Reducing Wait Times at Airport Security
减少机场安全等待时间

Sovijja Pou
Daniel Kunin
Daniel Xiang
COMAP Interdisciplinary Contest In Modeling
January 23, 2017
Brown University

翻译：周吕文

关注“数学模型”公众号，获取更多信息

[0]附录中所有代码均为本人（周吕文）根据论文和作者的提示重新编写（原文并没有附代码），并重新绘制了论文中的所有图。程序修正了部分原文的错误，加之模拟是随机的，因此部分图与原文有所差别。发现翻译错误或改进方案请发邮件告知。

Abstract

In this report, we develop and employ a queuing model for lines and servicing at TSA security in order to identify bottlenecks in the procedure and suggest improvements. We model the influx of passengers as a Poisson process, allowing us to sample from an exponential distribution to simulate inter-arrival times. To find probabilistic distributions on the service time spent in security stations, we take a nonparametric approach, using kernel density estimation to obtain probability densities from the given data. Using recurrence concepts from our queuing network, we develop a simulation method via dynamic programming as a means to verify the theoretical results and conduct further experiments. We apply these results to a cost-benefit analysis, which we use to identify the optimal allocation of TSA resources. Additionally, we consider different cultural norms regarding passenger interaction in queues, and add complexity in our model to account for them. These tools allow us to perform experiments, which are realistic within reason, and give suggestions for improvement in a manner justified by both the given and researched data.

在这份报告中，我们为运输安全机构（Transportation Security Agency, TSA）的安检通道和服务建立并运用了排队模型，以确定过程中的瓶颈并提出改进建议。我们用泊松过程模拟旅客的到达，这使我们能够从指数分布中随机抽样来模拟旅客到达的时间间隔。为了得到安检服务台服务时间的概率分布，我们采用非参数方法，使用核密度估计来从给定数据中获得概率密度。使用排队网络中的递推概念，我们通过动态规划建立了一种模拟方法，作为验证理论结果和进一步模拟的手段。我们将这些结果应用于成本收益分析，我们用这个分析来确定 TSA 资源的最优分配。此外，我们考虑不同的文化背景的旅客在队列中的相互影响，并增加我们模型的复杂性来解释它们。这些工具使我们能够在合理的范围内进行实际的模拟，提出改进建议并由给定的和研究的数据验证。

Summary

Passengers arrive at TSA security screening at seemingly random times, and the uncertainty in their arrival along with TSA inefficiencies can cause abnormally long and varying wait times. A major problem the agency must confront is the improvement of passenger throughput while maintaining a strict standard of security. Our solution consists of four main parts: a queuing model describing the security screening, a method of simulation for theory verification and experimentation, the application of these results to cost-benefit analysis, and modifications to the model to account for different traveler characteristics.

We decided that a reasonable measure for TSA inefficiency was the average time a passenger spent waiting in line. The queuing model was formulated by expressing the relationship between wait times in the queue recursively. We used kernel density estimation and exponential fitting to estimate distributions on the random variables involved in the model. We obtained explicit expressions for passenger wait times, which we used to reason about the system behavior as the number of passengers and arrival rates varied.

Using the above framework we developed a method for simulation that used dynamic programming to sample arrival times of passengers within the security system. The simulation numerically verified the results of the theory and allowed for further experimentation with varying parameters. The results of these experiments gave us insights on where the TSA procedure could be optimized and the passenger experience improved.

We applied cost-benefit analysis to the efficiency-cost tradeoff that TSA must consider when staffing employees and screening passengers. We found an optimal allocation of resources that maximized profit rate by plotting a profit objective function against the different configurations of TSA employees.

Finally, we considered distinctions between travelers and how they might affect the activity in the queuing system and each other's wait times as a result. We investigated alternative methods to order passengers in queue and found that this approach was not as successful as we expected.

Using a diverse array of techniques, we were able to successfully model the flow of passengers through airport security, identify bottlenecks in the screening process, and suggest improvements to the current procedure. As our world becomes more interconnected and airline activity increases, TSA will have to confront the issue of passenger throughput inefficiency. The use of mathematical models will undoubtedly become an invaluable tool in optimizing efficiency while maintaining passenger safety.

旅客以看似随机的时间到达 TSA 安全服务台，由于旅客这种到达时间的不确定性以及 TSA 安检效率的低下而有可能会导致较长且不确定的等待时间。TSA 机构必须面对的主要问题是提高旅客吞吐量，同时保持严格的安全标准。我们的解决方案由四个主要部分组成：一个描述安检的排队模型，一个可验证理论的测试模拟方法，将这些结果应用于成本效益分析，以及对模型进行修改以考虑不同旅客特征。

我们认为，TSA 效率的合理衡量标准是旅客排队等待的平均时间。排队模型是通过依次表示出旅客在队列中等待时间之间的关系而建立的。我们使用核密度估计和指数拟合来估计模型中涉及的随机变量的分布。我们获得了旅客等待时间的明确表达式，并以此推断旅客数量和到达率变化时系统的行为。

基于上述框架，我们建立了一种仿真方法，使用动态规划对旅客在安检系统内的到达时间进行抽样。该模拟以数值的方式验证了理论结果，并允许用对不同的参数进行进一步的测试。这些测试的结果为我们提供了 TSA 安检过程可以优化的地方以及改善旅客体验的建议。

我们将成本效益分析应用于 TSA 在员工聘用和旅客安检时必须考虑的效率-成本权衡。通过对 TSA 员工的不同配置绘制利润目标函数并最大化利润率，我们找到了一个最优化的资源配置。

最后，我们考虑了旅客之间的区别，以及他们如何影响排队系统的行为和彼此的等待时间。我们研究了旅客按照其它方式排队，发现这种方式并不像我们预期的那样成功。

通过多种方法，我们能够成功地模拟旅客通过机场安检的行为，找出安检过程中的瓶颈，并建议改进目前的安检流程。随着我们的世界相互联系变得更加紧密，航空公司的航班增加，TSA 将不得不面对旅客吞吐量低效的问题。数学模型的使用无疑将成为在保持旅客安全的同时优化效率的有效工具。

Contents

1 Introduction | 引言 **3**
- 1.1 Outline | 概述 . . . 3
- 1.2 Main Assumptions | 主要假设 . . . 3

2 A Queuing Model | 排队模型 **4**
- 2.1 Obtaining Probability Densities | 获得概率密度 . . . 6
  - 2.1.1 Exponential Fitting | 指数拟合 . . . 6
  - 2.1.2 Kernel Density Estimation | 核密度估计 . . . 7
- 2.2 Writing Down a Recurrence | 写出递推关系 . . . 10
  - 2.2.1 Definitions and Derivations | 定义和推导 . . . 10
  - 2.2.2 Average Wait Time Values | 平均等待时间 . . . 12

3 The Simulation | 模拟 **13**
- 3.1 A Dynamic Programming Method | 一种动态规划方法 . . . 13
  - 3.1.1 Pseudo code | 伪代码 . . . 15
- 3.2 Comparison to the Model | 与模型的比较 . . . 16

4 Evaluation/Results | 评价/结果 **17**
- 4.1 Bottlenecks in Passenger Throughput | 旅客吞吐量瓶颈 . . . 17
  - 4.1.1 ID Check Station | 身份检查服务台 . . . 17
  - 4.1.2 Scan Station | 扫描服务台 . . . 18
- 4.2 Variance in Passenger Wait Time | 乘客等待时间的方差 . . . 19
- 4.3 Modifications to the Model | 对模型的修改 . . . 20
  - 4.3.1 Cost-Benefit Analysis | 成本效益分析 . . . 20
  - 4.3.2 Sensitivity Analysis | 灵敏度分析 . . . 22

5 Improving the Model | 改进模型 **24**
- 5.1 Traveler Characteristics | 旅客特征 . . . 24
- 5.2 Different Queuing Disciplines | 不同的排队规则 . . . 24
- 5.3 Arrival Rate Regulation | 到达率调控 . . . 24
- 5.4 Improving Profit Measure | 提高利润措施 . . . 25

6 Conclusion | 结论 **25**

Appendices **27**

A Matlab code | Matlab 程序代码 **27**

1 Introduction | 引言

In this report, we present a model and simulation to optimize TSA security lines with respect to passenger throughput and variance in wait times. Specifically, we study the relationship between the number of ID check and scan stations and arrival rate to identify bottlenecks in the security process. We also define a cost measure to determine the optimal number of service stations. Further, we explore the implementation of a "virtual queue" and varying passenger traveling speeds and how they impact these bottlenecks. Our model provides the TSA valuable information for reducing wait time and variance in security lines, while maintaining the same standards of safety and security.

在这份报告中，针对在旅客吞吐量和等待时间差异方面，我们提出了一个模型和模拟来优化 TSA 安全过程。具体而言，我们研究了身份检查服务台和安检扫描服务台的数量与到达率之间的关系，以确定安检过程中的瓶颈。我们还定义了一个成本度量来确定服务台的最佳数量。此外，我们研究了“虚拟队列”的执行并改变旅客的到达率，以及它们如何影响这些瓶颈。我们的模型为 TSA 提供了有价值的信息：在保持相同的安检标准下减少安检流程的等待时间及方差。

1.1 Outline | 概述

Our objective is to develop and implement a model to explore the flow of passengers through airport security, identify bottlenecks in the current process, and suggest modifications. To realize this objective we will proceed as follows:

- Formulate a queuing model to explore passenger wait times as a measure of inefficiency.
- Implement a simulation method to verify the theoretical framework and perform experiments to gain further insights.
- Analyze simulation results to identify the optimal allocation of resources according to a cost-benefit analysis.
- Make modifications to the model based on cost-benefit analysis and cultural differences. Incorporate more complexity in the model to account for these considerations.

我们的目标是建立一个模型并应用该模型研究乘客通过机场安检的流程，找出当前过程中的瓶颈，并提出改进建议。为了实现这一目标，我们将按以下步骤进行：

- 建立一个排队模型，研究乘客等待时间（乘客等待时间是安检效率的一个衡量标准）。
- 执行模拟方法来验证理论框架并进行测试以获得更深层次的见解。
- 根据成本效益分析，分析模拟结果以确定资源的最佳分配。
- 根据成本收益分析和文化差异对模型进行修改。在模型中加入更多的复杂性来考虑这些因素。

1.2 Main Assumptions | 主要假设

Queuing systems are probabilistic in nature and as a result are difficult to describe and predict. In order to quantify the uncertainty in waiting times, it is necessary to make some simplifying assumptions. Listed below are several of the main assumptions we make regarding how we view the problem. Throughout the remainder of this paper we introduce additional assumptions as they become

排队系统本质上是概率性的，因此难以描述和预测。为了量化等待时间的不确定性，有必要做一些简化的假设。下面列出了我们就如何看待问题所做的几个主要假设。其他额外的相关假设，我们将会在本文用到的部分再介绍。

relevant.

- Transit times between stations and queues are negligible. We assume that the time spent moving from a queue to a station or a station to a queue is negligible. This is a reasonable assumption in that most TSA checkpoints are relatively compact.
- Queues operate on the “First In First Out” (FIFO) discipline. This principle states that the first person to line up in a queue will be the first person to be served.
- All employees are identical in terms of competency at their job. We assume that all TSA employees are equally capable. This isn’t entirely realistic because some workers are better at their jobs than others; however, this distinction doesn’t serve to provide any further meaningful insights into the problem.
- Passenger wait times are an accurate measure of throughput efficiency. Large passenger wait times in queues are the result of inefficiencies in the TSA screening process. Passenger throughput (given as passengers per unit time) is inversely proportional to time spent in the system by each passenger. Thus, maximizing passenger throughput is essentially the same as minimizing wait time.

- 服务台和队列之间的步行时间可以忽略不计。我们假设从一个队列移动到一个服务台或一个服务台移到一个队列的时间是可以忽略的。这是一个合理的假设，因为大多数 TSA 安检服务台都比较紧凑。
- 队列在“先到先服务”（FIFO）规则下运行。这个原则规定第一个到达队列旅客是将第一个接受服务。
- 所有员工的工作能力都是一致的。我们假设所有 TSA 员工都有同样的能力。这不是完全现实的，因为有些员工的工作比其他员工出色，然而，这个区别并不能为这个问题提供任何更有意义的见解。
- 乘客等待时间是吞吐量效率的准确度量。较长的乘客排队等待时间是 TSA 安检过程效率低下的结果。旅客吞吐量（以单位时间内旅客的数量计算）与每位乘客在系统中所花费的时间成反比。因此，旅客吞吐量的最大化基本等价于等待时间的最小化。

2 A Queuing Model | 排队模型

We model the airport security line as 2 nearly-identical parallel queuing networks with the FIFO discipline depicted below. The network with green entities corresponds to the TSA precheck lane and the network with blue entities corresponds to passengers in the regular lane. Both networks are comprised of a queue, Q_1, leading to a server S_1 (ID check station), leading to another queue Q_2, leading to another server S_2 (scan station, which includes mm wave scan, carry on screening, and extra screening if necessary). A pre-determined fraction of people will queue in the precheck lane while the rest will queue in the regular lane.

