Agent-based modelling and analysis of security and efficiency in airport terminals

Janssen, Stef; Sharpanskykh, Alexei; Curran, Richard

DOI
10.1016/j.trc.2019.01.012

Publication date
2019

Document Version
Accepted author manuscript

Published in
Transportation Research Part C: Emerging Technologies

Agent-based Modelling and Analysis of Security and Efficiency in Airport Terminals

Stef Janssen* Alexei Sharpanskykh Richard Curran

February 28, 2019

Delft University of Technology
Kluyverweg 1, 2629 HS Delft
s.a.m.janssen@tudelft.nl; o.a.sharpanskykh@tudelft.nl; r.curran@tudelft.nl

*Corresponding author.

Abstract

Both security and efficiency are important performance areas of air transport systems. Several methods have been proposed to assess security risks and estimate efficiency independently, but only few of these methods identify relationships between security risks and efficiency performance indicators. To analyze security, efficiency, and the relationships relations between them, an agent-based methodology was proposed in this work. This methodology combines an agent-based security risk assessment approach with agent-based efficiency estimation. The methodology was applied to a case study that analyzes security regarding an Improvised Explosive Device (IED) attack, different commonly used efficiency performance indicators in the aviation domain, such as queuing time for passengers, and the relationships between them. Results showed that reducing security risks and improving efficiency were not always conflicting objectives. Reducing the number of passengers before the security checkpoint was found to be an effective measure to reduce security risks and improve efficiency aspects. Furthermore, results showed that airports should attempt to spread passengers across the available space as much as possible to reduce the impact of an IED attack.

Keywords: Security Risk Management; Efficiency; Agent-based Modelling; Airport Terminal; Improvised Explosive Device

1 Introduction

Improving the security and efficiency of airports are two of the most important strategic objectives of the International Civil Aviation Organization (ICAO) [1] and airports. Apart from ICAO and airports themselves, the research community has shown interest in methods to estimate and improve both security and efficiency.

Airport terminal efficiency has been studied using a wide range of different approaches. For instance, data driven approaches utilized airport data to estimate their efficiency [2, 3], while Bayesian models were used to more efficiently process the vast amount of airport data [4]. Moreover, traditional simulation studies were used to estimate efficiency in current and hypothetical scenarios [5, 6]. Finally, agent-based simulation methods were used to more accurately incorporate heterogeneous passenger behavior [7, 8].

Airport security is driven by a large set of rules and regulations defined by a variety of institutes. For instance, ICAO has a security manual [9], the European Union has regulations [10, 11], and the United States has the Aviation and Transportation Security Act [12]. These rules and regulations form the basis for the implementation of security measures at airport terminals, but airports still have some freedom to implement these measures according to their preferences.

To assess (and/or improve) airport terminal security, many methods have been proposed in literature. Most commonly, the so-called threat-vulnerability-consequence (TVC) methodology is used in practice. Many variants of the TVC methodology exist: the Risk Analysis and Management for Critical Asset Protection (RAMCAP) approach [13], the ICAO security manual [9], the security risk assessment handbook [14], and the RAND terrorism risk estimation handbook [15]. In the TVC methodology, security risks are estimated based on three threat components: threat likelihood, vulnerability and consequence. These components are individually assessed by security experts, and are used as a guide to implement security measures. The TVC methodology heavily depends on security experts, who cannot take into account all complex processes and interactions at an airport terminal [16].

To overcome this dependency on security experts, researchers have developed computational methods to assess security risks. One such computational method is that of attack trees [17, 18]. In attack trees, security threats are represented in a tree structure. A successful executed security threat is represented as the root node of the tree. Leaf nodes of the attack tree are events that can be happen independently, while events represented by internal nodes depend their child nodes. Similar to attack trees, probabilistic methods [19] represent security threats using probabilistic events. The interaction of these events and their respective probabilities lead to risk estimations. Security games [20, 21] use game theory to represent security threats. In these games, two players are defined: the attacker and the defender. Using a game-theoretic analysis, the optimal strategy for the defender can be determined. Finally, a method using agent-based modelling was introduced recently [22]. In this work, agent-based models and Monte Carlo simulations are used to assess security risks.

Some of the above-mentioned studies recognize that security and efficiency are related. However, many of the security-oriented studies only consider efficiency as a constraint, while most efficiency-oriented studies model security measures only as an efficiency bottleneck [5]. A notable exception to this is the work of Wilson et al. [23], in which efficiency and security are estimated simultaneously using a simulation method. However, this work lacks a formal methodology and uses a basic notion of security by only incorporating vulnerability in their analysis. Moreover, the work of Kirschenbaum [24] investigates tradeoffs between security and efficiency using informal quantitative methods, but does not follow a formal computational methodology. Finally, Grant

and Stewart performed a traditional security risk assessment on an Improvised Explosive Device (IED) attack, while taking into account costs for the airport [25]. Their work concerned a higher-level tradeoff between costs and security, while other efficiency performance indicators may be of influence as well.

The goal of our work is to develop a formal methodology to analyze security, efficiency, and identify and quantify relationships between them, using agent-based modelling as a central paradigm. Agent-based modelling forms a promising paradigm, as it allows for detailed analysis of security, efficiency and their corresponding relationships, often hard in the above-mentioned modelling frameworks. Agent-based models are important tools to better understand complex systems, such as airports. Attackers and defenders can naturally be represented by agents with diverse strategies and non-linear interactions between them. Other security risk assessment methodologies often transform airport operations to a group of linear relations. Complex interactions, such as the detection of an ongoing attack by a behavior-detection employee cannot be modelled in such paradigms. Moreover, most other security risk assessment approaches either do not consider efficiency performance indicators at all, or consider efficiency as a constraint. Agent-based modelling forms a promising paradigm in which both efficiency and security can be estimated simultaneously.

The methodology proposed in this work consists of four steps: scope selection, agent-based model definition, security and efficiency estimation, and analysis of simulation results. In the methodology, an agent-based security risk assessment approach is combined with a typical agent-based approach to analyze efficiency of operations. We apply this methodology to a case study in which we analyze security regarding an IED attack, commonly used efficiency performance indicators at an airport terminal, such as queuing time for passengers and number of employees, and their corresponding relationships.

Section 2 introduces the proposed methodology, while the rest of the work applies the methodology to a case study described in Section 3. In Section 4 the corresponding agent-based model is introduced, and in Section 5 the estimation of security risks and efficiency performance indicators relative to the case study is described. Finally, in Section 6 the simulation results are analyzed and discussed.

2 Methodology

The methodology to analyze security risks, efficiency performance indicators and corresponding relationships contains four main steps, outlined in Figure 1. The first step is used to determine the scope of the analysis. It is further discussed in Section 2.1. The second step, agent-based model definition, forms the basis of the analysis. In this step, an agent-based model is defined that will be further used to estimate efficiency performance indicators and assess security risks. This step is further discussed in Section 2.2. Based on the defined models, security risks are assessed and efficiency performance indicators are estimated by means of Monte Carlo simulations in the third step of the methodology (Section 2.3). Finally, in the fourth step the simulation results are analyzed (Section 2.4). An overview of common definitions used in our work can be found in Appendix A.

Steps 1(a,c,d) and 3(b) are used in most variants of the TVC methodology. These steps are complemented by the additional steps 2(b) and 4(b), which were previously discussed in the agent-based security risk assessment method defined by Janssen and Sharpanskykh [22]. Furthermore, a typical agent-based approach for estimation of efficiency of operations follows steps 1(a-b), 2-4(a). This methodology integrates these approaches, while adding step 4(c) to find relationships.

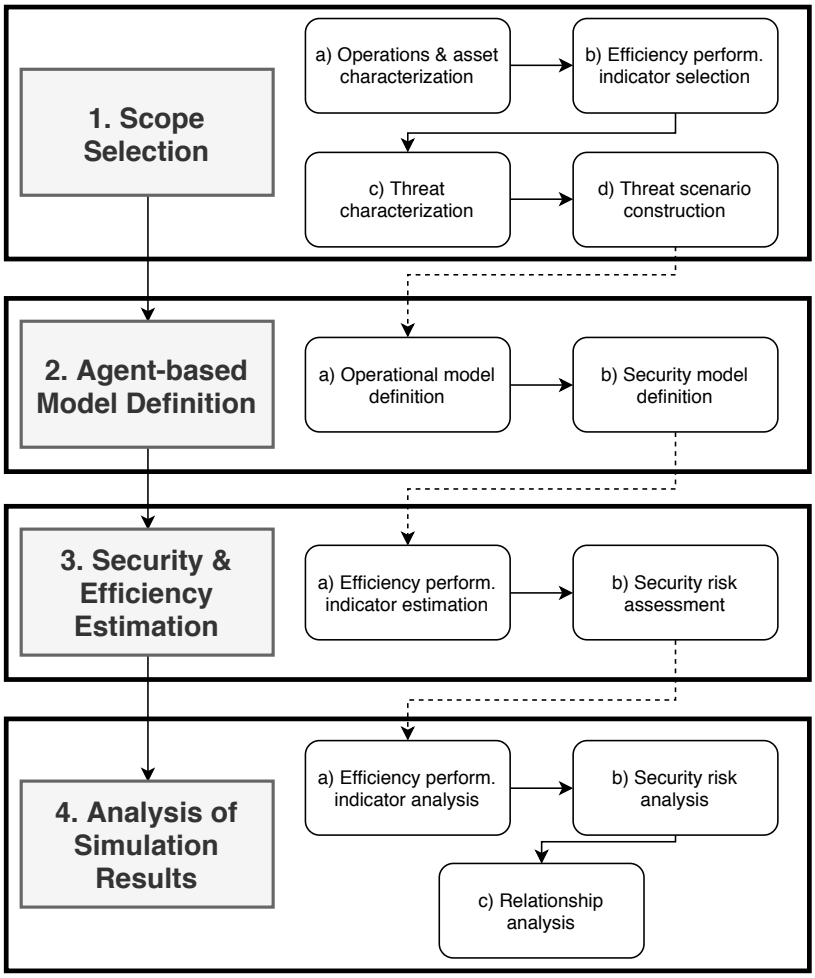


Figure 1: The methodology used in this work.

2.1 Scope Selection

In this first step the scope of the project is defined. The first step is the selection of the specific operational processes and assets to focus on. For the airport domain, an example process can be the check-in process at the airport terminal, while assets can be passengers or the airport terminal building. Based on the selected domain, a set of efficiency performance indicators has to be selected and a set of security threats have to be characterized. Based on the characterized security threats, specific threat scenarios for each of the threats are constructed. Efficiency performance indicators are used to quantify a specific element of efficiency in the selected domain, related to efficiency goals of the airport. In the airport domain, this can for example be the average queuing time for passengers. An example threat scenario is the following: a single attacker brings an IED to a regional airport and detonates it in a publicly accessible area of the airport.

