An Investigation of the Effectiveness of Interdiction Regimes Against Terrorist Attacks in an Urban Transport Hub

D. Keep [a], **I. Piper** [a] **and A. Green** [a]

[a]*School of Computer Science and Software Engineering, University of Wollongong, Northfields Ave, Wollongong, NSW 2500*
Email: daniel.keep@gmail.com

Abstract: This paper examines the effect of varying attack and interdiction strategies, both alone and in combination, in an urban transport hub. Particular attention is paid to the potential disruption to normal commuter services resulting from an intrusive stop and search regime.

The work presented here represents a qualitative investigation in that many parameters relating to the details of the interdiction mechanisms are first-order approximations. However, the background against which the investigation is conducted has been constructed to be as realistic as possible.

The simulation is performed using our generalised microsimulation framework *Simulacron*, along with a repertoire of simulation modules. Changes and additions to these modules required for the present study are described.

The station is modelled using 454 distinct locations, most of which are interconnected to form a directed graph to permit commuter movement. An average of some 8,300 commuters move through the station, outbound and inbound. Figure 1 is a frame from the baseline simulation of the station, generated by a custom post-processing tool and rendered by LightWave, showing commuters and trains. The train schedule is constructed from the real schedule for the 3rd of June, 2013. It is from this that the commuter population is derived.

Five attack strategies (including no attack) are matrixed against four interdiction strategies (including no protection) to produce 20 scenarios. Some key results are presented, along with brief remarks on the remainder. An additional 30 variant scenarios were used to examine a stop and search interdiction strategy. For these scenarios, impact on commuters was inferred from the number of outbound commuters missing their intended train, how long commuters spent waiting in checkpoint queues, and the size of these queues, on average and at peak. Congestion at the search point was also used as an indication of the increased vulnerability resulting from stop and search.

Our conclusions from this work are that overt, intrusive protection schemes appear to be more efficacious than passive or covert means. However, the former present their own problems in both disruption to commuter activities and the creation of new potential targets. In the case of stop and search, this takes the form of unacceptable commuter delays and congestion at the checkpoints. As a result of this, the overall interdiction regime must be adjusted to protect the new target. Doing so without introducing additional targets may prove challenging. It seems that the "best" likely outcome is to redirect the terrorist attack to a softer target.
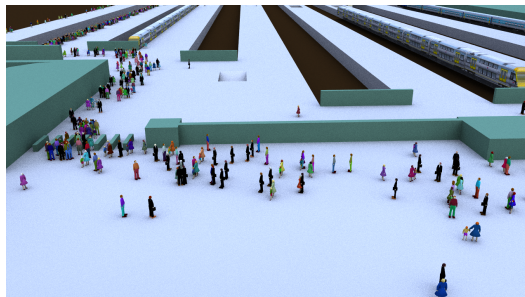
Figure 1. Visualisation of the baseline simulation.

Keywords: Simulation, terrorism, attack, interdiction.
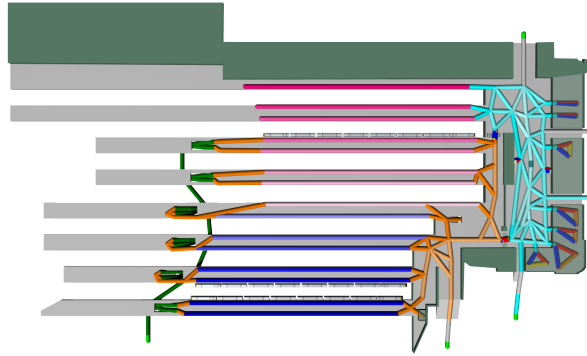
Figure 2. Central Station showing cell structure superimposed on (decorative) station geometry.

1 INTRODUCTION

Microsimulation is a valuable technique for analysing systems with reasonably well-understood micro-level processes, but whose macro-level observable results are not directly predictable (Merz, 1991). It has been applied in a number of realms including the modelling of terrorist attacks (Henson et al., 2009). This paper builds directly upon our previous work (Keep et al., 2011) in which we modelled the Sydney Central Station transportation hub. The goal of this work is to conduct a qualitative examination of various attack and interdiction strategies focusing particularly on their disruptive effects, rather than how commuters react to such hypothetical attacks (one such attack was the subject of our previous work). We present the details of the construction of our scenarios, the results and a discussion of the implications. The choice of a transportation hub is based upon the critical nature of such facilities and their vulnerability to attack (Szyliowicz, 2009).

