TRUSTS: Scheduling Randomized Patrols for Fare Inspection in Transit Systems using Game Theory

Zhengyu Yin and **Albert Xin Jiang** and **Milind Tambe**

University of Southern California, Los Angeles, CA 90089, USA
{zhengyuy, jiangx, tambe}@usc.edu

Christopher Kiekintveld	**Kevin Leyton-Brown**	**Tuomas Sandholm**	**John P. Sullivan**
University of Texas at El Paso	University of British Columbia	Carnegie Mellon University	LA County Sheriff's Department
El Paso, TX 79968, USA	Vancouver, BC V6T 1Z4, Canada	Pittsburgh, PA 15213, USA	Los Angeles, CA 91754, USA
cdkiekintveld@utep.edu	kevinlb@cs.ubc.ca	sandholm@cs.cmu.edu	jpsulliv@lasd.org

Abstract

In *proof-of-payment* transit systems, passengers are legally required to purchase tickets before entering but are not physically forced to do so. Instead, patrol units move about the transit system, inspecting the tickets of passengers, who face fines if caught fare evading. The deterrence of fare evasion depends on the unpredictability and effectiveness of the patrols.

In this paper, we present TRUSTS, an application for scheduling randomized patrols for fare inspection in transit systems. TRUSTS models the problem of computing patrol strategies as a leader-follower Stackelberg game where the objective is to deter fare evasion and hence maximize revenue. This problem differs from previously studied Stackelberg settings in that the leader strategies must satisfy massive temporal and spatial constraints; moreover, unlike in these counterterrorism-motivated Stackelberg applications, a large fraction of the ridership might realistically consider fare evasion, and so the number of followers is potentially huge. A third key novelty in our work is deliberate simplification of leader strategies to make patrols easier to be executed. We present an efficient algorithm for computing such patrol strategies and present experimental results using real-world ridership data from the Los Angeles Metro Rail system. The Los Angeles County Sheriff's department is currently carrying out trials of TRUSTS.

Introduction

In the Los Angeles Metro Rail system and other proof-of-payment transit systems worldwide, passengers are legally required to buy tickets before boarding, but there are no gates or turnstiles. There are, quite literally, no barriers to entry, as illustrated in Figure 1. Instead, security personnel are dynamically deployed throughout the transit system, randomly inspecting passenger tickets. This proof-of-payment fare collection method is typically chosen as a more cost-effective alternative to direct fare collection, i.e., when the revenue lost to fare evasion is believed to be less than what it would cost to directly preclude it.

For the LA Metro, with approximately 300,000 riders daily, this revenue loss can be significant; the annual cost has been estimated at $5.6 million (Booz Allen Hamilton

Figure 1: Entrance of an LA Metro Rail station.

2007). The Los Angeles Sheriffs Department (LASD) deploys uniformed patrols onboard trains and at stations for fare-checking (and for other purposes such as crime prevention), in order to discourage fare evasion. With limited resources to devote to patrols, it is impossible to cover all locations at all times. The LASD thus requires some mechanism for choosing times and locations for inspections. Any predictable patterns in such a patrol schedule are likely to be observed and exploited by potential fare-evaders. The LASD's current approach relies on humans for scheduling the patrols. However, human schedulers are poor at generating unpredictable schedules (Wagenaar 1972; Tambe 2011); furthermore such scheduling for LASD is a tremendous cognitive burden on the human schedulers who must take into account all of the scheduling complexities (e.g., train timings, switching time between trains, and schedule lengths). Indeed, the sheer difficulty of even enumerating the trillions of potential patrols makes any simple automated approach—such as a simple dice roll—inapplicable.

The result of our investigation is a novel application called TRUSTS (**T**actical **R**andomization for **U**rban **S**ecurity in **T**ransit **S**ystems), for fare-evasion deterrence in urban transit systems, carried out in collaboration with the LASD. We take a *game-theoretic* approach, which studies systems with multiple self-interested parties and aims to predict the likely outcomes of the system under rational behavior of the players. In particular, we model this problem as a *Stackelberg game* with one leader (the LASD) and many followers, in which each metro rider (a follower) takes a fixed route at a fixed time. The leader precommits to a mixed patrol strategy (a probability distribution over all pure strategies), and riders observe this mixed strategy before deciding whether

to buy the ticket or not (the decision to ride having already been made), in order to minimize their expected total cost, following for simplicity the classic economic analysis of rational crime (Becker and Landes 1974). Both ticket sales and fines issued for fare evasion translate into revenue to the government. Therefore the optimization objective we choose for the leader is to maximize total revenue (total ticket sales plus penalties).