2.2 Agent-based Model Definition

For the above selected scope of the project, the agent-based models M and $M_1, \ldots, M_n$ are defined. The operational model M is defined to model the selected operations of the domain and is used to estimate efficiency performance indicators selected in the previous

step.

Model M defines an environment that represents the environment of the domain area. Then, a set of agents that execute the standard operations in the domain is defined. In an airport, this can for example be passengers or check-in employees. Finally, a set of defender agents is defined. In the context of airports, these can for instance be behavior-detection employees or X-ray officers. These defender agents can additionally have operational task, such as helping passengers find directions.

The model forms the basis for security models $M_1, \ldots, M_n$. These models are used to represent the n threat scenarios in S, which are in turn used to estimate security risks related to the corresponding threat scenario. Each model M_i defines a non-empty set of attacker agents, on top of the components already present in M. The attacker agents execute the attacker behavior in threat scenario s_i, while the defender agents try to prevent the attackers from being successful.

Both a modelling language and an agent architecture need to be selected to specify the models. A modelling language should at least include the following abilities: (1) representation of time; (2) representation of stochastic processes; (3) specification of both qualitative and quantitative aspects; and (4) representation of behavioral and cognitive properties of agents and interaction between agents. The following elements should at least be present in an agent architecture: (1) observation and action; (2) storage of information; (3) maintenance of goals; and (4) reasoning. The Temporal Trace Language (TTL) [26] and LEADSTO [27] are example languages. The BDI architecture [28], and the Desire architecture [29] are example architectures. A more extensive discussion on language selection and architecture selection is provided by Janssen et al. [16].

2.3 Security & Efficiency Estimation

The third step of the methodology is the estimation of efficiency performance indicators and assessment of security risks from simulation results. A set of efficiency performance indicators and security risks are generated, that are used to identify and quantify relationships in the next step.

2.3.1 Efficiency Performance Indicator Estimation

Efficiency performance indicators are estimated by performing Monte Carlo simulations. These Monte Carlo simulations are performed with model M. By extracting relevant information from simulation results of M, each of the efficiency performance indicators defined in 1(b) are estimated. For example, the average queuing time of passengers can be obtained by averaging over the queuing time for each of the passengers present in the simulation model.

2.3.2 Security Risk Assessment

For each threat scenario $s_i \in S$ defined in step 1(d), a corresponding security risk r_i is calculated based on simulation results of model M_i defined in step 2.

An agent-based security risk management methodology is used following the work of Janssen and Sharpanskykh [16]. A security risk r_i is defined for some time period T as a function of *Threat Likelihood* and *Conditional Risk*, as outlined below.

$$R(s_i, T) = f(P(s_i, T), R_c(s_i))$$

Risk $R(s_i, T)$ (or r_i in short) is the risk value for threat scenario s_i in time period T. Conditional risk $R_c(s_i)$ is estimated as follows. For each threat scenario s_i and asset a_l (as defined in the scope selection), a real-valued Consequence function $C(M_i^j, a_l)$ is

defined. This function is used to determine the Consequence value for some simulation run j in model M_i. This Consequence function incorporates estimates of direct losses and indirect losses. Direct losses for instance include fatalities and physical damages of a simulated threat scenario. Indirect losses, such as decreased number of future passengers and business disruptions, are then based on the estimated direct losses and historical data.

Monte Carlo simulations are performed to estimate conditional risk based on a set of N simulation runs. This is done by calculating the following estimate of conditional risk for some scenario:

$$\hat{R}_c(s_i) = \frac{\sum_{j=1}^{N} \sum_{a_l \in A} C(M_i^j, a_l)}{N}$$

where $C(M_i^j, a_l)$ is the consequence for asset l in simulation run j of model M_i. $\hat{R}_c(s_i)$ is the estimated conditional risk for scenario s_i. By calculating the ratio between the number of nonzero consequence values and N (i.e., the total number of consequence values), the vulnerability of the scenario can be obtained. The mean of the nonzero consequence values corresponds to the consequence of the scenario.

Threat likelihood $P(s_i, T)$ for threat scenario s_i is estimated independently from model M_i. Commonly, crime databases and intelligence data are used to estimate the Threat Likelihood [25].

2.4 Analysis of Simulation Results

Simulation results are analyzed following a structured approach. First, the influence of model parameters on efficiency performance indicators is established using statistical analysis techniques. For instance correlation analysis, or more advanced methods such as (global) sensitivity analysis [30, 31, 32] and uncertainty analysis [31] can be used. Similarly, the influence of model parameters on security risks is established using the same techniques.

Relations between model parameters, security risks, and efficiency performance indicators are obtained in this step. This is done by determining which parameters influence both security risks and efficiency performance indicators. By analyzing emergent effects in the defined agent-based models, unexpected relationships can be identified as well.

3 Case Study

The remainder of this work applies this methodology to analyze security and efficiency, and identify and quantify relationships between them in the domain of a small airport terminal. The reference airport handles under 2 million passengers per year and has a centralized security checkpoint. The operations that are included in the study are: check-in, facility visits, security checkpoint operations, queuing, gate processes and the movement of passengers between these processes. We focus on a single asset: humans (i.e., all passengers and employees). A visualization of the airport terminal used in this case study is shown in Figure 2.

We focus on a single threat: a bomb attack in the open areas of the airport terminal, as for instance seen at the Atatürk Airport attack and the Zaventem Airport attack. Based on this threat, two threat scenarios in which an attacker aims to detonate an IED in the open areas of the airport are represented: an early attack and a late attack.

Five efficiency performance indicators are defined: number of employees n, mean time in checkpoint queue over all passengers T_{queue}, mean time to gate over all passengers T_{gate}, number of missed flights $miss$, and monetary loss $loss$.

We focus this case study on three main research questions, as outlined below.

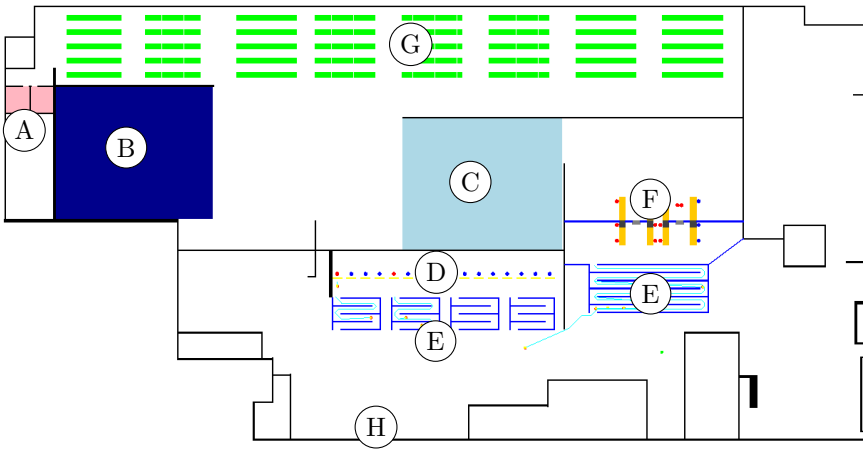


Figure 2: The airport layout of the case study, with indicators for different areas. A, B and C are facility areas. D is the check-in area and E are queuing areas. F is the checkpoint area, G is the gate area and H is the entrance area.

- How does the number of passengers influence the identified efficiency performance indicators and the security risk with respect to the security threat?
- How does the number of checkpoint lanes influence the identified efficiency performance indicators and the security risk with respect to the security threat?
- How does the number of behavior-detection employees and their respective strategies influence the identified efficiency performance indicators and the security risk with respect to the security threat?

4 Agent-based Model

Three agent-based models for the above selected scope are defined. We refer to the operational model as M, while the model that includes the threat scenario is referred to as M_{ied}. The modelling language is discussed in Section 4.1, and the agent architecture is discussed in Section 4.2. The operational model and the security models are discussed in Section 4.3 - Section 4.4. Section 4.5 finally describes the parameters used in the models.

4.1 Modelling Language

To specify the dynamics of a multiagent system, the order-sorted predicate logic-based language called LEADSTO is used [27]. This language allows both discrete and continuous modelling of a system at different aggregation levels. Furthermore, one can express both qualitative and quantitative aspects of a system using LEADSTO.

Dynamics in LEADSTO are represented as evolution of states over time. A state is characterized by a set of properties that do or do not hold at a certain point in time. To specify state properties for system components, ontologies are used that are defined by a number of sorts, sorted constants, variables, functions and predicates (i.e., a signature). For every system component A, a number of ontologies can be distinguished: the ontologies $IntOnt(A)$, $InOnt(A)$, $OutOnt(A)$, and $ExtOnt(A)$ are used to express respectively internal, input, output and external state properties of the component A. For a given ontology Ont, the propositional language signature consisting of all state

ground atoms based on *Ont* is denoted by $APROP(Ont)$. State properties are specified based on such ontology by propositions. Propositions are formed, combining ground atoms by logical operators such as conjunction, negation, disjunction, and implication. Input ontologies contain elements for describing perceptions of an agent from the external world, such as the observed function *obs*: $IntOnt(A) \rightarrow APROP(IntOnt(A))$. Output ontologies describe actions and communications of agents. To this end, the function *performed*: $ACTION \rightarrow APROP(OutOnt(A))$ is introduced. Then, a state S is an indication of which atomic state properties are true and which are false: S: $APROP(Ont) \rightarrow \{true, false\}$.

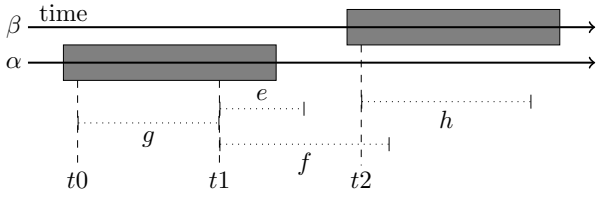


Figure 3: Timing relationships for LEADSTO expressions.

LEADSTO enables modeling of direct temporal dependencies between two state properties in successive states, also called dynamic properties. A specification of dynamic properties in LEADSTO is executable and can be depicted graphically. The format is defined as follows. Let $\alpha 1$ and $\alpha 2$ be state properties of the form 'conjunction of atoms or negations of atoms', and e, f, g, h non-negative real numbers. In the LEADSTO language the notation $\alpha 1 \twoheadrightarrow_{e,f,g,h} \alpha 2$ means: if state property $\alpha 1$ holds for a certain time interval with duration g, then after some delay (between e and f) state property $\alpha 2$ will hold for a certain time interval of length h (Fig. 3). To indicate the type of a state property in a LEADSTO property we shall use prefixes $internal(c)$, $input(c)$, $output(c)$ and $external(c)$, where c is the name of a component. Consider an example dynamic property:

$$input(A)|obs(arrest\_fail) \twoheadrightarrow_{0,0,1,1} output(A)|performed(detonate())$$

Informally, this example expresses that if agent A observes a failed arrest during some time unit, then A will detonate an IED in the following time unit. Next, a *trace* or *trajectory* γ over a state ontology *Ont* is a time-indexed sequence of states over *Ont* (where the time frame is formalized by real numbers). A LEADSTO expression $\alpha 1 \twoheadrightarrow_{e,f,g,h} \alpha 2$, holds for a trace γ if:

$$\begin{aligned}
&\forall t1[\forall t[t1 - g \leq t < t1 \Rightarrow \alpha 1 \text{ holds in } \gamma \text{ at time } t] \\
&\Rightarrow \exists d[e \leq d \leq f \& \forall t'[t1 + d \leq t' \leq t1 + d + h \\
&\Rightarrow \alpha 2 \text{ holds in } \gamma \text{ at time } t']]
\end{aligned}$$

More details on the semantics of the LEADSTO language can be found in [27].