我们将机场安检流程模拟为 2 个几乎相同的并行 FIFO 排队系统，其中 FIFO 规则如下所示。具有绿色实体（圆）的排队系统对应于 TSA 预检通道，具有蓝色实体（圆）的排队系统对应于常规通道中的乘客。两个系统都由一个队列 Q_1 连接一个服务台 S_1（身份检查服务台），再连接一个队列 Q_2，最后连接一个服务器 S_2（扫描服务台，其中包括微波扫描，行李扫描，以及必要时进行的额外扫描）组成的。预先确定的一小部分旅客将在预检通道中排队，而其余部分旅客将在常规通道中排队。

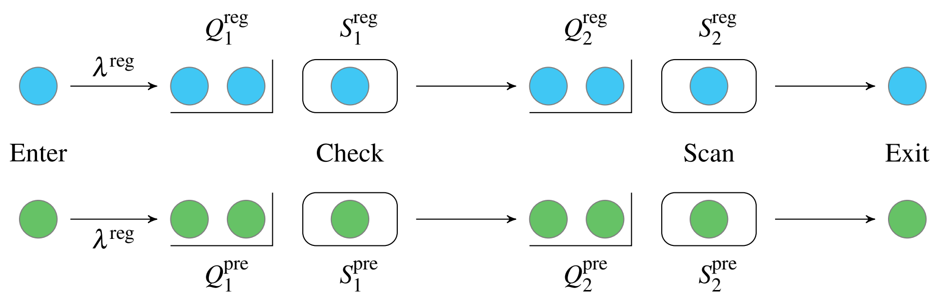

Figure 1: A schematic of the queuing model. The superscripts “pre” and “reg” stand for “precheck” and “regular”, respectively. | 排队模型的示意图。上标 “pre” 和 “reg” 分别代表 “预检旅客” 和 “常规旅客”。

Assumption 1. *Time spent in additional screening (Zone D) was included in the “Time to get scanned property” column in the given data file.*

Remark. *Data was given as total time until picking up scanned property (column H), which would be done after the additional Zone D screening. This implies that column H incorporates any time spent in Zone D.*

假设 1. 在额外的扫描（D 区）中花费的时间包含在给定数据文件中的“取到扫描物品的时间”列中。

说明：数据给出的是直到取回扫描的物品的总时间（H 列），这包含了在 D 区额外的扫描时间。这意味着 H 列中的时间包含了在 D 区花费的任何时间。

Assumption 2. *For the first version of our model, we assume that the system includes only the top row in Figure 1, and that there is only one service spot at each of S_1 (ID check) and S_2 (scan station). Later on we will remove this assumption.*

Remark. *In order to explicitly solve the recurrence relations we develop later on, it suffices to assume that service stations S_1 and S_2 each can only service one entity at a time. When we eventually take a dynamic programming approach, we will vary the number of ID check and scan stations and select a configuration which optimizes the cost-benefit tradeoff (See “Cost-Benefit Analysis” section).*

假设 2. 对于我们模型的第一个版本，我们假设系统只包括图 *1* 中的第一行，并且在 S_1（身份检查服务台）和 S_2（扫描服务台）中的每一个处仅有一个服务台。稍后我们将去掉这个假设。

说明：为了明确地解决我们后来发展的递推关系，假设服务台 S_1 和 S_2 每一次只能服务一个乘客。当我们最终采取动态编程的方法时，我们将改变身份检查服务台和扫描服务台的数量，并选择一个优化成本效益权衡的配置（参见“成本效益分析”部分）。

2.1 Obtaining Probability Densities | 获得概率密度

2.1.1 Exponential Fitting | 指数拟合

Depicted below are histograms of the passenger inter-arrival times extracted from the given data file. They strongly resemble the shape of an exponential probability density, motivating the next key assumption.

下面描述的是从给定数据文件中提取的乘客到达时间间隔的直方图。它们非常类似于指数概率密度的形状，这启发了我们做出下一个关键假设。

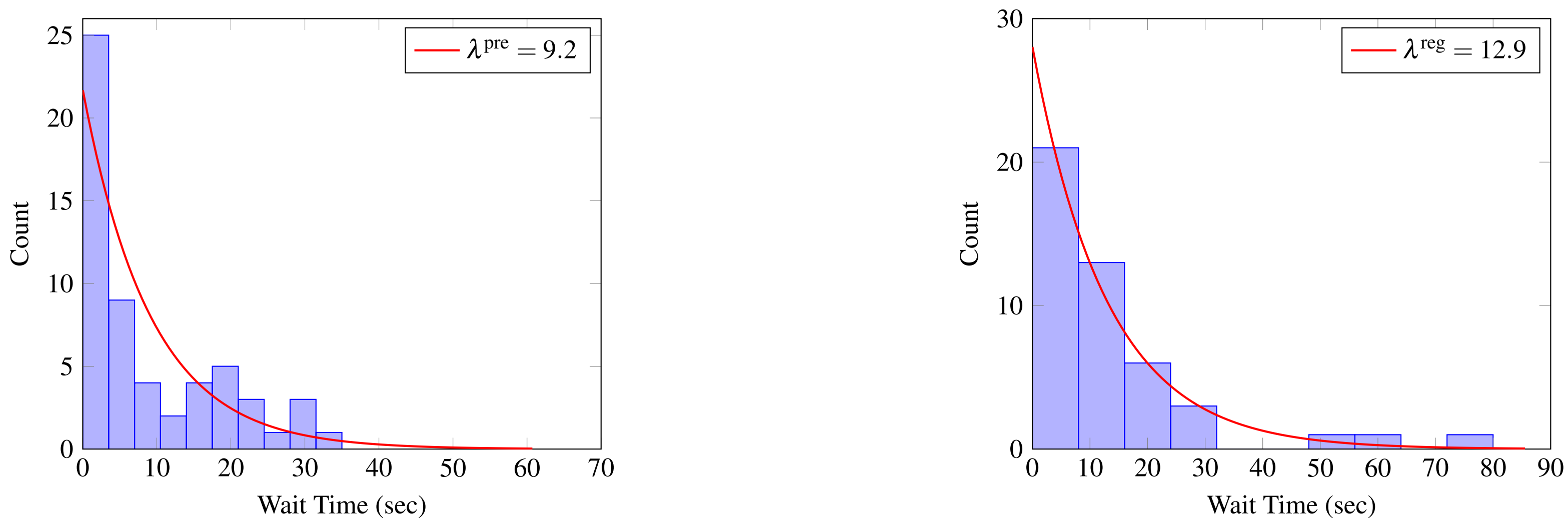

Figure 2: The exponential fit (using Matlab's expfit function) to the inter-arrival times of precheck (left) and regular (right) passengers overlaid onto the corresponding histograms. | 预检（左）和常规（右）旅客的到达间隔时间的指数拟合（使用 Matlab expfit 函数）曲线重叠在相应的直方图上。

Assumption 3. *The inter-arrival times are distributed i.i.d. exponential with rate $\lambda^{reg} = 12.9$ and $\lambda^{pre} = 9.2$ (seconds per passenger).*

Remark. *We chose to model the incoming passengers as a Poisson Process, since the histogram of the inter-arrival times resembles an exponential distribution, a defining characteristic of the inter-arrival times of a Poisson Process. The estimate for the rate parameter we used was the sample mean of the inter-arrival times, as given by maximum likelihood estimation [1].*

假设 3. 到达时间间隔服从参数为 $\lambda^{reg} = 12.9$（秒/人）和 $\lambda^{pre} = 9.2$（秒/人）的指数分布。

说明：我们选择泊松过程来模拟旅客的到达过程，因为到达时间间隔的直方图类似于指数分布，这符合泊松过程对到达间隔时间的定义特征。由于样本均值就是泊松分布参数的最大似然估计 *[1]*，我们使用的速率参数的估计值是到达时间的样本均值。

2.1.2 Kernel Density Estimation | 核密度估计

We define the following random variables.

$$
\begin{aligned}
T^{(j)}_{\text{total}} &\doteq \text{The total time entity } j \text{ spends at TSA check in.}\\
T^{(j)}_{Q_i} &\doteq \text{The wait time of entity } j \text{ in queue } i.\\
S^{(j)}_i &\doteq \text{The time spent of entity } j \text{ in service } i.\\
A_j &\doteq \text{The arrival time of entity } j.\\
I_j &\doteq \text{The inter-arrival time between passenger } j{-}1 \text{ and } j.
\end{aligned}
$$

The A_j's can be written as sums of the first j inter-arrival times, which we assumed to be distributed i.i.d. exponential. To find the distributions on the $S^{(j)}_i$, we perform kernel density estimation on the given data using Matlab's `ksdensity` function for estimation of a density over a positive support.

我们定义下面的随机变量。

$$
\begin{aligned}
T^{(j)}_{\text{total}} &\doteq \text{乘客 } j \text{ 在 TSA 检查中花费的总时间。}\\
T^{(j)}_{Q_i} &\doteq \text{乘客 } j \text{ 在队列 } i \text{ 中的等待时间。}\\
S^{(j)}_i &\doteq \text{乘客 } j \text{ 花在服务 } i \text{ 上的时间。}\\
A_j &\doteq \text{乘客 } j \text{ 的到达时间。}\\
I_j &\doteq \text{乘客 } j-1 \text{ 和 } j \text{ 之间的到达时间间隔。}
\end{aligned}
$$

A_j 可以写成前 j 个到达间隔时间的总和，我们假定这个时间间隔是服从指数分布。为了找到 $S^{(j)}_i$ 的分布，我们使用 Matlab 的 ksdensity 函数对给定数据进行核密度估计，以估计支撑集为正实数集的密度。

Assumption 4. *Time spent in the scan station is smaller for precheck passengers since they do not need to remove as many personal items.*

Remark. *This is a reasonable assumption, as precheck passengers pay an extra fee in order to exercise this privilege.*

假设 4. 由于无需移除尽可能多的私人物品，所以在扫描站中预检旅客花费的时间更短。

说明. 这是一个合理的假设，因为预检旅客为了行使这种特权而支付额外的费用。

To find the distribution on the time spent in service station 2 (scanning) for

为了得到预检旅客在服务台 2（扫描）花费的时间分布，

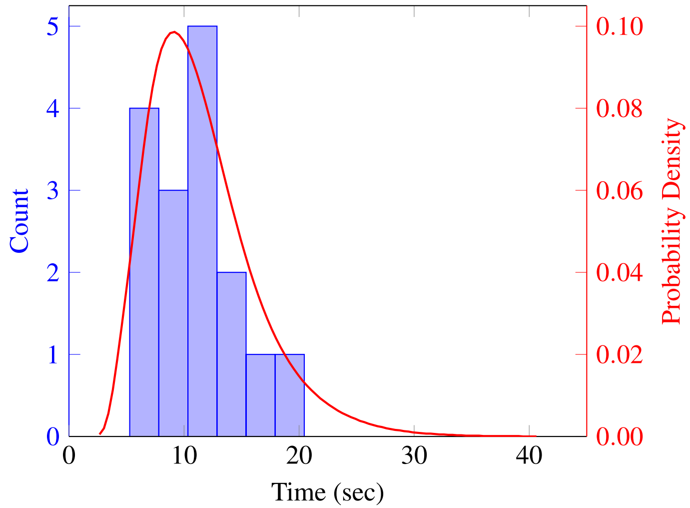

Figure 3: The kernel density estimate of both regular and precheck passenger time spent in service station 1 (ID check) overlaid onto the corresponding histogram. | 常规和预检旅客在服务台 1（身份检查）花费时间的核密度估计曲线重叠在相应的直方图上。

precheck passengers, we noted that since they don't have to remove personal items, the time they spend in the scanning station, S_2^{pre} is the maximum of the time spent in the mm wave scan and the time spent on luggage scanning. For instance, if the luggage finishes scanning first, then as soon as the passenger exits the mm wave scan, they can immediately pick up their luggage and go.

$$
\begin{aligned}
X &\doteq \text{Time spent in the mm wave scan.} \\
W &\doteq \text{Time spent scanning a piece of luggage.} \\
Y &\doteq 2.26 \cdot W \\
S_2^{\text{pre}} &\doteq \max(X, Y)
\end{aligned}
$$

Since the provided data file reports the times that each piece of luggage exits the scan machine, we must multiply the random variable representing this time by the number of bags that the average passenger carries on. According to the TSA, a passenger brings 2.26 bags on average [2].