There are exponentially many possible pure patrol strategies, each subject to both the spatial and temporal constraints of travel within the transit network. Explicitly representing a mixed strategy would be impractical. To remedy this difficulty, TRUSTS uses the *transition graph*, which captures the spatial as well as temporal structure of the domain, and solves for the optimal (fractional) flow through this graph, using linear programming (LP). Such a flow can be interpreted as a marginal coverage vector. Additionally, we show that a straightforward approach to extracting patrol strategies from the marginals faces important challenges: it can create infeasible patrols that violate the constraint on patrol length, and it can generate patrols that switch too frequently between trains, which can be difficult for patrol personnel to carry out. Thus, we present a novel technique to overcome these difficulties using an extended formulation on a *history-duplicate transition graph* that (1) forbids patrols that are too long and (2) penalizes patrols with too many switches.

Finally, we perform simulations based on actual ridership data provided by the LASD, for four LA Metro train lines (Blue, Gold, Green, and Red). Our results suggest the possibility of significant fare evasion deterrence and hence prevention of revenue loss with very few resources. The LASD is currently testing TRUSTS in the LA Metro system by deploying patrols according to our schedules and measuring the revenue recovered.

Related Work

There has been research on a wide range of problems related to game-theoretic patrolling on graphs. One line of work considers games in which one player, the patroller, patrols the graph to detect and catch the other player, the evader, who tries to minimize the detection probability. This includes work on *hider-seeker games* (Halvorson, Conitzer, and Parr 2009) for the case of mobile evaders, and *search games* (Gal 1979) for the case of immobile evaders.

Another line of research considers games in which the patroller deploys resources (static or mobile) on the graph to prevent the other player, the attacker, from reaching certain target vertices. There are a few variations depending on the set of possible sources and targets of the attacker. *Infiltration games* (Alpern 1992) considered one source and target. *Asset protection* problems (Dickerson et al. 2010) consider multiple sources and multiple equally weighted targets. *Networked security games* (Tsai et al. 2010) consider targets with different weights.

The leader-follower Stackelberg game model has been the topic of much recent research (Tambe 2011) and has been applied to a number of real-world security domains, including the Los Angeles International Airport (Jain et al. 2010),

the Federal Air Marshals Service (Jain et al. 2010), and the Transportation Security Administration (Pita et al. 2011).

Urban transit systems, however, present unique computational challenges. First, unlike in existing work on graph patrolling games, and unlike in previous deployed applications on counterterrorism, here the followers we seek to influence are potentially very many: large numbers of train riders might plausibly consider fare evasion. Booz Allen Hamilton (2007) estimates that 6% of riders are ticketless in the metro system overall; anecdotal reports suggest that on some lines this percentage could be far greater, even a majority. Second, the leader has exponentially many possible patrol strategies, corresponding to all the feasible trips within the transit network subject to certain restrictions and preferences. Similar to FAMS (Jain et al. 2010), we represent patrol strategies compactly as a marginal coverage vector. But unlike the FAMS problem in which a patrol consists of very limited number of flights (often a pair of flights), TRUSTS allows much more complex patrols and thus uses a novel compact representation based on *history-duplicate transition graphs*.

Game Theory Background

We begin with a brief overview of the game-theoretic concepts that we use in this paper. At a high level, game theory studies *games*, which are mathematical models of interactions among multiple players, each trying to advance their self-interest by choosing among a set of strategies. Formally, a game has a set N of players; each player $i \in N$ has a set S_i of *pure strategies*; the outcome of the game is determined by the strategy profile $s \in S$, where $S = \prod_{i \in N} S_i$ is the Cartesian product of the player's sets of pure strategies. Each player i's preference over the outcomes of the game is specified by her utility function $u_i : S \to \mathbb{R}$, and her objective is to maximize her expected utility. Each player can choose a pure strategy, or a mixed strategy which is a probability distribution over pure strategies. Player i's *best response* strategy, given strategies of the other players, is a strategy of i that maximize i's expected utility. It is always possible to pick a best response that is a pure strategy.

In a *Stackelberg game* between a leader and a follower, the leader commits to a mixed strategy, and the follower observes leader's mixed strategy and plays a best response (Conitzer and Sandholm 2006). This setting has been applied to security domains where the defender's daily patrol strategies are observed by attackers. The optimal mixed strategy for the leader to commit to, together with the follower's best response, form a *Stackelberg equilibrium* of the game.

