Demand Prediction and Dynamic Workforce Allocation to Improve Airport Screening Operations

Girish Jampani Hanumantha, Berkin T. Arici, Jorge A. Sefair, and Ronald G. Askin
School of Computing, Informatics, and Decision Systems Engineering
Arizona State University, Tempe, AZ
{gjampani, barici, jorge.sefair, ron.askin}@asu.edu

Abstract: Workforce allocation and configuration decisions at airport security checkpoints (e.g., number of lanes open) are usually based on passenger volume forecasts. The accuracy of such forecasts is critical for the smooth functioning of security checkpoints where unexpected surges in passenger volumes are handled proactively. In this article, we present a forecasting model that combines flight schedules and other business fundamentals with historically observed throughput patterns to predict passenger volumes in a multi-terminal multi-security screening checkpoint airport. We then present an optimization model and a solution strategy for dynamically selecting a configuration of open screening lanes to minimize passenger queues that at the same time determine workforce allocations. We present a real-world case study in a US airport to demonstrate the efficacy of the proposed models.

Key words: Airport security screening, passenger volume forecasting, workforce planning, queueing optimization

1. Introduction

The US Aviation and Transportation Security Act of 2001 established the Transportation Security Agency (TSA) with the mission to “protect the nation’s transportation systems to ensure freedom of movement for people and commerce” [1]. As part of that mission, TSA screens departing passengers and baggage at every airport in the country. In 2018, TSA screened over 2 million travelers a day nationwide, which accounts for a total of more than 813 million passengers and crew members screened during the entire year [51]. These screening operations are of great importance for national security due to the severe consequences of potential malicious attacks, like those perpetrated in September 11, 2001.

Thorough screening is time consuming and may constitute an important portion of the passenger’s time spent at the airport. Moreover, long waits at security queues are a source of passenger discomfort and frustration. Due to the increasing passenger volumes in the airline industry, maintaining a balance between wait times at screening facilities and security levels is a current challenge for TSA. Critical decisions for efficient screening include the number of screening lanes open, and hence required number of Transportation Security Officers (TSOs) needed at any time. Due to airport space
constraints, expected growth in air travel, and increasing security threats, new demand prediction and workforce management tools are needed to improve TSA’s operational efficiency.

TSA conducts screening procedures for all departing passengers and carry-on baggage prior to boarding. To execute these procedures, US airports have a number of Security Screening Check Points (SSCPs) that are the only access point to departure gates. Each SSCP has one or more queues leading up to Travel Document Checkers (TDCs), who verify every passengers boarding pass against their identity. Upon identity verification, TSOs at the TDC stations direct passengers to the primary inspection lanes for screening. Typically, each inspection lane has its own conveyor belt feeding a scanner, a walk-through metal detector (WTMD) and an Advanced Imaging Technology (AIT) body scanner. A person or an item that fails to pass the primary inspection is directed to a secondary inspection (open bag or pat-down). Randomly selected passengers may also be asked to pass additional checks such as a chemical residue test. In 2013, TSA launched TSA Pre-Check that gives low-risk travelers the convenience of going through a dedicated lane without the need to remove selected items (such as laptop, shoes, etc.). New technologies and passenger classes are constantly being considered by TSA to enhance security and passenger convenience.

Despite current efforts to improve their efficiency, the operation of SSCP is still challenging due to the uncertainty in the number of passengers arriving at a checkpoint at any given time. Although the number of passengers on a flight is known to some degree of accuracy, the time at which they arrive at the SSCP prior to departure is unknown. Other factors such as flight delays, gate changes, and number of passengers making a connection also contribute to the demand uncertainty. To keep security lanes as short as possible in this uncertain environment, TSA dynamically opens and closes screening lanes according to labor constraints and predicted (or observed) demands. Although ideal from a passenger perspective, keeping all lanes open all the time at an SSCP is not economically efficient, whereas having few lanes open may lead to long passenger wait times. As a result, accurate demand predictions and automated models to determine optimal SSCP configurations may have great impact of TSA operations.

In this article we focus on two critical tasks to improve SSCP operations: passenger demand forecasting and SSCP configuration decisions including workforce allocations. In Section 2, we review the existing literature on similar problems and frame our contribution. Section 3 discusses our proposed passenger forecasting model, which combines inherent variables to the travel industry (e.g., flight schedules, aircraft occupancy, expected arrival time at the SSCP prior to departure) and a learning model to capture historical trends. In Section 4, we embed the passenger forecast from Section 3 into an optimization model that seeks optimal SSCP configurations to minimize worst-case queue lengths and determine the required workforce allocations. Our optimization model also captures several realistic operational aspects such as available labor, additional TSOs allocations
(e.g., part-time), and operational constraints to prevent drastic changes in the SSCP configuration. We also propose a solution strategy to incorporate other business metrics such as minimization of additional TSOs needed and minimization of queue lengths at non-peak times. Section 5 describes existing data sources in the US that could be used to estimate the parameters of our models. Section 6 describes a case study based on our experience in Phoenix (Arizona) Sky Harbor airport. Although our results focus on a particular airport, our models could be applied to any airport with commercial passenger traffic. Section 7 presents the final remarks and outlines future work.

2. Literature Review

The adoption of risk based airport security led to modelling efforts to facilitate optimal allocation of security resources while allocating passengers classified into different risk levels to classes (sequence of screening operations). Discrete optimization models (deterministic and stochastic) have been used extensively to gain insight on how allocation of passengers to screening classes can improve the effectiveness of screening operations. Babu et al. [4] presents an integer program to randomly allocate passengers to classes (or sequence of screening operations) with an objective of minimizing false alarms while adhering to an Federal Aviation Authority (FAA) mandate on false clears under the assumption that all passengers have the same risk level. McLay et al. [32] propose a Multi-level Allocation Problem (MAP) to study the optimal allocation of passengers of various risk levels to security classes (or sequence of screening operations) under budget constraints to maximize the screening efficacy. McLay et al. [33] develop an enhancement to the MAP, the Multi-level Passenger Screening Problem (MPSP), with the additional assumptions of preexisting screening devices infrastructure and their availability. Nikolaev et al. [36] develop a two stage stochastic optimization model to simultaneously make screening device allocation and passenger assignment decisions. McLay et al. [34] refine the model developed by Nikolaev et al. [36] to facilitate real-time assignment of passenger to classes through a Markov Decision Process framework. Lee et al. [25] model the real-time assignment of passengers to security classes as a nonlinear control problem, whereas Nikolaev et al. [37] examines optimal policies for assignment of passengers to classes in a multistage screening process under the assumption that perceived risk levels of passengers are dynamically updated at each stage. We assume the formation of passenger classes and assignment is known prior to passengers arriving at the airport, however, such dynamic assignments could be considered in the future to expand our decision model.

