Simulations and Queueing Theory: The Effects of Randomly Bypassing Security

Emily Ortmann
Dakota State University

Follow this and additional works at: https://scholar.dsu.edu/theses

Part of the Other Applied Mathematics Commons

Recommended Citation
https://scholar.dsu.edu/theses/314

This Thesis is brought to you for free and open access by Beadle Scholar. It has been accepted for inclusion in Masters Theses & Doctoral Dissertations by an authorized administrator of Beadle Scholar. For more information, please contact repository@dsu.edu.
SIMULATIONS AND QUEUEING THEORY: THE EFFECTS OF RANDOMLY BYPASSING SECURITY

EMILY ORTMANN

Abstract. We discuss queueing theory in the setting of airport security and customs. By developing queueing simulations based on mathematical models, we explore a variety of questions related to optimal queue design with respect to efficiency, feasibility, priority, and other prescribed/variable constraints.

1. Introduction and Examples

1.1. Introduction. Queueing theory, the mathematical study of waiting lines, is an extremely useful tool for analyzing and designing operational models. The performance characteristics of these models (e.g., average wait time or queue length) are utilized by users (system designers/business executives) to improve processes and keep systems running smoothly and cost effectively. In particular, queueing models can be used to predict or evaluate a system’s performance and may differ depending on the users goals and needs. For example, a business that chooses to prioritize profit over customer satisfaction may operate under different assumptions than a business that values their customers’ experience.

A queueing model typically looks at studying average customer wait time, average time customers spend in service, average queue length, and average time servers spend idle. To help decrease the time customers spend in service, a business may consider introducing the possibility that a customer may bypass the queue or a server group. Other modifications may include various priority statuses or multiple servers/service rates.

In this research we study security and customs lines at an airport. As customers arrive, they wait in a single line until the next security server is available. The person at the front of the line then goes into service. Our models and simulations implement the possibility that a customer may bypass security by including a ‘button push’ before customers enter service. The button push is random with a set probability, \( P(G) \), that a customer will bypass security, essentially operating as random selection. This allows some customers at the airport to move past security, in return opening up space for the remaining customers. Denoting the average number of customers entering the system by \( \lambda \) and the average number of customers serviced by \( \mu \), we construct a cost function (1)

\[
C = c\lambda E(W) + s\lambda P(G)
\]

that allows us to value the wait time of our customers, \( c \), to the security risk of a passenger bypassing security, \( s \). In particular, we are interested in determining an optimal probability, \( P(G) \), that minimizes this cost function (1).

1.2. Examples. Queueing theory can be applied to many different systems. The following example concentrate on different aspects of the queueing problem.
1.2.1. Airports. An airport contains many lines, from initial security to getting on the plane. In this application, customers stand in a single-file line for metal detectors and wait for the next available metal detector to open. Airports also implement the possibility to bypass a server group by using random selection, forcing some customers to go through extra security.

1.2.2. Hospitals. In a hospital application of queueing theory, customers will typically operate as one line. The first customer to arrive will be treated first. However, if someone with more severe symptoms arrives, the staff will treat him or her as quickly as possible - modeling a priority queueing system. Hospitals may also operate using an appointment system. This makes the amount of time customers spend in service fixed because the time interval is scheduled.

1.2.3. Grocery Stores. A grocery store models a system where each server has his or her own line. Customers choose which line to stand in, which adds another layer of variability to the system. Some grocery stores include faster lines for customers with a smaller number of items, giving the system two rates at which customers go through service. Moreover, grocery stores also change the number of registers they have open depending on the number of customers in the store/queue.

1.2.4. Telecommunications and Networking. In telecommunication and networking, wait time is very important. As technology develops, users expect immediate service when using a computer or phone. A queueing model in this field focuses on scheduling work so the users wait time goes unnoticed. This application may allow preemption, meaning one process can terminate another if it has a higher priority to give better service to the user.

