Bottleneck Detection of Security Checking Process for Airport Throughput Optimization

Cui-hua LU, Hong-qing ZHOU* and Shu-hu WANG
Naval Aeronautical University, Yantai China 264001
*Corresponding author

Keywords: Bottleneck detection, Security checking process, Grey system theory.

Abstract. There exists a trade-off between high throughput of passengers at airport security checkpoints and passenger safety. Our task is to optimize the throughput with the constraint of safety. First we define a couple of assumptions for a typical security checking system. Second, we create a bottleneck-finding model to predict the probability of congestion using grey system theory. Finally, we find that belongings-removal step is the bottleneck of security checking process by applying apagoge on each possible step.

Introduction

Passengers suffer a lot waiting in the long line when facing congestion in the airport security checkpoint. However, it is necessary to assure the safety of the flight by thorough check of individuals. Therefore, the pursuit of convenience and the need for safety conflicted. In order to enhance the performance of airport security checking system, it means increase the throughput and lower the difference among passengers’ wait time while the safety standard is maintained, the security checking procedure is required to be optimized.

Different approaches are used to explore the dynamic of passenger flow and help improve the performance of the airport security checkpoint. Queuing network theory has been a staple of operations research since Jackson (1963) identified the need for stochastic modeling of queuing in multipurpose production systems and proceeded to establish joint probability distribution and create the M/M/1 queue. Lee and Jacobson (2011) performed a study that analyzed the patterns of aviation checkpoint queues on optimizing security screening effectiveness [1].

Considered as an effective tool, DES is widely used to expedite the passengers’ process in the Security Screening Checkpoint (SSCP) [2]. Multiple researchers have used DES in the realm of aviation transportation and security. Most studies where DES was used on airport operations either specifically investigate the check in process, the checked baggage screening process, or the entire airport departure experience, only a handful of studies have been performed on SSCPs. Arena simulation software which was used in this study was commonly used throughout the literature for a DES approach. A study by Pendergraft et al. (2004) used DES to understand operational dynamics of both checked baggage screening and the SSCP at a major U.S. airport during peak operating hours [3]. Wetter, Lipphardt, and Hofer (2010) in their research used DES to assess throughput of SSSCs by examining internal and external factors [4].

In order to optimize the throughput of security checking system, we should try to find out the step with the lowest process rate, and thus explain what mainly contributes to the passenger congestion. Thus, it is the problem of bottleneck detection at airport security checking procedure.

The Security Check Process

The security check process for a typical airport security checkpoint is displayed in Figure 1[5].

To simplify the problem, we propose some general assumptions in the model, each of which is justified.

- Staffs and equipment have maintain the same efficiency in each part of process.
- All the passengers follow the first-in first-out (FIFO) rule from Zone A to Zone C.
We ignore the time of placing items on the belt going into x-ray and the time transit time between process stations in the wait time.

Additional check/screening will not congest the line.

After security check finished, the passenger will not returned to the security checkpoint.

Figure 1. The layout of security checking system.

The Bottleneck-Finding Model

As Figure 2 shows, we use the flow chart to simulate the process, which can be decomposed into several independent time-costing steps. The bottleneck, defined as the step contributes most to congestion. In the security checking process, passengers first enter checkpoint at Zone A and queue up waiting for processing ID Check Step. After ID check, they enter Zone B and choose one line to wait for screening. When in the top of the queue, the passenger will remove their belongings and get ready for Millimetre Scan for physical body as well as X-ray Scan for luggage. If the officer suspects there is something forbidden for boarding, passenger will have an additional check at Zone D.

Figure 2. The flow chart of security checking process.

It is obvious to see that the potential bottleneck should be in the four process colored crimson, which are ID Check Step, Remove Belongings (shoes etc. & laptop etc.) Step, X-Ray Scan Step, Millimeter Scan Step.
The Bottleneck Assumption

According to the published dataset [6], we are able to get passengers’ arrival times, ID-check processing times, Millimetre wave scan times, X-ray scan times and time to get scanned property. In the Excel data, it has been revealed the interval of two passengers enter the airport (column A/B), the average time every passenger spend on ID check step (column C/D), the average time of X-ray scan step (column F/G) and the average time of Millimeter Scan Step (column E). However, the time of removing belongings is not existed.

According to the information we had, we find out the bottleneck step following this procedures:

● First, we assume each step as bottleneck step.
● Second, we use Grey system theory to predict the data of ID Check Step, X-Ray Scan Step, and Millimeter Scan Step.
● Third, we use the prediction results of each step to verify that whether the step is the bottleneck step. If we can prove that none of the three step is the bottleneck step, the Remove Belongings Step will be that.

Data Prediction of Each Step

Take the Millimeter Scan Step as the example.

Assume that the Millimeter Scan Step is the bottleneck step, and other three steps are not. The speed of passenger arrival exceeds that of Millimeter Scan Step, causing the congestion.

Because the data is limited and random, we can apply Grey System Theory [8] to predict when the congestion will happen under the assumption. If the congestion happens frequently, we can conclude that Millimeter Scan Step is the bottleneck. Otherwise, it is not the bottleneck.