The relation between screening operations decisions and passenger satisfaction have been studied through statistical models. Gkritza et al. [18] study the relation between security levels and passenger satisfaction through the use of multinomial-logit models. Sakano et al. [42] employ a statistical framework to understand how screening operations affect passenger satisfaction and their perception
of public transit safety. Other papers have focused on assessing quantitative models employed in practice. Peterson et al. [40] detailed how analytical models developed by the United States Commercial Aviation Partnership (USCAP), leveraged by the Transportation Security Administration (TSA) in various facets of aviation security, balance the impact of screening procedures on passengers and airlines. Lee et al. [26] presented an exhaustive survey on applications of Operations Research (OR) to aviation security and Wu and Mengersen [54] provide an exhaustive review of quantitative models being employed in various facets of airport operations with a discussion on airport security planning. Kirschenbaum [22], Stewart and Mueller [47] and Gillen and Morrison [16] examine the costs and economic aspects associated with airport security operations and identified potential opportunities for savings. Bagchi and Paul [5] apply game theoretic principles to study how a combination of profiling and screening strategies can be employed to improve airport security.

Passenger forecasting in air transport systems has been widely studied due to its importance in strategic, tactic, and operational airport decisions. Existing models in the literature use multiple forecasting techniques, including classic regression analysis, machine learning techniques, and hybrid models combining multiple approaches, among others. Xie et al. [55] use a hybrid approach that combines time-series with least-squares support vector regression to predict short-term air passenger volumes. For short-to-medium term air passenger demand predictions (i.e., one year ahead or closer), Dantas et al. [11] use a model combining bootstrapping aggregation with a more traditional Holt-Winter method and Nieto and Carmona-Benítez [35] propose a combination of auto-regressive time-series models with bootstrap sampling. Gelhausen et al. [15] develop a co-integration theory model as an alternative to the traditional models used by the German Aerospace Center to predict long-term passenger and flight volumes at German airports. Scarpel and Pelicioni [43] present an ensemble Mixture-of-Experts Model (MEM) to predict congested days at São Paulo’s Airport in Brazil. Additional closely related works include [29] and [41], who use flight schedules and industry specific parameters to create passenger arrival patterns for simulation processes.

Queuing models have been employed to improve system design while analyzing trade-offs between risk and congestion. Gilliam [17] is one of the earliest works on modelling airport screening operations and analyzing these operations by employing steady state (equilibrium) queueing results. In the aftermath of the September 2001 attacks, Pendergraft et al. [39] developed a detailed discrete event simulation which captured “curb-to-gate” activities of passengers to facilitate passenger checkpoint redesign and understand operational dynamics at the Baltimore/Washington International (BWI) airport. Wilson et al. [53] present a 2-dimensional spatially-aware discrete event simulation model that incorporates physical space concerns, passenger behavior, and queueing dynamics aiming to perform what-if analyses in security checkpoint layout decisions. Leone and Liu [27] study a single lane airport checkpoint as a two stage serial open queueing network of $M/M/1$ queues. Zhang et al.
[56] analyze trade-offs between customer service and security levels at security check wait lines through the use of a stylized two stage queuing network. They assumed the service distribution at the first stage to be a two phase Phase-type distribution with the second phase representing additional secondary inspection. Lee and Jacobson [24] model screening operations as a multiclass queueing system of parallel servers with a single server dedicated exclusively to a single class. They presented optimal passenger assignment policies to classes under steady-state and transient conditions. Stolletz [48] computes time-dependent performance measures at airport terminals by modelling screening operations as a nonstationary $M(t)/G/c(t)$ queue.

To improve airport operations, calculated demands must inform staffing and queueing models to determine workforce levels at every stage of the travel experience (e.g., airline counters, security checkpoints, baggage processing) that are according to the pursued quality of service. Although the literature on staffing and workforce allocation problems for general systems is vast, specific applications to airport operations are scattered. For this reason, we first review the existing general staffing models that can be applicable to airport operations. In particular, we focus on queueing systems characterized by non-stationary arrivals. Defraeye and Nieuwenhuyse [13] provide an extensive literature review, analysis, and classification of such existing approaches. Ingolfsson et al. [20] present an algorithmic solution to a scheduling problem that brings queueing theory and optimization together. Atlason et al. [3] emphasize the importance of accurate queueing models and the inter-temporal effect of decisions, where staffing levels in one period can affect the quality of service in subsequent periods. To this end, they propose an optimization model coupled with a simulation procedure to evaluate the queue performance under multiple staffing levels. Liu and Whitt [31] provide a systematic formula-based approach to determine the number of servers to open for each period throughout a planning horizon under time-varying arrival rates. Similarly, Bhandari et al. [6] study a call center with adjustable service rate determined by temporary hires as needed to meet performance requirements. They combine a Markov decision process with an integer programming model to find optimal staffing levels with a state dependent threshold policy. Defraeye and Van Nieuwenhuyse [14] develop an iterative staffing algorithm that uses a queue simulation to determine staffing levels while controlling the probability of excessive wait times. Al-sultan [2], Bruno and Genovese [7], Parlar and Sharafali [38], and Stolletz [48] develop staffing and rostering models at passenger airline check-in counters. Littler and Whitaker [30] study a similar staffing problem at an airport’s immigration checkpoint.

Staffing and workforce allocation models for queue networks of screening facilities have received less attention in the literature. De Barros and Tomber [12], Wang et al. [52], and Leone and Liu [28] analyze the security processes at an airport from a queueing-theoretic and simulation point of view. Soukour et al. [45] and Soukour et al. [46] study staffing problems at airport security checkpoints,
focusing on the assignment of employees to shifts based on skills and contractual requirements. Similarly, Lange et al. [23] investigates the cost savings of accurate passenger arrival estimations, which is coupled with a simulation-based personnel scheduling model under a virtual queueing setting. Kierzkowski and Kisiel [21] utilizes airline data to forecast the passenger arrivals, which are fed into a simulation model to determine service levels. These decisions are dynamically made according to an algorithm that tracks the observed system state as the simulation advances.

Although some related works exist, there is no approach in the literature that can be used for the specific problem tackled in this article. Existing forecasting models focus on predicting general passenger or flight volumes at an airport, but not specifically passenger arrivals at a security checkpoint. These are more difficult to predict, as they are needed at a finer scale (i.e., 10-minute intervals) and are noisier due to last minute gate changes and flight delays, which may occur after a passenger is already at the airport. There is no approach in the literature that combines a causal (nonlinear) mechanistic prediction with machine learning models to capture both the business fundamentals and historical trends. As a result, the advantage of the presented model is that it enhances the causal prediction using observed patterns. Further, we provide a new mathematical programming description of a queueing network model describing multiple stages within an airport’s security checkpoint. Existing models in the literature for general queueing systems either use oversimplified queue descriptions or rely on simulation-optimization techniques, which may be computationally expensive. Our proposed model presents a good balance between realism and tractability, including some operational constraints that make our models unique.

3. Passenger arrival estimation

In this section we describe our approach to estimate the volume of passengers arriving at each Security Screening CheckPoint (SSCP). The approach consists of two components. The first component is a mechanistic model to predict passenger arrivals based on business fundamentals such as flight departure schedules, airplane capacities, expected number of passengers, among others. The second component is a learning model to adjust the mechanistic prediction based on historical patterns. Throughout this article we refer to the volume of departing passengers arriving at each SSCP as passenger arrivals.