The scenarios investigated in this paper were simulated using our microsimulation package called *Simulacron*. Simulacron is divided into a generalised framework, and a number of simulation modules. Together, these modules provide the actual simulation behaviour. The world is modelled using *cells* and *peeps*; cells are abstract locations and peeps are abstract entities within those locations. Further details can be found in our previous publications (Keep et al., 2011; Piper et al., 2010); a more complete description of Simulacron can be found in Keep (2012).

The most significant new addition to our simulation capabilities is the introduction of stop and search mechanics to the *TPC* (Terrorist, Police, Citizen) module. This allows the creation of checkpoints where a proportion of peeps are queued and subsequently subjected to either a cursory or a thorough search.In addition to the stop and search mechanic, a number of other changes and additions were made. *TPC* was modified to allow terrorists to attack specific cells, in addition to the previous timed attack behaviour. The new *Peep cell actions* module allows actions to be executed whenever a peep arrives in a given cell and was used to implement patrolling behaviours.[1] Other minor changes were made to improve the fidelity and functioning of the model. Although the setting for this work is still the Sydney Central railway transport hub, the details of the model are significantly different from those used in Keep et al. (2011). Revisions made to the model include:

- The cells which make up the station were rebuilt, changing their number, position and connectivity to better reflect the layout of the station and pedestrian flows. This was greatly assisted by the development of a new tool allowing us to convert a 3D model created in a standard modelling package[2] into a usable form. As part of this, the shape and size of the model was adjusted to more closely match maps of the station. Figure 2 shows this new layout. Colour is used to group cells with similar properties.

- The template, which is used to construct the input data for the simulation, was rebuilt in order to accommodate the more complicated construction of the model. Large amounts of complex template code were replaced with external Python scripts, allowing for less redundancy in specification and more complex transformations to be performed. As an example of this, all commuters and trains are procedurally generated from a simple listing of the train timetable. Python scripts were also used to automate the generation, simulation and post-processing of the scenarios. Allowances for the greater number of variations in the scenario were also made. All variants are described by a scenario file, each of which can

---

[1]This module differs from *Induced peep actions* in that the sets of actions are attached to peeps as opposed to cells.
[2]LightWave version 11.5.

override any given aspect or parameter in the base scenario. Most scenarios are created by combining individual attack and interdiction strategy descriptions.

- The simulation time was reduced from two hours to ninety minutes, with the first half hour used to prepare the station.[3] The final sixty minutes was the reporting period for all simulations.

- The number of people in the simulation overall was changed to more accurately reflect actual traffic. Generating the population using the number and capacity of the trains in the simulated period (16:00-17:00 on Monday, 3rd of June, 2013) produced roughly 8,500 commuters.

- Intermediate goals (shops, toilets and cafés) were redesigned to take time to pass through. In a related change, purchasing a ticket now takes a fixed amount of time rather than using a queue. This change, although arguably reducing the realism of the simulation, was adequate for the problem at hand.

- Trains, both inbound and outbound, were added to the model (some of which can be seen in figure 2 standing at platforms 4, 13 and 14). The real-world timetable[4] for 3rd of June, 2013 was used to construct the simulated timetable. In addition, commuters who miss their train will automatically reroute to whichever platform the next appropriate train will depart from. This was done without introducing new simulation modules or mechanics.

- The behavioural parameters were removed entirely. Since this work was not concerned with *how* people reacted to the terrorist attack, they were unnecessary.

- A number of other small changes were made, which included: making the ticket barriers faster to pass and decreasing the walking speed of peeps to 1 ms^{-1} (Guerrier and Jolibois, 1998).

For each outbound train, a number of commuters (the total per train is sampled from the normal distribution $\mu = 896$, $\sigma = 50$) intending to board that train is generated. Each commuter arrives at the station through a random entrance and proceeds to purchase a ticket if required (50%). They then go through zero or more intermediate goals (visiting a shop, a café and/or a bathroom) in a random order. Finally, they proceed to the noticeboard, which directs them to the appropriate platform to await and/or board their train. Each commuter's arrival time T_a is determined by: $T_a = T_b - t_e - (t_t + t_m)$ where: T_b is the time at which the commuter intends to board the train, uniformly sampled from the standing interval of the train. t_e is the estimated amount of time it will take the commuter to get from their ingress to the platform. This estimate is currently fixed at 4.8 minutes. t_t is the estimated time to purchase a ticket, if one is needed. This estimate is currently fixed at 0.6 minutes. t_m is a time margin within which the commuter may pursue their intermediate goals. The margin is sampled uniformly from the range $[4.8, 30]$ minutes.