我们注意到，由于他们不需要移除个人物品，他们在扫描服务台花费的时间 S_2^{pre} 是在微波扫描中花费的时间和在行李扫描上花费的时间的最大值。例如，如果行李先完成扫描，那么一旦乘客离开微波扫描，他们可以立即拿起行李走人。

$$
\begin{aligned}
X &\doteq \text{在微波扫描花费的时间。} \\
W &\doteq \text{扫描一件行李的时间。} \\
Y &\doteq 2.26 \cdot W \\
S_2^{\text{pre}} &\doteq \max(X, Y)
\end{aligned}
$$

由于提供的数据文件给出了每件行李退出扫描机器的时间，因此我们必须将表示这一时间的随机变量乘以旅客平均携带的行李数量。根据 TSA 的报道，一名乘客平均携带 2.26 个行李 [2]。

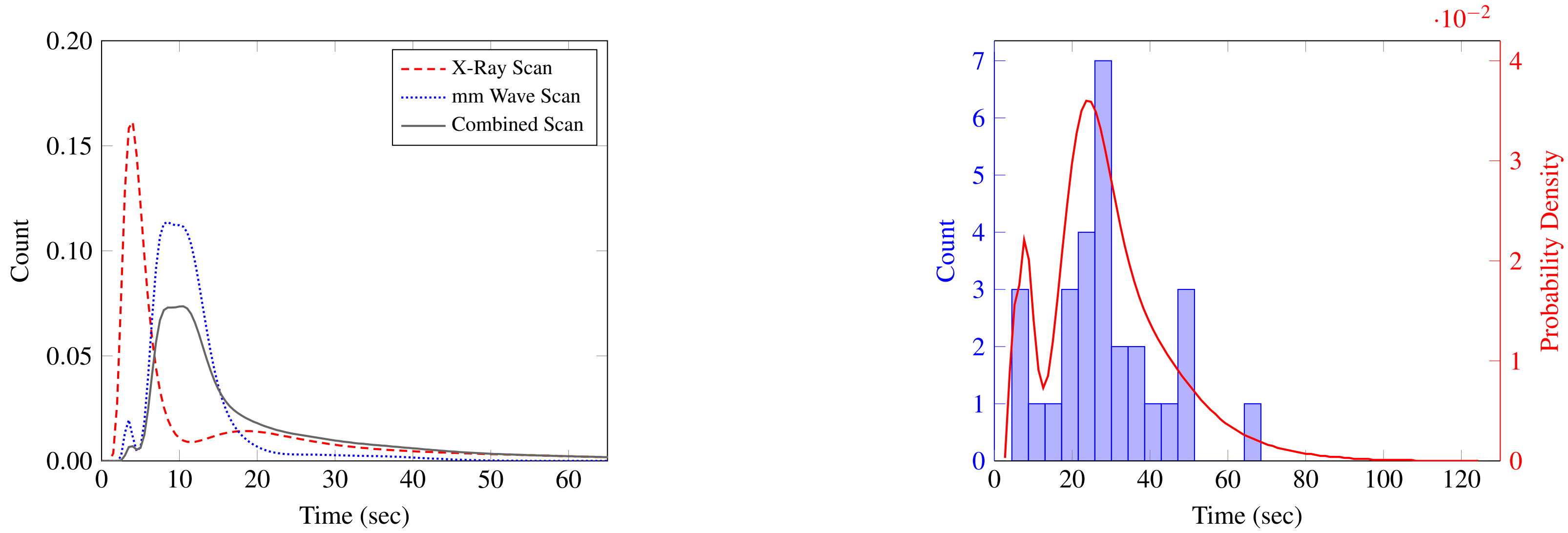

Figure 4: The kernel density estimate of precheck (left) vs. regular (right) passenger time spent in service station 2 (scanning). | 预检（左）与常规（右）旅客在服务台 2（扫描）花费时间的核密度估计。

Assumption 5. *X and Y are independent.*

Remark. *This is a reasonable assumption, as scan times for luggage machines and mm wave scan machines don't influence each other.*

假设 5. X 和 Y 是相互独立的。

说明. 这是一个合理的假设，因为行李扫描仪和微波扫描仪的扫描时间不会相互影响。

We estimate the distributions on X and Y using kernel density estimation (depicted in the left plot of Figure 4 by the dotted densities). To obtain the distribution on S_2^{pre}, we write

我们使用核密度估计来估计 X 和 Y 的分布（在图 4 的左图中用虚线表示密度）。要获得 S_2^{pre} 的分布，我们有

$$F_S(s) \doteq P(S_2^{\text{pre}} \leq s) = P\big(\max(X, Y) \leq s\big) = P(X \leq s, Y \leq s) = F_X(s)F_Y(s)$$

where the last equality follows from independence. Taking the derivative to obtain the density on S_2^{pre}, we have

最后等号是基于 X 和 Y 是相互独立的。取导数得到 S_2^{pre} 的密度分布，我们有

$$f_S(s) \doteq \frac{\mathrm{d}}{\mathrm{d}s}F_S(s) = f_X(s)F_Y(s) + f_Y(s)F_X(s)$$

This density was computed and plotted as the black curve against the kernel density estimates of f_X and f_W on the left plot in Figure 4.

计算该密度并将其绘制成黑色实线，并与 f_X 和 f_W 的核密度估计值一同绘制于图 4 中左图上。

2.2 Writing Down a Recurrence | 写出递推关系

2.2.1 Definitions and Derivations | 定义和推导

We define the total time spent by passenger n in the system as the sum of time spent in the queues and the service stations.

我们将乘客 n 在系统中花费的总时间定义为排队和在服务台花费的时间总和。

$$T_{\text{total}}^{(n)} = T_{Q_1}^{(n)} + T_{Q_2}^{(n)} + S_1^{(n)} + S_2^{(n)} \tag{1}$$

We define the wait times in the first queue $\{T_{Q_1}^{(i)}\}_{i\geq 1}$ recursively. The time that passenger i spends in queue 1 is the time at which they arrive subtracted from the time at which passenger $i-1$ leaves the first service station. The latter quantity can be written as the sum of the following quantities of the previous passenger: arrival time, time spent in line, and time spent getting serviced. This recurrence relation is illustrated below.

我们在第一个队列中顺序地定义等待时间 $\{T_{Q_1}^{(i)}\}_{i\geq 1}$。乘客 i 在队列 1 中花费的时间等于其到达的时刻减去乘客 $i-1$ 离开第一个服务台的时刻。后面乘客的数量可以写成之前乘客的以下数量的总和：到达时刻，排队花费的时间、接受服务花费的时间。这个递推关系如下所示。

$$\begin{aligned}
T_{Q_1}^{(1)} &= 0 \\
T_{Q_1}^{(2)} &= A_1 + T_{Q_1}^{(1)} + S_1^{(1)} - A_2 \\
&\vdots \\
T_{Q_1}^{(n)} &= A_{n-1} + T_{Q_1}^{(n-1)} + S_1^{(n-1)} - A_n
\end{aligned}$$

Assumption 6. *When passengers arrive at either queue, there is always someone in line to wait behind (except in the case of the first passenger to arrive).*

Remark. *In our definition for the $T_{Q_1}^{(i)}$'s, we should be taking the max of the expression displayed above and 0, since we could possibly have $A_i > A_{i-1} + T_{Q_1}^{(i-1)} + S_1^{(i-1)}$, i.e. passenger i arrives after passenger $i-1$ leaves the service station. This would lead to a negative waiting time. However, omitting the max operation allows for an explicit solution to the recurrence $T_{Q_1}^{(n)}$ and simplifies calculations immensely. Later on we will remove this assumption.*

假设 6. 当乘客到达任何一个队列时，总是有人排队等待（除了第一个乘客到达的情况）。

说明. 在我们对 $T_{Q_1}^{(i)}$ 的定义中，我们应该取上面显示的表达式的最大值和 *0*，因为我们可能有 $A_i > A_{i-1} + T_{Q_1}^{(i-1)} + S_1^{(i-1)}$，即乘客 i 在乘客 $i-1$ 离开服务站之后才到达。这会导致负的等待时间。然而，省略取最大值操作 max 允许显式的求解递推 $T_{Q_1}^{(n)}$ 并且极大地简化了计算。稍后我们将去掉这个假设。

Solving the recurrence, we obtain an explicit form for the time passenger n waits in queue 1,

通过解决递推问题，我们获得了在队列 1 中乘客 n 等待时间的明确形式，

$$T_{Q_1}^n = \left(\sum_{i=1}^{n-1} S_1^{(i)}\right) - A_n + A_1$$

Note that A_n can be written as a sum of interarrival times $\{I_j\}_{j=1}^n$ distributed i.i.d. exponential (λ), so that the expectation of $T_{Q_1}^{(n)}$ is given by

请注意，A_n 可以写为服从指数（λ）分布的到达间隔时间 $\{I_j\}_{j=1}^n$ 和的形式，所以 $T_{Q_1}^{(n)}$ 的期望可由下式给出

$$\mathbb{E}\left(T_{Q_1}^{(n)}\right) = \mathbb{E}\left(A_1 - A_n + \sum_{i=1}^{n-1} S_1^{(i)}\right) = \mathbb{E}\left(A_1 - \sum_{i=1}^{n} I_i + \sum_{i=1}^{n-1} S_1^{(i)}\right) = \lambda - \sum_{i=1}^{n}\lambda + \sum_{i=1}^{n-1}\mathbb{E}\left(S_1^{(i)}\right) = \lambda(1-n) + \mathbb{E}\left(S_1^{(i)}\right)(n-1)$$

where the third equality follows from linearity of expectations and since A_1 and I_i are distributed $\exp(\lambda)$. Simplifying, we have

其中第三个等式遵循期望值可线性叠加，并且由于 A_1 和 I_i 服从 $\exp(\lambda)$ 分布。通过简化，我们有

$$\mathbb{E}\left(T_{Q_1}^{(n)}\right) = (n-1)\left(\mathbb{E}\left(S_1^{(1)}\right) - \lambda\right) \tag{2}$$

Remark. This expression makes sense intuitively since if the expected service time per customer, $\mathbb{E}(S_1^{(1)})$, is large relative to time per passenger arrival, λ, then the line will build up, so the expected time spent waiting in the first queue grows with n, the number of passengers. We also note an inconsistency in the expression, namely that it can be negative as it is currently written. In reality, time can never be negative. This negativity is a result of Assumption 6, since we don't take the max of the quantity and 0 in our calculations. This remark is applicable to all

说明. 这个表达式是直观的，因为如果每个乘客的期望服务时间 $\mathbb{E}(S_1^{(1)})$ 大于每乘客到达时间间隔 λ，那么就会形成队列，所以在第一个队列中等待时间的期望值随乘客人数 n 增加。我们也注意到表达方式的不一致性，也就是说，它可能是现在写出来的形式的负值。事实上，时间永远不会是负的。这个负值情况是假设6的一个结果，因为在我们的计算中我们没有在数值和 0 之间取最大值。此说明适用于所有

boxed equations (5), (6), and (7). Once Assumption 6 is removed in section 3, the times will be strictly non-negative.

带方框的方程 (5)、(6) 和 (7)。一旦在小节 3 中删除假设 6，时间将严格为非负数。

Similarly, we can express the time spent waiting in queue 2 recursively, and solve for $T_{Q_2}^{(n)}$. The process is very similar to that of $T_{Q_1}^{(n)}$, so we omit the calculations and go straight to the final expression.

同样，我们可以递推地表示出在队列 2 中等待的时间，并求解出 $T_{Q_2}^{(n)}$。该过程与 $T_{Q_1}^{(n)}$ 非常相似，所以我们省略了计算并直接给出了最终表达式。

$$T_{Q_2}^{(n)} = \left(\sum_{i=1}^{n-1} S_2^{(i)} - \sum_{i=1}^{n} T_{Q_1}^{(i)}\right) + A_1 + S_1^{(1)} - A_n - S_1^{(n)}$$

原文公式有误，正确形式为：

$$T_{Q_2}^{(n)} = \sum_{i=1}^{n-1} S_2^{(i)} - \left(T_{Q_1}^{n} + A_n + S_1^{(n)}\right) + \left(A_1 + S_1^{(1)}\right)$$ [1]

By the linearity of expectations and since $S_1^{(1)} \overset{(d)}{=} S_1^{(n)}$,

通过期望的可线性叠加以及 $S_1^{(1)} \overset{(d)}{=} S_1^{(n)}$，

$$\mathbb{E}\left(T_{Q_2}^{(n)}\right) = \left(\sum_{i=1}^{n-1} \mathbb{E}\left(S_2^{(i)}\right) - \sum_{i=1}^{n} \mathbb{E}\left(T_{Q_1}^{(i)}\right)\right) + \lambda(1-n)$$

$$\mathbb{E}\left(T_{Q_2}^{(n)}\right) = \sum_{i=1}^{n-1} \mathbb{E}\left(S_2^{(i)}\right) - \mathbb{E}\left(T_{Q_1}^{(n)}\right) + \lambda(1-n)$$

Plugging in (2) for $\mathbb{E}\left(T_{Q_1}^{(n)}\right)$ and simplifying using the i.i.d. assumption of the $S_2^{(i)}$'s, we have

代入式 (2) 中的 $\mathbb{E}\left(T_{Q_1}^{(n)}\right)$，并使用 $S_2^{(i)}$ 服从独立同分布的假设简化，我们有

$$\mathbb{E}\left(T_{Q_2}^{(n)}\right) = (n-1)\left(\mathbb{E}\left(S_2^{(1)}\right) - \frac{1}{2}\left(\mathbb{E}\left(S_1^{(1)}\right) - \lambda\right) - \lambda\right)$$

$$\mathbb{E}\left(T_{Q_2}^{(n)}\right) = (n-1)\left(\mathbb{E}\left(S_2^{(1)}\right) - \mathbb{E}\left(S_1^{(1)}\right)\right) \tag{3}$$

Thus, the expected total time that passenger i spends in TSA checking is

因此，乘客 i 在 TSA 安检全过程中花费的期望总时间是

$$\mathbb{E}\left(T_{\text{total}}^{(n)}\right) = \mathbb{E}\left(T_{Q_1}^{(n)}\right) + \mathbb{E}\left(T_{Q_2}^{(n)}\right) + \mathbb{E}\left(S_1^{(n)}\right) + \mathbb{E}\left(S_2^{(n)}\right) \tag{4}$$

where $\mathbb{E}\left(T_{Q_1}^{(n)}\right)$ and $\mathbb{E}\left(T_{Q_2}^{(n)}\right)$ are given by (2) and (3).

其中 $\mathbb{E}\left(T_{Q_1}^{(n)}\right)$ 和 $\mathbb{E}\left(T_{Q_2}^{(n)}\right)$ 由式 (2) 和 (3) 给出。

2.2.2 Average Wait Time Values | 平均等待时间

To compute the average time that any passenger spends waiting in queue 1, $T_{Q_1}^{\text{avg}}$, we take the average of the waiting times. This will yield an expression in terms of the $T_{Q_1}^{(i)}$'s, which are random variables. We then take the expectation to

为了计算任意乘客在队列 1 中等待的平均时间，我们取所有乘客等待时间的平均值。这将以随机变量 $T_{Q_1}^{\text{avg}}$ 的形式产生一个表达式。然后，我们用其期望值来得到队列中的平

[1] 本文中所有给出的红色公式都是原文中有误的公式，蓝色公式是本人（周吕文）修正后的公式，具体参照原文。

obtain the average time spent in the queue.