3.1. Mechanistic passenger arrival model

We estimate passenger arrivals in each 10-minute interval for each day during the period of analysis. We describe these dynamic arrivals using a causal model based on inherent factors to the travel industry. The most evident driver is the number and type of flights that are scheduled to depart in
upcoming time intervals. Different departure schedules capture seasonal variations in airline operations, and consequently yield different passenger arrival patterns. We refer to this mechanistic factor as a schedule. Additionally, aircraft capacity associated with departing flights is crucial. Aircraft equipment type and seat configuration define the maximum number of passengers. However, many flights are not full and we include the expected percentage of occupancy, or load factor, into our mechanistic model.

In the US, most airlines use a hub-and-spoke model for their operations. As a result, direct flights are not always available and passengers may have to change flights at a hub airport. Passengers making domestic connections usually stay in the secured side of the terminal and thus avoid passing through a SSCP a second time. We refer to this factor as the percentage of originating passengers. Depending on their preferences, passengers arrive at the SSCP at some time prior to the scheduled departure time. These arrival times may vary considerably and can drastically affect the passenger arrival predictions. We incorporate these behaviors by using an earliness arrival distribution, which provides the proportion of passengers arriving at various times prior to departure. Finally, departing flights are associated with a SSCP based on the location of their predicted departure gate.

To predict passenger arrivals at a specific SSCP, the proposed mechanistic model estimates the expected number of passengers on each scheduled flight and distributes them into time intervals prior to departure according to the earliness arrival distribution. This prediction is further adjusted by including the originating percentage and the expected distribution of flights into gates, which determines the SSCP used by passengers. Mathematically, we define $D$ as the set of days for which predictions are needed (e.g., days during the next two weeks), $T$ as the set of time periods in which each day is partitioned (e.g., 10-minute intervals), $F_j$ as the set of scheduled flights on day $j \in D$, and $G$ as the set of SSCPs. For any day $j \in D$, we denote the capacity, load factor, and percentage of originating passengers of flight $k \in F_j$ by $c_k, l_k, \text{ and } o_k$, respectively. Moreover, we use parameter $\alpha_{kt}$ to represent the probability a passenger arrives at time period $t \in T$ prior to departure of flight $k \in F_j$. If the departure time of flight $k$ is $\bar{t}$, then $\alpha_{kt} = 0, \forall t > \bar{t}$. We distribute flights across security checkpoints depending on their gate assignment. To do so, parameter $r_g$ represents the percentage of flights assigned to security checkpoint $g \in G$. Using these elements, we calculate the estimated number of passengers at security checkpoint $g \in G$ for time interval $t \in T$ on day $j \in D$ as

$$p_{gjt} = r_g \sum_{k \in F_j} c_k l_k o_k \alpha_{kt}. \quad (1)$$

Note that for any given flight $k \in F_j$, the term $c_k l_k o_k$ in Equation (1) calculates the estimated number of passengers after correcting the aircraft capacity by the load factor and the percentage of originating passengers. We estimate the parameters required in Equation (1) using information on scheduled flights as well as data from the same day and time from previous years. We describe possible data sources to estimate the parameters in (1) in Section 5.
3.2. Learning approach to adjust mechanistic model

A challenging issue in the mechanistic approach is the disparity in the time scales for different input data. As a result, the predicted variability achieved by the mechanistic model is limited by the granularity of the input data. For example, available load factors from the Bureau of Transportation Statistics [8] are the same for all days of the week and hour of the day. Intuitively, load factors may change throughout the day and across days. Moreover, unpredictable factors like gate changes, variability in the earliness arrival curves per type of flight (e.g., business vs. leisure trips), and flight delays, among others, are not captured by the mechanistic approach due to the absence of available data. Thus, for instance, we approximate $\alpha_{kt}$ by $\alpha_k$.

We propose a simple learning model based on historical passenger volumes to improve the predictive performance of the mechanistic model. This model is able to capture a higher level of detail that is not possible with available input data. By doing so, available information is adjusted from a monthly level—which is the frequency of some of the mechanistic model’s input data—to a daily or even hourly level. Moreover, this approach allows us to capture unpredictable factors that are not included the mechanistic model but that play a role in the passenger arrival predictions. This allows accommodation of factors such as airport staff passing through the SSCPs. Our learning approach is a hybrid forecasting model that uses mechanistic model estimates at a prior date and compares them with observed passenger arrivals at each SSCP. Because passenger arrivals at the beginning of the SSCP’s queue are not generally recorded, we use the number of screened passengers as a proxy (i.e., passengers that passed through an AIT full body scanner or walk-through metal detector) [50]. We then estimate a set of adjusting factors by minimizing the sum of squared errors between the mechanistic prediction and the observed throughput under certain rules to prevent overfitting. These multiplicative factors are used to adjust current mechanistic predictions, which are validated using a reserved sample of observed throughput (see Section 6).

Formally, let $H$ be the set of hours of the day and let $S_j$ be a sample of prior days to train the adjusting factors for day $j \in D$. For instance, to calculate the adjusting factors for a given Monday in the future, the corresponding $S$-set may include the same day in prior years. The construction of these sets is flexible so the analyst can include any subset of days that are relevant for the training. To perform the data training, we require the throughput data as well as the information needed to estimate the parameters of the mechanistic model to be available for each day in $S_j$ for all $j \in D$. Moreover, we assume that throughput data is available per hour, so that the mechanistic predictions are at a finer scale (i.e., $|T| \geq |H| = 24$). For each day $j \in D$, we define $\hat{p}_{gsh}$ as the aggregated mechanistic predictions for hour $h \in H$ of day $s \in S_j$ at security checkpoint $g \in G$. That is, if the time scale in the mechanistic model is set to 10-minute intervals, then $\hat{p}_{gsh}$ is the sum of $p_{gst}$
over all six time periods $t$ belonging to hour $h$. We use $\tau_{gsh}$ to denote the observed throughput at security checkpoint $g \in G$ during hour $h \in H$ of day $s \in S_j$. The adjusting factors for the mechanistic predictions for hour $h \in H$ of day $j \in D$ at terminal $g \in G$ are given by $\beta_{gjh}$.

The estimation of the adjusting factors for hour $h \in H$ of day $j \in D$ at security checkpoint $g \in G$ is performed by the constrained least-squares training model $[T_{gjh}]$ shown in (2)–(3). In this model, parameters $l_{gh}$ and $u_{gh}$ control the strength of the correction applied to the mechanistic model to prevent overfitting during the training phase, which could deteriorate the quality of the prediction. For instance, an $l$-parameter equal to 0.75 allows the mechanistic prediction to be reduced by no more than 25%, whereas an $u$-parameter equal to 1.25 allows the mechanistic prediction to be increased by no more than 25%.

\[
[T_{gjh}] \min \sum_{s \in S_j} (\tau_{gsh} - \beta_{gjh} \hat{p}_{gsh})^2 
\]

\[
\text{s.t. } l_{gh} \leq \beta_{gjh} \leq u_{gh} \quad (3)
\]

The objective function in (2) minimizes the sum of squared errors between the observed throughput and the corrected mechanistic prediction for each of the training days in $S_j$. Note that there is a single adjusting parameter for each checkpoint-day-hour combination that needs to be estimated to minimize the error across training days subject to the bound constraints in (3). Further, model $[T_{gjh}]$ is a convex univariate optimization problem with linear constraints that can be solved using first-order optimality conditions. Solving $[T_{gjh}]$ for each checkpoint-day-hour combination provides a full set of $\beta$-parameters that can be used to adjust mechanistic predictions.