Inbound commuters are similar except that they arrive on a platform from a train, do not purchase a ticket and do not visit the noticeboard. They can, like outbound commuters, have intermediate goals. After this, each inbound commuter will then proceed to a randomly chosen exit.[5]

In total, the final model consisted of 454 cells and an average of 8,392 peeps.[6] The number of commuters entering and leaving the station is in general agreement with reported data (BTS, 2012). The approximately 600 kilobytes of template code, parameters files and supporting scripts and assets expand to over 250 input data set instances taking over one gigabyte of storage, and half a terabyte of output databases. A single 90 minute run takes approximately 20 minutes on a normal consumer desktop machine.

2 INITIAL SCENARIOS

In the first round of simulations conducted, a number of attack and interdiction strategies were modelled independently, then combined in every possible permutation. The attack strategies were:

- **Nothing**: there is no attack.

- **AttackConcourse**: a single terrorist attacks the noticeboard.

---

[3]This pre-simulation time allowed a realistic distribution of commuters throughout the station, which would otherwise start empty.
[4]Sourced from `http://131500.com.au/`.
[5]Commuters entering the concourse and bound for the underground platforms are not included in the current model but these represent a small proportion of the pedestrian traffic during the evening peak.
[6]Simulations were run across multiple instances of the template. Each of these yielded a different population.

- **AttackPlatform**: the terrorist attacks a specific platform location, intending to detonate shortly before an outbound train is scheduled to commence boarding.

- **AttackQueue**: the terrorist directly attacks the ticket barrier queue.

- **AttackPawn**: a bomb with a 10 minute timer is planted on an unsuspecting outbound commuter. This action occurs outside the simulation.

The interdiction strategies modelled were:

- **Nothing**: there is no attempt at interdiction.

- **Patrols**: there are two officers patrolling different routes through the station.

- **EntranceGuard**: a single officer is stationed at each of the entrances to the station.

- **EntranceGuardAndPatrols**: a combination of the previous two strategies.

For the **EntranceGuard*** strategies, one guard is placed at each of the entrances where they attempt to identify and arrest the terrorist. The ***Patrols** strategies involve two patrolling guards, with similar behaviours. Of note is that all of the above interdiction strategies are "passive" in that they do not introduce any disruption to the flow of commuters, unless a terrorist detonates on being detected. Also, the parameters used for these interdiction strategies are first-order approximations only. No formal sensitivity analysis has been undertaken to tune them.

The combination of these yielded 20 different scenarios. Each of these scenarios was instantiated six times, with each instance producing a different population. It should be noted that the *same* population is used for corresponding instances of each scenario. All instances were simulated a single time with a consistent random seed. There was no significant value in performing multiple runs of each instance.

2.1 Results and discussion

None of the results obtained from these simulations showed any meaningful deviation from what might be reasonably anticipated. No formal validation or verification was performed as this is a proof-of-concept, rather than a precise simulation.

The traffic patterns seen were consistent with informal expectations. There was some clustering around the noticeboard, ticket window and ticket machine, with some minor clustering around the ticket barriers. One unanticipated observation was that peeps would sometimes choose unusual, but optimal, paths through the station.[7] The peak number of people in each location was recorded for the baseline (*i.e.* no attack, no interdiction) runs. It was found that the highest average concentration of commuters was on the departing carriages (each of which holds approximately 150 people). The next highest concentrations (around 100 people) were to be found on the platforms where commuters were waiting to board a train, just before boarding commenced. Next were the incoming carriages (approximately 60 people per carriage),[8] and finally the concourse itself, with a comparable peak of about 60, and typical values in the 20s and 30s. The average peak at the noticeboard was 46.