4.2 Agent Architecture

Agents are modelled following an adapted version of the AATOM architecture visualized in Figure 4. The architecture is loosely based on a framework of Blumberg [33], Hoogendoorn [34] and Reynolds [35]. It is described in detail in a technical report [36].

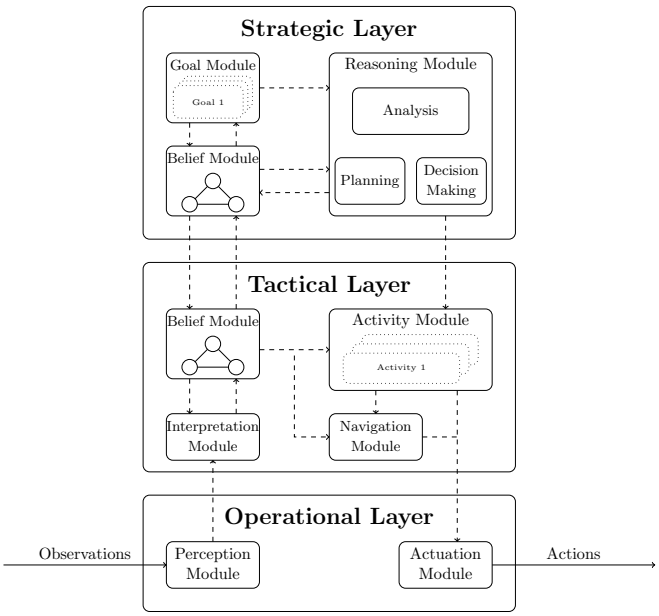


Figure 4: The AATOM architecture and its different modules.

In this architecture, three layers are distinguished, namely the operational layer, the tactical layer and the strategic layer. Each of these layers has a set of modules that execute specific tasks. The operational layer is responsible for doing observations (perception module) and performing actions (action module). Communication with other agents is also executed by the action module. Based on observations, actions and internal states the belief module maintains a belief in the tactical layer. That layer is also responsible for navigation (navigation module) and activity execution (activity module). Finally, the strategic layer maintains a higher level belief (strategic belief module) and generates a plan (planning module). A plan is defined as an ordered sequence of activities that are executed by the agent. For each agent in the model, relevant modules are described in more detail.

Activities form a central concept in this architecture. They have a starting condition, a set of actions that have to be executed and an ending condition. Based on these conditions, an activity is defined to be in either of the three different activity states: *not_started*, *in_progress*, *finished*. All activities start in the *not_started* state and switch to the *in_progress* state when the starting condition is met. Finally, they switch to the *finished* state when the ending condition is met. The activity state is represented as follows: *activity_state*: $ACTIVITY \times ACTIVITY\_STATE \rightarrow APROP(IntOnt(A))$. In addition, an activity can be the next activity in the planning of an agent (determined by the planning module). This is defined in the following function: *next_activity*: $ACTIVITY \rightarrow APROP(OutOnt(A))$ is introduced.

Employee agents and attacker agents only have a single activity they can perform, while passenger agents can execute more activities. They therefore plan their activities following a set of simple rules, explained in more detail in Section 4.4.1.

4.3 Environment

The airport terminal environment consists of several elements, categorized into four different categories: physical objects, IEDs, areas and flights. A visualization of the airport terminal environment is shown in Figure 2.

Two types of physical objects, *wall* and *desk*, are defined. An IED is defined by its location, the number of particles and mass. It is carried by an attacker, denoted $carried\_by(ied, attacker)$. Areas are used to specify functionality of regions in the airport terminal, where *check-in_area*, *checkpoint_area*, *facility_area*, *queuing_area*, *gate_area* and *entrance_area* are the types of areas present in the model. Some areas, such as the *gate_area*, are accessible to passengers only after execution of the *checkpoint_activity* (airside), while others, such as the *entrance_area*, are publicly accessible (landside). Finally, a *flight* is defined to be an abstract concept with the following properties: $departs\_at(\ flight, f_{time})$, *has_gate(flight,gate_area)* and *has_desk(flight,desk)*. The value f_{time} is the time at which the flight departs. The flight also has at least one *desk* that passengers use for checking in and exactly one *gate_area*.

4.4 Agents

The model M contains three types of agents, namely: passengers, operational employees and behavior-detection employees (BDE). The last two agent types are also the defender agents in the model. We assume that there are no other persons, such as visitors, as they form a very small part of the population in the airport under consideration. All agents are human agents and are designed using the framework discussed in Section 4.2. These agents are discussed in more detail in subsequent sections.

4.4.1 Passenger Agent

Passengers are agents that depart with some flight f in the environment. They are characterized by the following five properties: arrival time $t_{arrival}$, level of disorientation d, suitability of luggage s, checked-in c and facility visitor y.

The arrival time $t_{arrival}$ is the time at which the passenger is generated (in the *entrance_area*). The level of disorientation d refers to how disoriented or confused the passenger appears in the airport, and the suitability of luggage s refers to how well the luggage of the passenger fits the appearance of the owner. For example, a business traveller with a large suitcase has a low suitability of luggage. Both these properties are conceptualized with a real number. These properties are important indicators that are used in the SPOT program of the TSA [37, 38]. In the SPOT program, officers assign points to passengers to quantify their danger to the airport. If the points assigned to a passenger exceed a threshold, a secondary screening is initiated.

Checked-in c is a Boolean value indicating whether the passenger is already checked-in on arrival, and facility visit y indicates which facility the agent will visit (*none*, *bathroom*, *restaurant*, *shop*). Passengers can observe physical objects and other agents that are in line of sight within a radius r_{obs}. Furthermore, passengers can observe the area that they are in, and the flight they are taking. Finally, a wait request communicated by other agents can be observed.

Based on these observations, passengers find a collision free path between the different activity locations using the Jump Point Search pathfinding algorithm [39], sometimes used in pedestrian simulators [40]. This is executed by the navigation module, and done when all activities are in the *not_started* activity state or when an activity switched from *in_progress* to *finished*. Passengers follow their generated path (using the action module) by changing their location point using the Social Force model defined by Helbing and Molnar [41]. Passengers can also wait for a specified time t_{wait}.

Passengers can perform the following activities: *check-in_activity*, *checkpoint _activity*, *facility_activity* and *gate_activity*. These activities are planned (in the order as they appear) by the planning module. The checkpoint and gate activity are always executed by agents, while the *check-in* and $facility\_activity$ are only executed if the property checked-in c is *false* or the property facility visit f is not *none*, respectively. If the

check-in_activity or *checkpoint_activity*, cannot be executed (when all activity areas are occupied), passengers perform a wait action in the nearest queuing area until an activity area becomes free. Passengers are removed from the model when $t = F_{time}$.

The *check-in_activity* is executed in a check-in area and consists of a wait action. The activity starts when the passenger observes a wait communication of an employee. The checkpoint activity is executed in a checkpoint area and consists of the same steps as the check-in activity. The facility activity is executed at a facility and also consists of a wait action. The time of the wait action is dependent on the type of facility visit *f*. Finally, the gate activity is executed in the gate area of the flight of the passenger and consists of a single wait action until the flight leaves. The LEADSTO properties below formalize the gate activity.

$input(A)|obs(flight) \wedge obs(gate\_area)$
& $external(A)|has\_gate(flight,gate\_area)$
& $internal(A)|next\_activity(gate\_activity) \rightarrow\!\!\!\!\rightarrow_{F_{time}-t,F_{time}-t,1,1}$
$output(A)|performed(wait(F_{time} - t))$

$output(A)|performed(wait(F_{time} - t)) \rightarrow\!\!\!\!\rightarrow_{0,0,1,1}$
$internal(A)|activity\_state(gate\_activity, finished)$

4.4.2 Attacker Agent

The attacker agent is modelled in the models $M_{ied-early}$ and $M_{ied-late}$. It is a human agent, like passengers, characterized by its arrival time $t_{arrival}$ and the level of disorientation d, suitability of luggage s. In $M_{ied-early}$, the attacker has an early $t_{arrival}$, while this is late in $M_{ied-late}$. The attacker agent has a single goal: achieve as many fatalities at the airport as possible.

To achieve this goal, it can observe physical objects, passengers and attackers in radius r_{obs}. The attacker can further determine the area it is currently in. The number of passengers at the checkpoint area and the check-in area can also be observed. Finally, the attacker can observe that it is being arrested (successfully or unsuccessfully) by a BDE.

The attacker carries an IED that it uses to cause fatalities. To be able to be successful (from an attacker's perspective), the attacker executes the *attacker_activity*. The activity consists of three phases: target selection, movement to target and execution of attack. The target selection is based on a single criterion, namely the observed number of people in the *checkpoint_area* and the *check-in_area*. In the second phase, the attacker moves from the arrival location to the target area. The attacker can then be observed by a BDE (if present), resulting in one of two outcomes. With a probability of p_{arrest} the attacker is arrested and cannot finish the attack, while otherwise the attacker detonates the IED on the spot. This was for instance seen in attacker behavior at the Atatürk Airport attack of 2016 [42]. If the attacker was not observed by any BDE, it continues moving to the target area, where phase three is initiated. In this phase, the

attacker detonates the IED. The LEADSTO properties below formalize the activity.

$t_{arrival} = t \twoheadrightarrow_{0,0,1,1} internal(A)|path(target)$

$internal(A)|path(target)) \twoheadrightarrow_{1,t_{move},1,1}$
$prob(output(A)|performed(move(target)), p)$ & $prob(input(A)|obs(arrest), 1-p)$

$input(A)|obs(target) \vee obs(arrest\_fail) \twoheadrightarrow_{0,0,1,1}$
$output(A)|performed(detonate())$

$output(A)|performed(detonate())$ || $input(A)|obs(arrest) \twoheadrightarrow_{0,0,1,1}$
$internal(A)|activity\_state(attacker\_activity, finished)$

4.4.3 Operational Employee Agent

The operational employee can observe a single passenger at a time in a small radius. It can execute a single action, namely the communication of a wait request. This observation and action is used in the single activity the standard employee executes: the *employee_activity*. This activity consists of the communication of a wait order (of a specified time t_{wait}) to the passenger, when a passenger is observed. The standard employee interacts with passengers that either perform the *check-in_activity* or the *checkpoint_activity*.