Problem Setting

TRUSTS addresses the challenge of generating randomized schedules for LASD patrols for four LA Metro lines (Blue, Gold, Green, and Red). For simplicity, TRUSTS treats the LA Metro System's multiple lines as independent. Indeed, currently, LA's metro lines have just a few transfer points. Dealing with the impact of transfers will be a topic for future work. We model the patrol scheduling problem on each line as a leader-follower Stackelberg game with one leader

(a) Map of the Gold Line.
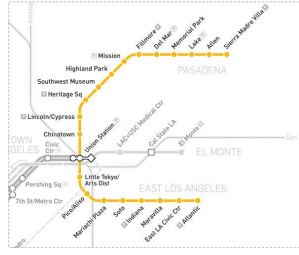
(b) A snippet of the Gold Line's timetable.

Atlantic	East LA Civic Center	Indiana	Soto	Mariachi Plaza	Little Tokyo/Arts District	Union Station	Chinatown
—	—	—	—	—	—	3:40A	3:43A
—	—	—	—	—	—	4:10	4:12
—	—	—	—	—	—	4:28	4:30
4:21A	4:23A	4:31A	4:34A	4:35A	4:41A	4:45	4:47
—	—	—	—	—	—	4:56	4:58
4:41	4:43	4:51	4:54	4:55	5:01	5:05	5:07
—	—	—	—	—	—	5:14	5:16
5:00	5:02	5:10	5:13	5:14	5:20	5:24	5:26
—	—	—	—	—	—	5:32	5:34
5:18	5:20	5:28	5:31	5:32	5:38	5:42	5:44
—	—	—	—	—	—	5:48	5:50
5:30	5:32	5:40	5:43	5:44	5:50	5:54	5:56
—	—	—	—	—	—	6:00	6:02
5:42	5:44	5:52	5:55	5:56	6:02	6:06	6:08
5:48	5:50	5:58	6:01	6:02	6:08	6:12	6:14
5:54	5:56	6:04	6:07	6:08	6:14	6:18	6:20

Figure 2: LA Metro's Gold Line.

(the LASD) and multiple followers (riders). In this game, a pure leader strategy is a *patrol*, i.e., a sequence of *patrol actions* (defined below), of constant bounded duration. The two possible pure follower strategies are buying and not buying. There are many types of followers, one for each source, destination, and departure time triple (corresponding to the set of all riders who take such a trip).

Train System: The train system consists of a single line (e.g., the Gold line as shown in Figure 2(a)) on which trains travel back and forth, in general with multiple trains traveling simultaneously. The system operates according to a fixed daily schedule, with trains arriving at stations at (finitely many) designated times throughout the day. Figure 2(b) shows a snippet of the Gold line's timetable taken from http://www.metro.net. Therefore we can model time as discrete, focusing only on the time steps at which some train arrival/departure event occurs. We use the (directed) *transition graph* $G = \langle V, E \rangle$ to encode the daily timetable of the metro line, where a vertex $v = \langle s, t \rangle$ corresponds to some pair of station s and time point t. An edge in G represents a possible (minimal) action. In particular, there is an edge from $\langle s, t \rangle$ to $\langle s', t' \rangle$ if:

• s' is either the predecessor or successor of s in the station sequence and $\langle s, t \rangle$ and $\langle s', t' \rangle$ are two consecutive stops for some train in the train schedule (*traveling action*), or
• $s' = s$, $t < t'$, and there is no vertex $\langle s, t'' \rangle$ with $t < t'' < t'$ (*staying action*).
We refer to the entire path that a given train takes through G, from the start station to the terminal station, as a *train path*.

Patrols: There are a fixed number γ of deployable patrol units, each of which may be scheduled on a patrol of duration at most κ hours (with, e.g., $\kappa = 7$). There are two sorts of patrol actions, which a given patrol unit can alternate between on its shift: *on-train* inspections (in which patrollers ride the train, inspecting passengers), and *in-station* inspections (in which they inspect passengers as they exit the station). A pure patrol strategy is represented mathematically as a path in G for each patrol unit, in which an edge e represents an *atomic* patrol action, i.e., inspecting in-station from the time of one train event at that station to the next (at that station) or inspecting on-train as it travels from one station to the next. Each edge e has a length l_e equal to the corresponding patrol action duration and an effectiveness value f_e, which represents the percentage of the relevant ridership inspected by this action. For both in-station and on-train in-

spections, f_e depends on the ridership volume at that location and time of day and on the duration. A valid pure patrol strategy is a set of paths $P_1, ..., P_\gamma$, each of size at most κ.