Most of the attack strategies resulted in a predictable number of casualties ranging from single digits at the ticket barriers, to around 30 at the noticeboard, up to 60 on the platform. The exception to this was the AttackPawn strategy which delivered casualties ranging from 2 (on a platform) to 371 (aboard a Newcastle-bound train). AttackPawn is a particularly insidious strategy as the unsuspecting individual carrying the bomb is unlikely to exhibit any tell-tale suspicious behaviours, making them far more difficult to identify without invasive searching.

The effectiveness of interdiction was almost entirely predictable from the chosen parameters. At this scale of simulation, and with the reasonably realistic numbers of police officers, there is little interaction between the police and the terrorist. As a result, patrols do not appear to be very useful as, with any reasonable patrol length, there is a low chance of encountering the terrorist. This could, in part, be an artefact of the patrol model, as officer and terrorist must occupy the same cell for detection to occur. In essence, all of the police officers

---

[7]This further reinforces our suspicion that real people do not use A*.

[8]The difference in inbound and outbound occupancies reflects the afternoon, peak period traffic.

are *extremely* near-sighted. The entrance guards, on the other hand, seem to work relatively well as everyone entering the station must go past one of them. At that point, however, the effectiveness of this strategy depends on the observational skills of the guard versus the evasion skills of the terrorist; these were set to 0.25 and 0.8 respectively, which yields an approximately 25% chance of detection per 5 second tick.

The interdiction and attack strategies appear to be, at least in the current model, completely orthogonal. The proportion of successful attacks is simply 0.75^t where t is the cumulative number of ticks in which a terrorist and a police officer are co-located. Patrols contribute a relatively small proportion of this time. The potential exists to develop a more complex behavioural model for the terrorists and police, but we have not attempted this to date. As a result, we decided not to run attack strategies against the stop and search strategy.

3 STOP AND SEARCH SCENARIOS

The second round of simulations were designed to investigate the impact of an intrusive interdiction strategy (stop and search). This impact was measured by examining the number of outbound commuters who missed their desired trains and how long they spent waiting in checkpoint queues, as well as the average and peak length of such queues.

In addition to the previously given reason for not explicitly considering the terrorist in these scenarios, it should be noted that there are only four plausible responses, on the part of the terrorist, to the introduction of a stop and search strategy. They are: ignore the checkpoints and proceed as intended; attack somewhere before the checkpoint; attack the checkpoint itself; and do not attack at all. For the purposes of this study, none of these responses change the simulation in any meaningful fashion. The likely outcome to each of these can be estimated directly from the existing cell census counts in the output.

Two checkpoints were created, one at each of the ticket barriers, on the concourse side, checking outbound commuters. The length of the queues is effectively unbounded. Parameters of interest (and their base values) include the chance for a commuter to be added to the search queue (10%), the proportion of queued commuters to be given a thorough (as opposed to a cursory) search (10%), and the times for both the cursory (15 seconds) and thorough (45 seconds) search procedures.

A number of derived checkpoint strategies were created which varied these parameters. Specifically:

- **CheckpointBase**: The baseline checkpoint scenario with all parameters set to default.

- **MoreSearch**: Thorough search chance was multiplied by a factor of 3 to 30%.

- **SlowSearch**: Thorough search time was multiplied by a factor of 3 to 2:15. This scenario is designed to look at the effect of a more thorough search, assuming that longer searches proportionally improve detection chance.

- **MoreQueue_xN** $[2, 9]$: Chance to be queued multiplied by a factor of N.

- **QueueScale_xN** $[2, 9]$: Chance to be queued multiplied by a factor of N, with search times (both cursory and thorough) reduced by the same factor.

- **MoreSearchers_xN** $[2, 5]$: Search times (both cursory and thorough) are divided by a factor of N to simulate multiple parallel searchers.

Each scenario was instantiated three times, using the same set of three distinct populations. Each of these was simulated once, as varying the population provided better variation than changing the simulation seed.

3.1 Results

Figure 3 shows the average time spent in a queue for outbound commuters passing through the checkpoints, graphed against the search chance. The first two series show the times for the MoreQueue_xN scenarios. The "scaling" series show the same for the QueueScale_xN scenarios. Note that the 10% search chance data points are taken from the CheckpointBase scenario in both cases, as it is the "x1" variant of each.