Data Preparation. We will elaborate the data-generation process to Millimeter Wave Scan times as an example, and the two steps, that are ID Check and X-ray Scan, share the same procedure. First of all, abnormal data is extracted. We define the time series of Millimeter Scan as $x^1(i)$ where $i$ represent that the $i$th passenger. Subsequently, the time interval between $x^1(i)$ and $x^1(i + 1)$ is recorded as $x^0(i)$. From the given data, the average interval between two passengers’ arrival at the airport is 5 seconds. In our model, there are 3 lanes for Millimeter Scan, which means that the bottleneck margin of Millimeter Scan time is $t_{max} = 3 \times 5 = 15$ s. In other words, Millimeter Scan could be suspected to be the potential bottleneck if the scanning time exceed $t_{max}$. Based on the Grey System Theory, we first find abnormal times that exceed $t_{max}$ in the given series. The corresponding series $n_0$ we get is:

$$n_0 = (11 13 18 20 32)$$

After the Accumulated Generating Operation (AGO), we get is:

$$n_1 = (11 24 42 62 94)$$

Here, we use GM (1,1) model [6] which is the most commonly used model in Grey Theory to generate data. To test whether this model is appropriate for using, we need to first conduct Class Ratio Test, as shown below:

$$\lambda^0(k) = n^0(k - 1)/n^0(k)$$

Acceptable class model’s $\lambda$ should in the range of (0.716, 1.33) [6]. After calculation, the $\lambda^0$ is:

$$\lambda^0 = (0.85 0.72 0.9 0.625)$$

Thus, $n_0$ is qualified to be replaced by our generated series $N_0$ and $\lambda^{0r}$.

$$N^0 = (22 24 29 41 53)$$

$$\lambda^{0r} = (0.92 0.83 0.94 0.72)$$

After AGO, we get

$$N^1 = (22 46 75 116 169)$$
**GM (1,1) Modeling.** The GM (1,1) modeling is based on first order differential equation, that it believes the interval \( z^1(k) \) between two neighbour numbers can be described as

\[
z^1(k) = 0N^1(k) + (1 - \theta)N^1(k - 1), \quad k=2,3,4,5
\]

Here we let \( \theta = 0.5 \) as an empirical value from the observation through original series data. Solve the following equation:

\[
u = (B^TB)^{-1}B^TY
\]

where

\[
u = (a, b)^T
\]

\[
Y = (N^0(2), N^0(3), N^0(4), N^0(5))^T
\]

\[
B = \begin{bmatrix} -z^1(2) & \cdots & -z^1(n) \\ 1 & \cdots & 1 \end{bmatrix}^T
\]

According to least-square method, we should minimize the function \( J(u) = (Y - Bu)^T(Y - Bu) \), so we get

\[
u = (a, b)^T = (-0.193 \ 16.659)^T
\]

Solve the following differential equation:

\[
\frac{dN^1}{dt} + aN^1(t) = b
\]

The result is:

\[
N^1(k + 1) = \left( N^0(1) - \frac{b}{a} \right) e^{-ak} + \frac{b}{a}
\]

\[
N^1(k + 1) = 108.316e^{-0.193k} - 86.316
\]

\[
N^0(k) = N^1(k + 1) - N^1(k)
\]

\[
n^0(k) = N^0(k) - 11
\]

**Validation.** To validate our model, we conduct the residual test as shown in Table 1. The deviation of class ratio test is shown in Table 2. We can see that either \( \varepsilon_k \) or \( \rho_k \) is lower than the empirical bound 0.2, so the prediction is reliable.

<table>
<thead>
<tr>
<th>Predicted value</th>
<th>Actual value</th>
<th>Residual (( \varepsilon_k ))</th>
</tr>
</thead>
<tbody>
<tr>
<td>11</td>
<td>11</td>
<td>0</td>
</tr>
<tr>
<td>12.06</td>
<td>13</td>
<td>-7.2%</td>
</tr>
<tr>
<td>16.97</td>
<td>18</td>
<td>-5.7%</td>
</tr>
<tr>
<td>22.92</td>
<td>20</td>
<td>14.2%</td>
</tr>
<tr>
<td>30.14</td>
<td>32</td>
<td>-5.8%</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Actual value</th>
<th>( \lambda_k )</th>
<th>( \rho_k )</th>
</tr>
</thead>
<tbody>
<tr>
<td>22</td>
<td>0.917</td>
<td>-0.11</td>
</tr>
<tr>
<td>24</td>
<td>0.828</td>
<td>-0.005</td>
</tr>
<tr>
<td>41</td>
<td>0.935</td>
<td>-0.135</td>
</tr>
<tr>
<td>53</td>
<td>0.721</td>
<td>0.124</td>
</tr>
</tbody>
</table>
Conclusion

To sum up, from the previous modeling, we are able to simulate which passenger will spend over 15 seconds. So when $k = 21$, $n^0(k) = 1083.49$, there is only $\frac{21}{1083.49} \approx 2\%$ chance can Millimetre Scan be the bottleneck. Hence our initial assumption is wrong.

After doing the same assumptions and data prediction with ID Check Step, X-Ray Scan Step, the results can prove they are not either. Therefore, none of the three steps is the bottleneck step. The Remove Belongings Step is actually the bottleneck step in the security checking process.

Summary

In this paper, the bottleneck-finding model is built to detect bottleneck of the airport security checking system using Grey System Theory. Based on this model, we can further simulate the passenger flow in the process so as to optimize the throughput and waiting time variance. As a result, the performance of the airport security checkpoint will be improved.

Reference