4.4.4 Behavior-Detection Employee Agent

The behavior-detection employee (BDE) can observe physical objects, passengers and attackers in radius r_{obs} and in direct line of sight. They cannot be observed to be a BDE by attackers or passengers, as it operates undercover.

Three different strategies can be employed by the BDE: static observation, dynamic observation and intelligent observation. When performing static observation, the BDE positions itself at the queue in front of the security checkpoint and executes its job there. For dynamic observation, the BDE constantly moves between two areas: the *checkpoint_area* and the *check-in_area*. Finally, when performing intelligent observation, the BDE estimates every $t_{intelligent}$ seconds which area has most passengers. The BDE will then move to the area with the highest number of passengers and performs its job there.

The BDE randomly chooses one agent of these observed agents (that it did not evaluate yet) to evaluate if it is an attacker or not. To do that, the BDE assigns points to the observed agent based on the SPOT program [37, 38, 43]. First, a threshold $d_{threshold}$ is defined for level of disorientation d. If the observed agent has a level of disorientation $d > d_{threshold}$, two points are assigned. Moreover, the suitability of luggage s is compared against a threshold $s_{threshold}$. If the agent exceeds the threshold, three points are assigned. Finally, if the difference between the arrival time $t_{arrival}$ and the flight time of an agent exceeds the threshold $f_{threshold}$, one point is assigned. If the number of points exceeds four, the BDE attempts to arrest the agent. If the agent is a passenger, the passenger is arrested and the BDE will leave the airport terminal with the passenger. If the agent is an attacker, the *arrest* action is executed with a success rate of p_{arrest}, while the *arrest_fail* action is executed otherwise. If the arrest action is executed, the attacker is stopped and will not detonate the IED. If the arrest was not successful, the attacker detonates the IED on the spot.

Table 1: The model parameters that were varied in the experiments.

Parameter	**Values**
Number of flights f	1, 2, 3 flights
Number of checkpoint lanes open l	2, 3, 4 lanes
Number of check-in desks open k	3, 5 desks
Number of BDEs d	0, 1, 2 empl.
BDE strategy	*static*, *dynamic*, *intell.*
Attacker time	*early*, *late*

It takes some time $t_{evaluation}$ to evaluate the agent. This time is calculated as follows:

$$t_{evaluation} = t_{max} - (c_1 \cdot abs(d_{threshold} - d) + c_2 \cdot abs(s_{threshold} - s))$$

where t_{max} is the maximum time that a BDE spends on evaluation of agents, and the c_i's are constant. This relationship indicates that passengers with traits close to the threshold take longer to evaluate than passengers that are not.

The BDE uses the above described observations and actions to execute the *behavior_detect_activity*. In this activity, the BDE moves between a list of locations *location_list* in the airport terminal, while checking if it observed an attacker. When this is the case, the employee tries to arrest the attacker.

It is noted that both the attacker and the BDEs can be modelled to be more complex than the current form. For example, more strategic behavior (i.e., a small decoy attack) in both the attacker and the BDEs can be included. Collaboration between teams and camera observations could also be added. For now, this is beyond the scope of this work.

4.5 Model Parameters

Five model parameters were defined and shown in Table 1. Other internal parameters of the models are discussed Section 6.1.

Passenger arrival at the airport follows a distribution based on the number of flights f and data collected at the regional airport. This has a direct influence on the number of passengers present within the model over time. The number of checkpoint lanes open refers to the number of passengers that can perform the *checkpoint_activity* simultaneously. This influences the number of employees directly as follows: $n_{checkpoint} = 4l + mod(l, 2)$. This relationship indicates that it is beneficial to open checkpoint lanes in pairs, as also recommended by IATA [44]. The number of check-in desks open refers to the number of check-in desks through which a passenger can check in. An open check-in desk requires a single employee. The number of BDEs present influences the number of employees present, and potentially the effectiveness of the defense. Furthermore, three BDE strategies are defined: static, dynamic and intelligent. Some of these parameters cannot be influenced by the airport directly. For example, the number of flights also depends on airlines, and the number of BDEs has to be determined in collaboration with regulators. Finally, the attacker time t_{attack} defines the time that the attacker executes its attack.

5 Estimation of Security and Efficiency

The third step of the proposed methodology estimates security risks and efficiency performance indicators based on the agent-based models described above. They are discussed in detail below.

5.1 Efficiency Estimation

The efficiency performance indicators, as defined in Section 3 are calculated as follows. The time in checkpoint queue for passengers T_{queue} is measured by calculating the time a passenger spends in the *queuing_area* closest to the *checkpoint_area*. A passenger is considered to have missed its flight if it is not in the *gate_area* at time f_{time}. We define loss as follows:

$$loss = ((|P_{max}| - |P|) \cdot rev_p - miss \cdot c_{miss}$$

where $|P_{max}|$ is the maximum number of passengers that the airport can process. P is the set of passengers that arrived on the flight day and rev_p is the mean revenue per passenger. Furthermore, c_{missed} is the costs that an airport has for each passenger that misses a flight. The other efficiency performance indicators, number of employees and time to gate, are trivially obtained from the simulation results. For each of the defined efficiency performance indicators it holds that lower is better.

5.2 Security Risk Assessment

As defined in Section 2.3.2, the Consequence function needs to be defined. Furthermore, Threat Likelihood has to be estimated independently from the models. Both of these elements are described in more detail below.

5.2.1 IED Consequences

As an IED attack at an airport terminal is modelled, a Consequence model is defined to estimate the number of lives lost after an attack. The model is based on the work of Pope [45], who designed a prediction tool that is able to quickly assess the human injury after a terrorist attack. The Consequence model described below forms the Consequence function $C(M^j_{ied})$.

It is argued that there are two main causes for fatalities after an IED attack: blast wave propagation and fragmentation injuries. While other factors are of influence on human injuries, only these two elements are considered in this model.

Blast Wave Prediction The explosion of an IED causes the release of a lot of energy, resulting in the propagation of a blast wave. Rapid changes in pressure are associated with this blast wave and can cause injury or death. Kingery and Bulmash [46] show that there is a relation between the mass of the explosive, the distance to the explosive, and the incident pressure P. This relation is outlined below:

$$\begin{aligned} z &= \frac{d}{mass^{1/3}} \\ U &= k_0 + k_1 \log_{10} z \\ P &= c_0 + c_1 U + c_2 U^2 + \ldots + c_n U^n \end{aligned}$$

where d is the distance in meters between the IED and the target and $mass$ is the IED mass in kg. The k_i's and c_i's are constants, while P refers to the incident pressure in kPa. The relationship above assumes an unobstructed path between the IED and the target, while in practice walls and other physical objects can reflect the pressure wave. This is modelled by generating imaginary IEDs on a commensurate location on the other side of the wall. Walls are then ignored and the pressure contributions from both sources are superimposed to find the total pressure at a specific location.

The incident pressure at the location of each human agent is recorded and translated to a fatality probability, based on the work of Zipf and Cashdollar [47]. Finally, a random number is drawn to determine if the agent survived or not. The number of fatalities caused by the incident pressure is referred to as c_{blast}.

Fragmentation Prediction Apart from fatalities due to pressure changes, injuries and fatalities can arise due to the presence of fragments. Two types of fragments are distinguished: primary fragments and secondary fragments. Primary fragments are the fragments that are present within the IED, while secondary fragments are the fragments that originate from the environment (i.e., ceiling or other objects in the environment). Here, only a set of K primary fragments originating from the IED are considered. The initial direction Θ_{init} of a fragment is determined using a uniform distribution, while the initial speed v_{init} is set to be a constant.

The fragment will then move around the environment following a Newtonian motion model. If the path of the fragment intersects with a human, the distance that it covers within the human body (called depth of penetration, DOP) is recorded. A truncated linear relation between fatality probability and DOP is assumed. Finally, a random number is drawn to determine if the human survives or not, for each human that survived the blast impact. The number of human fatalities caused by fragmentation is referred to as c_{frag}.

Consequence Function The Consequence function is then defined to be the sum of the fatalities caused by the blast wave and the fragmentation.

$$C(M^j_{ied}, a_1) = c_{blast} + c_{frag}$$

In this function only the fatalities are taken into account. A more extended approach could also take into account injuries, damages to physical structures and indirect consequences, but this is currently beyond the scope of this work.

5.2.2 Threat Likelihood

Threat likelihood is based on the work of Grant and Stewart [25], which in turn is based on historic data originating from a terrorist database [48]. From this database, it was obtained that historically there were an average of 1.7 IED attacks on airport terminals in Western countries each year [49]. Based on an estimate of 100 to 200 large hub airports, Grant and Stewart finally obtain an estimate of 0.5-2.0%. This percentage means that there is between 0.5% and 2% chance per airport terminal per year that someone attempts to attack it. As small airports seem less likely to be a target for terrorists, we chose a conservative likelihood of 0.5% for such an attack. As this estimate is based on historical data, it may very well be inaccurate. Data from intelligence agencies can provide more accurate estimates of threat likelihood.

6 Experiments & Results

Experiments performed with the above discussed model are presented in this section. The setup of the experiments is discussed in Section 6.1 and the results are discussed in Section 6.2.

6.1 Model Calibration & Experimental Setup

We have calibrated the model based on airport data, literature data, and assumptions if no data could be obtained. The calibrated parameters are found in Appendix B. We simulate a flight morning, between 05:00-07:00, where 05:00 corresponds to $t = 0$ sec. All flights are defined with the same departure time, which is standard practice in the airport under consideration. This is due to noise restrictions that are enforced on the airport. We assume a load factor of 0.75 for all aircraft, leading to 135 passengers

per flight. The layout of the airport was shown in Figure 2. Revenue per passenger is based on an ACI economics report [50], while the costs per missed flights are based on assumptions. The proportion of checked-in passengers was based on estimates of airport managers. The actual proportion can be obtained from airline data, which was unavailable for airport managers. No data was available for the facility visits at the airport, so this was based on assumptions.

The desired speed was assumed to be 1 m/s, and only individual passengers were considered. We assumed a single carry-on luggage for passengers that were checked-in, and an additional checked luggage for passengers that were not checked-in. Based on discussions with airport managers, we assumed that 20% of passengers arrive in the first half hour, 60% of passengers arrive in the second half hour, and the remaining 20% of passengers arrive in the third half hour. Passengers in these blocks are generated using a Poisson distribution with an arrival rate that ensures that the right number of passengers arrive. Check-in times were based on estimates by airport managers. The checkpoint parameters were obtained by fitting a distribution over 102 manually collected checkpoint processing times between 05:00 and 07:00 at the airport on March 22nd 2017.

