# CSE 6010 Final Report

# A Discrete Event Simulation of Waiting Time at a Fast Food Restaurant

Tianyi Zheng
Yu Zheng
Xiaoyu Yang

## Introduction

As the name suggests, this is a discrete time simulation of the wait time of a fast food restaurant. Traditionally, a fast food restaurant can be broken down into three different stations: the servers, the cashiers and the cooks. One such example is the Panda Express at the Georgia Tech Student Center. The wait time at this restaurant is highly dependent on the time of day. We aim to provide the restaurant managers, as well as the customers, information regarding the estimated wait time based on different levels of traffic, which can be represented as the length of the waiting queue. For an example, a low traffic scenario may consist of 0.5 incoming customers per minutes and a high traffic scenario may consist of 2 incoming customers per minute. Thus, the simulation will be developed using the structure type of a discrete time event that consists of an application side and a simulation engine. The application schedules different events based on a time stamp and an event type. And the simulation engine stores the events using a priority queue. This project implements 6 different event types and using 6 event handlers and 12 states. The simulation results are visualized to show the change of several statistical measurements, such as the average wait time, against different levels of traffic.

## Project Architecture

This project involves two core parts: simulation engine and simulation application. The application part involves a FIFO (first in first out) queue and a state machine. The two parts communicate using an interface header file.

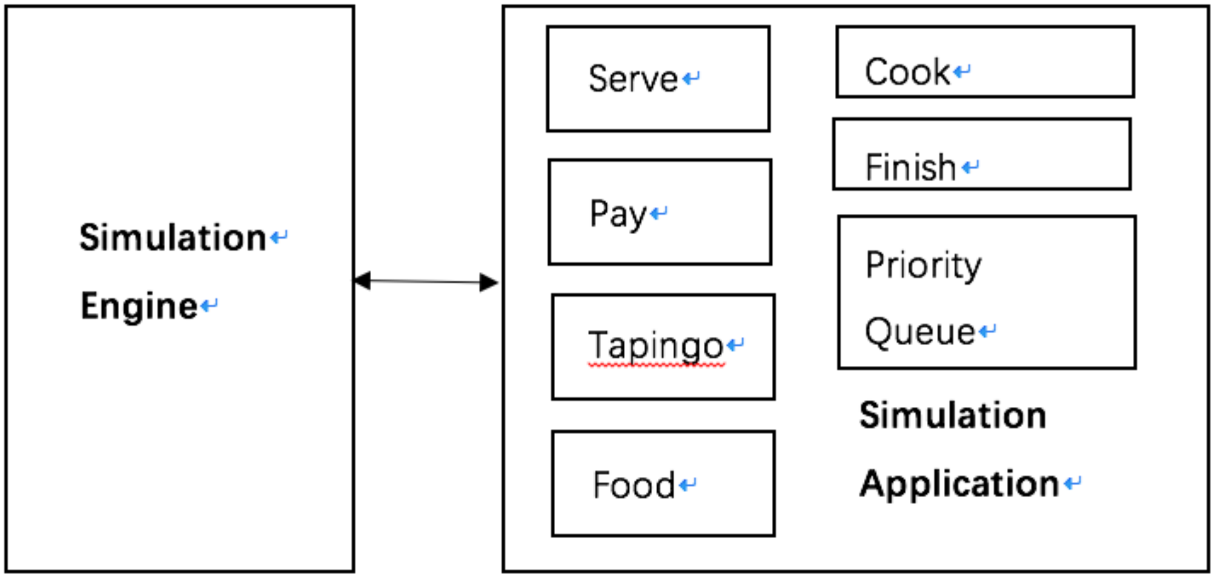


Figure 1. Project Architecture

**Simulation Application**

The application was implemented with 12 states and 6 event handlers. Its states are based on the panda express model. Since customers are served in the order of their arrival time, three FIFO queues were used to represent the waiting lines of customers: serving, paying and waiting-for-food queue. Each node in the queue represents a customer. It records the customers' attributes, such as payment type (paying by BuzzCard is quicker) and waiting time. The performance of this system is measured with the statistics of the waiting time.

The incoming customers' schedule are constructed using a Poisson distribution while varying the incoming rate of customers during normal operations and peak times. All of the data for setting up the simulation, such as the incoming rate and the average serving durations, were collected from panda express itself.

First of all, the application has the serve and pay functions. There are two types of staffs who can serve the customers: the server and the cashier. We implemented a way to allow the servers and cashiers to help each other based on the current situations, which means they will always be scheduled a task based on the current situation in the waiting queues. It was found that the way these staffs are assigned tasks and the total number of them greatly affect the performance of this system. The serve handler can also schedule different pay events based on the customer's information such as paying method and type of staff. The pay handler handles the paying and schedule the finishing event for when the customer leaves the restaurant based on the payment methods.