From the MoreQueue_xN scenarios, it can be seen that with a single searcher, wait times, especially for the busier checkpoint, are intolerable. For this checkpoint, the average wait time for 10% search chance is 8 minutes. These times rapidly increase as search chance increases. The scaling scenarios demonstrate that the effects of increasing the number of searchers and increasing the search chance cancel each other out. With
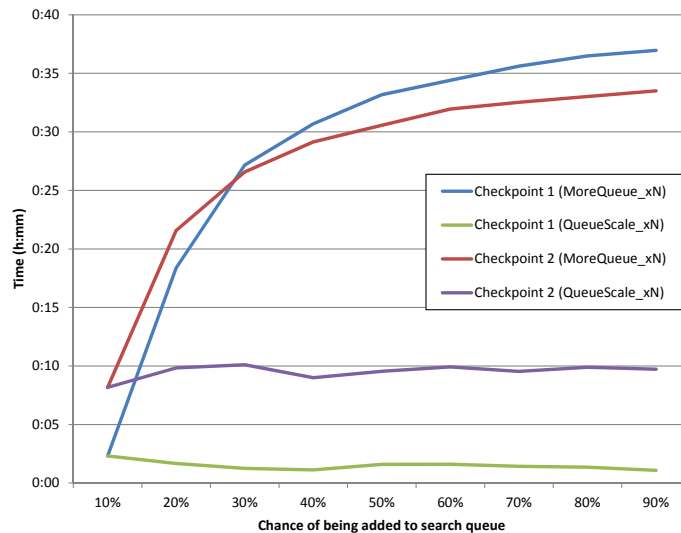
Figure 3. Average time in queue

the MoreSearchers_xN scenarios, we found that two searchers are sufficient to comfortably cover 10% search, with queue times dropping to around 40 seconds at each checkpoint. Beyond two searchers, search times drop to and stay around 20 seconds. For the MoreSearch and SlowSearch scenarios, the average time spent in the two queues became (7:13, 16:34), and (11:54, 19:04) respectively for each of the two checkpoints in turn. From this we observe that, in regards to minimising disruption, it is better to search a larger proportion of people than it is to search a small proportion for longer, assuming the improvement to detection chance is comparable.

In the base scenario with no checkpoints, there are, on average, four commuters who miss their intended train. For the MoreQueue_xN and QueueScale_xN scenarios, the average number of commuters who miss their train scales linearly with N, ranging from 14 at 10% search to 2,909 at 90%, and from 14 at 10% to 115 at 90%, respectively. In the MoreSearchers_xN scenarios, the value jumps from 4 (in the base scenario) to 14 for 1x, before diminishing to 4 thereafter. Both MoreSearch and SlowSearch demonstrate a significant increase in this metric, with 88 and 133 commuters missing their trains respectively. Again, increasing the proportion of commuters searched appears less disruptive than applying a more thorough search regime to a fixed proportion.

The average queue lengths increase linearly with N, ranging from $(4, 31)$ at 10% search to $(891, 1211)$ at 90% for MoreQueue_xN, and from $(4, 31)$ at 10% to $(14, 324)$ at 90% for QueueScale_xN. A similar pattern was observed with the peak queue length, with values ranging from $(17, 106)$ to $(1885, 2984)$ for MoreQueue_xN, and from $(17, 106)$ to $(55, 931)$ for QueueScale_xN. With the MoreServer_xN scenarios, for both measures, the lengths dropped rapidly as N increased. The average length dropped to 0 at 2x, with peak length falling to single-digits by 3x. Similarly, the average queue lengths are $(17, 65)$ for MoreSearch and $(30, 77)$ for SlowSearch. The peak queue lengths are $(43, 171)$ and $(63, 201)$ respectively.

Subsequent to this, a set of scenarios derived from the MoreSearchers_xN set were constructed, in which the search chance was increased to 90%, and the number of searchers was varied between 10 and 20.[9]

In order to eliminate any increase in the number of commuters missing their trains, more than 10 searchers would be needed.[10] For the busier checkpoint, more than 14 searchers would bring the average checkpoint queue length below 20 people. In order to eliminate the queues, a minimum of 18 would be required. In order to bring the peak queue length below 50 on the busiest checkpoint, more than 16 searchers are required. With more than 14 searchers, average time spent queuing drops to below one minute.

4 DISCUSSION

The above results clearly demonstrate that overt, intrusive interdiction, while more effective, has a significant downside in that it has the potential to create massive disruption (up to 75% of outbound commuters missing their train) in addition to providing terrorists with an attractive target of opportunity.