The observation radius r_{obs} of agents was assumed to be equal to 10 meter. The behavior-detection employee parameters were calibrated as follows. We assumed that a BDE arrests 0.025 passenger per hour, which falls within the range provided by the United States Government Accountability Office report [37]. Assuming that both passenger disorientation d and passenger luggage suitability s follow a normal distribution with mean 0 and variance 1, the BDE thresholds d_{thres} and s_{thres} become 2.395. We assumed that following the SPOT program, 75% of the time an attacker is observed. This leads the attacker disorientation d and attacker luggage suitability s to follow a normal distribution with mean 3.5, and we assumed the same variance as for passengers. Based on a CNN news report [51], we assumed that a BDE takes up to 20 seconds to evaluate the characteristics of a passenger. The corresponding evaluation constants c_i were based on assumptions. The arrest probability p_{arrest} was set to 0.8, based on the work of Price and Forrest [52].

The mass of the IED was based on a report by the Department of Homeland Security [53]. The number of particles and their initial speed were finally based on assumptions. Some of the constants found in Table 2 will benefit from more extensive sensitivity analysis in the future. The output variables are the number of employees n, mean time in checkpoint queue T_{queue}, the mean time to gate T_{gate}, the number of missed flights $miss$, the monetary loss $loss$ and the risks $r_{ied-early}$ and $r_{ied-late}$ of the threat scenarios, as set out in Section 5.

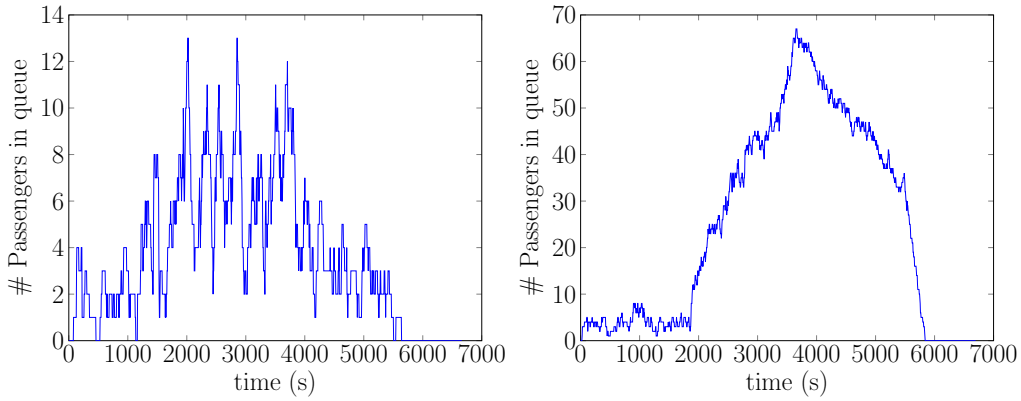
For the implementation of the model, we developed the AATOM simulator [54], a Java-based open-source agent-based airport terminal operations simulator. This simulator contains a large library of airport terminal related components, and basic implementations of attacker agents. A visualization of an AATOM simulation was shown in Figure 2. For each combination of model parameters, 500 simulation runs were executed.

6.2 Experimental Results

In this section, the results of the experiments are discussed. We first analyze the influence of the model parameters on efficiency, followed by an analysis of the influence on security. Finally, we discuss some of the relationships that were found between these performance areas. This constitutes to the fourth and last step in the proposed methodology.

6.2.1 Efficiency Performance Indicators

Figure 5 shows two typical buildups of passengers over time in the checkpoint queue, where Figure 5a shows the buildup under low passenger conditions, while Figure 5b shows a setup in saturated passenger conditions. From this figure the arrival pattern of passengers can be observed. When the slope of the figures changes, a different arrival rate of passengers is observed. This effect is more clearly visible in the three flight setup, as a larger queue buildup is observed there. This is due to the number of passengers in the queue being directly related to the mean queuing time T_{queue}.



(a) $f = 2$ flights.

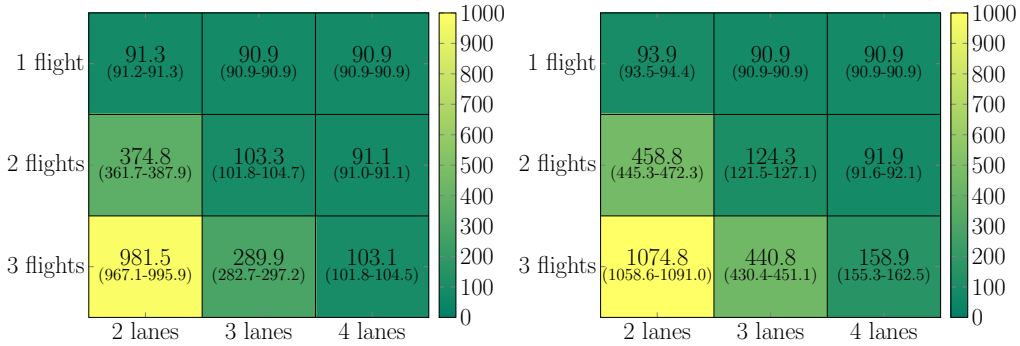
(b) $f = 3$ flights.

Figure 5: The number of passengers in the checkpoint queue over the flight morning. Graphs show a configuration of $l = 3$ checkpoint lanes and $k = 3$ check-in desks. Note that the scale of the y-axis is different for both configurations.

If we consider mean checkpoint queuing times T_{queue} for different airport setups (see Figure 6), it can be observed that three check-in desk setups mostly have shorter queuing times than five check-in desk setups. In the three check-in desk setups, the passengers arrive at the checkpoint queue more gradually due to longer waiting times at the check-in, leading to shorter queuing times. While not shown in the figure, it should be noted that five check-in desk setups generally lead to shorter times to gate for passengers. Furthermore, opening more checkpoint lanes leads to a higher number of employees present, but opening too few checkpoint lanes can lead to an increase in missed flights. Determining the number of checkpoint lanes and check-in desks is an important tradeoff that airports have to make on a regular basis with respect to these efficiency performance indicators. However, these decisions do not only influence efficiency of the airport but also security, as discussed in Section 6.2.2.

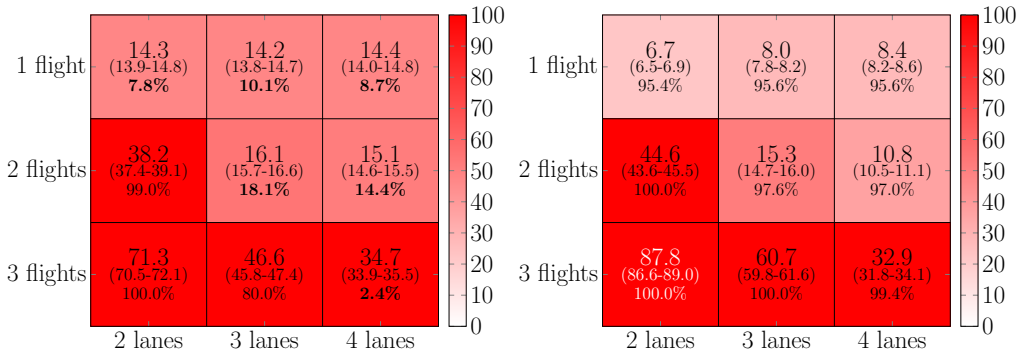
6.2.2 Casualties without Defenders

Figure 7 shows the mean number of casualties (in the case of a late attack) for different airport configurations. This corresponds to the conditional risks of the different threat scenarios. It further shows the choices of attacker (i.e., detonate IED at check-in or checkpoint) between the different configurations. In the three check-in desk setups, the attacker mostly chooses the check-in desks as a target, as most passengers are present in that area. However, this does not hold for the setups with two or three flights and two checkpoint lanes open and the setup with three checkpoint lanes open and three flights. The attacker has a strong preference for the checkpoint as a target in the five check-in desk setup. It nearly always chooses for this location as a target.



(a) $k = 3$ check-in desks. (b) $k = 5$ check-in desks.

Figure 6: The mean queuing time (in seconds) of passengers in the flight morning for different airport configurations. The values between brackets are the 95% confidence intervals.



(a) $k = 3$ check-in desks open. (b) $k = 5$ check-in desks open.

Figure 7: The number of casualties in a late attack for different airport configurations. The values between brackets are the 95% confidence intervals, and the percentages correspond to the proportion of times the attacker chooses for the checkpoint queue. Percentages smaller than 50% are shown in bold.

More flights generally lead to more casualties per flight as well. This is mainly caused by a nonlinear increase in queue lengths for increasing numbers of flights. When comparing the number of casualties with the number of checkpoint lanes, it can be observed that a higher number of checkpoint lanes results in a lower number of casualties per flight. This does not hold for the single flight case, as the number of casualties remains constant or even increases when more checkpoint lanes are opened. In this case, any number of checkpoint lanes is sufficient to prevent a buildup of passengers in the queue. The extra casualties (for the configuration with five check-in desks) are caused by the higher number of employees that are present at the checkpoint. In this situation, it is beneficial from both a security and efficiency perspective to reduce the number of checkpoint lanes open as much as possible. In all the other situations, it is beneficial from a security perspective to open more checkpoint lanes, but that clearly increases the number of employees. At the same time, mean queuing time T_{queue} is reduced. This constitutes to an important tradeoff that has to be made by airport managers.

If we compare the setups in which the check-in area was preferred by the attacker in

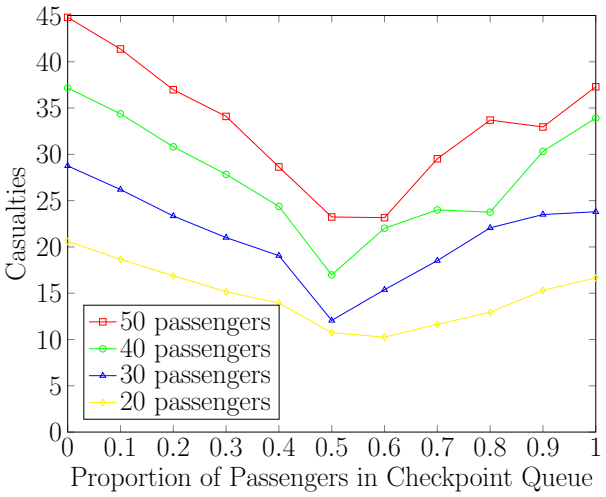


Figure 8: The relationship between the ratio of queue lengths and the number of casualties under different passenger loads.

the three check-in desk setups with the corresponding five check-in desk setups, it can be observed that with five check-in desks the number of casualties is reduced. This is the case, because the total number of passengers in the queue that is attacked is reduced. In general it holds that the size of the longest queue (i.e., checkpoint queue or check-in queue) is a good linear indicator for the expected number of casualties ($R = 0.72$). This also somewhat holds for the total number of passengers present in the open areas of the airport ($R = 0.59$), but not in situations in which at least ten passengers are present in the shorter queue ($R = 0.27$). To minimize the expected casualties, the airport should therefore minimize the size of the longest queue. Ideally, this is done by reducing the size of both queues. However, airport managers might consider the increased number of employees (i.e. efficiency decrease), not worth the reduced expected number of casualties. Alternatively, the size of the queues could be balanced as much as possible, by choosing the right number of check-in desks and checkpoint lanes. This is a result similar to the results of Grant and Stewart, who argue that distributed security queuing "will offer casualty reductions when used in preference to centralized security queuing" [25]. Figure 7 shows how to minimize the expected casualties in our reference airport.