Example 1. *A simple scenario with* 3 *stations* (A, B, C) *and* 4 *discrete time points (6pm, 7pm, 8pm, 9pm) is given in Figure 3. The dashed lines represent staying actions; the solid lines represent traveling actions. There are* 4 *trains in the system; all edge durations are 1 hour. A sample train path here is* $\langle A, 6pm \rangle \rightarrow \langle B, 7pm \rangle \rightarrow \langle C, 8pm \rangle$. *In this example, if* $\kappa = 2$ *and* $\gamma = 1$, *then the valid pure leader strategies (pure patrol strategies) consist of all paths of length* 2.
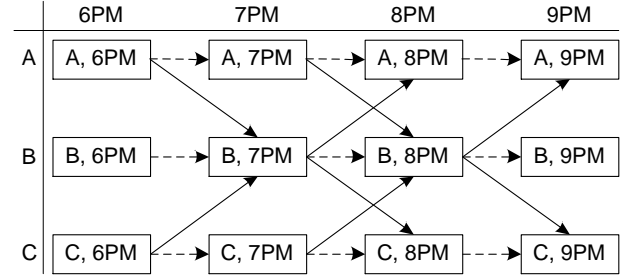

Figure 3: The transition graph of a toy problem instance.

Riders: The riders are assumed to be daily commuters who take a fixed route at a fixed time. Horizon Research Corporation (2002) estimates that more than 82% of riders use the system at least 3 days a week. The ticket price (for any trip within the transit system) is a nominal fee ρ, with the fine for fare evasion τ much greater. As the riders follow the same route every day, they could estimate the likelihood of being inspected, based on which they make a decision as to whether to buy a ticket. We assume the riders know the inspection probability perfectly, and are rational, risk-neutral economic actors (Becker and Landes 1974), who make this choice in order to minimize expected cost.

A rider's *type* λ is defined by the path he takes in the graph. Because there is a single train line, riders never linger in stations, i.e., do not follow any "stay" edges (staying at a station) mid-journey; the last edge of every follower type is a (short) stay edge, representing the action of "exiting" the destination station, during which the rider may be subject to in-station inspection. Therefore the space Λ of rider types corresponds to the set of all subpaths of train paths. (When

G is drawn as in Figure 3, all rider paths are "diagonal" except for the last edge.) A metro line with N stops and M scheduled trains will have $O(MN^2)$ rider types.

Given a pure patrol strategy of the γ units, $(P_1, \ldots, P_\gamma)$, the inspection probability for a rider of type $\lambda \in \Lambda$ is:

$$\min\{1, \sum_{i=1}^{\gamma} \sum_{e \in P_i \cap \lambda} f_e\}, \qquad (1)$$

and therefore his expected utility is the negative of the expected amount he pays: $-\rho$ if he buys the ticket and $-\tau \min\{1, \sum_{i=1}^{\gamma} \sum_{e \in P_i \cap \lambda} f_e\}$ otherwise. The inspection probability for a mixed strategy is then the expectation of Equation (1), taken over the distribution of pure strategies.

We justify the inspection probability in Equation (1) as follows. First, consider on-train inspections. The fraction of the train that is inspected in a given inspection action is determined by f_e (which depends on ridership volume). The key is that in the next inspection action, a patrol will not reinspect the fraction of the train that is already inspected in a previous inspection action. Therefore, unlike in settings where patrollers may repeatedly draw a random sample from the same set of train passengers to inspect, in our setting, the probabilities f_e are added rather than multiplied. Now also consider in-station inspections. Since a rider taking a journey only exits a single station, a rider will encounter at most one in-station inspection. Finally, when multiple patrol units cover the same edge e, the inspection probability given by (1) is the sum of the contributions from each patrol unit, capped at 1. This is a reasonable assumption when the number of patrol units on each edge e is small, as multiple patrol units on the same train could check different cars or different portions of the same car, and multiple patrol units inspecting at the same station could be checking different exits.

Objective: The leader's utility, equal to total expected revenue, can be decomposed into utilities from bilateral interactions with each individual follower; furthermore the followers do not directly affect each other's utilities. As a result, the game can be equivalently formulated as a two-player *Bayesian Stackelberg game*, in which there is just one follower taking on the role of one of the passengers. The type of this passenger is known to the follower but not to the leader, and is drawn from a probability distribution such that the probability p_λ of a follower type $\lambda \in \Lambda$ is proportional to its ridership volume.