均等待时间。

$$T_{Q_1}^{\text{avg}} \doteq \mathbb{E}\left(\frac{1}{n}\sum_{i=1}^{n} T_{Q_1}^{(i)}\right) = \frac{1}{n}\sum_{i=1}^{n}(i-1)\left(\mathbb{E}\left(S_1^{(1)}\right)-\lambda\right)$$

where the last equality follows from (2). Simplifying, we have

其中后一个等式来自式 (2) 。通过简化，我们有

$$\boxed{T_{Q_1}^{\text{avg}} = \frac{n-1}{2}\left(\mathbb{E}\left(S_1^{(1)}\right)-\lambda\right)} \tag{5}$$

A similar calculation that yielded average wait time for queue 2 will give

类似计算可以给出队列 2 中的平均等待时间

$$\boxed{T_{Q_2}^{\text{avg}} = \frac{n-1}{2}\left(\mathbb{E}\left(S_2^{(1)}\right)-\frac{1}{2}\mathbb{E}\left(S_1^{(1)}\right)-\frac{1}{2}\lambda\right)} \qquad \boxed{T_{Q_2}^{\text{avg}} = \frac{n-1}{2}\left(\mathbb{E}\left(S_2^{(1)}\right)-\mathbb{E}\left(S_1^{(1)}\right)\right)} \tag{6}$$

To find the average total waiting time, we combine the average waiting times (5) and (6) with expression (1).

为了得到总的平均等待时间，我们将式 (5) 和式 (6) 的平均等待时间与式 (1) 结合起来，

$$\boxed{T_{\text{total}}^{\text{avg}} = T_{Q_1}^{\text{avg}} + T_{Q_2}^{\text{avg}} + \mathbb{E}\left(S_1^{(1)}\right) + \mathbb{E}\left(S_2^{(1)}\right)} \tag{7}$$

We compute the boxed expressions (5), (6), and (7) to compare with the simulation in the following section. This will serve to validate our method of simulation.

我们计算带方框的表达式 (5)、(6) 和 (7) 来与下一小节的模拟进行比较。这将有助于验证我们的模拟方法。

3 The Simulation | 模拟

3.1 A Dynamic Programming Method | 一种动态规划方法

In the previous section, assumptions on the number of ID check and scan stations were made in order to simplify calculations. As a result we were able to obtain explicit formulas. In the following section we no longer make this simplify-

在上一节中，为了简化计算，对身份检查和扫描服务台的数目进行了假设。结果，我们得到了解析的公式。在接下来的小节中，我们不再做这个简化假设，而是表示为

ing assumption and instead denote

$$c \doteq \text{Number of service spots at check stations}$$
$$k \doteq \text{Number of service spots at scan stations}$$
$$n \doteq \text{Number of passengers}$$

We denote the number of passengers by n even though our model isn't restricted to any finite number of passengers. For the sake of the simulation, we sample for n passengers, where n can be made arbitrarily large.

Every passenger has an arrival time, a start/end time for the ID checking process, and a start/end time for the scanning process. Since our expression for these times is recursive, we take a dynamic programming approach to fill out a 2-D array, whose rows represent time events and whose columns represent passengers.

Since inter-arrival times are distributed i.i.d. exponential(λ), we simulate arrival times by keeping a running sum of i.i.d. exponential random variables. Since there are c spots for ID checking, we inspect the ID check leaving times of the previous c passengers to determine which is most readily available. If the arrival time of the current passenger exceeds this quantity, then the passenger can directly enter one of the ID check spots. Otherwise they must wait until a spot opens up. The start time for the scan process is computed similarly using k, the number of scan stations, in place of c. The end times for both service stations are computed by sampling from the kernel density estimates for the service times and adding these to the start times.

To compute the average waiting times, we take the difference between service start times and the times at which passengers start waiting, which is either their arrival time or their ID check end time depending on the service for which they are in queue. We then take the average over all passengers and return this result. A more concise set of instructions is given below as pseudo code.

$$c \doteq \text{身份检查服务台的数量}$$
$$k \doteq \text{扫描服务台的数量}$$
$$n \doteq \text{乘客数量}$$

尽管我们的模型不限于任何有限数量的乘客，我们用 n 来表示乘客的数量。为了仿真，我们抽样 n 个乘客，其中 n 可以任意大。

每个乘客都有到达时刻，身份检查过程的开始/结束时刻，以及扫描过程的开始/结束时刻。由于我们对这些时刻的表达是递推的，所以我们采用一种动态规划的方法来填充一个二维数组，其行代表时间事件，列代表乘客。

由于到达时间间隔服从指数分布（λ），我们用指数分布随机数的总和来模拟到达时刻。由于身份检查有 c 个服务台，我们查看前面 c 个乘客接受身份检查的离开时间，以确定哪个服务台是最先可用的。如果当前乘客的到达时刻晚于服务台的可用时刻（即空闲时刻），则乘客可以直接进入其中一个空闲的身份检查服务台。否则，他们必须等到一个服务台重新开放。扫描过程的开始时刻的计算与身份检查类似，使用 k（扫描服务台的数量）代替 c。两种服务台的结束时刻是通过从服务时间的核密度估计抽样并将其加到开始时刻来计算的。

为了计算平均等待时间，我们考虑服务开始时刻和乘客开始等待的时刻之间的时间差，乘客开始等待的时刻可以是他们到达时刻或者他们的身份检查结束的时刻，这取决于他们所在的队列。然后我们取所有乘客的平均值并返回这个结果。下面以伪代码的形式给出一组更简洁的指令。

3.1.1 Pseudo code | 伪代码

Algorithm 1 TSA Security Simulation

Result: a vector representing the average time spent in queues and the entire system.

First, initialize a two dimensional array with 5 rows and $\max(c,k)+n$ columns. Each row represents either passenger arrival time, time entering check station, time exiting check station, time entering scan station, or time exiting scan station (finished).

for $i \in \{\max\{c,k\},\cdots,n\}$ **do**

$$\text{Arrival}^{(i)} = \text{Arrival}^{(i-1)} + \text{sample}_{\exp(\lambda)}$$

$$\text{Check}_{\text{start}} = \max\left\{\text{Arrival}^{(i)}, \min\left(\text{Check}_{\text{end}}^{(i-c)},\cdots,\text{Check}_{\text{end}}^{(i-1)}\right)\right\}$$

$$\text{Check}_{\text{end}}^{(i)} = \text{Check}_{\text{start}}^{(i)} + \text{sample}_{\text{id check}}$$

$$\text{Scan}_{(i)}^{\text{start}} = \max\left\{\text{Check}_{(i)}^{\text{end}}, \min\left(\text{Scan}_{\text{end}}^{(i-k)},\cdots,\text{Scan}_{\text{end}}^{(i-1)}\right)\right\}$$

$$\text{Scan}_{\text{end}}^{(i)} = \text{Scan}_{\text{start}}^{(i)} + \text{sample}_{\text{scan}}$$

end for

$$T_{Q_1}^{\text{avg}} = \frac{1}{n}\text{sum}\left(\text{Check}_{\text{start}} - \text{Arrival}\right)$$

$$T_{Q_2}^{\text{avg}} = \frac{1}{n}\text{sum}\left(\text{Scan}_{\text{start}} - \text{Check}_{\text{start}}\right)$$

$$T_{\text{Total}}^{\text{avg}} = \frac{1}{n}\text{sum}\left(\text{Scan}_{\text{end}} - \text{Arrival}\right)$$

return $\left(T_{Q_1}^{\text{avg}}, T_{Q_2}^{\text{avg}}, T_{\text{Total}}^{\text{avg}}\right)$

结果：一个表示在队列和整个系统中花费的平均时间的矢量。

首先，初始化一个 5 行 $\max(c,k)+n$ 列的二维数组。每一行代表一个乘客到达时间、进入身份检查服务台的时刻、离开身份检查服务台的时刻、进入扫描服务台的时刻、或者退出扫描服务台的时间（完成）。

for $i \in \{\max\{c,k\},\cdots,n\}$ **do**

$$\text{到达}^{(i)} = \text{到达}^{(i-1)} + \text{sample}_{\exp(\lambda)}$$

$$\text{检查}_{\text{开始}}^{(i)} = \max\left\{\text{到达}^{(i)}, \min\left(\text{检查}_{\text{结束}}^{(i-c)},\cdots,\text{检查}_{\text{结束}}^{(i-1)}\right)\right\}$$

$$\text{检查}_{\text{结束}}^{(i)} = \text{检查}_{\text{开始}}^{(i)} + \text{sample}_{\text{检查}}$$

$$\text{扫描}_{(i)}^{\text{开始}} = \max\left\{\text{检查}_{(i)}^{\text{结束}}, \min\left(\text{扫描}_{\text{结束}}^{(i-k)},\cdots,\text{扫描}_{\text{结束}}^{(i-1)}\right)\right\}$$

$$\text{扫描}_{\text{结束}}^{(i)} = \text{扫描}_{\text{开始}}^{(i)} + \text{sample}_{\text{扫描}}$$

end for

$$T_{Q_1}^{\text{avg}} = \frac{1}{n}\text{sum}\left(\text{检查}_{\text{开始}} - \text{到达}\right)$$

$$T_{Q_2}^{\text{avg}} = \frac{1}{n}\text{sum}\left(\text{扫描}_{\text{开始}} - \text{检查}_{\text{开始}}\right)$$

$$T_{\text{Total}}^{\text{avg}} = \frac{1}{n}\text{sum}\left(\text{扫描}_{\text{结束}} - \text{到达}\right)$$

return $\left(T_{Q_1}^{\text{avg}}, T_{Q_2}^{\text{avg}}, T_{\text{Total}}^{\text{avg}}\right)$

3.2 Comparison to the Model | 与模型的比较

We implemented the simulation described above in Matlab and compared the simulated average waiting times with their explicit formulas which were derived in the previous section, namely, expressions (3), (6), and (7).

我们在 Matlab 中实现了上面所描述的模拟，并将模拟的平均等待时间与上一小节中推导出的平均等待时间公式，即表达式 (3)、(6) 和 (7) 作比较。

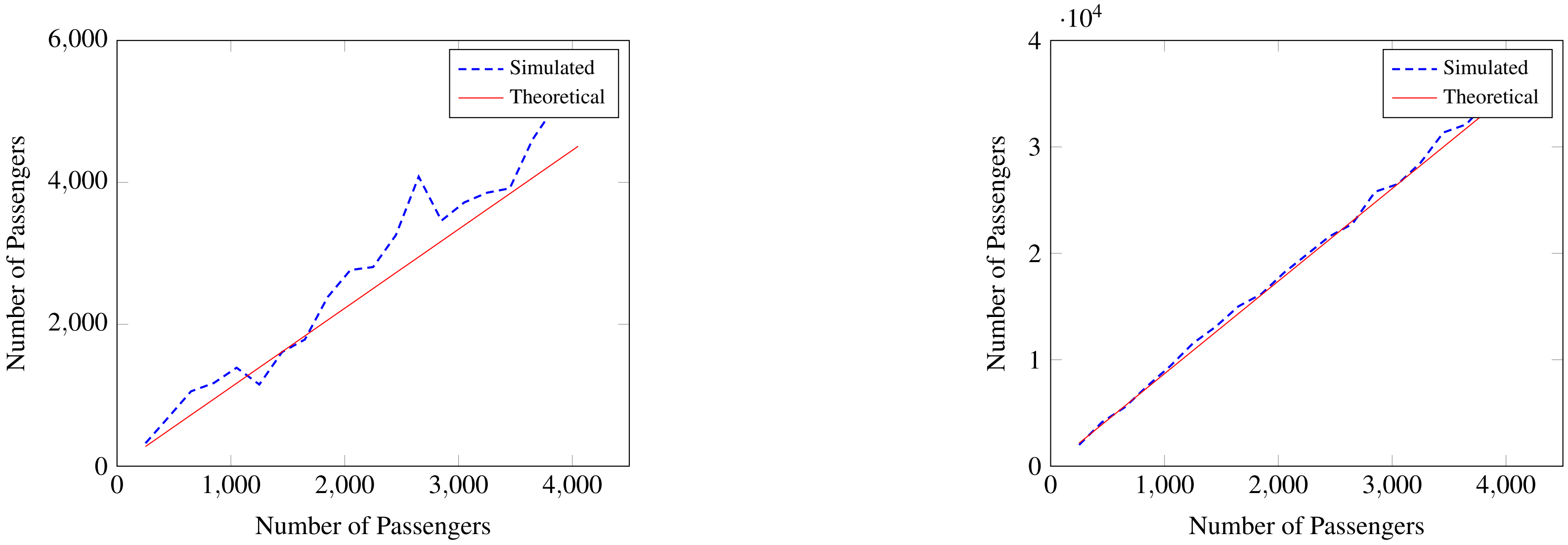

Figure 5: Plot of the average time spent in queues 1 (left) and 2 (right). Simulated and theoretical time averages are depicted in blue and orange respectively. | 在队列 1（左）和 2（右）花费的平均时间的图。模拟和理论时间平均值分别用蓝色和橙色表示。

Figure 5 shows the calculated results from our model against the simulated results. It appears that they follow the same trend on average as the number of passengers increases from 50 to 4050. As a result, we are more inclined to believe the results obtained from using our simulation in the proceeding experiments. The simulation also seems to vary from the theoretical results more drastically in the waiting time for queue 1. However, looking at the scale on which these waiting times are plotted, we reason that we see this relatively large variance because the waiting times for queue 1 are much smaller in magnitude than those for queue 2. The plot of simulated versus theoretical values for total time spent in the system

图5显示了我们的模型计算结果与模拟结果。当旅客人数从 50 人增加到 4050 人时，它们似乎也遵循相同的趋势。最终，我们更倾向于相信使用我们的模拟测试所得到的结果。对于在队列 1 中的等待时间，模拟似乎与理论结果有较大差异。但是，考虑这些被画出的等待时间的量级，我们推断我们看到的这个相对大的差异的原因是队列 1 的等待时间在数量级上要小于队列 2。在系统中花费的总时间的模拟值与理论值的图被省略，因为它基本上是上述两个图的总和，因此，系统中花费的总时间遵循相同的趋势。

is omitted since it is essentially the sum of the above two plots and as a result, follows the same trend.