1.2.5. Traffic Lights. Traffic lights focus on keeping a system balanced. The major street at an intersection is given a longer green light while allowing the minor street occasional windows to clear any queue that has formed. A queueing model can be used to find a break-even point between the length of one queue and the wait time of the other.

2. Probability Theory

2.1. Exponential Random Variables. In this section, we review needed facts from probability theory. A random variable is a function that associates a number with each point in an experiment’s sample space. We denote random variables by capital letters (e.g., $X$, $Y$, or $Z$).

For a discrete random variable, we denote the probability that $X = x_i$ by $P(X = x_i)$, the probability mass function (pmf) for the discrete random variable, $X$. For example, if $X$ denotes the outcome of a standard 6-sided die roll, then the probability mass function (pmf) for $X$ is $P(X = i) = \frac{1}{6}$, for $i = 1, 2, 3, 4, 5, 6$.

**Definition 1.** The cumulative distribution function $F$ for any random variable $X$ is defined by

$$F(x) = P(X \leq x).$$

In many applications, we allow events to occur at any time (e.g., inter-arrival times and service times). If a random variable, $X$, can assume all values on a given interval, then $X$ is called a continuous random variable. For continuous random variables, a density curve describes the overall pattern of the distribution.

**Definition 2.** A function $f$ is a probability density function for the continuous random variable $X$, defined over the set of real numbers, if

1. $f(t) \geq 0$ for all real $t$.
2. $\int_{-\infty}^{\infty} f(t) \, dt = 1$
3. $P(a < X < b) = \int_{a}^{b} f(t) \, dt$.

**Theorem 1.** Let $X$ be a continuous real-valued random variable with density function $f(x)$. Then the function defined by

$$F(x) = \int_{-\infty}^{x} f(t) \, dt$$

is the cumulative distribution function of $X$ and

$$F'(x) = f(x).$$
Proof. Let \( A = (-\infty, x] \). By definition of \( F \), we have
\[
F(x) = P(X \leq x) = P(X \in A) = \int_A f(t) \, dt = \int_{-\infty}^x f(t) \, dt.
\]
Moreover, the Fundamental Theorem of Calculus gives \( F'(x) = f(x) \).
\[\square\]

Definition 3 (Uniform Distribution). The density of the uniform distribution on \([0, 1]\) is given by
\[
f(t) = \begin{cases} 1 & 0 \leq t \leq 1 \\ 0 & t > 1, \end{cases}
\]
and the associated cumulative distribution function is
\[
F(x) = \begin{cases} x & 0 \leq x \leq 1 \\ 1 & x > 1. \end{cases}
\]

There are many models and applications where we want events to occur at random times and we will use a continuous random variable \( X \) to denote the time between events (e.g., arrivals and departures). In particular, we model these events with exponential random variables.

Definition 4. The density of the exponential distribution with parameter \( \lambda > 0 \) is given by
\[
f_\lambda(t) = \lambda e^{-\lambda t}, \quad t > 0.
\]
Moreover, the cumulative distribution function is given by
\[
F_\lambda(x) = P(X \leq x) = \int_0^x f(t) \, dt = 1 - e^{-\lambda x}, \quad x > 0.
\]

The number \( \lambda \) is a non-negative number whose reciprocal represents the average value of \( X \), that is \( E(X) = \frac{1}{\lambda} \) (see Theorem 2 below). For example, if the average time between arrivals (or departures) is 10 minutes, then \( \lambda = \frac{1}{10} \).

\[\text{Figure 2. Graphs of } y = f_\lambda(t) \text{ for } \lambda = 1, \lambda = 5.5, \text{ and } \lambda = 10.\]

\[\text{Figure 3. Graphs of } y = F_\lambda(t) \text{ for } \lambda = 1, \lambda = 5.5, \text{ and } \lambda = 10.\]

Theorem 2. Let \( X \) be an exponential random variable with parameter \( \lambda \). Then \( E(X) = \frac{1}{\lambda} \).