Furthermore, although in general solving Bayesian Stackelberg games is NP-complete (Conitzer and Sandholm 2006), the utility functions of this game satisfy the *zero sum* property, i.e., the utility gained by the leader always equals to the utility lost by the follower. For such zero-sum Bayesian games, the Stackelberg equilibrium is equivalent to the *maximin* solution, which are solvable by linear programming formulations of (Ponssard and Sorin 1980) or (Koller, Megiddo, and von Stengel 1994). However, these LP formulations explicitly enumerate the pure strategies of the leader; since our game of interest has an exponential number of leader pure strategies, even storing these formulations would require exponential space.

Linear Program Formulation

In this section, we formulate a linear program which finds a maximum-revenue (mixed) patrol strategy. As noted above, the leader's space of *pure* strategies is exponentially large, even with a single patrol unit. We avoid this difficulty by compactly representing mixed patrol strategies by marginal coverage on edges x_e of the transition graph (the marginal strategy), i.e., by the expected numbers of inspections that will occur on these edges. Subsequently, we construct a mixed strategy (i.e., a probability distribution over pure strategies) consistent with the marginal coverage.

For expository purposes, we first present a basic LP formulation of our approach of compactly representing the problem using marginal coverage. This basic formulation also illustrates the key issues that make it difficult for the end-user to deploy the patrol strategies computed. We then introduce an extended formulation to address these issues.

Basic Formulation

We denote the set of possible starting vertices in the transition graph $G = \langle V, E \rangle$ by $V^+ \subset V$, and the set of possible ending vertices by $V^- \subset V$. For algorithmic convenience, we add to the transition graph a source v^+ with edges to all vertices in V^+ and a sink v^- with edges from all vertices in V^-. We assign these additional dummy edges zero duration and zero effectiveness.

Based on this graph, we provide a linear program (shown in Figure 4) to provide an upper bound on the optimal revenue achievable. Here u_λ denotes the expected value paid by a rider of type λ, and so $p_\lambda u_\lambda$ is the expected total revenue from riders of this type; x_e is the expected number of inspections on edge e. Constraint (4) bounds the total flow entering and exiting the system by γ, the number of total patrol units allowed. Constraint (5) enforces conservation of flow, which clearly is satisfied by any mixed patrol strategy. Constraint (6) limits the total number of time units to $\gamma \cdot \kappa$, and also bounds x_e for each e. Since x_e is the expected coverage of e, it could theoretically go up to as much as γ. On the other hand our assumption that the inspection probability is added across patrol units (Equation (1)) is valid only when a small number of units are on each edge. Thus we have another parameter $\alpha \in [1, \gamma]$ as the upper-bound of x_e.

Finally, let us consider Constraint (3), which indicates that the rider will best respond, by bounding the expected cost to a rider of type λ by both the ticket price ρ and $\tau \min\{1, \sum_{e \in \lambda} x_e f_e\} = \min\{\tau, \tau \sum_{e \in \lambda} x_e f_e\}$, the formulation's estimate of the expected fine if the rider chooses not to buy. However, the latter is only an *overestimate* of the actual expected fine of not buying. This is because the expression $\min\{1, \sum_{e \in \lambda} x_e f_e\}$ only caps the expectation (over its pure strategies) of the inspection probability at 1, but allows a pure strategy $(P_1, \ldots, P_\gamma)$ in its support to achieve $\sum_{i=1}^{\gamma} \sum_{e \in P_i \cap \lambda} f_e > 1$, whereas according to (1) the inspection probability of each pure strategy should be at most 1. This results in an overestimate of the actual inspection probability (and thus the leader's utility). As a result the solution of this LP provides only an upper bound on the optimal revenue. Fortunately, once we generate the

$$\max_{\mathbf{x},\mathbf{u}} \sum_{\lambda \in \Lambda} p_\lambda u_\lambda \qquad (2)$$

$$\text{s.t. } u_\lambda \leq \min\{\rho, \ \tau \sum_{e \in \lambda} x_e f_e\}, \text{ for all } \lambda \in \Lambda \qquad (3)$$

$$\sum_{v \in V^+} x_{(v^+,v)} = \sum_{v \in V^-} x_{(v,v^-)} \leq \gamma \qquad (4)$$

$$\sum_{(v',v) \in E} x_{(v',v)} = \sum_{(v,v^\dagger) \in E} x_{(v,v^\dagger)}, \text{ for all } v \in V \quad (5)$$

$$\sum_{e \in E} l_e \cdot x_e \leq \gamma \cdot \kappa, 0 \leq x_e \leq \alpha, \forall e \in E \qquad (6)$$