For further illustration, we provide the steps required by our learning approach to estimate passenger arrivals for a set of days $D$ in 2019, assuming that the training set for any day $j \in D$ (i.e., $S_j$) consist of all same days of the week for the same month in 2018. For instance, the training set of a Monday in March 2019 consists of all Mondays in March 2018. This approach may be useful in airports with strong seasonal patterns so that weekly or monthly airline operations are similar across consecutive years (i.e., same number of flights due to seasonal patterns).

**Step 1:** Using (1), construct previous year $y - 1$ (e.g. $y - 1 = 2018$) mechanistic estimations for each training day $s \in S_j$, SSCP $g \in G$, and time interval $t \in T$. To calculate the year $y - 1$ parameters needed in (1), use data from $y - 2$ or $y - 1$ that is available prior to day $s \in S_j$, for each $j \in D$. As a result, obtain $y - 1$ values of $p_{gst}$, for each $g \in G$, $s \in S_j$, and $t \in T$.

**Step 2:** Construct aggregated mechanistic predictions $\hat{p}_{gsh}$ by adding $p_{gst}$ over all time periods $t$ belonging to each hour $h$. Observed throughput for checkpoint $g$, training day $s$, and hour $h$ must be available.
**Step 3:** Using observed throughput and aggregated mechanistic predictions from Step 2, solve $[T_{gjh}]$ for each checkpoint $g$, day $j$, and time interval $h$ required for the year $y$ prediction. Obtain adjusting factors $\beta_{gjh}$.

**Step 4:** Using (1), construct mechanistic estimations for year $y$. To keep the estimation consistent, use the same data sources and estimation methods as in Step 1. Obtain year $y$ mechanistic predictions $p_{gjt}$.

**Step 5:** Using the adjusting factors from Step 3, calculate the corrected mechanistic prediction for year $y$, which are given by $\lambda_{gjt} = \beta_{gjh}p_{gjt}$ for each checkpoint $g$, day $j$, and time interval $t$. Note that the factor used to adjust $p_{gjt}$ is that for which time period $t$ belongs to hour $h$.

The importance in the consistency requirement in Step 4 is that the learning method is designed to correct missing aspects of the prediction provided by the mechanistic model in (1). If the mechanistic estimations are constructed using different methodologies, then the adjusting factors may be misleading as they were trained to correct something that the current mechanistic predictions may not be lacking. The trade-off between model complexity and generalizability is an important aspect of machine learning models. Model complexity for the proposed learning model is controlled by varying upper and lower bounds $l_{gh}$ and $u_{gh}$ in (3), which are viewed as hyper-parameters. Model selection is performed using cross-validation to determine the best values of $l$- and $u$-parameters. Cross validation is a procedure employed to avoid over-fitting the training data and ensure the chosen bounds result in a model which generalizes well [19]. To do so, training is performed by removing one training day at a time, fitting the adjusting factors, and then predicting the throughput for the removed day. The values with minimal cumulative prediction error are selected. Because of the small sample size of the training data, the applicability of more sophisticated cross validation procedures (such as $k$-fold cross validation or holdout sampling) is limited. Based on initial empirical limits and our understanding of the factors excluded from the mechanistic model, a grid search was conducted for bounds from 0 to 2 with a step size of 0.1. Once the empirical bounds are obtained, the optimization problem $T_{gjh}$ is solved using all training days in $S_j$ to obtain the final adjusting factors.

4. **Optimal checkpoint configuration and workforce allocation**

Accurate passenger arrival predictions are important as they can inform staffing decisions seeking a high quality of service at each SSCP even under high volumes of passengers. In Section 4.1, we propose an optimization model to find an optimal configuration and staffing of each checkpoint in terms of the number of TDC stands and identical parallel screening lanes operating at any given time. Our model incorporates operational requirements such as available labor, business rules to
open and close a TDC or a screening lane, and queue dynamics for each candidate configuration. In Section 4.2, we propose a solution approach to explore the alternative optimal solutions provided by the model in Section 4.1. The approach incorporates multiple desired outcomes such as minimum worst-case queue length across the day, minimum additional TSOs needed to achieve a service level, and minimum queue length at any time.

4.1. Mathematical programming model

Our model evaluates the performance of each checkpoint using the expected queue length for each time interval of the day, which is driven by the passenger arrivals and the chosen shift configuration. From a managerial point of view, it is justifiable to minimize the maximum queue length across the day as the quality of service perceived by a passenger is directly related to their own wait time rather than an overall daily performance. To calculate queue lengths, our model embeds the governing equations of a multi-server queueing network model, whose parameters are a function of the configuration decisions. We model a SSCP as a sequence of two stages, TDC and baggage/passenger screening, at which passengers can spend time in queue. As baggage screening is typically slower than passenger screening (i.e. passage through an AIT or WTMD), we utilize baggage processing rates per passenger as screening rates. A configuration is defined by the number of TDCs and screening lanes open at a point in time. Mathematically, we define $C$ as the set of TDC areas in the airport and $A$ as the set of screening areas. Each checkpoint consists of one TDC area and one screening area. A TDC area has the capacity for multiple TDC stands and a screening area has multiple screening lanes. TDC and screening areas have a set of possible configurations given by $K_i$, for $i \in C \cup A$. The finite set of candidate configurations is determined by each checkpoint’s layout and available equipment (e.g., TDC stations, AIT machines). To prevent drastic changes in a checkpoint configuration over time, we define $\bar{K}_i(\ell)$ as the set of incompatible configurations with configuration $\ell$ at area $i \in C \cup A$. If configuration $\ell$ is selected at time $t$ for area $i$, then none of the configurations in $\bar{K}_i(\ell)$ can be selected at $t+1$ for the same area. These rules prevent checkpoint’s reconfigurations that are undesirable in practice. Decision variable $x_{ilt}$ equals one if configuration $\ell$ is used at area $i$ during time period $t$, and is equal to zero otherwise.