Proof. The definition of \( f_\lambda \) along with an integration by parts gives
\[
E(X) = \int_{-\infty}^\infty tf_\lambda(t) \, dt = \int_0^\infty t\lambda e^{-\lambda t} \, dt = -\frac{\lambda t e^{-\lambda t}}{2} \bigg|_0^\infty + \frac{1}{\lambda^2} = \frac{1}{\lambda}.
\]
\[\square\]
An additional feature of the exponential distribution is the following ‘memoryless property.’ Essentially, the following result says that the probability of an arrival in the next $h$ minutes is the same if there have been no arrivals for the last $t$ minutes or if an arrival just occurred.

**Theorem 3** (Memoryless property of the exponential distribution). Let $X$ be an exponential random variable with parameter $\lambda$. For non-negative $t$ and $h$,

$$P(X > t + h | X \geq t) = P(X > h).$$

**Proof.** We follow the proof of Lemma 1 in [6, Chapter 20]. By (4) we have

$$P(X \geq h) = 1 - P(X \leq h) = \int_{h}^{\infty} \lambda e^{-\lambda t} dt = -e^{-\lambda t}|_{t=h} = e^{-\lambda h}.$$

Thus,

$$P(X > t + h | X \geq t) = \frac{P(X > t + h \cap X \geq t)}{P(X \geq t)} = \frac{e^{-\lambda (h+t)}}{e^{-\lambda t}} = e^{-\lambda h} = P(X \geq h)$$

which completes the proof. \hfill \square

3. Theory of the M/M/1 Model

3.1. M/M/1 Model. In this section, we analyze the queueing model with exponential inter-arrival times (time between arrivals) with mean $\frac{1}{\lambda}$, exponential service times with mean $\frac{1}{\mu}$, and one server, the so-called M/M/1 queueing model. We follow the notation and presentation of [1] and [6]. For further information about M/M/$n$ queueing models, see [2], [3], and [4].

Throughout the section, we assume that

$$\rho = \frac{\lambda}{\mu} < 1,$$

which guarantees our queue length and system does not explode (see (7) below). If $\rho < 1$, then $\rho$ is called the occupation rate or server utilization, since it represents the fraction of time the server is working. (For multi-server systems, the occupation rate is given by $\rho = \frac{\lambda}{n\mu}$.)

3.2. Arrival and Departure Process and Steady-State Probabilities. For a given time $t$, we define the number of people in a queueing system as the state of the queueing system at time $t$. For $t = 0$, the state of the system is equal to the number of people initially present in the system. In our models and simulations, we typically take this to be zero, e.g., assuming there are zero people in security when an airport opens.

Let $p_n(t)$ denote the probability that at time $t$, there are $n$ customers in the system, $n = 0, 1, 2, \ldots$. Note that $p_n(t)$ is analogous to the $n$-step transition probability, $p_n(t_n)$, that denotes the probability that after $m$ transitions, a Markov chain will be in state $n$. Similar to Markov chains, for many queueing models, these probabilities will approach limiting or equilibrium probabilities. That is, as $t \to \infty$, $p_n(t) \to p_n$ where $p_n$ are the limiting or equilibrium probabilities. The behavior of $p_n(t)$ before the steady state is reached is called the transient behavior. In what follows, we analyze the behavior of a queueing system in which the steady state has been reached. For detailed analysis of the system’s transient behavior see, [3].

Most queueing systems with exponential inter-arrival times and service times can be modeled as a birth-death process. A birth-death process is a continuous-time stochastic process where the system’s state at any time is a non-negative integer. For queueing systems, a ‘birth’ is an arrival, a ‘death’ is a departure, and arrivals and departures are independent of each other.

Consider the following flow diagram where an arrow indicates possible transitions. A forward arrow (with rate $\lambda$) denotes a transition from state $n$ to $n + 1$ (an arrival) and a backwards arrow (with rate $\mu$) denotes a transition from state $n$ to $n - 1$ (a departure or service completion).