Figure 4: Basic Formulation

$$\max_{\mathbf{x},\mathbf{y},\mathbf{u}} \sum_{\lambda \in \Lambda} p_\lambda u_\lambda - \beta \sum_{e \in \mathcal{E}} c_e y_e \qquad (7)$$

$$\text{s.t. } u_\lambda \leq \min\{\rho, \ \tau \sum_{e \in \lambda} x_e f_e\}, \text{ for all } \lambda \in \Lambda \qquad (8)$$

$$\sum_{v \in \mathcal{V}^+} y_{(v^+,v)} = \sum_{v \in \mathcal{V}^-} y_{(v,v^-)} \leq \gamma \qquad (9)$$

$$\sum_{(v',v) \in \mathcal{E}} y_{(v',v)} = \sum_{(v,v^\dagger) \in \mathcal{E}} y_{(v,v^\dagger)}, \text{ for all } v \in \mathcal{V} \quad (10)$$

$$x_e = \sum_{e' \in \Gamma(e)} y_{e'}, \forall e \in E, 0 \leq x_e \leq \alpha, \forall e \in E \quad (11)$$

Figure 5: Extended Formulation

patrols from the marginals we are able to compute the actual best-response utilities of the riders. Our experiments show that the differences between the actual utilities and the upper-bounds given by the LP formulation are small. The remaining task is to construct a γ-unit mixed strategy whose marginals match the marginal strategy $\mathbf{x}$.

Proposition 1. *Given a marginal strategy* $\mathbf{x}$*, a γ-unit mixed strategy for the leader that produces the same coverage on each edge e as* $\mathbf{x}$ *does can be constructed in polynomial time.*

Proof. First, we construct a set Υ of weighted patrol paths, by extracting distinct source-to-sink flows from $\mathbf{x}$ through the following iterative procedure.
1. Find a path P from v^+ to v^- where $x_e > 0$ for all $e \in P$. If no such path exists, terminate because x_e must then be 0 for all $e \in E$ (due to Constraint (5)). Otherwise go to step 2.
2. Let $x^* = \min_{e \in P}\{x_e\}$. Add path P with weight x^* to the set Υ. Deduct x^* from x_e for all $e \in P$. Go to step 1.

Since every iteration removes a complete source-to-sink flow, constraint (5) is maintained throughout the execution of this procedure. The procedure's running time is polynomial because at least one new x_e is set to 0 in each iteration.

Finally, we create a mixed strategy of joint patrol paths (with γ units) that matches exactly the set of weighted patrol paths Υ obtained in the procedure above, and thus the marginal strategy $\mathbf{x}$. To do this, we could assign a path of weight x^* to the γ units independently, each with an equal probability of $\frac{x^*}{\gamma}$. Since $x^* \leq \gamma$, we have $\frac{x^*}{\gamma} \leq 1$. □

Issues with the Basic Formulation

There are two fundamental issues with the basic formulation. First, the mixed strategy constructed can fail to satisfy the patrol length limit of κ, notwithstanding Constraint (6) on the sum of the lengths of all patrols, and hence be infeasible. In fact, the marginal strategy computed in the basic formulation may not correspond to any feasible mixed strategy in which all patrols have length at most κ. Consider the counterexample in Figure 6. Edges $v_1 \rightarrow v_2$ and $v_2 \rightarrow v_3$ represent two real atomic actions, each with duration 1. Patrols must start from either v_1 or v_3, but can terminate at any of v_1, v_2 and v_3. This is specified using

v^+ and v^-, the dummy source and sink respectively. We assume $\kappa = 1$ and $\gamma = 1$. It can be verified that the marginal strategy shown in Figure 6 satisfies constraints (4) through (6). However, the only corresponding mixed strategy is to take $v^+ \rightarrow v_3 \rightarrow v^-$ with 50% probability and $v^+ \rightarrow v_1 \rightarrow v_2 \rightarrow v_3 \rightarrow v^-$ with 50% probability. (To see this, observe that we cannot put any probability mass on the path $v^+ \rightarrow v_1 \rightarrow v_2 \rightarrow v^-$ because the edge $v_2 \rightarrow v^-$ has zero flow; the path $v^+ \rightarrow v_1 \rightarrow v^-$ is excluded by a similar argument.) This mixed strategy is infeasible since its second patrol has duration greater than 1. This patrol length violation arises because the basic formulation only constrains the average patrol length, and therefore allows the use of overlong patrols as long as some short patrols are also used.
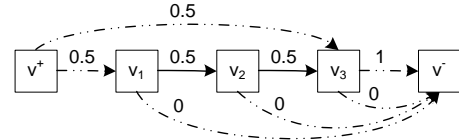

Figure 6: Example of an infeasible marginal strategy.