To model the queueing process, we define $Q_{it}$ as the queue length at the start of an interval and $\mu_{it}$ as the effective service rate at area $i$ during time $t$, respectively. Effective service rate is defined by the configuration (number of active lanes) and the TSA standard throughput rate per lane. Note that these factors can be adjusted based on specific SSCP characteristics such as space limitations or specific policies of the airlines served by that SSCP. We use parameter $\lambda_{it}$ to denote the estimated passenger arrivals to TDC area $i$—i.e., to the SSCP where area $i$ is located—during time period $t$. This parameter is estimated using the results in Section 3. The number of processed passengers at
any area is bounded by the number of passengers available and the processing capacity. To formally model this relationship, we define \( c_{it} \) as the number of passengers processed in area \( i \) at time \( t \) and write these conditions as

\[
c_{it} = \begin{cases} 
\min \{Q_{i,t-1} + \lambda_{it}, \mu_{it}\}, & i \in C \\
\min \{Q_{i,t-1} + c_{i-1,t}, \mu_{it}\}, & i \in A,
\end{cases}
\]

where \( c_{i-1,t} \) is the number of passengers processed at the TDC area preceding screening area \( i \) at time \( t \). In other words, \( c_{i-1,t} \) for \( i \in A \) is the number of passengers entering screening area \( i \) after clearing the TDC procedures. The processing rate of area \( i \) under configuration \( \ell \) is given by \( r_{i\ell} \) and the maximum queue length observed across all times \( t \in T \) is given by \( \hat{Q} \).

To capture existing labor constraints, we define \( L_t \) as the number of TSOs assigned to the SSCP under analysis at time \( t \). Moreover, \( l_{i\ell} \) denotes the number of TSOs required to operate area \( i \) under configuration \( \ell \). We also assume that a number of flexible TSOs is available to work at the SSCP if needed. However, their working shifts need to be determined depending on the SSCP needs. Our model includes these decisions using a set of predetermined shifts. Each shift is a collection of times in which a TSOs is on duty. The set of shifts, which we denote by \( \Omega \), reflects agency-specific constraints such as shift duration (i.e., part- and full-time), breaks, among other features. We use the decision variable \( h_f \) to determine the number of flexible TSOs working under shift \( f \in \Omega \). The number of available flexible TSOs is no more than \( B \) (i.e., beyond those included in \( L_t \)). Our proposed optimization model to determine optimal SSCP configurations and workforce allocation decisions is shown in (5)–(19).

\[
\begin{align*}
\min & \quad \hat{Q} \\
\text{s.t.} & \quad \sum_{\ell \in K_i} x_{i\ell t} = 1, \quad i \in C \cup A, \quad t \in T \\
& \quad \sum_{\ell \in K_i} r_{i\ell} x_{i\ell t} = \mu_{it}, \quad i \in C \cup A, \quad t \in T \\
& \quad \sum_{i \in C \cup A} \sum_{\ell \in K_i} l_{i\ell} x_{i\ell t} \leq L_t + \sum_{\{f \in \Omega : t \in f\}} h_f, \quad t \in T \\
& \quad \sum_{f \in \Omega} h_f \leq B \\
& \quad Q_{it} \leq \hat{Q}, \quad i \in C \cup A, \quad t \in T \\
& \quad Q_{i,t-1} + \lambda_{it} \geq c_{it}, \quad i \in C, \quad t \in T \\
& \quad Q_{i,t-1} + c_{i-1,t} \geq c_{it}, \quad i \in A, \quad t \in T \\
& \quad \mu_{it} \geq c_{it}, \quad i \in C \cup A, \quad t \in T \\
& \quad Q_{i,t-1} + \lambda_{it} - c_{it} = Q_{it}, \quad i \in C, \quad t \in T
\end{align*}
\]
\begin{align}
Q_{i,t-1} + c_{i-1,t} - c_{it} &= Q_{it}, \ i \in A, \ t \in T \\
x_{it} &\leq 1 - x_{i,\ell,t+1}, \ i \in A, \ \ell \in K_i, \ \bar{\ell} \in \bar{K}_i(\ell), \ t \in T \\
x_{it} &\in \{0,1\}, \ i \in C \cup A, \ \ell \in K_i, \ t \in T \\
\mu_{it}, Q_{it}, c_{it} &\geq 0, \ i \in C \cup A, \ t \in T \\
h_f &\in \mathbb{N}, \ f \in H
\end{align}

Constraints (6) guarantee that exactly one configuration is selected for each area and time period. The effective processing rate at each area is calculated in Constraints (7) as a function of the chosen configuration. The labor availability is imposed in Constraints (8) for each time period, where the left hand side is the number of officers needed airport-wide given the chosen configurations and the right hand side is the number of TSOs available including those under flexible schedule. Constraint (9) imposes a limit on the maximum number of flexible TSOs available, which in this case we model as a cardinality constraint because we assume that any officer earns the same wage. Note that the flexible TSO feature could also serve to model use of overtime. Constraints (10) enforce that \( \hat{Q} \) is greater than or equal to the maximum queue length observed in any area at any time, which together with the objective function in (5) helps minimize the maximum queue length. Constraints (11)–(13) linearize Equation (4) for each area and time. Note that given the model’s objective to limit queue lengths, one constraint between (11) and (12) or (13) will be binding depending on the area. The checkpoint’s queue dynamics are governed by Constraints (14) and (15). Intuitively, these constraints capture that the difference between the number of passengers and the number of passengers processed at an area at a given time determines the passengers in queue for the next time period. Constraints (17)–(19) define the nature of the model’s decision variables.

We incorporate the operational rules related to changes in the SSCP configuration by using Constraints (16), which guarantee that consecutive configuration changes (i.e., configurations at times \( t \) and \( t + 1 \)) in an area follow the compatibility rules, meaning that no drastic changes are allowed. In this case, if \( x_{it} = 1 \), then all the variables related to the selection of incompatible configurations for the same area at time \( t + 1 \) must be equal to zero. For example, if the configuration for screening area \( i \) at time period \( t \) consist of three screening lanes, then the allowed configurations for \( t + 1 \) may be two, three, or four lanes only, and no other configuration can be selected.

Constraints (16) narrow down the possibilities for a configuration change between consecutive time periods. However, they allow repeated changes every time period within the set of compatible options. This situation may be undesirable if setup costs exist, dictating that once a configuration is selected, then it needs to be operated for some time. To address this issue, we introduce parameter \( M \) to describe the minimum number of time periods that a configuration needs to be maintained.
after being selected. Moreover, we introduce binary decision variables \( \delta_{a}^{i \ell t} \) and \( \delta_{n}^{i \ell t} \) to record whether configuration \( \ell \) of area \( i \) is changed at time \( t \). In particular, \( \delta_{a}^{i \ell t} \) determines whether configuration \( \ell \) is adopted, whereas \( \delta_{n}^{i \ell t} \) indicates that configuration \( \ell \) is no longer in use. Under this new approach, we need to remove Constraints (16) and add the following new constraints.

\[
\begin{align*}
    x_{i \ell t} - x_{i, \ell, t+1} + \delta_{a}^{i \ell t} - \delta_{n}^{i \ell t} &= 0, \quad i \in C \cup A, \quad \ell \in K, \quad t = 1, \ldots, |T| - 1 \\
    \sum_{\ell \in K} \delta_{a}^{i \ell t} &\leq 1 - \sum_{\ell \in K} \delta_{n}^{i \ell t}, \quad i \in C \cup A, \quad t = 1, \ldots, |T| - M, \quad m = 1, \ldots, M, \quad u, v \in \{ a, n \} \\
    \delta_{a}^{i \ell t} &= 0, \quad i \in C \cup A, \quad \ell \in K, \quad t = |T| - M + 1, \ldots, |T|, \quad u \in \{ a, n \} \\
    \delta_{n}^{i \ell t} &\in \{0, 1\}, \quad i \in C \cup A, \quad \ell \in K, \quad t = 1, \ldots, |T| - M, \quad u \in \{ a, n \}
\end{align*}
\]