4 Evaluation/Results | 评价/结果

4.1 Bottlenecks in Passenger Throughput | 旅客吞吐量瓶颈

From our model we identified two major bottlenecks in the flow of passengers through airport security: ID check stations and scan stations. Through simulation we investigated how the number of service spots at each station affects the total time a passenger waits.

从我们的模型中，我们确定了乘客通过机场安检的两个主要瓶颈：身份检查服务台和扫描服务台。通过模拟，我们研究了每种服务台的数量如何影响乘客等待的总时间。

4.1.1 ID Check Station | 身份检查服务台

The first bottleneck a passenger experiences upon reaching airport security is the ID check station. Both TSA precheck and regular passengers receive service at the same rate from this station and therefore, when investigating this bottleneck we considered only one service distribution. However, TSA precheck and regular passengers arrive at different rates to this station, so we investigated how varying arrival rates and the number of check stations would affect the wait time in the checking queue. The results from our simulation can be seen in figure 6.

身份检查服务台是乘客在抵达机场安检时遇到的第一个瓶颈。TSA 预检和普通乘客都能以同样的速度从服务台接收服务，因此在研究这个瓶颈时，我们只考虑一种服务时长分布。然而，TSA 预检和普通乘客以不同的速率到达该服务台，因此我们研究了不同的到达率和检查服务台的数量将如何影响检查队列中的等待时间。我们的模拟结果见图 6。

The average wait time of a passenger in the checking queue appears to be independent of the number of scanning stations, k, an intuitive result. However, there is a clear relationship between the number of open spots at the ID check station, c, and the average wait time of a passenger in the ID check queue. The wait time undergoes a significant transition, decreasing drastically, as the arrival rate of passengers slows (λ increases). Using the queuing model described previously, the λ value at which this transition takes place can be estimated by the following relation.

显然，身份检查队列中乘客的平均等待时间似乎与扫描服务台的数量 k 无关。然而，身份检查服务台的开放数量 c 与身份检查队列中的乘客的平均等待时间之间有明确的关系。等待时间经历了显着的转变，随着乘客到达速率的减缓（λ 增加）急剧下降。使用之前给出的排队模型，可以通过以下关系来估计发生该转变的 λ 值。

$$\lambda = \frac{\mathbb{E}\left(S_1^{(i)}\right)}{c} \tag{8}$$

Remark. This relation can be understood as the condition for when the average arrival rate (λ) is equal to the average service rate ($\mathbb{E}(S_1^{(i)})/c$). When passengers

说明：这个关系可以理解为当平均到达率（λ）等于平均服务率（$\mathbb{E}(S_1^{(i)})/c$）的条件。当乘客的到达速率快于在身份

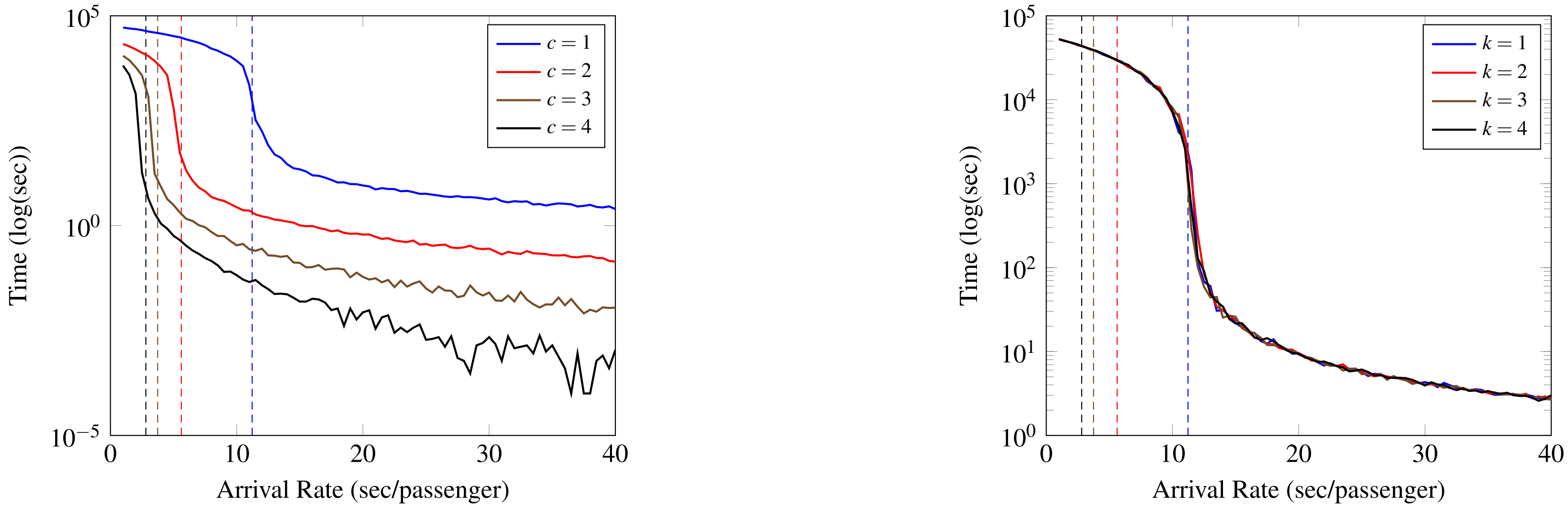

Figure 6: Plot of wait time spent in ID check queue for various values of c (left) and k (right). | 在 c（左）和 k（右）的各种值中花费在 ID 检查队列中的等待时间的图。

arrive faster than they are being serviced at the ID check stations, a queue forms, causing a bottle neck on passenger throughput. Conversely, when passengers are serviced more quickly than they arrive, a queue becomes less likely to form. In the case of $c = 1$, this is equivalent to setting the expected waiting time in Q_1 given by (2) equal to 0.

检查服务台的处理速度时，就会形成一个队列，造成乘客吞吐量的瓶颈。相反，当乘客接受服务的时长比抵达的时间间隔更短时，则不太可能形成队列。在 $c = 1$ 的情况下，这相当于将由（2）给出的 Q_1 中的等待时间期望设置为 0。

The estimated λ transition values given by (8) are plotted as vertical lines on the same plot in Figure 6. One can see that these points correspond very closely to the observed transitions from our simulation.

由（8）给出的 λ 转变值的估计在图 6 中绘制为垂直线。可以看到，这些点非常接近我们模拟观察到的转变。

4.1.2 Scan Station | 扫描服务台

The second bottleneck a passenger will experience in airport security is the scan station. We conducted an analysis similar to the one we performed on the first bottleneck. Unlike the ID check station, TSA precheck and regular passengers experience different service rates at this station; however, the general trend remains

扫描服务台是旅客在机场安检中遇到的第二个瓶颈。我们进行了一个类似于我们在第一个瓶颈上进行的分析。与身份检查服务台不同，TSA 预检和普通乘客在本服务台遇到不同的服务率；不过，总的趋势依然如此。我们对普通乘客的

the same. The results from our simulation of regular passengers can be seen in figure 7.

模拟结果见图 7。

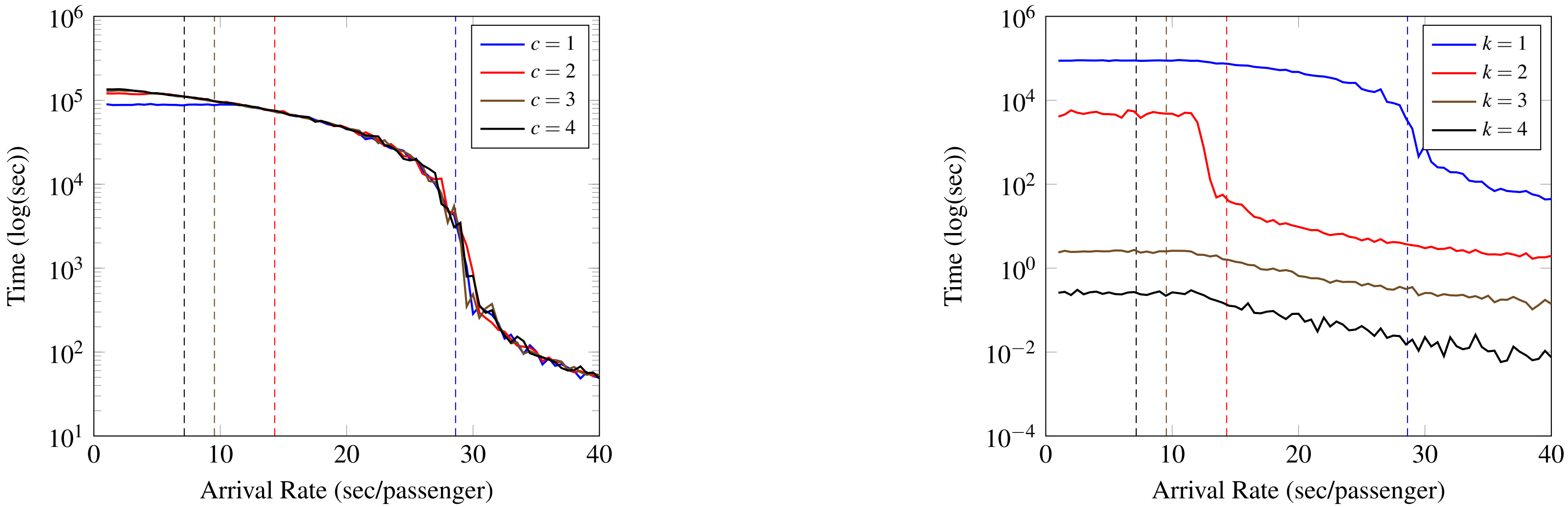

Figure 7: Plot of average time spent in the regular scan queue with varying arrival rates for various values of c (left) and k (right). | 在 c（左）和 k（右）不同值的不同到达率下在正常扫描队列中花费的平均时间的图。

The average wait time of a passenger in the scanning station is dependent on both c and k. We observe that the concept of λ transition values, discussed above in the first bottleneck, are important as they mark significant transitions in $T_{Q_2}^{(n)}$, where the analogous values are given by

扫描站中乘客的平均等待时间取决于 c 和 k。我们观察到 λ 过渡值的概念，在上面讨论的第一个瓶颈中是重要的，因为它们标记了 $T_{Q_2}^{(n)}$ 中的显着过渡，其中类似的值由

$$\lambda = \frac{\mathbb{E}\left(S_2^{(i)}\right)}{c} \tag{9}$$

4.2 Variance in Passenger Wait Time | 乘客等待时间的方差

Increasing passenger throughput and decreasing wait times is only part of the story when it comes to improving airport security. Equally important is reducing the variance in passenger wait times to ensure that passengers know generally how

提高旅客吞吐量和减少等待时间只是提高机场安检效率的一部分。同样重要的是减少乘客等待时间的方差，以确保乘客通常知道安检将会需要多长时间。使用我们的模拟，我

long security lines will be. Using our simulation, we investigated how the number of check stations (c) and scan stations (k) affected the variance of passenger wait times in queues. We found that the number of check stations had a larger affect on the variance of wait times in both queues than the number of scan stations.