Second, the paths selected according the constructed mixed strategy may *switch* between trains or between instation and on-train an impractically large number of times, making the patrol path difficult to implement and error-prone. This is an important issue as we want real LASD officers to be able to carry out these strategies. The more switches there are in a patrol strategy, the more instructions the patrol unit has to remember, and the more likely they will miss a switch due to imperfections in the train schedule and/or the unit's mis-execution of the instructions. For example, in Example 1, $\langle A, 6pm \rangle \rightarrow \langle B, 7pm \rangle \rightarrow \langle A, 8pm \rangle$ and $\langle C, 6pm \rangle \rightarrow \langle B, 7pm \rangle \rightarrow \langle C, 8pm \rangle$ each do 1 switches while $\langle A, 6pm \rangle \rightarrow \langle B, 7pm \rangle \rightarrow \langle C, 8pm \rangle$ and $\langle C, 6pm \rangle \rightarrow \langle B, 7pm \rangle \rightarrow \langle A, 8pm \rangle$ each do none. Both path pairs cover the same set of edges, making the second preferable because it is easier to implement.

Extended Formulation

Now we present a more sophisticated formulation design to address the two aforementioned issues. The difficulty involved in imposing constraints on the patrol paths (i.e., penalizing or forbidding certain paths) in the marginal representation is that paths themselves are not represented, instead being encoded only as marginal coverage.

Hence the key idea is to preserve sufficient path history information within vertices to be able to evaluate our constraints, while avoiding the exponential blowup creating a node for every path would cause. We construct a new graph, called the *History-Duplicate Transition graph* (HDT graph), by creating multiple copies of the original vertices, each corresponding to different values of history information.

We first explain how to construct the HDT graph from a transition graph G in order to forbid patrol paths longer than κ. The HDT graph is composed of multiple restricted copies of G (i.e., subgraphs of G), corresponding to different possible starting time points. For the copy corresponding to starting time point t^*, we only keep the subgraph on vertices $v = \langle s, t \rangle \in V$ where $t^* \leq t \leq t^* + \kappa$. Thus, in each restricted copy of G, the length of any path is guaranteed to be less than or equal to κ. Since there are a finite number of distinct possible starting time points (i.e., all distinct discrete time points in V^+), the new graph is a linear expansion of G. An approximation can be obtained by taking one starting time point every δ time units. In this case, an original vertex (edge) will be kept in at most $\lceil \kappa/\delta \rceil$ copies, implying the new graph is at most $\lceil \kappa/\delta \rceil$ times larger than G.

Figure 7(a) shows the HDT graph (the shaded portion further explained below) of Example 1 with $\kappa = 2$ and 2 starting time points, $6pm$ and $7pm$. The HDT graph is thus composed of two restricted copies of the original transition graph. In each vertex, the time shown in parenthesis indicates the starting time point. For example, the original vertex $\langle A, 7pm \rangle$ now has two copies $\langle A, 7pm, (6pm) \rangle$ and $\langle A, 7pm, (7pm) \rangle$ in the HDT graph. For the starting time point of $6pm$, the patrol must end at or before $8pm$, hence we do not need to keep vertices whose discrete time point is $9pm$. For the starting time point of $7pm$, the patrol must start at or after $7pm$, hence we do not need to keep vertices whose discrete time point is $6pm$. The two restricted copies *are not two separate graphs* but a single graph that will be tied together by the dummy source and sink.

Next, we explain how to further extend the HDT graph to penalize complex patrol paths. The idea is to have each vertex encode the last action occurring prior to it. Specifically, we create multiple copies of a vertex v, each corresponding to a different edge that leads to it. If v is a possible starting vertex, we create an additional copy representing no prior action. If there is an edge from v to v', we connect all copies of v to the specific copy of v' whose last action was (v, v'). A new edge is called a switching edge if the recorded last actions of its two vertices are of different types (e.g., inspecting different trains), unless one of the two vertices is a "no prior action" vertex. As can be verified, the number of switches of a patrol path in the new graph is the number of switching edges it has. To favor simple patrol paths, we demand a cost $\beta > 0$ for using switching edges. Varying the value of β lets

us trade off between solution quality (greater revenue) and patrol preference (lower average number of switches).