Constraints (20) track whether configuration \( \ell \) in area \( i \) is changed between times \( t \) and \( t + 1 \). If this is the case, \( x_{i \ell t} - x_{i, \ell, t+1} \) is equal to one or negative one, which requires \( \delta_{a}^{i \ell t} \) or \( \delta_{n}^{i \ell t} \) to be equal to one. As a result, a \( \delta \)-variable equal to one indicates a change in configuration. Constraints (21) indicate that if there is a change in the configuration of area \( i \) at time \( t \) (i.e., \( \sum_{\ell \in K} \delta_{a}^{i \ell t} = 1 \) or \( \sum_{\ell \in K} \delta_{n}^{i \ell t} = 1 \)), then no change can happen in the next \( M \) periods of time (i.e., \( \sum_{\ell \in K} \delta_{a}^{i \ell, t+m} = 0 \) and \( \sum_{\ell \in K} \delta_{n}^{i \ell, t+m} = 0 \), for \( m = 1, \ldots, M \)). Constraints (22) prevent any SSCP re-configuration within the last \( M \) periods of the time horizon and Constraints (23) establish the binary nature of \( \delta \)-variables.

### 4.2. Solution approach

The optimization model in (5)–(19) minimizes the worst-case queue length across periods in \( T \). The optimal objective function value, which we denote by \( \hat{Q}^* \), provides an achievable performance metric given the workforce constraints and available checkpoint configurations. However, if Constraint (9) is not binding, there is no incentive in the model to use the minimum number of flexible TSOs to achieve the optimal queue performance. Moreover, there is no incentive in the model to maintain short queues at times other than those producing the worst-case performance (i.e., at those times when constraint (10) is not binding) because this will not improve the maximum queue length. To overcome these issues, we propose a three-stage \( \epsilon \)-constraint approach that iteratively solves (5)–(19), adding constraints at a time to prevent the deterioration in the objective at previous stages [44, 10]. Our proposed three-stage solution approach is described below.

**Stage 1:** Solve model in (5)–(19) and obtain the optimal objective function value \( \hat{Q}^* \)

**Stage 2:** Solve the problem \( z = \min \sum_{f \in \Omega} h_f \) subject to (6)–(19), but replacing Constraints (10) with \( Q_u \leq \hat{Q}^*, \quad i \in C \cup A, \quad t \in T \). Obtain the optimal objective function value \( z^* \).

**Stage 3:** Solve the problem \( \min \sum_{i \in C \cup A, t \in T} Q_u \) subject to the same constraints as the model in Stage 2 and the additional constraint \( \sum_{f \in \Omega} h_f \leq z^* \).
In Stage 2, queue lengths are bounded by the optimal value of Stage 1, which prevent their deterioration. That is, no solution in Stage 2 or 3 will have a maximum queue length worse than $\hat{Q}^*$. The objective in Stage 2 is to minimize the number of flexible TSOs used, maintaining the same queue performance as in Stage 1. Stage 3 seeks to minimize the summation of queue lengths across all areas and time periods, aiming to bring queue lengths to their minimum throughout the entire day, even at times where queues are not at their worst-case length. The problem solved in Stage 3 enforces that the number of flexible TSOs is no more than the optimal level from Stage 2 and that no queue at any time has length worse than the optimal value from Stage 1. Stage 3 also discourages the model from keeping unnecessary queues and will utilize processing capacity whenever passengers are present in the TDC or screening queues. Lastly, note that the described procedure prioritizes maximum queue length. The model can easily be adjusted to allow a modest increase in maximum queue length to reduce the use of flexible TSOs or average queue length by allowing $\hat{Q}$ to increase by some level $\epsilon$ in Stage 2 or 3.

5. Data Sources

In this section, we describe possible data sources to estimate the parameters required by our models. Whenever possible, we discuss the benefits and challenges of each source based on our experience. Due to the competitive nature of the commercial aviation industry, airlines are reluctant to publish historical data related to load factors. The US Department of Homeland Security (DHS), however, requires certain information to be publicly disclosed by airlines. This information is mostly in aggregated form in order to protect airline’s operational information from public disclosure. Additionally, for protection of customer and citizen privacy, personal information to accurately estimate earliness arrival times is never recorded.

One of the major sources of information in this field is the US Department of Transportation’s Bureau of Transportation Statistics (BTS). Several databases used in this study are publicly available through BTS and contain reasonably up-to-date information. Other sources of information include DHS reports, third-party agencies (e.g., Official Aviation Guide–OAG), and airport-specific agencies. We summarize the characteristics of each potential data source in the reminder of this section.

**BTS T-100 Segment.** This database reports monthly route information for individual airlines aggregated over a month. This data can be used to estimate the load factor for a given airline, route, and month by using the data on airline, capacity (seats), passengers, origin, and destination.

**BTS DB1B Market.** This database is also referred to as Airline Origin and Destination Survey. A 10% sample of the reported tickets belonging to customers along with the itinerary details of their travel is available quarterly. This data can be used to derive quarterly percentages of people who originated from a particular airport to a specific destination and on an airline by comparing
the counts of the entries where that particular airport is a connection itinerary versus a travel start point.

**BTS On-time Statistics.** This database reports historical flight schedules. A disadvantage of the database is that international flights are not reported. Reported schedules also have the tail number of the aircraft on which a particular flight was operated. Tail numbers can potentially be used to obtain the number of available seats for each reported aircraft.

**TSA Reports.** Earliness arrival distributions are available in some reports from the Transportation Security Administration (TSA) [49]. These reports focus on national averages across all US airports and provide earliness arrival distributions for early morning, peak hour, non-peak, and international departures. However, these distributions are general and ignore specific aspects of the type of passengers (business persons vs. tourists), destinations, and time of the day, among other airport-specific aspects.

**TSA Throughput Data.** TSA reports hourly throughput counts at security checkpoints throughout the country. Because these counts are obtained in the screening area (once a passenger goes through the AIT or WTMD), this data is lagged compared to the passenger arrivals given the queueing process. Our assumption is that, since the wait times typically are significantly less than one hour, the lag between arrivals and throughput can be neglected and throughput data is a good estimate of the actual arrivals.

**OAG.** OAG is a private company specializing on providing aviation data. This database reports flight schedules that can be used to obtain $F_j$ values for (1). OAG also reports load factors and percentage of originating passengers. However, the percentage of originating passengers was the same for all flights in most of the sample days we analyzed. Moreover, load factors seem to be estimated based on historical information rather than actual ticket purchases.

**Additional Open Data.** Additional data sources are available for local customization of the model. For instance, in Section 6, we describe a case study for Phoenix Sky Harbor Airport. In this case, the city of Phoenix publishes the Worldwide Time Table for all flights departing Sky Harbor. However, the data is only updated monthly and effectively only has accurate flight schedules for the beginning of the month. Also available is the published *Aviation Flight Information* database with gate assignments for departures, allowing estimation of passenger loads at specific SSCPs.