A Tapingo(an online food ordering service) event handler was also implemented for people who orders food online. Unlike normal customers, these customers only need to be scheduled serving event. It was found that they do not add much pressure to the performance of the system.

Another case when a waiting queue develops is when food has ran out. This is checked and handled in the serve state with the food and cook handlers. We assume everyone select the food randomly and checks against the remaining portion and available cooks. If the remaining portion is below a threshold and there is a cook available, a food event is scheduled. The cooking time is assumed to be the average of the cooking time of the entire menu.

Finally, the finish event is scheduled when a customer leaves the restaurant after paying the cashier or has received a Tapingo order. The finish handler also checks the cashier queue and tries to schedule pay event in the future.

**Simulation Engine**

The engine is implemented using a heap data structure. Heap is a complete binary tree, and every node's timestamp is less than or equal to that of each child node. It supports two main operations: delete and insert. For the delete operation, it first replaces the root with the last node in the last level. After that, it migrates the root down the tree as far as necessary to maintain heap property. For the insert operation, it adds the new node to the next available place in the last level. Then, it moves the new node up to restore the heap property. The heap structure used in this study is shown as Figure 2.

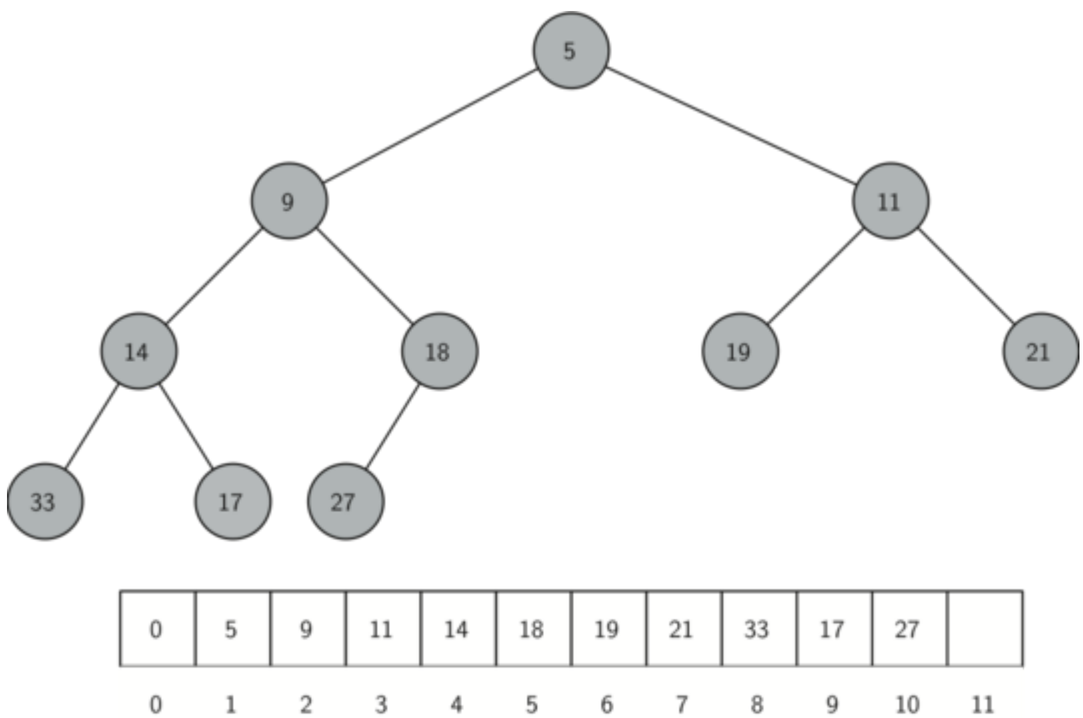


Figure 2. Heap Structure

In this project, each insert operation represents a new customer arriving and waiting in the queue. When a customer get served and leaves the waiting queue, we do a delete operation.