To illustrate that the size of the queues should be balanced as much as possible, we performed a controlled experiment in which the total number of passengers is set to a constant, while distributing the passengers over the different queues according to different ratios. Figure 8 shows the number of casualties for different proportions of passengers in the checkpoint queue. In this figure, a minimum number of casualties was observed at a ratio of around 0.5. In this case, the queues are equally balanced. This is a result that can be generalized to similar situations in other airports as well. The consequence of an IED attack can be lowered by distributing passengers over the available space as well as possible.

The number of casualties was found to be a bit higher when all passengers are in the check-in queues as compared to the checkpoint queue. This is the case, because the attacker can better position himself between the passengers than in the checkpoint queue (as can be seen from Figure 2). This trend is reversed (although not shown in the figure) for very low passenger numbers, as also discussed above. It should be noted that this strategy of balancing queues might lead to increased security risks of other threat

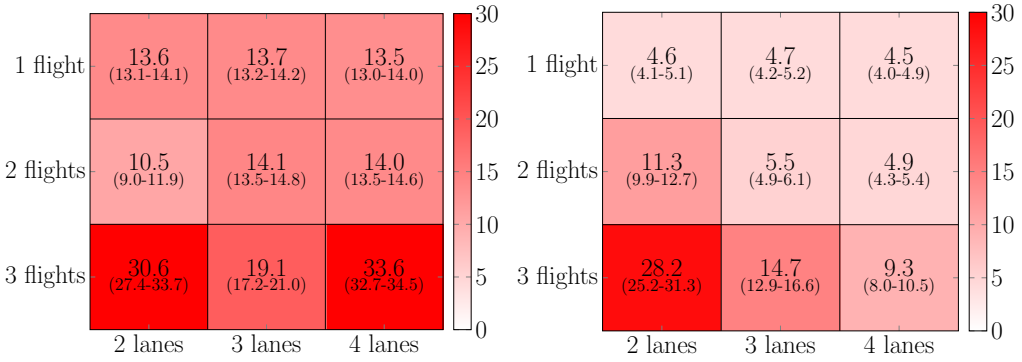
scenarios, not considered in this work. This forms an interesting direction for future research.

Different Passenger Types We analyzed the effect of different passenger types on our simulation results. We consider two passenger types in isolation: senior passengers and family passengers. The luggage drop time of senior passengers was calibrated to follow a normal distribution with mean 63.7 and variance of 35.1. Their luggage collect time follows a normal distribution with mean 59.4 and variance of 48.2. The luggage drop time of family passengers then follows a normal distribution with mean 69.2 and variance of 36.1, while their luggage collect time follows a normal distribution with mean 80.6 and variance of 53.0. These distributions were based on manually collected checkpoint processing times on four days in March and April 2018. The classification of passenger type was performed manually as well. It should be noted that these distributions already include the effects on processing speed for different amounts of luggage. Furthermore, a large part of the senior passengers considered fly several times per year from the airport under consideration.

It was found that the number of casualties is reduced with 12.0% for senior passengers on average. This is due to faster collection of luggage for this type of passengers, as compared to the passengers considered in the rest in this work. Contrary, family passengers move through the security checkpoint slower than the default passenger. This leads to an increase of 3.4% of casualties on average. The mix of passenger types has a large influence on security risk and efficiency performance indicators. Airports therefore need to consider the passenger mix they serve when making decision related to both security and efficiency.

6.2.3 Behavior-Detection Employee

In all airport setups and threat scenarios, the number of casualties is reduced when a (set of) BDE(s) is hired. This holds regardless of the strategy of the BDE. In general, the intelligent BDE is best capable of defending against attacks of different types. The agent is most frequently found at the area in which the attack will take place, and therefore performs more arrests than the other BDE types. A typical example of the performance of BDEs with different strategies is shown in Figure 9.



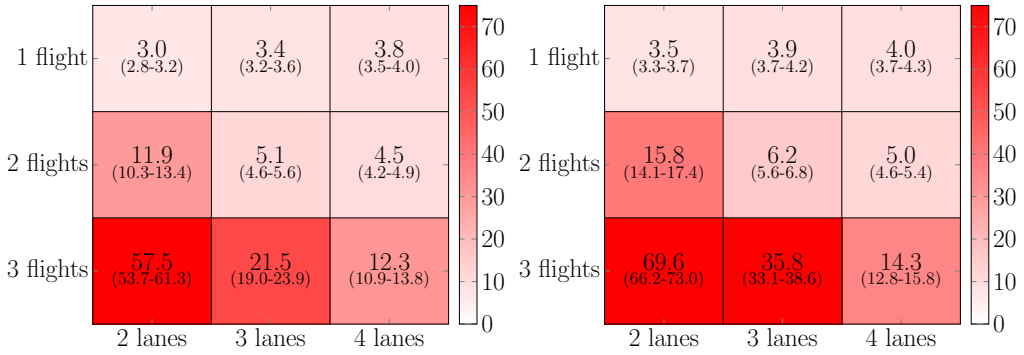
(a) Two static defenders.

(b) Two intelligent defenders.

Figure 9: The number of casualties in a late attack with three check-in desks for different types of defenders. The values between brackets are the 95% confidence intervals.

However, the intelligent defender is not always better capable of defending against attacks. Figure 10 shows the mean number of casualties in a late attack for two different

defender strategies: dynamic and intelligent. The static defender performs similar to the intelligent defender and is therefore not shown. In this case, the dynamic defender performs better than the intelligent defender.



(a) One dynamic defender.

(b) One intelligent defender.

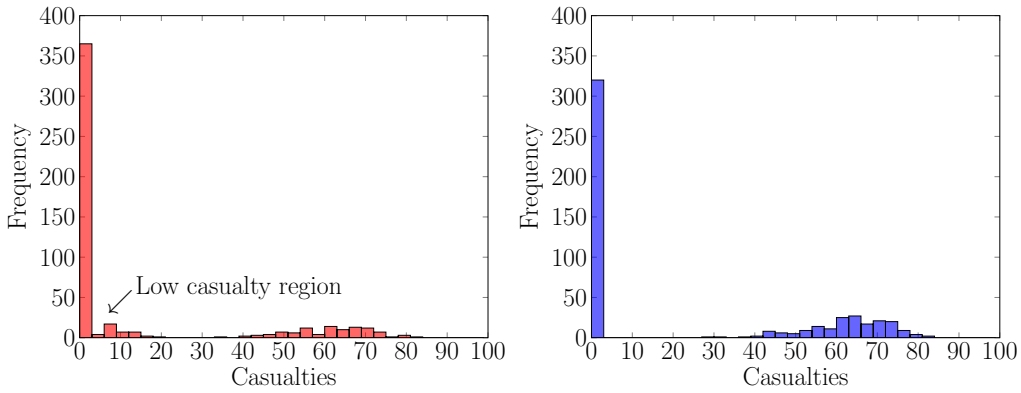
Figure 10: The number of casualties in a late attack with five check-in desks for different types of defenders. The values between brackets are the 95% confidence intervals.

This can be explained as follows. Figure 11 shows a histogram of casualties for the configuration with three flights, three checkpoint lanes, five check-in desks and a single BDE with different strategies. Note that in this configuration the checkpoint queue is much larger, and therefore the attacker always chooses this as a target. From Figure 11 it can be observed that the dynamic BDE make a higher number of arrests (zero casualties), and has a region in which a very low number of casualties is observed. This is the case, as the dynamic BDE moves between the check-in area and checkpoint area, while the other BDEs only perform their work in the checkpoint area. As the queue is long there, these other defenders do not have the time to assess every passenger, and therefore the attacker might be missed. The dynamic defender might observe the attacker (at the entrance area), as few other passengers are present in the check-in area. Note that these two areas are close together, and that the BDE can therefore observe passengers in both areas while it is in the check-in area. The region of very low casualties is caused by the failed arrests in this region. As only few passengers are around, fewer casualties are observed. On the contrary, when the intelligent defender (and also the static BDE) performs a failed arrest, the detonation of the IED occurs close to the checkpoint queue. This then leads to a higher number of casualties in the case of a failed arrest.

While not modelled in this work, observant passengers may also help prevent an ongoing attack to become successful. This was for instance seen in the 2018 Belgium train attack [55]. This forms an interesting direction for future research.

6.2.4 Security and Efficiency

To be able to determine the sensitivity of the estimated efficiency and security outputs to the model parameters, Spearman's rank correlation test was performed. This test assesses monotonic relationship between the parameters and outputs. Conditional risk ($R_c(M_{ied})$) is used as an output parameter, as Threat Likelihood remains constant for all parameter combinations. Figure 12 shows the results of this test and indicates insignificant results ($p \leq 0.05$) crossed out.



(a) One dynamic defender.

(b) One intelligent defender.

Figure 11: Histogram of casualties for two different defender strategies: intelligent and dynamic. Results are shown for a configuration with three flights, three checkpoint lanes, five check-in desks and a single defender.