Notice that the flow from $n$ to $n + 1$ (the number of transitions per unit time from $n$ to $n + 1$) is equal to $\lambda p_n$ (arrival rate $\times$ percentage of time the system is in state $n$). By equating the in-flow
and out-flow between states, we see these steady state probabilities satisfy the following equations

\begin{align}
0 &= -\lambda p_0 + \mu p_1 \\
0 &= \lambda p_{n-1} - (\lambda + \mu) p_n + \mu p_{n+1}, \quad n = 1, 2, 3, ...
\end{align}

Using (4), we can express \( p_1 \) in terms of \( p_0 \), giving

\[ p_1 = \frac{\lambda}{\mu} p_0. \]

Substituting this relation into (5) for \( n = 1 \) gives

\[ p_2 = \left( \frac{\lambda}{\mu} \right)^2 p_0. \]

Moreover, a standard induction argument gives,

\[ p_n = \left( \frac{\lambda}{\mu} \right)^n p_0, \quad n = 0, 1, 2, ... \]

Since

\[ \sum_{n=0}^{\infty} p_n = \sum_{n=0}^{\infty} \left( \frac{\lambda}{\mu} \right)^n p_0 = p_0 \sum_{n=0}^{\infty} \left( \frac{\lambda}{\mu} \right)^n = p_0 \left( \frac{1}{1 - \frac{\lambda}{\mu}} \right), \]

and the probabilities \( p_n \) must satisfy \( \sum_{n=0}^{\infty} p_n = 1 \), it follows that

\[ p_0 = 1 - \frac{\lambda}{\mu}. \]

Notice if \( \frac{\lambda}{\mu} \geq 1 \), the infinite sum in (7) blows up. Thus, no steady-state distribution exists if \( \frac{\lambda}{\mu} \geq 1 \).

**Theorem 4.** In an \( M/M/1 \) queueing model with arrival rate of \( \lambda \) and service time of \( \mu \), the steady state probabilities are given by

\[ p_n = \left( \frac{\lambda}{\mu} \right)^n \left( 1 - \frac{\lambda}{\mu} \right). \]

### 3.3. Steady-States and Little’s Law

From Theorem 4, we can easily obtain explicit formulae for a variety of interesting values in the \( M/M/1 \) queueing model with arrival rate of \( \lambda \) and service time of \( \mu \). For ease of reading/referencing, we define key operating characteristics of a system by the following:

- \( E(L) \) = average number of customers present in the queueing system
- \( E(L_q) \) = average number of customers waiting in line
- \( E(L_s) \) = average number of customers in service
- \( E(W) \) = average time a customer spends in the system
- \( E(W_q) \) = average time a customer spends in line
- \( E(W_s) \) = average time a customer spends in service.

**Theorem 5.** In an \( M/M/1 \) queueing model,

\[ E(L) = \frac{\lambda}{\mu - \lambda} \quad E(L_q) = \frac{\lambda^2}{\mu(\mu - \lambda)} \quad E(L_s) = \frac{\lambda}{\mu} \]

**Proof.** We follow the arguments of Section 4 in [6, Chapter 20].

For an \( M/M/1 \) model where a steady state has been reached, the average number of customers in the system, \( E(L) \), is given by

\[ E(L) = \sum_{k=0}^{\infty} kp_k. \]
Applying Theorem 4, we obtain,

\[ E(L) = \left(1 - \frac{\lambda}{\mu}\right) \sum_{k=0}^{\infty} k \left(\frac{\lambda}{\mu}\right)^k. \]

Letting \( S = \sum_{k=0}^{\infty} k \left(\frac{\lambda}{\mu}\right)^k \), we see that

\[ \frac{\lambda}{\mu}S = \sum_{k=1}^{\infty} k \left(\frac{\lambda}{\mu}\right)^k = \left(\frac{\lambda}{\mu}\right)^2 + 2 \left(\frac{\lambda}{\mu}\right)^3 + 3 \left(\frac{\lambda}{\mu}\right)^4 + \ldots \]