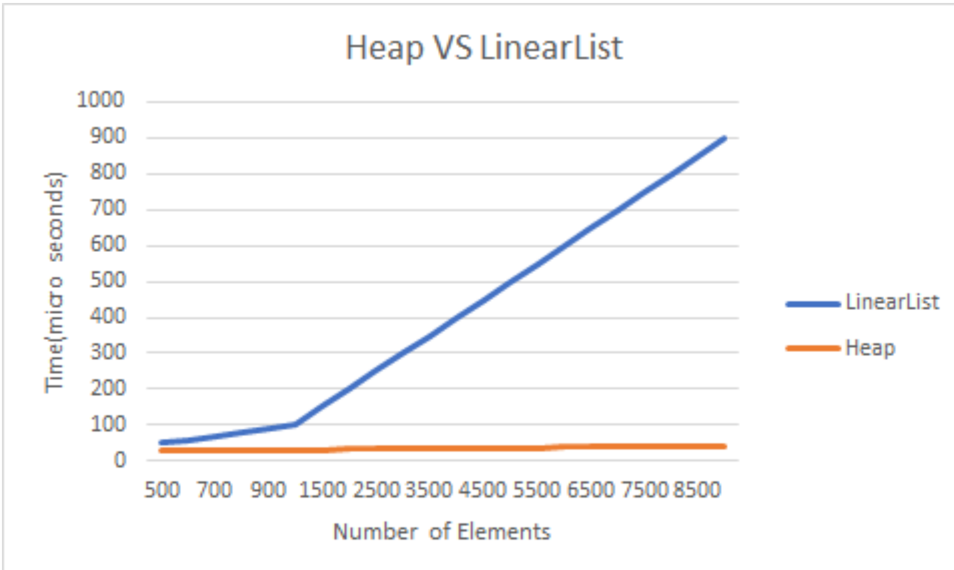


Figure 3. Performance comparison

We tested the performance of the engine by examining its hold time versus the number of original elements stored in the queue. The hold time is defined to be the time for putting one element into and getting one element out of the queue. It shows that the hold time increases linearly as the queue size grows in the linear list while increasing in a logarithmic way in the heap. In conclusion, the heap is an efficient data structure for the implementation of simulation engine.

**Existing research and date collecting**

Waiting time is especially important in fast food restaurants. Due to this reason, great number of studies have proposed various models and methods to simulate and reduce waiting and serving time for fast food restaurant. Jones and Peppiatt (1996) identified that the key aspects of queue management with regards to waiting time is that the wait times should be less than five minutes for most of the express restaurants. Dharmawirya et.al (2012) conducted a series of surveys to analyze the expected and actual waiting time in fast food restaurants. The queueing model he used has Poisson distributed arrivals, exponentially distributed service time. In this study, we conducted a survey at the Panda Express in Georgia Tech to collect realistic data during various time periods for a single day. The data collection sheet used for this study is shown as Table 1.

Table 1. Data Collection Sheet

| ID | Arrival Time | End Waiting | Begin Service | End Service | Payment Method | Whether Try Sample |
|---|---|---|---|---|---|---|
| | | | | | | |

Based on the collected data, the notations required for queueing models which are proposed by Kendall (1953) are summarized as followings:

a = Arrival (or inter arrival) distribution. We use Poisson distribution for the arrival distribution and use mean $\lambda$ to control the incoming rate of customers.
b = Service time distribution. We use random value between maximum and minimum service time to simulate the service time distribution.
c = Number of service channels in the system. There are two types of service, which are serving and paying. The number of servers for each service type are the optimization parameters in this study.
d = Service discipline. The service discipline is described in the application part.
e = Maximum number of customers allowed in the system. There are no certain limits for the maximum number of customers in this study.

Except for the notations required by Kendall's model (1953), detailed queueing and service information are also collected by using the data collection sheet. The main parameters for this study are listed as Table 2.

Table 2. Average values for parameters

| | Food Service | | Payment | | | |
|---|---|---|---|---|---|---|
| Type | 3 Servers | 2 Servers | Buzz Card | Visa | Cash | Tapingo |
| Average ServiceTime/s | 20 | 30 | 25 | 36 | 60 | 5 |

**Analysis and optimization**

To test the accuracy of the proposed simulation method, we conducted a series of tests to simulate the food service of the Panda Express with both low and high incoming rates. We used a standard employee deployment strategy for our test, where there are 3 servers and 1 cashiers. The incoming rate during the peak time was 0.1, while the rate during normal service hour was set to 0.02. For each test case, total 100 minutes were simulated. The test results are shown in Table 3.

Table 3. Simulated results with default employee deployment strategy (3 servers, 1 cashier)

| | | Average Waiting Time (min) | Total Serve Number per 100 Min |
|---|---|---|---|
| High Coming Rate | Simulation Result | 28 | 339 |
| | Real Situation | 24 | >250 |
| Average Coming Rate | Simulation Result | 2 | 121 |
| | Real Situation | 4 | 100 - 200 |

By analyzing the average waiting time caused by serving and paying, we found that customers spent much more time waiting for paying rather than food serving. As a result, the main reason for the long waiting time is the limited number of cashiers. As a result, in order to increase the service efficiency and reduce waiting time, we need to increase the number of cashiers.

The first strategy is to transfer one of the servers as a cashier. This will cause longer serving time. The second strategy is to increase one more cashier while keep the original number of servers. This strategy will cause the restaurant to pay more to hire new employee. Both alternatives and original strategy were simulated with peak incoming rate for 100 minutes' duration. The test results are shown as Figure 4 and Figure 5.