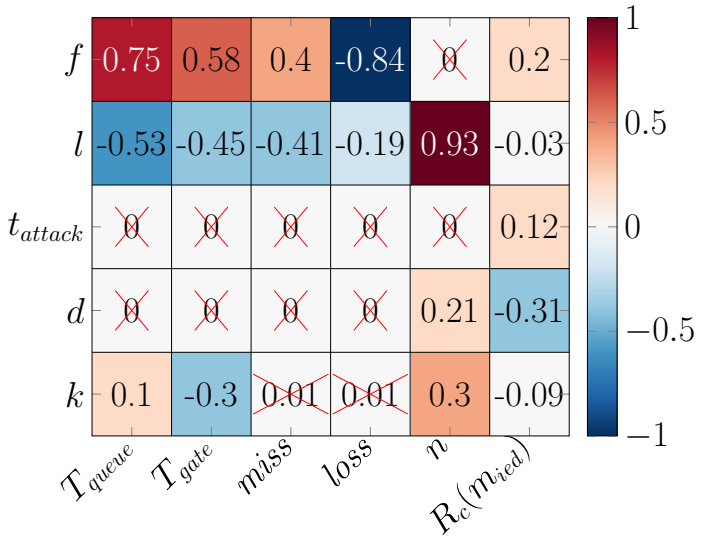
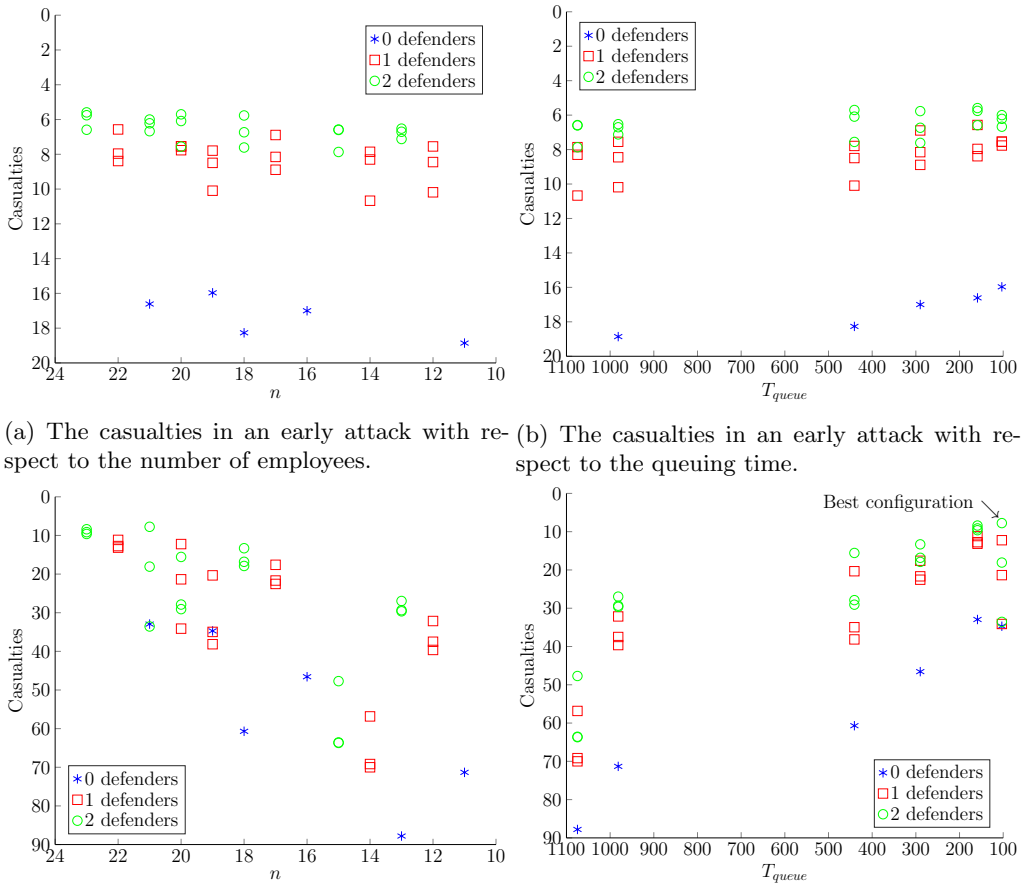


Figure 12: Spearman's rank correlation plot between model parameters and efficiency performance indicators and conditional security risks. The rows of this figure show the different model parameters, while the columns show the output parameters (efficiency performance indicators and conditional security risk). Insignificant results ($p \leq 0.05$) are crossed out.

Results show that the number of flights f had a positive correlation with each of the output parameters, with an exception of monetary loss. The number of checkpoint lanes open l shows opposite relationships with the parameters. For instance, fewer checkpoint lanes open results in longer time to gate T_{gate} and more casualties. This makes sense, as fewer checkpoint lanes open result in longer queues and longer queuing times. This in turn results in higher passenger densities in the queuing area, resulting in a higher number of fatalities. Furthermore, it shows that both the number of check-in desks open and the presence of a BDE have a low influence on most output parameters. However, the number of BDEs does have a negative correlation with the number of casualties.

We show this effect in more detail in Figure 13. Figure 13a-13b show the number of casualties in an early attack in relationship to queuing time and number of employees, while Figure 13c-13d show the same relationships for casualties in a late attack. Each of these results are shown for a three flight setup. It can be seen that the number of employees and queuing time do not have a strong relationship to the number of casualties in an early attack. However, in a late attack, the relationship becomes stronger. There is a strong negative relationship between the number of employees present and the expected number of casualties. This is a clear tradeoff that has to be made by airport managers, as also mentioned before. They have to choose how many more potential casualties they are willing to accept for a reduced number of employees. In contrary, the mean queuing time for passengers at the checkpoint has a positive relationship with the expected number of casualties. If we only consider these two output parameters, it is beneficial for airports to choose for configurations that lead to low casualties and queuing times. There is only one such configuration that minimizes both objectives: the configuration with four checkpoint lanes, two intelligent defenders, and three check-in desks (see also the configuration indicated with an arrow Figure 13). However, this is a configuration in which 21 employees are present; only two fewer than the maximum number. Similar results are found when T_{queue} is replaced with T_{gate}. Pareto analysis can further be used to determine which configurations are optimal with respect to the defined objectives.



(a) The casualties in an early attack with respect to the number of employees.

(b) The casualties in an early attack with respect to the queuing time.

(c) The casualties in a late attack with respect to the number of employees.

(d) The casualties in a late attack with respect to the mean queuing time.

Figure 13: The number of casualties in a three flight setup in relationship to the number of employees and queuing time. Note that the axes are reversed.

7 Conclusions & Future Work

Understanding security, efficiency and the relationships between them is essential, as airport managers regularly have to make decisions that influence these performance areas. Important decision regarding security and efficiency are often made based on experience and assumptions. This paper introduced a novel methodology to analyze security, efficiency, and relationships between these performance areas using agent-based modelling. It combines an agent-based security risk assessment methodology and a typical agent-based approach to analyze efficiency of operations. The methodology is capable of analyzing security, efficiency and their relationships in detail, and therefore forms a promising way to investigate different tradeoffs between security and efficiency.

The methodology was applied to a case study in a regional airport terminal. Relationships between risks regarding an IED attack and efficiency performance indicators, such as the average queuing time for passengers and number of employees, were quantified. Results showed that airports should attempt to spread passengers across the available space as much as possible. Furthermore, it was found that reducing security risks and improving efficiency were not always conflicting objectives. For example, de-

creasing the number of passengers in the open areas of the airport was found to be an effective measure to reduce security risks and improve different efficiency aspects.

Human behavior is far more complex than modelled in the discussed case study. More research is needed to include this complexity in the agent behavior. Furthermore, more extensive analysis, such as causal analysis [56] can be performed on results of the study. Another interesting possibility for further research is to integrate the proposed methodology with security games. This work could be used to determine payoff values in a security game, while the framework of security games can be used to find optimal defender policies. Furthermore, Pareto analysis can be performed to determine a set of dominant airport configurations. Another future direction of this work is that of applying the methodology to different domains, such as shopping malls and stadiums. Different threat scenarios, such as a shooting, and efficiency performance indicators, such as facility revenue, can also be investigated. Finally, the methodology could be generalized to identify relationships that also include other performance areas such as safety [57], resilience [58] and environmental impact [59].

A Definitions

The definitions below are adapted from the following references: [13, 60, 61].

Definition 1 (Security risk)**.** The potential for loss or harm due to the likelihood of an unwanted event and its adverse consequences.

Definition 2 (Threat)**.** Any indication, circumstance, or event with the potential to cause the loss of, or damage to, an asset.

Definition 3 (Threat Scenario)**.** A set of events, associated with a specific threat or multiple threats, partially ordered in time.

Definition 4 (Vulnerability)**.** Any weakness in an asset's or infrastructure's design, implementation, or operation that can be exploited by an adversary.

Definition 5 (Threat Likelihood)**.** The probability that an undesirable event will occur.

Definition 6 (Consequence)**.** The outcome of an event occurrence, including immediate, short- and long-term, direct and indirect losses and effects.

Definition 7 (Conditional Risk)**.** A measure of risk that focuses on consequences, vulnerability, and adversary capabilities, but excludes intent.

Definition 8 (Asset)**.** Item, thing or entity that has potential or actual value to an organization.

Definition 9 (Control)**.** Measure that is modifying risk.

B Calibration

Table 2: The calibrated model parameters.

Parameter	**Value**	**Origin**
Simulation parameters		
Simulation runs N	500 per configuration	-
Airport parameters		
Departure time F_{time}	7200 sec	Airport Data
Passengers per flight	135	Assumptions
Airport Layout	See Figure 2	Airport Data
Revenue per passenger rev_p	\$21.22	[50]
Missed flight costs c_{miss}	\$212.20	Assumption
Agent parameters		
Prop. passengers checked-in c	0.5	Airport Data
Prop. facility visit f ($none/bathroom/rest./shop$)	0.25/0.25/0.25/0.25	Assumption
Desired speed v_{des}	1 m/s	Assumption
Arrival Distribution ($early/middle/late$)	20%/60%/20%	Airport Data
Check-in time	$Norm(60, 6)$ sec	Airport Data
Luggage drop time	$Norm(54.60, 36.09)$ sec	Airport Data
Physical check time	$Norm(43.00, 20.96)$ sec	Airport Data
ETD check time	$Norm(34.80, 15.17)$ sec	Airport Data
Luggage collect time	$Norm(71.50, 54.95)$ sec	Airport Data
Observation radius r_{obs}	10 m	Assumption
Pass. disorientation d	$Norm(0, 1)$	[37]
Pass. luggage suitability s	$Norm(0, 1)$	[37]
Att. disorientation d	$Norm(3.5, 1)$	[37]
Att. luggage suitability s	$Norm(3.5, 1)$	[37]
Att. arrival time t_{attack}	1900 sec or 3900 sec	-
BDE threshold $d_{threshold}$	2.395	[37]
BDE threshold $s_{threshold}$	2.395	[37]
BDE threshold $f_{threshold}$	3600	[37]
BDE arrest prob. p_{arrest}	0.8	[52]
BDE maximum evaluation time t_{max}	20	[51]
BDE evaluation constants c_i	2.5	Assumption
IED parameters		
IED mass m	5 kg	[53]
Number of particles K	50	Assumption
Initial particle speed v_{init}	1000 m/s	Assumption

References

[1] ICAO, Annual report of the icao council: 2016, https://www.icao.int/annual-report-2016/Pages/default.aspx (2016).

[2] E. Fernandes, R. Pacheco, Efficient use of airport capacity, Transportation research part A: Policy and practice 36 (3) (2002) 225–238.

[3] R. Martens, Benchmarking the efficiency of terminal processes at regional airports, in: Air Transport and Operations: Proceedings of the Second International Air Transport and Operations Symposium 2011, IOS Press, 2011, p. 327.

[4] J. Pitchforth, P. Wu, C. Fookes, K. Mengersen, Processing passengers efficiently: An analysis of airport processing times for international passengers, Journal of Air Transport Management 49 (2015) 35–45.

[5] A. Kierzkowski, T. Kisiel, Simulation model of security control system functioning: A case study of the wroclaw airport terminal, Journal of Air Transport Management 64 (2017) 173–185.