**TSA Secure Flight.** This database is based on the Secure Flight airline passenger pre-screening program implemented in the US since 2009. Airlines report to TSA authorities the information of every passenger for every flight several days prior to departure. This is used to compare passengers identities with watch lists maintained by the federal government. This database provides more accurate short-term load factors. However, this information is not available (or is highly inaccurate) for longer term predictions than a week ahead. Access to this information is restricted and needs to be anonymized.
6. Results and Discussion

In this section we present a case study at Terminal 4 (SSCPs A and B) in Phoenix (Arizona) Sky Harbor airport. We first describe the results of the passenger arrival estimations as well as two validation procedures: a backtesting using flight departures as they occurred and a test using projected flight departures. The goal of the first validation is to assess the model accuracy under complete information, whereas the purpose of the second validation is to replicate a real prediction exercise using the available information at the decision-making time. In both cases, we compare our results with observed passenger throughput. We then illustrate the use of the proposed workforce allocation models.

6.1. Validation of passenger arrival estimations

6.1.1. Parameter tuning. We select the values of \( l \)- and \( u \)-parameters in (2)–(3) using hourly observed throughput, adjusted predictions by our methods, and TSA forecasts from October 21–31, 2018. We explore three different types of models in the cross-validation process: general, daily, and hourly. The general model restricts the cross-validation process to use the same lower and upper bounds within the interval \([0, 2]\) for the entire period of analysis. The daily model is more relaxed, restricting the model to use the same bounds within a day but allowing different values for different days, if needed. The hourly model allows the cross validation process to choose the best bounds for each hour of a day. We use the mean absolute error (MAE) as a measure of prediction quality. The boxplots shown in Figures 1a and 1b summarize the parameter tuning results for SSCP A and B, respectively. The general model has the smallest average error and spread of absolute error for SSCP A, whereas the daily model is slightly better than the general model for SSCP B. These models show an improvement in the average error compared to TSA projections, while the overall spread remains similar. We select the models with the smallest MAE in our predictions.

6.1.2. Backtesting. We use the proposed mechanistic and learning models to predict passenger arrivals for October 2018. We use the data sources from Section 5, including historical flight departure schedules from BTS Airline On-time Statistics [9] and parameter tuning results from Section 6.1.1. Figures 4a and 4b show the mechanistic and adjusted projections for October 22, 2018, as well as the observed passenger throughput per hour. In this case, these figures show that the adjustment process improves the mechanistic model for both SSCP A and B. These models show an improvement in the average error compared to TSA projections, while the overall spread remains similar. We select the models with the smallest MAE in our predictions.

Table 1 shows the MAE for the mechanistic, adjusted, and TSA predictions against observed throughput. The adjusted mechanistic model presents the smallest error compared to other prediction methods, with an average error of \( \sim 112 \) and \( \sim 109 \) passengers per hour in SSCP A and B, respectively. To put these numbers in perspective, Table 1 also shows in parenthesis the percentage
of the MAE error relative to passenger volumes per hour from 7:00AM to 7:00PM, which is the busiest period at Sky Harbor airport. Using this metric, the adjusted model shows the best performance for SSCP A, whereas the TSA predictions show the smallest relative error for SSCP B. This discrepancy lead us to produce a ensemble model averaging the adjusted and TSA predictions. The last column of Table 1 reports the performance of this model, which turns out to be the model with the best predicting performance. This result may be due to the nature of the estimation models, where TSA predictions are mostly based on short-term moving averages and Secure Flight data while the adjusted model relies on long-term data from prior years.

We further explore the quality of our models by calculating the MAE for those times where predictions underestimate the throughput. The motivation of this analysis comes from TSA opera-
tional decisions in which a short-staffed SSCP due to passenger underestimation is undesirable as it translates into long wait times. Table 2 shows that the ensemble model has the smallest average underestimation error for SSCP A, accounting for a relative error of 7.6% of the passenger volume at busy times. The mechanistic model presents the smallest error for SSCP B with a relative error of 3.9% of the passenger volume at the same times. However, it should be noted that the mechanistic model almost always overestimates passenger volumes. Depending on the analyst’s preferences, the adjusted or ensemble model may offer a more balanced prediction error.

### Table 2 Underestimation MAE and relative performance

<table>
<thead>
<tr>
<th></th>
<th>Mechanistic</th>
<th>Adjusted</th>
<th>TSA</th>
<th>Ensemble</th>
</tr>
</thead>
<tbody>
<tr>
<td>SSCP A</td>
<td>107.7 (14.8%)</td>
<td>51.7 (11.2%)</td>
<td>42.7 (8.0%)</td>
<td>39.4 (7.6%)</td>
</tr>
<tr>
<td>SSCP B</td>
<td>30.6 (3.9%)</td>
<td>60.8 (8.3%)</td>
<td>101.9 (18.3%)</td>
<td>73.2 (10.8%)</td>
</tr>
</tbody>
</table>

Underestimation MAE relative to passenger throughput from 7:00AM – 7:00PM in parenthesis

**6.1.3. Forecasting using current data.** As a second test, we use our models to predict passenger volumes for the last three weeks of March 2019. We recreate a realistic prediction exercise by only using data available in the first week of March 2019, including flight departure schedules from OAG. Similar to Section 6.1.2, we evaluate the quality of our models by calculating the MAE against observed throughput from TSA. Figures 4a and 4b show the mechanistic and adjusted predictions for March 21, 2019, as well as the observed passenger throughput per hour. The improvement of the adjusted model over the mechanistic is clear for SSCP B, but not so obvious for SSCP A.

Table 3 shows that the TSA prediction model performs better than the mechanistic and adjusted models for both SSCPs. However, the ensemble model has the best predictive performance of all models for both terminals, with MAEs of ~99 and ~100 passengers per hour. The MAE relative to hourly passenger volumes for the ensemble model is equal to 15.9% and 17.5% for SSCPs A and B, respectively. Table 4 shows the model performance when passenger arrivals are underestimated. In this case, the best performance is given by the ensemble model for SSCP A and the mechanistic model for SSCP B. Because of the results in Table 3, this result means that the mechanistic approach is overestimating most of the time, so similar to the backtesting exercise in Section 6.1.2, the ensemble model has the best overall performance for both SSCPs. Figure 4 shows the performance of the ensemble model for March 21, 2019.
Table 3  MAE and relative performance

<table>
<thead>
<tr>
<th></th>
<th>Mechanistic</th>
<th>Adjusted</th>
<th>TSA</th>
<th>Ensemble</th>
</tr>
</thead>
<tbody>
<tr>
<td>SSCP A</td>
<td>124.8 (17.5%)</td>
<td>119.1 (19.2%)</td>
<td>101.5 (17.4%)</td>
<td>89.5 (15.9%)</td>
</tr>
<tr>
<td>SSCP B</td>
<td>195.3 (43.1%)</td>
<td>115.8 (19.0%)</td>
<td>104.0 (20.3%)</td>
<td>99.5 (17.5%)</td>
</tr>
</tbody>
</table>

