Simulation - Dr. Ali Hadi

THE AMERICAN
UNIVERSITY IN CAIRO

Final Project

Simulation of Queueing Systems

1 Teller 1 Queue VS 2 Tellers 1 Queue

Farida Simaika - 900201753

Katia Gabriel - 900202272

Acknowledgement

We would like to thank and express our sincere gratitude for our professor Dr. Ali Hadi for the constant encouragement and support throughout this course. We would also like to thank him for enriching our knowledge up until the completion of this final project.

Introduction

In this fast-paced world we are living in and with the technological advancements and the emergence of data–driven decisions, it is of huge importance to take the right decisions to be able to cope with day-to-day challenges and minimize harms. However, sometimes dealing with huge problems is difficult especially with limited time, resources and money. Accordingly, the process of simulation has become an inevitable method and practice to help yield efficient and effective decision making in almost all fields.

Simulation is simply put the replication of real-world problems to mimic certain behaviors and phenomena that are complex to execute both analytically and manually. It is of huge importance to grasp simulation as a powerful tool to formulate answers to specific problems, in a timely manner, with more accuracy and less costs.

In light of this, the following project is an example of how simulation can be used to replicate, analyze and make solid decisions. The computer simulation in this project was done using the programming language R and a seed was set to make sure that the project is constant and can be replicated yielding the same results for comparison purposes. The code and the simulation will be discussed and explained thoroughly in this report.

Problem Statement

Today, banks are one of the most important public organizations. A major problem that arises in most of them is queuing. Long waiting times are the main source of dissatisfaction among customers. Customer satisfaction plays a significant role in banks and allows them to sustain competitiveness. An obvious solution to this problem would be to increase the number of tellers and branches of a certain bank. However, randomly increasing the number of tellers without sufficient evidence to prove the effectiveness of such a decision would have major economic costs to the management. The extra unnecessary tellers will have to be paid no matter the amount of time where they remain idle.

Therefore, managers and decision-makers must focus on analyzing queues and customer flow to optimize their operations and decrease customers' waiting time. To help decision-makers make optimal decisions and reduce customers' waiting time, we built two queuing models involving different numbers of tellers. This project aims to compare two queueing models with exponentially distributed random arrival and service times in order to choose the model that fits best in maximizing both customer and management satisfaction and minimizes the employees' idle times.