Subtracting \( S \) and \( \left(\frac{\lambda}{\mu}\right)S \), we obtain

\[ S - \left(\frac{\lambda}{\mu}\right)S = \frac{\lambda}{\mu} + \left(\frac{\lambda}{\mu}\right)^2 + \left(\frac{\lambda}{\mu}\right)^3 + \cdots = \sum_{k=0}^{\infty} \left(\frac{\lambda}{\mu}\right)^k = \frac{\lambda}{1 - \frac{\lambda}{\mu}}. \]

Thus,

\[ S = \frac{\lambda}{\left(1 - \frac{\lambda}{\mu}\right)^2} \]

and

(9) \[ E(L) = \left(1 - \frac{\lambda}{\mu}\right)S = \frac{\lambda}{1 - \frac{\lambda}{\mu}} = \frac{\lambda}{\mu - \lambda}. \]

For \( E(L_q) \), we notice that if 0 or 1 people are in the system, then nobody is in line (all in service), and if there \( q > 1 \) people in the system, there are \( q - 1 \) people in line. Thus,

\[ E(L_q) = \sum_{k=1}^{\infty} (k-1)p_k = \sum_{k=1}^{\infty} kp_k - \sum_{k=1}^{\infty} p_k = \sum_{k=1}^{\infty} kp_k - \sum_{k=0}^{\infty} p_k + p_0. \]

Applying (8) and (9), we obtain

(10) \[ E(L_q) = \frac{\lambda}{\mu - \lambda} - 1 + \left(1 - \frac{\lambda}{\mu}\right) = \frac{\lambda^2}{\mu(\mu - \lambda)}. \]

Lastly, for the expected number of customers in service, we have

\[ E(L_s) = \sum_{k=0}^{\infty} p_k = \sum_{k=0}^{\infty} p_k - p_0 = 1 - \left(1 - \frac{\lambda}{\mu}\right) = \frac{\lambda}{\mu}. \]

Little’s law gives an extremely important relation between the means of the previous result and \( E(W), E(W_q), E(W_s), \) and \( \lambda \). For proof of Little’s Law, see [5].

**Theorem 6** (Little’s Law). For any queueing system in which a steady-state distribution exists, the following relations hold:

\[ E(L) = \lambda E(W) \quad E(L_q) = \lambda E(W_q) \quad E(L_s) = \lambda E(W_s). \]

Little’s law combined with Theorem 5 gives

(11) \[ E(W) = \frac{1}{\lambda} E(L) = \frac{1}{\lambda} \frac{\lambda}{\mu - \lambda} = \frac{1}{\mu - \lambda} \]

(12) \[ E(W_q) = \frac{1}{\lambda} E(L_q) = \frac{1}{\lambda} \frac{\lambda^2}{\mu(\mu - \lambda)} = \frac{\lambda}{\mu(\mu - \lambda)} \]

(13) \[ E(W_s) = \frac{1}{\lambda} E(L_s) = \frac{1}{\lambda} \frac{\lambda}{\mu} = \frac{1}{\mu} \]
3.4. Non-preemptive Priority. In this section, we briefly discuss the $M/M/1$ system with two different types of customers, e.g., type 1/VIP and type 2/Non-VIP. Type 1 and type 2 customers arrive at different inter-arrival rates, denoted $\lambda_1$ and $\lambda_2$ respectively. We keep the service times of all customers the same, that is all customers go through service with a service rate of $\mu$. To ensure the system does not explode, we assume that

$$\rho_1 + \rho_2 < 1,$$

where $\rho_i = \lambda_i / \mu$. In an $M/M/1$ queueing system with non-preemptive priority, type 1 customers have priority over type 2 customers. Priority means that when a type 1 customer arrives, he or she ‘jumps’ any type 2 customers waiting in line. (If type 2 customers are currently in service, then the type 1 customer must wait until space is available. Systems where type 1 customers are able to interrupt the service of type 2 customers are called preemptive priority queueing models.)

The mean wait time of type 1 customers, $E(W_1)$, is given by

$$E(W_1) = \frac{1}{\mu} E(L_1) + \rho_2 \frac{1}{\mu},$$

where $E(L_1)$ gives the mean queue length for type one customers. Here, the last term represents the situation when a type 1 customer finds a type 2 customer already in service ($\rho_2$ gives the fraction of time the server spends on type 2 customers).