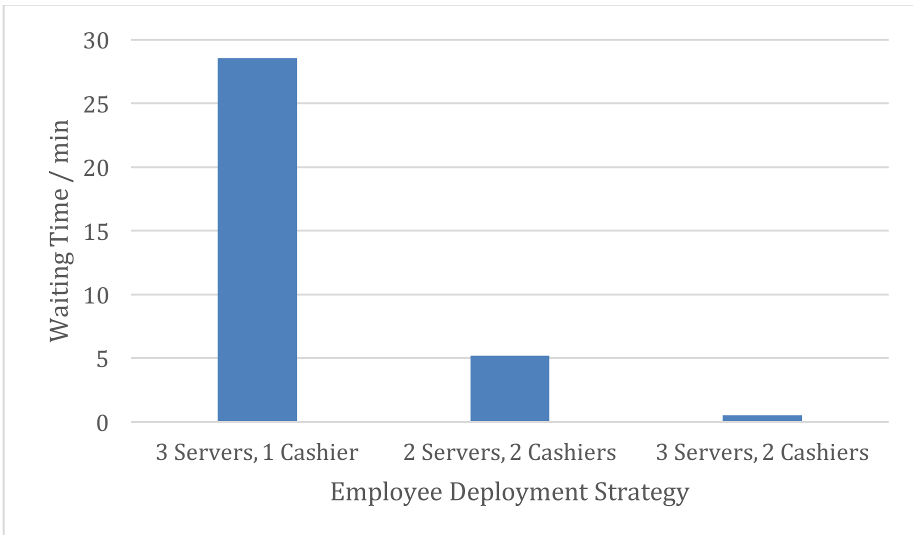


Figure 4. Average Waiting Time for Various Employee Deployment Strategy

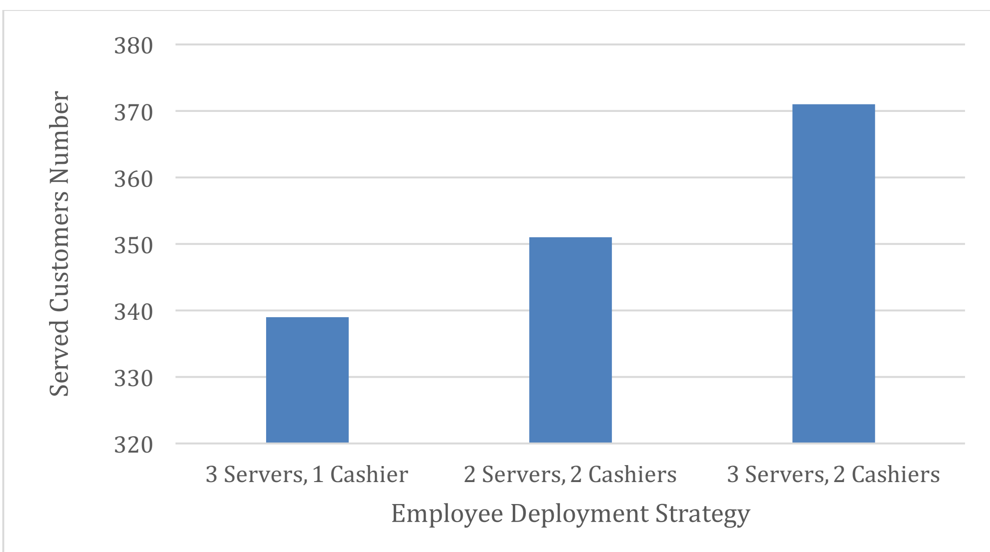


Figure 5. Served Customer Number for Various Employee Deployment Strategy

It is clear that by increasing the number of cashier, the Panda Express can increase total serving customers while reduce the average waiting time for each customer during peak time.

## Future Improvements

Several aspects of the simulation application can be further improved to better conform to reality. First, a stochastic generation of more customer attributes can help increase randomness. Second, the application can implement serving stations where each server is in charge of some specific types of food. And third, treat each type of food uniquely by gathering information on the consumption amount can better help with the scheduling of cooking. Furthermore, more data can be collected to validate the model.

## Conclusions

By comparing the simulated results and the real condition, we found out that our simulation model can represent the real situation with limited error. Meanwhile, by analyzing the average waiting time and served customers, we find out that the current deployment strategy for employee is not optimal and the payment process is the threshold. By switching one employee from serving to cash desk can reduce both average waiting time and increase the number of served customers.

## Reference:

Jones, P. and Peppiatt, E. (1996). "Managing perceptions of waiting times in service queues." International Journal of Service Industry Management, Vol. 7 No. 5, pp. 47-61.

Dharmawirya, M., Oktadiana, H., & Adi, E. (2012). Analysis of expected and actual waiting time in fast food restaurants.

Kendall, D. G. (1953). Stochastic processes occurring in the theory of queues and their analysis by the method of the imbedded Markov chain. *The Annals of Mathematical Statistics*, 338-354.