Our Approach

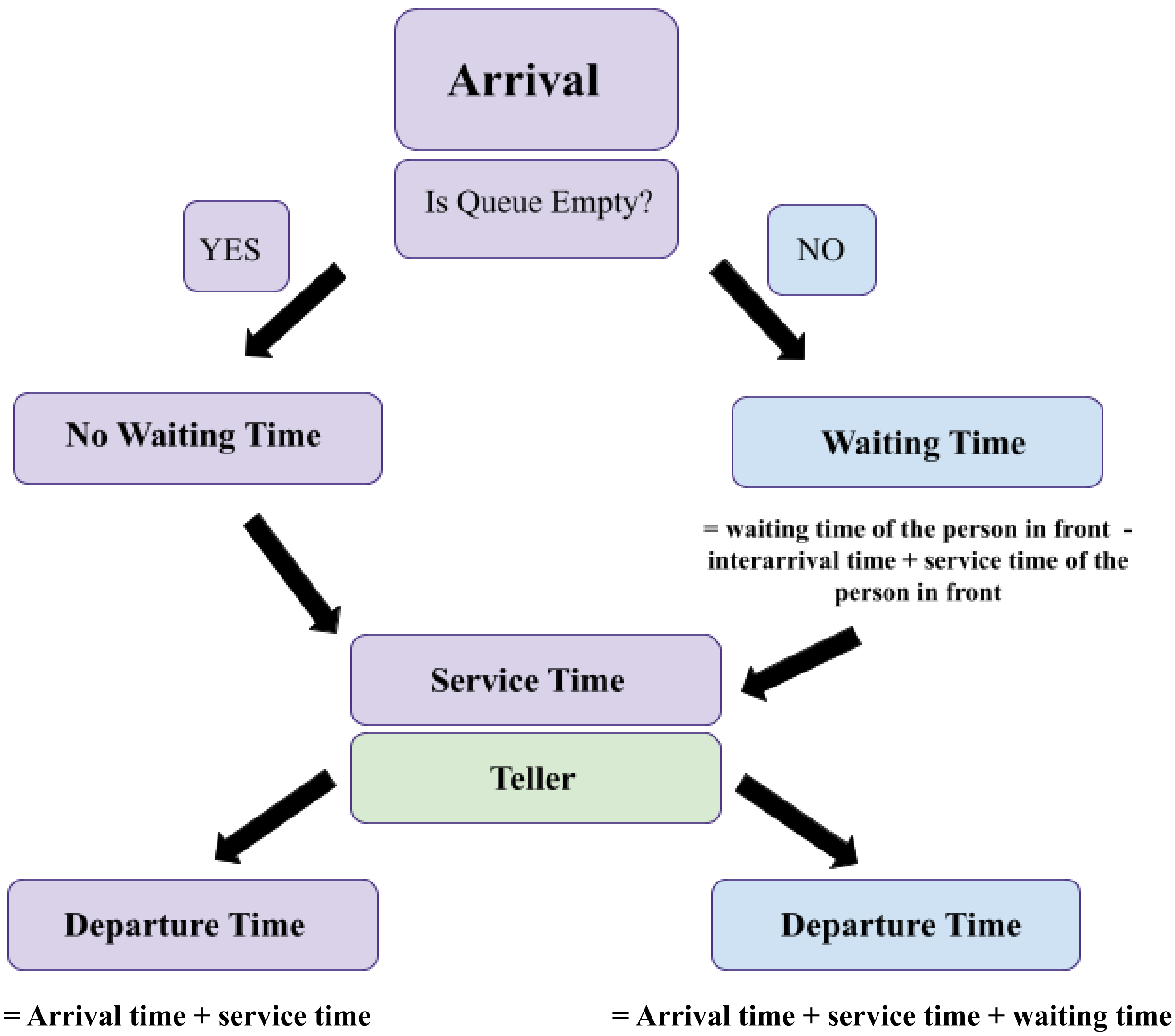
In order to find a solution to the queuing problem faced by banks, we have tried to apply some fundamental concepts of queuing theory. Queuing theory is basically a mathematical approach used for the analysis of waiting times. Using queuing theory, we have tried to develop and compare between two queuing systems with the aim to choose the most efficient model that reduces customer waiting time and increases the number of customers that can be served.

We started to simulate both queuing models using a small sample size. Before increasing the number of customers served, we wanted to make sure that the mathematics and logic behind the formulas that we have used for arrival, waiting, service and departure times were actually correct. After making sure of the relevance of the output obtained, we started to simulate on a wider scale. We encountered many problems with our codes especially with the for and while loops. We identified the mistakes and successfully debugged the code for our two models.