[6] D. R. Pendergraft, C. V. Robertson, S. Shrader, Simulation of an airport passenger security system, in: Proceedings of the 36th conference on Winter simulation, Winter Simulation Conference, 2004, pp. 874–878.

[7] M. Schultz, H. Fricke, Managing passenger handling at airport terminals, in: 9th Air Traffic Management Research and Development Seminars, 2011.

[8] L. Cheng, C. Fookes, V. Reddy, P. K. Yarlagadda, Analysis of passenger group behaviour and its impact on passenger flow using an agent-based model, in: Simulation and Modeling Methodologies, Technologies and Applications (SIMULTECH), 2014 International Conference on, IEEE, 2014, pp. 733–738.

[9] ICAO, Aviation security manual (doc 8973 – restricted), Montreal, Canada: ICAO.

[10] Council of European Union, Council regulation (EU) no 300/2008, http://data.europa.eu/eli/reg/2008/300/oj (2008).

[11] Council of European Union, Council regulation (EU) no 1998/2015, http://data.europa.eu/eli/reg_impl/2015/1998/oj (2008).

[12] 107th Congress, Aviation and transportation security act, https://www.gpo.gov/fdsys/pkg/PLAW-107publ71/pdf/PLAW-107publ71.pdf (2001).

[13] A. Washington, All-Hazards risk and resilience: prioritizing critical infrastructures using the RAMCAP Plus [hoch] SM approach, ASME, 2009.

[14] D. J. Landoll, D. Landoll, The security risk assessment handbook: A complete guide for performing security risk assessments, CRC Press, 2005.

[15] H. H. Willis, A. R. Morral, T. K. Kelly, J. J. Medby, Estimating terrorism risk, Rand Corporation, 2006.

[16] S. Janssen, A. Sharpanskykh, R. Curran, AbSRiM: an agent-based security risk management approach for airport operations, Risk analysis (in press).

[17] B. Schneier, Attack trees, Dr. Dobb's journal 24 (12) (1999) 21–29.

[18] O. Gadyatskaya, R. Jhawar, P. Kordy, K. Lounis, S. Mauw, R. Trujillo-Rasua, Attack trees for practical security assessment: ranking of attack scenarios with adtool 2.0, in: International Conference on Quantitative Evaluation of Systems, Springer, 2016, pp. 159–162.

[19] P. K. Chawdhry, Risk modeling and simulation of airport passenger departures process, in: Winter Simulation Conference, Winter Simulation Conference, 2009, pp. 2820–2831.

[20] M. Brown, A. Sinha, A. Schlenker, M. Tambe, One size does not fit all: A game-theoretic approach for dynamically and effectively screening for threats., in: AAAI, 2016, pp. 425–431.

[21] A. Schlenker, M. Brown, A. Sinha, M. Tambe, R. Mehta, Get me to my gate on time: Efficiently solving general-sum bayesian threat screening games., in: ECAI, 2016, pp. 1476–1484.

[22] S. Janssen, A. Sharpanskykh, Agent-based modelling for security risk assessment, in: International Conference on Practical Applications of Agents and Multi-Agent Systems, Springer, Cham, 2017, pp. 132–143.

[23] D. Wilson, E. K. Roe, S. A. So, Security checkpoint optimizer (sco): an application for simulating the operations of airport security checkpoints, in: Proceedings of the 38th conference on Winter simulation, Winter Simulation Conference, 2006, pp. 529–535.

[24] A. A. Kirschenbaum, The cost of airport security: The passenger dilemma, Journal of Air Transport Management 30 (2013) 39–45.

[25] M. J. Grant, M. G. Stewart, Benefit of distributed security queuing for reducing risks associated with improvised explosive device attacks in airport terminals, ASCE-ASME Journal of Risk and Uncertainty in Engineering Systems, Part B: Mechanical Engineering 3 (2) (2017) 021003.

[26] T. Bosse, C. M. Jonker, L. Van der Meij, A. Sharpanskykh, J. Treur, Specification and verification of dynamics in agent models, International Journal of Cooperative Information Systems 18 (01) (2009) 167–193.

[27] T. Bosse, C. M. Jonker, L. Van Der Meij, J. Treur, A language and environment for analysis of dynamics by simulation, International Journal on Artificial Intelligence Tools 16 (03) (2007) 435–464.

[28] M. Bratman, Intention, plans, and practical reason, Harvard University Press: Cambridge, MA.

[29] F. M. Brazier, B. M. Dunin-Keplicz, N. R. Jennings, J. Treur, Desire: Modelling multi-agent systems in a compositional formal framework, International Journal of Cooperative Information Systems 6 (01) (1997) 67–94.

[30] A. Saltelli, S. Tarantola, F. Campolongo, M. Ratto, Sensitivity analysis in practice: a guide to assessing scientific models, John Wiley & Sons, 2004.

[31] M. Fonoberova, V. A. Fonoberov, I. Mezić, Global sensitivity/uncertainty analysis for agent-based models, Reliability Engineering & System Safety 118 (2013) 8–17.

[32] J. C. Thiele, W. Kurth, V. Grimm, Facilitating parameter estimation and sensitivity analysis of agent-based models: A cookbook using netlogo and r, Journal of Artificial Societies and Social Simulation 17 (3) (2014) 11.

[33] B. M. Blumberg, T. A. Galyean, Multi-level direction of autonomous creatures for real-time virtual environments, in: Proceedings of the 22nd annual conference on Computer graphics and interactive techniques, ACM, 1995, pp. 47–54.

[34] S. P. Hoogendoorn, P. H. Bovy, Pedestrian route-choice and activity scheduling theory and models, Transportation Research Part B: Methodological 38 (2) (2004) 169–190.

[35] C. W. Reynolds, Steering behaviors for autonomous characters, in: Game developers conference, Vol. 1999, 1999, pp. 763–782.

[36] S. Janssen, A.-N. Blok, A. Knol, Aatom - an agent-based airport terminal operations model, https://pure.tudelft.nl/portal/en/publications/aatom--an-agentbased-airport-terminal-operations-model(eb03ba8b-a4ec-4754-9f63-41a0774c531a).html (2018).

[37] U. S. G. A. Office, TSA Should Limit Future Funding for Behavior Detection Activities, U.S. Government Accountability Office, 2013.

[38] J. Winter, C. Cora, Exclusive: Tsaś secret behavior checklist to spot terrorists, https://theintercept.com/2015/03/27/revealed-tsas-closely-held-behavior-checklist-spot-terrorists/, accessed: 2018-03-27 (2015).

[39] D. D. Harabor, A. Grastien, et al., Online graph pruning for pathfinding on grid maps., in: AAAI, 2011.

[40] F. T. Johora, P. Kraus, J. P. Müller, Dynamic path planning and movement control in pedestrian simulation, arXiv preprint arXiv:1709.08235.

[41] D. Helbing, P. Molnar, Social force model for pedestrian dynamics, Physical review E 51 (5) (1995) 4282.

[42] M. Pearson, What you need to know about the turkey airport attack, http://edition.cnn.com/2016/06/29/europe/turkey-attack-up-to-speed/index.html, accessed: 2017-09-18 (2016).

[43] B. A. Cotton, Strategic improvements to tsa spot program, Tech. rep., NAVAL POSTGRADUATE SCHOOL MONTEREY CA (2015).

[44] IATA, Checkpoint of the future - blueprint 2014, Report, IATA (2012).

[45] D. J. Pope, The development of a quick-running prediction tool for the assessment of human injury owing to terrorist attack within crowded metropolitan environments, Philosophical Transactions of the Royal Society of London B: Biological Sciences 366 (1562) (2011) 127–143.

[46] C. N. Kingery, G. Bulmash, Airblast parameters from TNT spherical air burst and hemispherical surface burst, US Army Armament and Development Center, Ballistic Research Laboratory, 1984.

[47] R. K. Zipf Jr., K. L. Cashdollar, Explosions and refuge chambers, https://www.cdc.gov/niosh/docket/archive/pdfs/niosh-125/125-explosionsandrefugechambers.pdf, accessed: 2017-09-18 (n.d.).

[48] G. LaFree, L. Dugan, Introducing the global terrorism database, Terrorism and Political Violence 19 (2) (2007) 181–204.

[49] START, Armed assault at los angeles international airport (lax), accessed: 2018-10-18 (Nov 2013).
URL https://www.start.umd.edu/pubs/STARTFactSheet_ArmedAssaultatLAX_Nov2013.pdf

[50] Airports Council International, Airport economics report and key performance indicators, http://www.aci.aero/News/Releases/Most-Recent/2016/03/07/ACI-releases-the-20th-edition-of-the-Airport-Economics-Report-and-Key-Performance-Indicators (2016).

[51] H. Handeyside, Be careful with your face at airports (opinion), accessed: 2018-10-18 (Mar 2015).
URL https://edition.cnn.com/2015/03/19/opinions/handeyside-tsa-spot-program/index.html

[52] J. C. Price, J. S. Forrest, Chapter 11 - the threat matrix, in: J. C. Price, , J. S. Forrest (Eds.), Practical Aviation Security (Third Edition), third edition Edition, Butterworth-Heinemann, Boston, 2016, pp. 461 – 513. doi:https://doi.org/10.1016/B978-0-12-804293-9.00011-4.

[53] National Acedemies, Ied attack: Improvised explosive devicesAccessed: 2018-10-18.
URL https://www.dhs.gov/xlibrary/assets/prep_ied_fact_sheet.pdf

[54] S. Janssen, AATOM - an agent-based airport terminal operations simulator, https://github.com/StefJanssen/AATOM (2017).

[55] R. Ellis, J. King, P. Dailey, A. Seshadri, 2 members of u.s. military stop islamist attacker on train in belgium, accessed: 2018-11-02 (Jan 2018).
URL https://edition.cnn.com/2015/08/21/europe/france-train-shooting/index.html

[56] D. R. Heise, Causal analysis., John Wiley & Sons, 1975.

[57] S. H. Stroeve, G. Bakker, H. A. Blom, Safety risk analysis of runway incursion alert systems in the tower and cockpit by multi-agent systemic accident modelling, in: Proceedings of 7th USA/Europe Air Traffic Management R&D Seminar, 2007.

[58] S. H. Stroeve, T. Bosse, H. A. Blom, A. Sharpanskykh, M. H. Everdij, Agent-based modelling for analysis of resilience in atm, Proceedings of the Third SESAR Innovation days. Stockholm (Sweden), Novermber.

[59] M. Hatzopoulou, J. Y. Hao, E. J. Miller, Simulating the impacts of household travel on greenhouse gas emissions, urban air quality, and population exposure, Transportation 38 (6) (2011) 871.

[60] ISO, Information technology – security techniques – information security management systems – overview and vocabulary, Tech. rep., International Organization for Standardization, Geneva, CH (2016).

[61] I. 251, Iso 55000:2014. asset management — overview, principles and terminology, Standard, International Organization for Standardization, Geneva, CH (2014).