---

[9]We already know that 9 searchers are inadequate from the results of the MoreSearchers_x9 scenario.

[10]The value of N for these scenarios was incremented by 2 A value of $N = 12$ was found to be sufficient.

Consideration of the baseline peak cell occupancies suggests that once a search queue exceeds approximately 50 people, it is arguably the best target in the station, excluding those beyond the checkpoint. For the busiest checkpoint, 10% search results in a peak queue length of over 100 people. In terms of average queue length over time, a search rate of 20% is sufficient to exceed 160 people. In order to avoid either checkpoint becoming a target at 10% search, a total of three searchers is required, with two at the busier checkpoint. As noted above, at a 90% search rate, at least 16 searchers are needed at the busier checkpoint to keep the peak queue length below 50 people. For the other checkpoint, 10 are sufficient.

Two remedies present themselves: reduce the size of the security checkpoint queues to acceptable levels; or provide sufficient passive surveillance to significantly lower the chance a terrorist will be able to reach the checkpoint. Both present logistical difficulties. The first strategy can be achieved by dropping the search rate and/or by increasing the number of searchers. For any reasonable queue size, this leads to either an unacceptably low search rate or an unacceptably high number of searchers. In regards to the second, for 90% coverage with acceptably low disruption, 26 searchers would be required. However, only 16 searchers could conveniently be accommodated with the current ticket barrier structure. Thus, the only solution in this case, for Central Station, would appear to be a significant expansion in the size of the ticket barriers.

In regards to the second remedy, two approaches present themselves. Either increase the effectiveness of each observer and/or increase the number of observers. Effectiveness could be increased by using sniffer dogs or fixed chemical sensors (on entrances). We cannot say how much the effectiveness or number of observers would need to be increased for an acceptable chance of detection, but it seems reasonable to assume the requirements would be not insignificant. If it is possible to provide sufficient protection for the checkpoint, the checkpoint itself can now only be justified if it is protecting a target of significantly higher value.

Sadly, any sufficiently effective interdiction strategy capable of protecting such a transport hub would probably cause the terrorist to direct their attentions to some other, more attractive target (e.g. shopping malls).

One avenue for future expansion on this work is to model multiple, coordinated attacks in which the second attacker exploits the confusion and congestion resulting from the first attack. Although we do not have such scenarios finalised at the time of writing, we anticipate presenting the results of these at the conference. Another area we would like to improve upon is the modelling of the interactions between the terrorists and the police, including terrorist evasion of conspicuous police presence and pursuit of suspected terrorists by police. Finally, there are various minor improvements that could be made such as more realistic queue processing, better intermediate goal modelling, and expanding the simulation to the full station.

REFERENCES

(2012). Compendium of Sydney Rail Travel Statistics, 8th. Technical report, Bureau of Transport Statistics.

Guerrier, J. H. and S. C. Jolibois (1998). The safety of elderly pedestrians at five urban intersections in miami. In *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, Volume 42, pp. 171–175. SAGE Publications.

Henson, K., K. Goulias, and R. Golledge (2009). An assessment of activity-based modeling and simulation for applications in operational studies, disaster preparedness, and homeland security. *Transportation Letters: the International Journal of Transportation Research 1*(1), 19–39.

Keep, D., R. Bunder, I. Piper, and A. Green (2011). Application of Microsimulation Towards Modelling of Behaviours in Complex Environments. In *Workshops at the Twenty-Fifth AAAI Conference on Artificial Intelligence*.

Keep, D., I. Piper, A. Green, and R. Bunder (2011). Modelling of Behaviours in Response to Terrorist Activity. In *Proceedings of the 2011 MODSIM International Congress on Modelling and Simulation*, pp. 475–481.

Keep, D. R. (2012). *Generalised Microsimulation*. Ph. D. thesis, University of Wollongong.

Merz, J. (1991). Microsimulation - A survey of principles, developments and applications. *International Journal of Forecasting 7*(1), 77–104.

Piper, I., D. Keep, T. Green, and I. Zhang (2010). Application of Microsimulation to the Modelling of Epidemics and Terrorist Attacks. *Informatica 34*, 141–150.

Szyliowicz, J. S. (2009). *Transportation Security Against Terrorism*, Chapter Terrorism, Mobility and Transportation Security, pp. 13–26. IOS Press.