Illustrations of the algorithm of the queuing systems

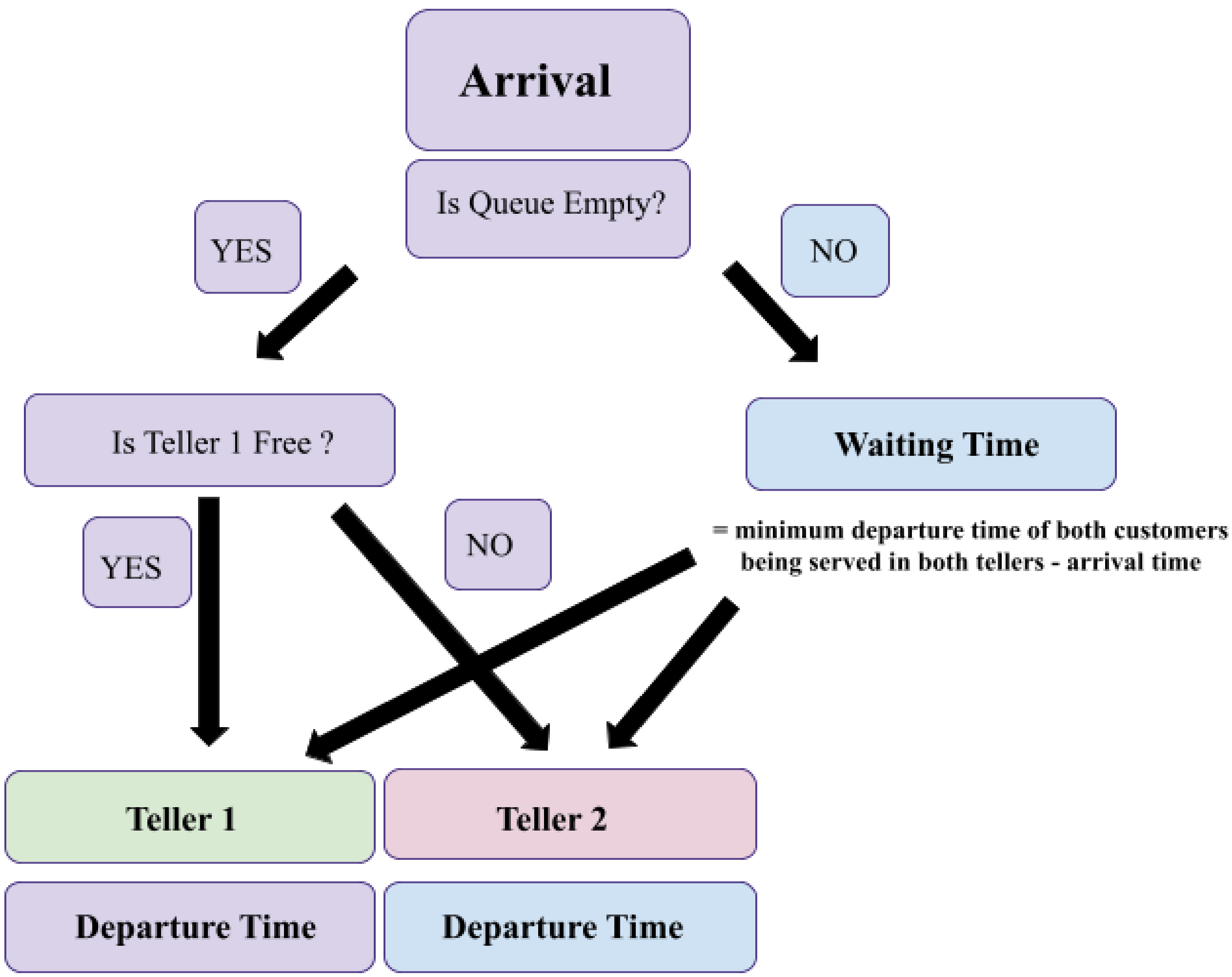
First Model :

The first model simulates the scenario where there is only one queue and one teller. The queuing model follows the first in first out concept, where the first customer that comes in gets served first and leaves. The challenge in this scenario is the long queues that might result in increased waiting times.



Second Model:

The second model simulates the scenario where there is one queue and two tellers that serve the respective customers. In this case the queueing depends on the first in first out concept. However, due to the multiple servers the departure time rather depends on the service times of the customers being served. The challenge in this scenario is the possibility of increased idle time of tellers or the long queuing which results in increased waiting times.



Description of the Variables

Variable Name	Description
n	Number of customers
interarrival_time	Time between successive customers entering the bank
service_time	Time spent by customer being served by the teller
next_waiting_time	Time spent by customer in the queue waiting to be served
next_arrival_time	Random exponentially distributed variable that indicates the arrival times of customers
arrival_times	Symbolizes the clock, which is the cumulative next arrival times of customers.
next_departure_time	Time where the customers leave the bank.
aqueue	Average time that a customer spends in the queue
bank	Average time customer spends in the bank whether in the queue or at the teller
nqueue	Average length of the Queue
total_idle_time	Total time where the tellers were free and unproductive
teller_num	Indicates which teller served the customer
queue	Number of customers in the queue