们研究了身份检查服务台（c）和扫描服务台（k）数量是如何影响队列中乘客等待时间的方差的。我们发现身份检查服务台的数量对两个队列中等待时间的变化影响比扫描服务台的数量的影响更大。

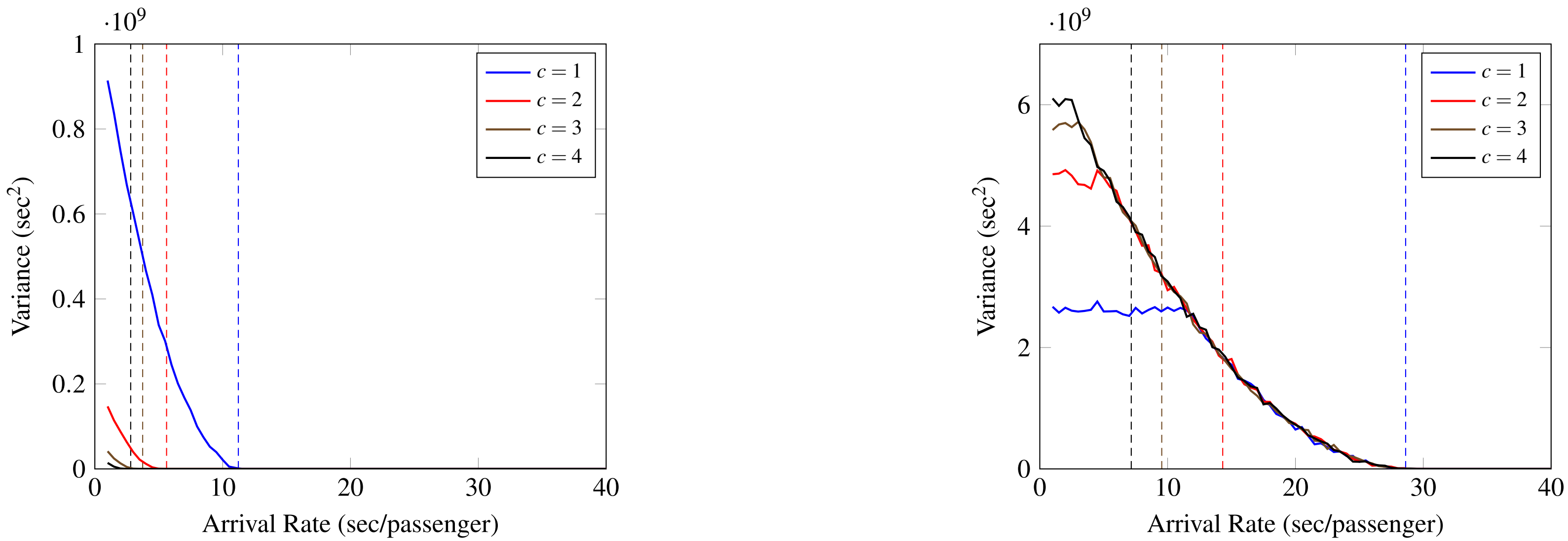

Figure 8: Plot of variance of passenger wait times spent in check station queue (left) and scan station queue (right) for various values of c and constant $k = 1$. | 固定 $k = 1$ 并改变 c 时，乘客在身份检查队列（左）和扫描队列（右）中花费的时间方差图。

As seen in figure 8, the variance in passenger wait times spent in either queue decreases as arrival rate increases. Major transitions occur at the same λ transition values given by (8).

如图8所示，每个队列中乘客等待时间的方差随着到达率的增加而降低。主要转变发生在由 (8) 给出的相同的 λ 转变值处。

4.3 Modifications to the Model | 对模型的修改

4.3.1 Cost-Benefit Analysis | 成本效益分析

In this section we confront the optimization problem of balancing the cost of employing TSA workers with the benefit of increasing passenger flow. Following the events of September 11, TSA began to include a $6.60 security fee per one-way-trip

在本小节，我们要面对平衡雇用 TSA 工作人员的成本和增加乘客流量的优化问题。在 9 月 11 日事件之后，TSA 在 2017 年开始包括每单程 6.60 美元的安检费 [3]。这笔费用用

in the year 2017 [3]. The fee goes to funding the TSA's airport security measures. In other words, increasing passenger throughput while limiting security costs can be profitable for TSA. In order to investigate this relationship, we describe an objective function for TSA profit using the following variables.

于资助 TSA 的机场安保措施。换句话说，增加旅客吞吐量同时限制安检成本对于 TSA 来说可能是有利的。为了研究这种关系，我们使用以下变量描述 TSA 利润的目标函数。

$$
\begin{aligned}
C_p &\doteq \text{September 11 fee per one-way-trip}\\
C_t &\doteq \text{The cost of a Transportation Security Officer (TSO).}\\
T_{\text{total}} &\doteq \text{Average time spent from entering to leaving the system.}
\end{aligned}
$$

$$
\begin{aligned}
C_p &\doteq \text{9 月 11 日单程费用}\\
C_t &\doteq \text{运输安检员工（TSO）的成本。}\\
T_{\text{total}} &\doteq \text{从进入系统到离开的平均时间。}
\end{aligned}
$$

Remark. In the ATL Hartsfield-Jackson airport, a TSO makes on average 18.5 dollars an hour. Converting this salary into units relevant to our calculations, the average TSO makes 0.0005139 dollars per second [4]. In addition, we assume that 1 TSO is required to operate an ID check station, and that 3 TSOs are required to operate a scan station: one for the mm wave scan, one for the X-Ray, and one for auxiliary help.

说明：在 ATL Hartsfield-Jackson 机场，一个 TSO 平均每小时赚 18.5 美元。把这个工资换算成与我们计算相关的单位，TSO 平均每秒挣 0.0005139 美元 [4]。另外，我们假设一个身份检查服务台需要 1 个 TSO 操作，而一个扫描服务台需要 3 个 TSO 操作：一个用于微波扫描，一个用于 X 射线，另一个用于辅助。

Assumption 7. *The only cost for running a TSA security check point is to employ TSA workers.*

Remark. *There are other costs including management, machine maintenance, and training of TSOs, but these costs per employee are marginal compared to employee salary.*

假设 7. 运行 *TSA* 安检服务台的唯一成本是聘用 *TSA* 工作人员。

说明：还有其他一些成本，包括管理、机器维护和 *TSO* 培训，但这些人均成本与员工工资相比是微不足道的。

Finally, we define TSA profit rate, P, as the difference in the rate at which TSA gains money through the September 11 fee and the rate at which TSA loses money by staffing employees. Combining the terms defined above in this way yields a measure of profit, P, whose units are dollars per second.

最后，我们将 TSA 利润率 P 定义为 TSA 通过 “$9 \cdot 11$” 费用获得的收益率与 TSA 由雇员的费用率之差。以这种方式结合上面定义的术语可以得出一个利润的衡量标准，单位是美元/秒。

$$
P = \frac{C_p}{T_{\text{total}}} - (c + 3k) \cdot C_t
$$

As in section 3 , c and k refer to the number of spots at ID check stations and scan stations, respectively. We wish to find the parameters c and k which maximize this objective function. Using the dynamic programming approach described in

如第 3 小节所述，c 和 k 分别是指身份检查服务台和扫描服务台的数量。我们希望找到使这个目标函数最大化的参数 c 和 k。使用第 3 节中描述的动态规划方法，我们对 T_{total}

section 3, we sample values for T_{total}, compute P for different values of c and k, and plot the resulting heat maps below.

进行采样，对于 c 和 k 的不同值计算 P，并绘制出下面的热图。

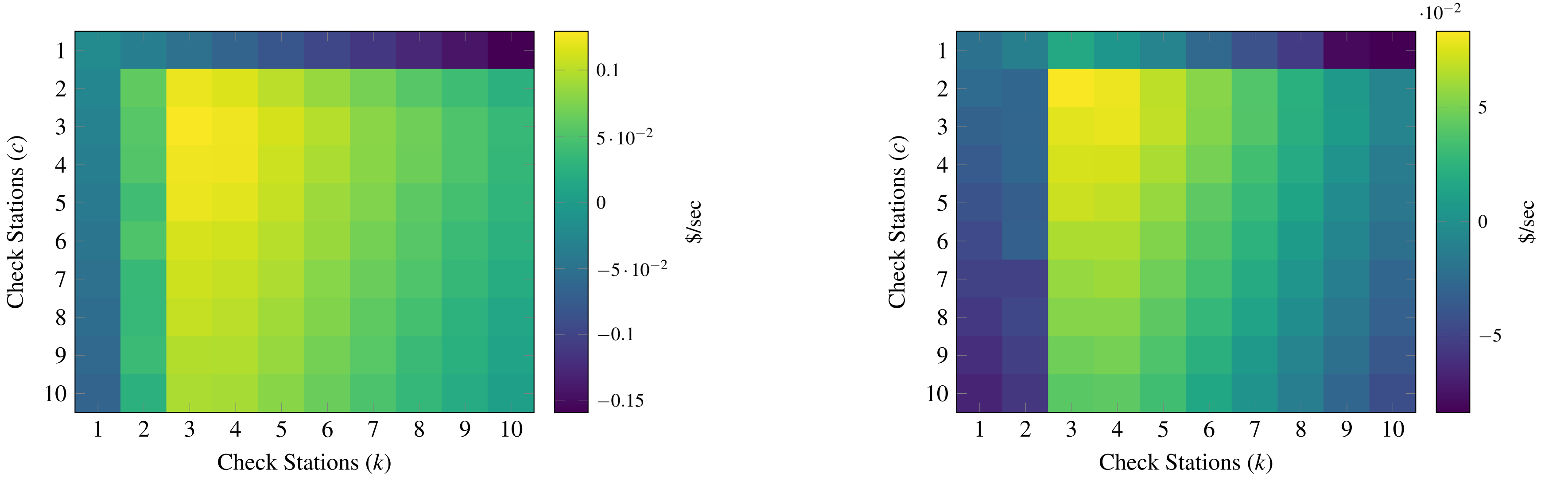

Figure 9: Profit as a function of the number of ID check and scan stations for TSA precheck (left) and Regular passengers (right). | 利润作为身份检查和扫描服务台数量的函数，TSA 预检乘客（左）和普通乘客（右）。

For the TSA precheck line, the most cost-efficient layout is to have 3 checking stations and 3 scanning stations to achieve a rate of $P = 0.140$ dollars per second. For the regular security line, the most cost-efficient layout is to have 2 checking stations and 3 scanning stations, yielding a profit rate of $P = 0.083$ dollars per second. TSA precheck gives the TSA a 68.6% gain in revenue per passenger compared to regular screening.

对于 TSA 预检通道，最具成本效益的设置是具有 3 个身份检查服务台和 3 个扫描服务台，此时利率可以达到每秒 $P = 0.140$ 美元。对于常规的安检通道，最具成本效益的设置是具有 2 个身份检查服务台和 3 个扫描服务台，此时利率可以达到 $P = 0.083$ 美元/秒。与常规安检相比，每位使用 TSA 预检通道的乘客都能使 TSA 得收入增加 68.6%。

4.3.2 Sensitivity Analysis | 灵敏度分析

Our other modification is the introduction of virtual queuing, a method first implemented in call centers for minimizing perceived wait times [5]. After a passenger checks in for their flight, the airline assigns them an arrival time and a

我们的其他修改是引入了虚拟排队，这种方法首先在呼叫中心实施，以最大限度地减少感知的等待时间 [5]。在乘客办理登机手续后，航空公司会在这段时间内为他们指定抵达

window around this time to enter the security process. The assigned boarding time instructs the passenger when to arrive at the first queue. The passenger now has more free time to shop at airport businesses instead of physically waiting in line. Virtual queuing not only allows for the reduction in perceived wait time but also for the implementation of non-FIFO disciplines.

We distinguish between two types of travelers: slow and fast (inexperienced and experienced), whose service times are multiplied by a predetermined factor. We define a new discipline, Priority, to describe the use of the virtual queue to group passengers of similar experience levels together, so that all slow passengers go first, followed by the quick passengers. This is done in contrast to the previous discipline, in which passengers of varying speeds arrive randomly.

时间和窗口，以进入安检程序。指定的登机时间指示乘客何时到达第一个队列。乘客现在有更多的空闲时间在机场的商店购物，而不是排队等候。虚拟排队不仅可以减少等待时间，而且可以实现非 FIFO 规则。

我们区分两种类型的旅客：慢速和快速（缺乏经验和有经验的），其服务时间乘以预设的因子。我们定义了一个新的方式，即“优先级”来描述使用虚拟队列将相似经验水平的乘客分组在一起，以便所有慢速乘客先行，然后是快速乘客。这是与之前的方式相反的，之前的方式是不同速度的乘客随机到达。

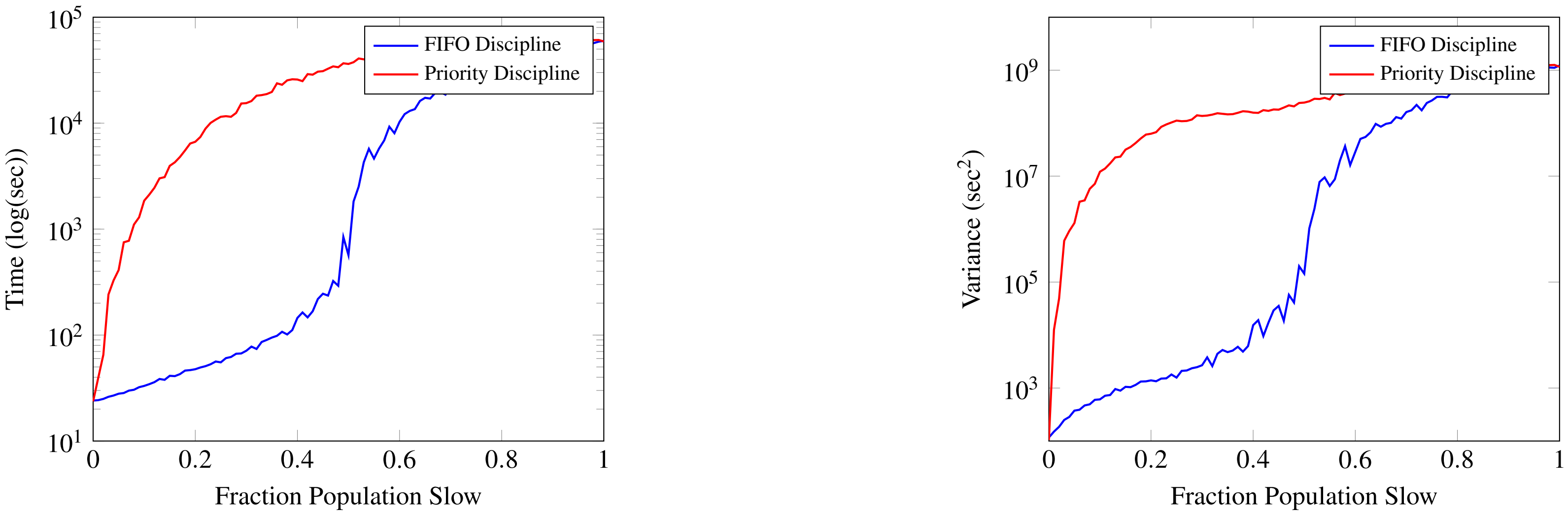

Figure 10: Plots of time (left) and variance (right) versus the fraction of fast passengers for the Priority (red) and FIFO disciplines (blue). | 等待时间（左）及方差（右）与优先快速乘客比例（红色）和 FIFO 规则（蓝色）的关系图。

In Figure 10, we plot wait time vs. the fraction of slow passengers. Contrary to our initial hypotheses, the Priority discipline generally led to both longer average total wait time and higher variance. A possible explanation is that if slow people

在图 10 中，我们绘制了等待时间与慢速乘客的比例关系。与我们最初的假设相反，优先规则通常导致平均总等待时间和方差更高。一个可能的解释是，如果慢的人先行，他

go first, they cause fast passengers to wait longer in line, increasing the average overall time waiting.