Table 4  Underestimation MAE and relative performance

<table>
<thead>
<tr>
<th></th>
<th>Mechanistic</th>
<th>Adjusted</th>
<th>TSA</th>
<th>Ensemble</th>
</tr>
</thead>
<tbody>
<tr>
<td>SSCP A</td>
<td>85.8 (13.0%)</td>
<td>53.3 (7.1%)</td>
<td>36.3 (4.9%)</td>
<td>34.4 (4.8%)</td>
</tr>
<tr>
<td>SSCP B</td>
<td>11.3 (0.1%)</td>
<td>46.3 (6.1%)</td>
<td>36.4 (4.7%)</td>
<td>36.2 (4.3%)</td>
</tr>
</tbody>
</table>
6.2. Optimal SSCP configuration and workforce allocation

We illustrate the use of our optimization models on a sample day, assuming that workforce allocation decisions need to be made in 10-minute intervals. We calculate the 10-minute passenger predictions—i.e., $\lambda$-parameters in model (5)–(19)—using the results from Section 3 for March 2019. Moreover, we set $L_t = 45$, for all $t \in T$, where $T$ is the set of all 10-minute intervals within a day. We also assume $B = 30$, meaning that no more than 30 flexible TSOs can be assigned to the SSCP in a day, beyond the baseline workforce of 45 TSOs. Flexible TSOs can be assigned to four-hour shifts that can start at any time $t \in T$. We use a processing rate for each TDC of 360 passengers per hour and for each lane of 200 passengers per hour. Moreover, we assume that each open TDC stand is staffed with one TSO and each screening lane with five TSOs. These processing rates and workforce demands change linearly with the number of TDCs and screening lanes open. This can be easily modified to replicate any pattern given that the checkpoint configurations are enumerated in advance. The set of configurations consider that in any SSCP there is a maximum number of four TDC stands and eight screening lanes.

Figure 5 shows the optimal configuration for each SSCP at the end of Stage 3, for each 10-minute interval of the chosen day. These configurations include the optimal number of TDC stands and number of screening lanes open. Figure 6 shows the optimal starting time and number of flexible TSOs allocated to the SSCP as well as the predicted passenger arrivals. As expected, flexible TSOs and checkpoint re-configurations are needed to anticipate the increase in passenger volume at busy times, for instance around 7:00AM ($t = 42$). This means that it is an optimal policy to proactively open TDCs and screening lanes (with their corresponding flexible TSOs) before peak times to prevent queue buildup.

To illustrate the utility of the proposed three-stage solution approach, Figure 7 displays the queue lengths at the end of Stage 1 (top) and Stage 3 (bottom). As expected, queue lengths after Stage 1 are not minimum at every time, given that the objective at this stage is to minimize the maximum length. At the end of Stage 3, queue lengths are minimized without deteriorating the maximum queue length from Stage 1. In this case, the optimal SSCP configurations shown in Figure 5 and flexible TSO allocations in Figure 6 result in a maximum queue length of no more than 55 passengers per time. In this case, the three-stage procedure is executed in no more than 5 minutes in a Dell Precision laptop with Intel Core i7 2.7 GHz and 16GB of RAM.

We explore the sensitivity of the optimal SSCP performance to changes in the number of available flexible TSOs. Intuitively, the maximum queue length of 55 passengers in Figure 4 could be improved if more TSOs are available to operate more TDCs and screening lanes at peak times. Figure 8 shows the impact of increasing the maximum number of flexible TSOs during the day on the maximum
Figure 5 Optimal SSCP configurations (Stage 3)

Figure 6 Optimal flexible TSO allocations and passenger arrivals (Stage 3)

queue length. The non-smoothness in the sensitivity profile is because adding one additional TSO may not have a huge impact in the SSCP performance as it can only be assigned to a TDC duty. Adding more TSOs allows the model to open new screening lanes, each of which requires five TSOs. Figure 4 also shows that there is a minimum (nonzero) queue length due to the fact that only a maximum number of eight screening lanes and four TDCs can be open at any time.

Figure 9 shows the optimal SSCP configurations after imposing the operational constraints (20)–(23). In this case, we show two scenarios, requiring a configuration to be in operation for at least 30 (i.e., $M = 2$) and 60 (i.e., $M = 5$) consecutive minutes if selected. Although these new constraints
reflect real SSCP operations, the lack of flexibility to re-configure the SSCP deteriorates the overall queue performance. This can be seen by examining the maximum queue length, which increases from 55 passengers in the solution shown in Figure 5, to 60 passengers when \( M = 2 \) and 76 passengers when \( M = 5 \). Moreover, imposing Constraints (20)–(23) implies at a higher computational cost, requiring 15 minutes and 3.3 hours to solve all the models in the three-stage approach for \( M = 2 \) and \( M = 5 \), respectively, in a Dell Precision laptop with Intel Core i7 2.7 GHz and 16GB of RAM.
7. Conclusions

The models proposed in this article aim to improve any airport’s SSCP operations by providing more accurate passenger arrival predictions and a flexible modeling approach to decide optimal checkpoint configurations and their corresponding workforce allocations. Our models can be used to support other types of decisions such as TSO reallocations and hiring (number and schedule), SSCP expansions, and new screening technology improving passenger processing rates (i.e., $r$-parameters). In practice, our models provide evidence that proactive queue management is critical to prevent long wait times, as they prevent queue buildups before peaks in demand. As a result, SSCP re-
configuration decisions with their corresponding workforce allocations are critical to maintain short queues and must be executed before expected surges in passenger volumes.

Multiple extensions can be pursued out of this work. From an optimization point of view, a natural extension to our models is to endogenously create shifts informed by operational constraints, which are currently taken as input parameters. Embedding such decisions in the model would require additional features such as TSO transit times between airport terminals, mandatory breaks, and TSO skills, among others. From a prediction perspective, the proposed methods perform fairly well even under unexpected fluctuations in passenger volumes due to flight delays, gate changes, and last minute bookings, among other factors. Ensemble models combining adjusted and TSA predictions proved to be more accurate than individual models. However, having passenger counts at the beginning of the TDC queue (i.e., actual arrivals to the SSCP) rather than throughput data will remove the effect of the SSCP configuration and may improve the quality of the predictions. To the best of our knowledge, such data is not currently recorded.

Although our analysis focuses on an airport system, our methods can be used in other contexts such as Customs and Border Protection (CBP) ports of entry and any other queueing network systems admitting optimal re-configuration and workforce allocation decisions.

8. Acknowledgments

This material is based upon work supported by the U.S. Department of Homeland Security under Grant Award Number 17STQAC00001-02-00. We also would like to thank Wesley Prochniewski, Matthew Alspaugh, and James Huffman from TSA Phoenix Sky Harbor for their continuous support along this project. Finally, we extend our gratitude to the Center for Accelerating Operational Efficiency (CAOE), a DHS Center of Excellence led by Arizona State University, for their support.

9. Disclaimer

The views and conclusions contained in this document are those of the authors and should not be interpreted as necessarily representing the official policies, either expressed or implied, of the U.S. Department of Homeland Security.
References