Code

```
#Katia Gabriel (900202272) & Farida Simaika (900201753)

#1-Teller, 1-Queue:

n=100
set.seed(20394)
interarrival_time = rep(0,n)
service_time = rexp(n,1/5)
next_waiting_time = rep(0,n)
arrival_times = rep(0,n)
next_arrival_time = rexp(n,1/3)
next_departure_time = rep(0,n)


next_waiting_time[1]=0
next_departure_time[1]=service_time[1]+arrival_times[1]



for (i in 2:n)
{
  arrival_times[[i]] = arrival_times[i-1]+next_arrival_time[i]
  interarrival_time[i]= arrival_times[i]-arrival_times[i-1]
  next_waiting_time[i] = next_departure_time[i-1]-arrival_times[i]
  if(next_waiting_time[i]<0)
  {
    next_waiting_time[i]=0
  }
  next_departure_time[i]=arrival_times[i]+next_waiting_time[i]+service_time[i]
}
round(arrival_times)
round(service_time)
round(next_waiting_time)
round(next_departure_time)

#statistics:

#1. Average Time that a customer spends in the Queue:

aqueue = sum(next_waiting_time, na.rm = TRUE)/n

#2. Average Time that a customer spends in the Bank whether in the Queue or at the Teller

bank = sum(next_waiting_time, na.rm = TRUE)/(n+((sum(service_time, na.rm = TRUE)/n)))

#3. Average length of the Queue

nqueue = ceiling(sum(next_waiting_time, na.rm = TRUE)/sum(next_arrival_time, na.rm = TRUE))
```

Results of the first model:

The model assumes that the bank welcomes 100 customers per day to be served. The parameters lambda for the exponential distribution were chosen to be ⅕ and ⅓ for randomly generating the service time and the next arrival time, respectively. The logic behind choosing the service rate to be less than the arrival rate is to guarantee the existence of a queue, as customers will be entering more frequently than the tellers finishing up serving customers which leads them to wait in the queue until their number is called. In light of the above and after executing the later code snippet, we reached the following findings:

1. The average time that a customer spends on the queue is around 140 minutes which is around 2.3 hours.
2. The average time that a customer spends in the bank whether on the queue or at the teller is around 132 minutes which is around 2.2 hours.
3. The average length of the queue is around 43 customers.

It is evident from the above findings that this model causes high waiting times. A customer visiting the bank for maybe a service that is not going to take more than 5 minutes has to stay on average around 2 hours until being served. Moreover, the bank is always crowded as the queue has on average 43 customers which makes the environment unwelcoming and unfavorable. With no doubt these circumstances increase customer's dissatisfaction and put an enormous threat on the bank's work flow and management. Therefore, this model is adverse and needs development. Accordingly a second model was simulated that increases the number of tellers.

Code

```
#2-Tellers,1 Queue:

n=100
set.seed(20394)
service_time = rexp(n,1/5)
next_arrival_time = rexp(n,1/3)
interarrival_time = rep(0,n)
next_waiting_time = rep(0,n)
next_departure_time = rep(0,n)
arrival_times = rep(0,n)
next_departure_times = rep(0,n)
queue = 0
teller1 = rep(0,n)
teller2 = rep(0,n)
teller_num = rep(0,n)
teller1_free = rep(0,n)
teller2_free = rep(0,n)
teller1_occupied = rep(0,n)
teller2_occupied = rep(0,n)
total_idle_time_=rep(0,n)

for (i in 1:(n))
{
  arrival_times[[i+1]] = arrival_times[i]+next_arrival_time[i+1]
  next_departure_times[[i+1]]=next_departure_times[i]+next_departure_time[i+1]
  interarrival_time[i+1]= arrival_times[i+1]-arrival_times[i]
}

next_waiting_time[1]=0
next_waiting_time[2]=0
next_departure_time[1]=service_time[1]
next_departure_time[2]=service_time[2]+arrival_times[2]
teller1[1]=1
teller2[1]=1


for(i in 3:n)
{
  if(arrival_times[i]>min(next_departure_time[i-1],next_departure_time[i-2]))
  {
    queue = 0
    if(arrival_times[i]>next_departure_time[i-2])
    {
      teller1[i]=1
      teller1_occupied[i]=arrival_times[i]
      teller1_free[i]=arrival_times[i]+service_time[i]
      next_waiting_time[i]=0
      next_departure_time[i]=arrival_times[i]+service_time[i]
      teller_num[i]=1
```

```
    }else if(arrival_times[i]>next_departure_time[i-2])
    {
      teller2[i]=1
      teller1[i]=1
      teller2_occupied[i]=arrival_times[i]
      teller2_free[i]=arrival_times[i]+service_time[i]
      next_waiting_time[i]=0
      next_departure_time[i]=arrival_times[i]+service_time[i]
      teller_num[i]=2
    }
  }else
  {
    queue = queue +1
    next_waiting_time[i] = min(next_departure_time[i-2],next_departure_time[i-1])-arrival_times[i]
    next_departure_time[i]= arrival_times[i]+next_waiting_time[i]+service_time[i]
    if((next_departure_time[i-1]-arrival_times[i])>(next_departure_time[i-2]-arrival_times[i]))
    {
      teller1[i]=1
      teller2[i]=1
      teller_num[i]=1
    }else{
      teller1[i]=1
      teller2[i]=1
      teller_num[i]=2
    }

  }
}

for (i in 2:n){

  total_idle_time=next_departure_time[i]-interarrival_time[i+1]


}

#1. Average Time that a customer spends in the Queue:

aqueue = sum(next_waiting_time, na.rm = TRUE)/n

#2. Average Time that a customer spends in the Bank whether in the Queue or at the Teller

bank = sum(next_waiting_time, na.rm = TRUE)/(n+((sum(service_time, na.rm = TRUE)/n)))

#3. Average length of the Queue

nqueue = ceiling(sum(next_waiting_time, na.rm = TRUE)/sum(next_arrival_time, na.rm = TRUE))
```