们会造成快速的乘客排队等候，增加平均等待时间。

In reality, inexperienced travelers (slow) are more likely to come early as they are new to the security process while experienced travelers (fast) are likely to come late. Thus, they will naturally group up into something similar to the conditions of our Priority discipline simulation. Because our simulations suggest that this decreases passenger throughput, airports can use virtual queuing and new disciplines to ensure a more uniform distribution of passengers to avoid bottlenecks.

事实上，没有经验的乘客（慢）更容易早到，因为他们是安检过程的新手，而经验丰富的乘客（快）很可能迟到。因此，他们自然会组成类似于我们优先规则的模拟条件。因为我们的模拟表明，这会降低旅客吞吐量，机场可以使用虚拟排队和新的规则，以确保更均匀的乘客分布来避免瓶颈。

5 Improving the Model | 改进模型

5.1 Traveler Characteristics | 旅客特征

In our model and modifications, we only account for travelers we deem slow and fast based on traveling experience. However, we can get more nuanced results by creating a system to profile people based on a variety of features like age, walking speed, families, etc. Perhaps there would be an optimal solution in creating different lanes designed for different types of travelers.

在我们的模型和改进中，我们只根据旅行经验来考虑我们认为缓慢和快速到达的旅客。但是，我们可以通过创建一个基于各种特征（如年龄、行走速度、家庭等）的系统来获得更多微妙的结果。也许针对不同类型的旅客设计不同的安检通道将是最佳的解决方案。

5.2 Different Queuing Disciplines | 不同的排队规则

We only considered the FIFO and Priority disciplines for our simulation. We can further explore new disciplines and ways of organizing passenger populations. These disciplines could possibly lead to more effective virtual queuing.

我们在我们的模拟中只考虑了 FIFO 和优先级规则。我们可以进一步研究新的乘客排队方式。这些排队方式可能会给出更有效的虚拟队列。

5.3 Arrival Rate Regulation | 到达率调控

Due to the nature of airport scheduling, it is unlikely that arrival rates follow a consistent Poisson distribution throughout the course of a day [6]. Virtual queuing gives us greater control over the arrival rate (λ) variation, because we set the passenger arrival times. In our simulations we show that total wait time and its variance is largely characterized by the fluctuation in λ. Therefore, we can explore methods to regulate the arrival rate while ensuring all passengers make their flight.

由于机场调度的性质，到达率在整个一天的过程中遵循一致的泊松分布是不太可能的 [6]。虚拟排队使我们能够更好地控制到达率（λ）的变化，因为我们设置了乘客到达时刻。在我们的模拟中，我们显示总等待时间及其方差主要是以 λ 的波动为特征的。因此，我们可以研究方法来调整到达率，同时确保所有的乘客及时登机。

5.4 Improving Profit Measure | 提高利润措施

There were many sources of cost and revenue which were not included in the construction of our profit measure. Some examples are machine maintenance, employee management, and variance in passenger wait times.

我们模型中忽略了很多成本和收入的来源。比如机器的维护，员工的管理和乘客等待时间的变化。

6 Conclusion | 结论

Our solution to the problem of excessive waiting times in airport security required the use of a diverse collection of techniques from the fields of statistics and computer science. The uncertainty in the dynamics of the TSA security check were described by a queuing network and simulated with the use of dynamic programming. The results of our simulations helped to indicate an optimal number of security officers to employ in order to maximize profit rate, while the sensitivity analysis hinted that our initial intuitions about line ordering based on passenger experience were flawed. We further identified where we could make our model more powerful by considering things such as traveler characteristics, creative queuing disciplines, and arrival time regulation. The real world continues to present all sorts of interesting problems for mankind to think about and develop mathematical tools to solve. Our attempt to gain insight into this problem closely follows this narrative, illustrating and verifying the importance of bringing theory to application and curiosity to exploration.

我们解决机场安检等待时间过长的问题需要使用统计学和计算机科学领域的多种技术。TSA 安检的动态不确定性由排队网络描述，并使用动态规划进行模拟。我们的模拟结果有助于确定最佳的安检工作人员数量，以使利润最大化。而灵敏度分析表明，我们最初的基于乘客经验的排队方式的想法是有缺陷的。我们通过考虑旅行者特征、制定排队规则和调控乘客到达时刻等因素，我们进一步确定了可以让我们的模型更强大的地方。现实世界继续呈现各种有趣的问题，以供人类思考并开发数学工具来解决。我们试图紧接着这篇论文之后深入了解这个问题，说明和验证了理论应用和好奇心对探索的重要性。

References

[1] George Casella and Roger L Berger. *Statistical inference*, volume 2. Duxbury Pacific Grove, CA, 2002.

[2] Transportation Security Administration. Tsa releases 2015 statistics. Web. 21 Jan. 2017.

[3] David Gillen and William G Morrison. Aviation security: Costing, pricing, finance and performance. *Journal of Air Transport Management*, 48:1–12, 2015.

[4] USAJOBS. Transportation security officer (TSO) job offering. Web. 21 Jan. 2017.

[5] Robert De Lange, Ilya Samoilovich, and Bo van der Rhee. Virtual queuing at airport security lanes. *European Journal of Operational Research*, 225(1):153–165, 2013.

[6] Stephen Louis Dorton. Analysis of airport security screening checkpoints using queuing networks and discrete event simulation: a theoretical and empirical approach. 2011.

Appendices

A Matlab code | Matlab 程序代码

readata.m

```
function [dt, name] = readata(type)
% READATE read time data from Data.xlsx file, converts to seconds, and then
% computes interarrival times. ICM 2017 Problem D.
%
% A TSA PreCheck Arrival Times    (PreArrT): Airport checkpoint recoding
%   individuals entering the pre-check queue.
% B Regular Arrival Times         (RegArrT): Airport checkpoint recoding
%   individuals entering the regular queue.
% C ID Check TSA officer 1        (IdChkT1): The time the arrival of the
%   passenger to the ID check station until the TSA officer calls the next
%   passenger forward.
% D ID Check TSA officer 2        (IdChkT2): Same as column C, but for a
%   different TSA officer.
% E mm wave scan times:           (MMScanT): Time stamps as passenger exited
%   the milimeter wave scanner.
% F X-Ray Scan Time 1             (XRayST1): Time stamps as bags exited the
%   x-ray screening.
% G X-Ray Scan Time 2             (XRayST2): Same as column F, but for a
%   different TSA officer.
% H Time to get scanned property  (GetProT): Time it takes people from
%   arriving at the belt to place items to be scanned, until they retrieved
%   their items off the post-xray belt.
%
% Zhou Lvwen: zhou.lv.wen@gmail.com
% January 8, 2018

d2s = 24 * 3600;

switch type
    case{'A','PreArrT','TSA PreCheck Arrival Times'}
        column = 'A'; isinter = 0; name = 'TSA PreCheck Arrival Times';
```

```
    case{'B','RegArrT','Regular Arrival Times'}
column = 'B'; isinter = 0; name = 'Regular Arrival Times';    case{'C','IdChkT1','ID Check TSA officer 1'}
        column = 'C'; isinter = 1; name = 'ID Check TSA officer 1';
    case{'D','IdChkT2','ID Check TSA officer 2'}
        column = 'D'; isinter = 1; name = 'ID Check TSA officer 2';
    case{'CD','IdChkT','IdChkT12','ID Check TSA officer 1 & 2'}
        dt = [readata('IdChkT1'); readata('IdChkT2')];
        name = 'ID Check TSA officer 1 & 2';
        return
    case{'E','MMScanT','mm wave scan times'}
        column = 'E'; isinter = 0; name = 'mm wave scan times';
    case{'F','XRayST1','X-Ray Scan Time 1'}
        column = 'F'; isinter = 0; name = 'X-Ray Scan Time 1';
    case{'G','XRayST2','X-Ray Scan Time 2'}
        column = 'G'; isinter = 0; name = 'X-Ray Scan Time 2';
    case{'FG','XRayST','XRayST12','X-Ray Scan Time 1 & 2'}
        dt = [readata('XRayST1'); readata('XRayST2')];
        name = 'X-Ray Scan Time 1 & 2';
        return
    case{'H','GetProT','Time to get scanned property'}
        column = 'H'; isinter = 1; d2s = 24 * 60;
        name = 'Time to get scanned property';
    otherwise
        error(['No data named ', type]);
end

data = xlsread('Data.xlsx',[column,':',column]) * d2s;
if isinter; dt = data; else; dt = diff(data); end
```

ArrivalTFit.m

```
function [lambdaPre, lambdaReg] = ArrivalTFit(isplot)
% The exponential fit to the interarrival times of precheck (left) and
% regular (right) passengers.
%
% Reference: Sovijja Pou, Daniel Kunin, Daniel Xiang. Brown University. ICM
% 2017 Problem D. Outstanding: Reducing Wait Times at Airport Security.
%
```

```
% Zhou Lvwen: zhou.lv.wen@gmail.com
% January 8, 2018
if nargin==0; isplot = 1; end

[PreArrT, PreName] = readata('PreArrT');
[RegArrT, RegName] = readata('RegArrT');

% Average inter-arrival time of precheck and regular passengers
lambdaPre = expfit(PreArrT); % [sec/passenger]
lambdaReg = expfit(RegArrT); % [sec/passenger]

% -----------------------------------------------------------------------

if ¬isplot; return; end
figure('position',[100,100,1000,450]);  % Figure 2

% Interarrival histogram (Regular PAX)
subplot(1,2,1);
hPre = histfit(PreArrT,10,'exponential');
legend(hPre(2),['Exponential Fit: lambda = ',num2str(lambdaPre)]);
xlabel('Wait time [sec]'); ylabel('Count'); title(PreName);

% Interarrival histogram (Precheck)
subplot(1,2,2);
hReg = histfit(RegArrT,10,'exponential');
legend(hReg(2),['Exponential Fit: lambda = ',num2str(lambdaReg)]);
xlabel('Wait time [sec]'); ylabel('Count'); title(RegName);
```

IDChkPdf.m

```
function [ti, fi] = IDChkPdf(isplot)
%
% Reference: Sovijja Pou, Daniel Kunin, Daniel Xiang. Brown University. ICM
% 2017 Problem D. Outstanding: Reducing Wait Times at Airport Security.
%
% Zhou Lvwen: zhou.lv.wen@gmail.com
% January 8, 2018

```

```
if nargin==0; isplot = 1; end
IdChkT = readata('IdChkT');[fi, ti] = ksdensity(IdChkT,'Support','positive');

% -------------------------------------------------------------------------

if ¬isplot; return; end
% Plots of ID Check process times 1 and 2
figure                             % Figure 3
hist(IdChkT,6); hold on
ax = plotyy(NaN,NaN, ti,fi,'plot');
xlim([0,45])

set(get(ax(1),'Ylabel'),'String','Count')
set(get(ax(2),'Ylabel'),'String','Probability Density')
xlabel('Time (sec)')
```