By Little’s Law ($\lambda E(W) = E(L)$), we also have that

$$E(W_1) = \frac{1}{\lambda_1 + \lambda_2} E(L_1).$$

Combining this with (14) and solving for $E(L_1)$, gives

$$E(L_1) = \frac{(1 + \rho_2)\rho_1}{1 - \rho_1} = \frac{\lambda_1}{\mu} \frac{\mu + \lambda_2}{\mu - \lambda_1}.$$

Notice that

$$E(L_1) + E(L_2) = \frac{\rho_1 + \rho_2}{1 - \rho_1 - \rho_2}.$$

Inserting (15) and solving for $E(L_2)$ gives the mean queue length for type 2 customers,

$$E(L_2) = \frac{(1 - \rho_1(1 - \rho_2))\rho_2}{(1 - \rho_1)(1 - \rho_1 - \rho_2)}.$$

4. Simulations/ Background of Simulation

4.1. Events and Run Time. For many interesting queueing models, there is not formulae available to directly compute operating characteristics of a system (e.g., mean wait time, mean queue length). Often, attempts to use analytical models to study these values in a queueing model require substantial simplifying assumptions on the systems - often resulting in an inadequate model for implementation. In these cases, we utilize simulations to explore interesting modifications.

By assumption, queueing models are dynamic and we need to keep track of the changing state of our system. In queueing models, there are typically only two events that can change the state of our system, arrival or departure. The time these events occur are recorded in a variable, often called ‘Run Time’ or ‘Clock Time’. If we have an arrival, we randomly generate an inter-arrival time and add it to the current clock time to obtain an arrival time, i.e.,

$$\text{Arrival Time} = \text{Clock Time} + \text{Randomly Generated Inter-Arrival Time}.$$ 

Similarly, if the event is a departure, we randomly generate a service time and add it to the current clock time to obtain the departure time, i.e.,

$$\text{Departure Time} = \text{Clock Time} + \text{Randomly Generated Service Time}.$$ 

This technique of simulation is called the next-event time-advance mechanism.

Many programming languages have a built in command for creating random numbers with uniform distribution. Using the density function for exponential distributions, we can easily simulate exponential random variables. In particular, for $x > 0$, the function $F_\lambda^{-1}$ (defined in (2)) is a strictly increasing function with inverse given by

$$F_\lambda^{-1}(x) = -\frac{1}{\lambda} \ln(1 - x).$$
Thus, to simulate values of random variables with exponential distribution from random variables with uniform distribution, we simply compute the value of the expression

$$\frac{1}{\lambda} \ln(1 - \text{RND}),$$

where RND represents a random number (with uniform distribution) between 0 and 1.

4.2. M/M/1 Queuing Model with Priority. In the next section, we explore queueing models with the ability to randomly bypass service/security and discuss the simulations needed for this modifications. Below we briefly compare known theoretical wait times (see (15) and (16)) to simulated wait times in the M/M/1 queueing model with non-preemptive priority.

<table>
<thead>
<tr>
<th>$\lambda_1$</th>
<th>$\lambda_2$</th>
<th>$\mu$</th>
<th>$\rho$</th>
<th>$E(W_1)$</th>
<th>$E(W_2)$</th>
<th>Sim $E(W_1)$</th>
<th>Sim $E(W_2)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>11</td>
<td>12</td>
<td>0.917</td>
<td>DNE</td>
<td>0.916</td>
<td>0.916</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>10</td>
<td>12</td>
<td>0.917</td>
<td>0.083</td>
<td>1</td>
<td>0.083</td>
<td>0.996</td>
</tr>
<tr>
<td>5.5</td>
<td>5.5</td>
<td>12</td>
<td>0.917</td>
<td>0.141</td>
<td>1.692</td>
<td>0.141</td>
<td>1.696</td>
</tr>
<tr>
<td>10</td>
<td>1</td>
<td>12</td>
<td>0.917</td>
<td>0.458</td>
<td>5.5</td>
<td>0.458</td>
<td>5.520</td>
</tr>
</tbody>
</table>

Figure 5. Performance characteristics for M/M/1 with non-preemptive priority along with simulation data from 100 simulations each consisting of 100,000 arrivals.