Results of the second model

The model also assumes that the bank welcomes 100 customers per day to be served. The parameters lambda for the exponential distribution were kept to ⅕ and ⅓ for randomly generating the service time and the next arrival time respectively to be able to compare both models. In light of the above and after executing the later code snippet, we reached the following findings:

1. The average time that a customer spends on the queue is around 4 minutes.
2. The average time that a customer spends in the bank whether on the queue or at the teller is around 3.5 minutes.
3. The average length of the queue is 2 customers.
4. The idle time of the tellers is around 2 hours.

It is obvious that the waiting times are reasonable. A customer visiting the bank will almost be served right away. Moreover, the bank is welcoming its customers in a structured environment with no crowdedness. With no doubt these circumstances increase customer's satisfaction and ease the bank's work flow and management. However, the problem with this model is that the idle time of the tellers is huge. One way to deal with this problem is to avail the second teller only during rush hours which is around afternoon and keep only one teller active early in the morning and late at night.

Comparison between both Models

Upon comparing both models we can confidently say that the second model that involves two tellers guarantees optimum bank effectiveness. In the presence of a second teller, waiting times decreased drastically from 2.2 hours to almost 3 minutes. In addition, having multiple tellers has significantly improved the average queue length decreasing it from an average of 43 customers to only 2 customers. This well-managed queue will undoubtedly have a tremendous impact on customer satisfaction and the bank's management efficiency. It is worth noting that the second model that involves two tellers has a major problem which is the idle time. However, it is clear that the benefits of having a second teller outweigh the risks of idle time.

Conclusion:

Collectively, this project was devoted to the application of queuing theory using two models involving different numbers of tellers. Through simulation, we were able to understand and determine how to model a multiple-server queue. A multiple-server queue made it possible to decrease waiting times and hence the average number of people standing in line. By decreasing waiting times, we were able to maximize the operational effectiveness and efficiency of the bank.

It is worth mentioning that the above models were simulated after assuming two parameters for lambda, yielding the results and findings mentioned above, however the generalizability of this simulation would be problematic, stating that a 1-Queue 2-Teller Model is the optimum solution. It is of huge importance to analyze the situation carefully and observe the trends in arrival times, service times and rush hours to be able to choose the appropriate parameters for the exponential distribution to successfully simulate a model that fits the respective queuing problem at hand. Queues have a major impact on customer satisfaction. Therefore, a well-managed queue is a major competitive advantage that a bank can have over its competitors. Managers should prioritize the analysis of queues and customer flow for optimum effectiveness.