ScanPdf.m

```
function [xi, fiPre, fiReg] = ScanPdf(isplot)
% The kernel density estimate of precheck and regular passenger time spent
% in service station 2 (scanning).
%
% Reference: Sovijja Pou, Daniel Kunin, Daniel Xiang. Brown University. ICM
% 2017 Problem D. Outstanding: Reducing Wait Times at Airport Security.
%
% Zhou Lvwen: zhou.lv.wen@gmail.com
% January 8, 2018

if nargin==0; isplot = 1; end

Nbags = 2.26; % A passenger brings 2.26 bags on average

MMScanT = readata('MMScanT');
XRaySTs = readata('XRayST') * Nbags;

dx = 0.5; xi = 0:dx:120;
fiM = ksdensity(MMScanT, xi, 'Support', 'positive'); ciM = cumsum(fiM)*dx;
fiX = ksdensity(XRaySTs, xi, 'Support', 'positive'); ciX = cumsum(fiX)*dx;
```

```
fiPre = fiX.*ciM + fiM.*ciX;
GetProT = readata('GetProT');fiReg = ksdensity(GetProT, xi, 'Support', 'positive');

% -------------------------------------------------------------------------

if ¬isplot; return; end
% Plots of X ray scan time 1 and 2 combined
figure('position',[100,100,1000,450]);  % Figure 4
subplot(1,2,1);
plot(xi,fiM, ':r', xi, fiX, '--b', xi, fiPre, '-k','linewidth',2)
legend('mm Wave Scan','X-Ray Scan','Combined Scan')

subplot(1,2,2);
hist(GetProT,15); hold on
ax = plotyy(NaN,NaN, xi,fiReg,'plot');
set(get(ax(1),'Ylabel'),'String','Count')
set(get(ax(2),'Ylabel'),'String','Probability Density')
xlabel('Time (sec)')
```

simQ.m

```
function [Tq1,Tq2,Ttot,Vq1,Vq2,Vtot,P] = simQ(lambda,n,c,k,tc,pdfc,ts,pdfs,mode,p)
% TSA Security Simulation
%     lambda = the rate for the arrival process
%          n = the number of people that will arrive
%          c = is the number of id checkers
%          k = the number of x ray conveyor belts
%          p = the proportion of passangers that are slow
%       mode = 'none', 'FIFO' or 'priority'
%
% Reference: Sovijja Pou, Daniel Kunin, Daniel Xiang. Brown University. ICM
% 2017 Problem D. Outstanding: Reducing Wait Times at Airport Security.
%
% Zhou Lvwen: zhou.lv.wen@gmail.com
% January 8, 2018

if nargin<9; mode='none'; end

```

```
n = n + max(c,k);
slow = 2; fast = 0.5;
switch mode
    case 'FIFO'      % First In First Out
        nslow = round(n*p); % the number of passangers that are slow
        fac = fast*ones(1,n); fac(randperm(n,nslow)) = slow;
    case 'priority' % slow first, followed by the quick passengers.
        nslow = round(n*p); % the number of passangers that are slow
        fac = fast*ones(1,n); fac(1:nslow) = slow;
    otherwise        % mode = 'none'
        fac = ones(1,n);
end

[Arrival, ChkSta, ChkEnd, ScnSta, ScnEnd] = deal(zeros(n,1));

Csample = fac.*datasample(tc,n,'Weights',pdfc);
Ssample = fac.*datasample(ts,n,'Weights',pdfs);
Asample = exprnd(lambda,n,1);

for i = max(c,k)+1:n
    Arrival(i) = Arrival(i-1) + Asample(i);             % Arrival time
    ChkSta(i) = max( Arrival(i), min(ChkEnd(i-c:i-1)) );% Time start check
    ChkEnd(i) = ChkSta(i) + Csample(i);                 % Time end check
    ScnSta(i) = max(  ChkEnd(i), min(ScnEnd(i-k:i-1)) );% Time start scan
    ScnEnd(i) = ScnSta(i) + Ssample(i);                 % Time end scan
end

Tq1  = mean(ChkSta - Arrival); % Average wait time in queue 1
Tq2  = mean(ScnSta - ChkEnd ); % Average wait time in queue 2
Ttot = mean(ScnEnd - Arrival); % Average total waiting time

Vq1  = var(ChkSta - Arrival);  % Variance of wait time in queue 1
Vq2  = var(ScnSta - ChkEnd );  % Variance of wait time in queue 2
Vtot = var(ScnEnd - Arrival);  % Variance of total wait time

Cp = 6.60;       % Fee per one-way-trip
Ct = 18.5/3600; % The cost of a Transportation Security Officer (TSO)
```

```
P = Cp / Ttot - (c + 3 * k) * Ct;
```

theoQ.m

```
function [Tq1, Tq2, Ttot] = theoQ(lambda, n)
%
% Reference: Sovijja Pou, Daniel Kunin, Daniel Xiang. Brown University. ICM
% 2017 Problem D. Outstanding: Reducing Wait Times at Airport Security.
%
% Zhou Lvwen: zhou.lv.wen@gmail.com
% January 8, 2018

tc = readata('IdChkT');
ts = readata('GetProT');

Tq1 = (n-1)/2 * (mean(tc) - lambda  );
Tq2 = (n-1)/2 * (mean(ts) - mean(tc));

Ttot = Tq1 + mean(tc) + Tq2 + mean(ts);
```

SimTheo.m

```
% Reference: Sovijja Pou, Daniel Kunin, Daniel Xiang. Brown University. ICM
% 2017 Problem D. Outstanding: Reducing Wait Times at Airport Security.
%
% Zhou Lvwen: zhou.lv.wen@gmail.com
% January 8, 2018
%

% rand('seed',0)

npass = 50 + 200*[1:20];
lambda = 9;
[simQ1, theoQ1, simQ2, theoQ2] = deal(zeros(size(npass)));

[tc,    pdfc] = IDChkPdf(0);
[ts, ¬, pdfs] = ScanPdf(0);

for i = 1:length(npass)
```

```
    [simQ1(i),   simQ2(i)] =  simQ(9, npass(i),1,1,tc,pdfc,ts,pdfs);
[theoQ1(i), theoQ2(i)] = theoQ(lambda, npass(i));end

figure('position',[100,100,1000,450]);  % Figure 5

subplot(1,2,1)  % Q1
plot(npass,simQ1,npass,theoQ1, 'linewidth',2)
xlabel('Number of Passengers'); ylabel('Time Spent in Q1 (sec)');
legend('Simulated', 'Theoretical')

subplot(1,2,2)  % Q2
plot(npass,simQ2,npass,theoQ2, 'linewidth',2)
xlabel('Number of Passengers'); ylabel('Time Spent in Q2 (sec)');
legend('Simulated', 'Theoretical')
```

QTcVar.m

```
% Plots of time in Queues as function of arrival rate with constant k
%
% Reference: Sovijja Pou, Daniel Kunin, Daniel Xiang. Brown University. ICM
% 2017 Problem D. Outstanding: Reducing Wait Times at Airport Security.
%
% Zhou Lvwen: zhou.lv.wen@gmail.com
% January 8, 2018

n = 40; m = 4; k = 1;

lambda = 1:0.5:n;

[tq1,tq2,vq1,vq2] = deal(zeros(m,length(lambda)));

[tc,    pdfc] = IDChkPdf(0);
[ts, ¬, pdfs] = ScanPdf(0);

hbar = waitbar(0,'Please wait...');
for i = 1:length(lambda)
    waitbar(lambda(i)/n,hbar);
    for c = 1:m
```

```
        [tq1(c,i),tq2(c,i),¬,vq1(c,i),vq2(c,i)] = simQ(lambda(i),1e4,c,k,tc,pdfc,ts,pdfs);
end;end
close(hbar)

% Mean and Variance of time in queue 1

muIdChk = mean(readata('IdChkT') );
muScan  = mean(readata('GetProT'));

figure('position',[100,100,1000,450]);
subplot(1,2,1)
semilogy(lambda,tq1,'LineWidth',3)
xlabel('Arrival Rate (sec/passenger)'); ylabel('Time (log(sec))');
legend('c = 1', 'c = 2', 'c = 3', 'c = 4')
hold on
plot(muIdChk*1./[1:4;1:4]',get(gca,'ylim'))

subplot(1,2,2)
plot(lambda,vq1,'LineWidth',3)
xlabel('Arrival Rate (sec/passenger)'); ylabel('Variance (sec^2)');
legend('c = 1', 'c = 2', 'c = 3', 'c = 4')
hold on
plot(muIdChk*1./[1:4;1:4]',get(gca,'ylim'))

% Mean and Variance of time in queue 2
figure('position',[100,100,1000,450]);
subplot(1,2,1)
semilogy(lambda,tq2,'LineWidth',3)
xlabel('Arrival Rate (sec/passenger)'); ylabel('Time (log(sec))');
legend('c = 1', 'c = 2', 'c = 3', 'c = 4')
hold on
plot(muScan*1./[1:4;1:4]',get(gca,'ylim'))

subplot(1,2,2)
plot(lambda,vq2,'LineWidth',3)
xlabel('Arrival Rate (sec/passenger)'); ylabel('Variance (sec^2)');
legend('c = 1', 'c = 2', 'c = 3', 'c = 4')
```

```
hold on
plot(muScan*1./[1:4;1:4]',get(gca,'ylim'))
```

QTkVar.m

```
% Plots of time in Queues as function of arrival rate with constant c
%
% Reference: Sovijja Pou, Daniel Kunin, Daniel Xiang. Brown University. ICM
% 2017 Problem D. Outstanding: Reducing Wait Times at Airport Security.
%
% Zhou Lvwen: zhou.lv.wen@gmail.com
% January 8, 2018

n = 40; m = 4; c = 1;

lambda = 1:0.5:n;

[tq1,tq2,vq1,vq2] = deal(zeros(m,length(lambda)));

[tc,    pdfc] = IDChkPdf(0);
[ts, ¬, pdfs] = ScanPdf(0);

hbar = waitbar(0,'Please wait...');
for i = 1:length(lambda)
    waitbar(lambda(i)/n,hbar);
    for k = 1:m
        [tq1(k,i),tq2(k,i),¬,vq1(k,i),vq2(k,i)] = simQ(lambda(i),1e4,c,k,tc,pdfc,ts,pdfs);
    end;
end
close(hbar)

% Time spent in Queue 1 and Queue 2

muIdChk = mean(readata('IdChkT') );
muScan  = mean(readata('GetProT'));

figure('position',[100,100,1000,450]);
subplot(1,2,1)
```

```
semilogy(lambda,tq1,'LineWidth',2)
xlabel('Arrival Rate (sec/passenger)'); ylabel('Time (log(sec))')legend('k = 1', 'k = 2', 'k = 3', 'k = 4')
hold on
plot(muIdChk*1./[1:4;1:4]',get(gca,'ylim'))

subplot(1,2,2)
semilogy(lambda,tq2,'LineWidth',2)
xlabel('Arrival Rate (sec/passenger)'); ylabel('Time (log(sec))')
legend('k = 1', 'k = 2', 'k = 3', 'k = 4')
hold on
plot(muScan*1./[1:4;1:4]',get(gca,'ylim'))
```

profit.m

```
% Plot of cost of system as a function of c, k
%
% Reference: Sovijja Pou, Daniel Kunin, Daniel Xiang. Brown University. ICM
% 2017 Problem D. Outstanding: Reducing Wait Times at Airport Security.
%
% Zhou Lvwen: zhou.lv.wen@gmail.com
% January 8, 2018

rand('seed',2)
[lambdaPre, lambdaReg] = ArrivalTFit(0);
[tc,    pdfc] = IDChkPdf(0);
[ts, pdfp, pdfs] = ScanPdf(0);

n = 10;
[Ppre, Preg] = deal(zeros(n));

hbar = waitbar(0,'Please wait...');
for c = 1:n
    for k = 1:n
        waitbar((c-1)/n+k/n^2,hbar);
        [¬,¬,¬,¬,¬,¬,Ppre(c,k)] = simQ(lambdaPre,1e4,c,k,tc,pdfc,ts,pdfp);
        [¬,¬,¬,¬,¬,¬,Preg(c,k)] = simQ(lambdaReg,1e4,c,k,tc,pdfc,ts,pdfs);
    end
end
```

```
close(hbar)
% pre[i,j] = find(Ppre == max(Ppre(:)));
fprintf('The optimal precheck cost is %d with c = %d and k = %d\n',Ppre(i,j),i,j);
% regular
[i,j] = find(Preg == max(Preg(:)));
fprintf('The optimal regular cost is %d with c = %d and k = %d\n',Preg(i,j),i,j);

figure('position',[100,100,1000,450]);
subplot(1,2,1) % pre
imagesc(Ppre)
h = colorbar; xlabel(h,'$/sec')
ylabel('Check Stations (c)'); xlabel('Scan Stations (k)')
set(gca,'YDir','normal'); axis([0.5,c,0.5,k])

subplot(1,2,2) % regular
imagesc(Preg)
h = colorbar; xlabel(h,'$/sec')
ylabel('Check Stations (c)'); xlabel('Scan Stations (k)')
set(gca,'YDir','normal'); axis([0.5,c,0.5,k])
```

FifoPrio.m

```
% Plots of time in System as function of proportion slow
%
% Reference: Sovijja Pou, Daniel Kunin, Daniel Xiang. Brown University. ICM
% 2017 Problem D. Outstanding: Reducing Wait Times at Airport Security.
%
% Zhou Lvwen: zhou.lv.wen@gmail.com
% January 8, 2018

p = 0:0.01:1;
c = 2; k = 2;

[¬, lambda] = ArrivalTFit(0);

[tsys, vsys] = deal(zeros(2,length(p)));

[tc,     pdfc] = IDChkPdf(0);
```

```
[ts, ¬, pdfs] = ScanPdf(0);
hbar = waitbar(0,'Please wait...');for i = 1:length(p)
    waitbar(p(i),hbar);
    [¬,¬,tsys(1,i),¬,¬,vsys(1,i)] = simQ(lambda,1e4,c,k,tc,pdfc,ts,pdfs,    'FIFO',p(i));
    [¬,¬,tsys(2,i),¬,¬,vsys(2,i)] = simQ(lambda,1e4,c,k,tc,pdfc,ts,pdfs,'priority',p(i));
end
close(hbar)

% Time
figure
semilogy(p,tsys,'LineWidth',3)
legend('FIFO Discipline','Priority Discipline')
xlabel('Fraction Population Slow')
ylabel('Time (log(sec))')
set(gca,'fontsize',14)

% Variance
figure
semilogy(p,vsys,'LineWidth',3)
legend('FIFO Discipline','Priority Discipline')
xlabel('Fraction Population Slow')
ylabel('Variance (sec^2)')
set(gca,'fontsize',14)
```