5. (Randomly) Bypass Security

5.1. Bypass Security Simulation. In our simulations, we implement a probability that a customer can bypass security (see Figure 1.1). Customers enter the system at a rate of $\lambda$. When customers reach the front of the line, they push the green/red button. The button is set at a probability of being green, denoted $P(G)$. For example, if $P(G) = 0.3$, there is a 30% chance that the button will be green and a 70% chance that the button will be red. If the button is green, the customer bypasses security and exits the system. If the button is red, the customer will enter the server group. Lastly, customers will go through service at a rate of $\mu$. After customers have gone through service, they exit the system.

Figure 6. Visual representation of the simulation code. A diamond shows a logical decision and a rectangle shows an action performed.

Above is a diagram of the code simulation for the $M/M/n$ queueing model with the addition of a bypass security button. We start by checking if the system is empty. If the system is empty, we know the next action will be an arrival because there is no one in the system to depart. Every time there is an arrival, we check to make sure a server is available to work with the customer. If there
is not a server available, the customer is added to the queue. If a server is available, the customer pushes the green/red button. If the button display is green, the customer exits the system.

If the system is not empty, then we must check if the action was an arrival or a departure. If it is an arrival, we do the same steps as described above. If the action is a departure, we make another server available because they are no longer with a customer. Next, we check if there is someone in line. If there is no one in line, the simulation waits for another event. If there is a line, the customer at the front pushes the green/red button. If the button is red, the customer is put with the newly available server. If the button is green, the customer exits the system and we go back and check if there is still a line and repeat the process until the button is red or until there is no one in line. While there is time left, the simulation will repeat these steps.

5.2. Bypass Security Cost Function. The reason an airport would want to include a random probability that customers can bypass security is to limit the amount of time their customers are waiting and decrease the length of their lines.

Naturally, a business would like if their customers did not wait at all - making it ideal for the business to set the probability that a customer get a green button to 100%. Without an appropriate cost function, the system would tend to allow all customers bypass security, resulting in a wait time of 0. We build a cost equation that takes into account the cost of customers waiting, $c$, as well as security penalty for each green button push, $s$.

\[ C = c\lambda E(W) + s\lambda P(G) \]  \hspace{1cm} (17)

We can think of the security penalty as the amount of money an airport is willing to pay for the possibility of letting a security rick through in order to decrease the wait time.

In the following plots, we view the cost function, $C$, as a function of the VIP Threshold, $p = P(G)$. Each node represents the value of the cost function averaged over 1,000 simulations with arrival rates, average service times, and number of servers matching the morning arrivals to Minneapolis - St. Paul International Airport (MSP) on March 21th, 2018 (see https://awt.cbp.gov).

![Figure 7](image_url)

**Figure 7.** Cost as a function of $p = P(G)$ for $\lambda = 375$, $\mu = 30$, $n = 8$, and $T = 2$.

For the left graph, we keep the cost of a customer waiting fixed at $20 and let the security cost range from $5 to $20. These graphs all emerge from a single point (due to equal wait costs). Each graph then decreases before experiencing a sharp up-tick. In these situations we see varying probabilities that minimize our cost function. For example, for high customer waiting time cost and low security cost the optimal green/red probability occurs around 0.4. However, for low customer customer waiting time cost and high security cost the optimal green/red probability occurs around 0.3. For extremely high security costs the optimal green/red probability is 0 (i.e., everyone goes through security).

For the right graph, we keep the security cost fixed at $5 and vary the cost of a customer waiting. We see that once the green button probability results in near zero wait time (in this data set around $p = 0.4$), the graphs merge on each other (since the wait cost is no longer a substantial factor).
REFERENCES