In Figure 7(b), we show how to apply this extension using the subgraph shown in the shaded box of Figure 7(a). Since there is only one edge leading to $\langle A, 7pm, (6pm) \rangle$, we create one copy of it representing the action of staying at A. There are 3 edges leading to $\langle B, 7pm, (6pm) \rangle$, so we create 3 copies of it representing the actions of taking train from A, staying at B, and taking train from C. The original edges are also duplicated. For example, $\langle B, 7pm, (6pm) \rangle \rightarrow \langle B, 8pm, (6pm) \rangle$ has 3 copies connecting the 3 copies of $\langle B, 7pm, (6pm) \rangle$ to the copy of $\langle B, 8pm, (6pm) \rangle$, representing the staying at B action. Among the three copies, only the "Stay" to "Stay" edge is not a switching edge.
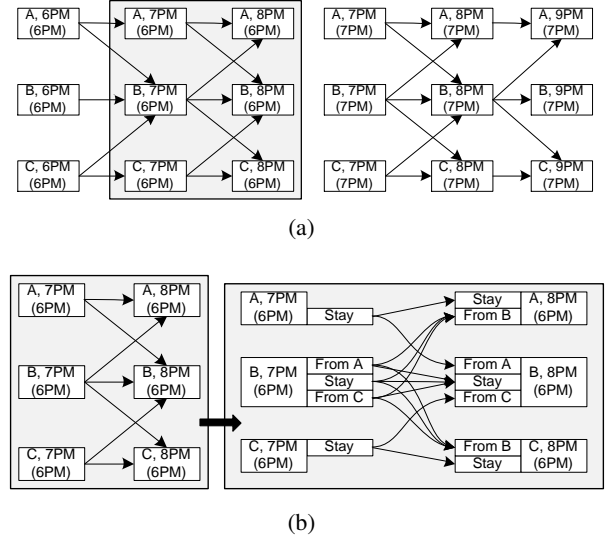

(a)

(b)

Figure 7: (a) HDT graph of Example 1 with two starting time points. (b) Extension storing the last action occurring.

Given the final HDT graph $\mathcal{G} = \langle \mathcal{V}, \mathcal{E} \rangle$, we provide an extended linear program formulation in Figure 5. We still use x_e to represent the marginal coverage of an original edge $e \in E$ being selected, but we now also use y_e to represent the marginal coverage of an HDT graph edge $e \in \mathcal{E}$ being selected. Let $\Gamma(e) \subset \mathcal{E}$ be the set of copies of e, then $x_e = \sum_{e' \in \Gamma(e)} y_{e'}$. Let $c_e = 1$ if $e \in \mathcal{E}$ is a switching edge and 0 otherwise. The set of possible starting vertices $\mathcal{V}^+$ is the set of copies of V^+ that are "no prior action" vertices. The set of possible ending vertices $\mathcal{V}^-$ is the set of all copies of V^-. We again add a dummy source v^+ leading to $\mathcal{V}^+$ and a dummy sink that can be reached from $\mathcal{V}^-$. Because the extended formulation enforces stricter restrictions on patrols allowed than the basic formulation, the LP of Figure 5, with β set to 0, provides a tighter upper bound on the optimal revenue than the LP of Figure 4.

A path in the HDT graph $\mathcal{G}$ trivially corresponds to a path in the transition graph G, since any edge in $\mathcal{G}$ is a duplicate of some edge in G. Therefore from the solution $\mathbf{y}^*$, we can use the same process described for the basic formulation to construct a mixed strategy. Because the length of any patrol path in the HDT graph is bounded by κ, the mixed strategy

must be feasible. In addition, since the number of switches in a patrol path equals the number of switching edges in it, the average number of switches of the constructed mixed strategy is always equal to $\sum_{e \in \mathcal{E}} c_e y_e^*$.

Real-World Evaluation

We present our evaluation based on real metro schedules and rider traffic data provided by the LASD. We solved the LP in the extended formulation using CPLEX 12.2 on a standard 2.8GHz machine with 4GB memory. We first describe the data sets we used, followed by our experimental results.

Data Sets

We created four data sets, each based on a different Los Angeles Metro Rail line: Red (including Purple), Blue, Gold, and Green. For each line, we created its transition graph using the corresponding timetable from `http://www.metro.net`. Implementing the LP requires a distribution of types of potential fare evaders (recall that a rider type corresponds to a 4-tuple of boarding station / time and disembarking station / time). We have access to hourly boarding and alighting counts provided by the Los Angeles Sheriff Department, and from this can estimate the distribution of population on the train system at each hour; but we do not know what percentage of the population are potential fare evaders. In our experiments, we assumed that potential fare evaders were evenly distributed among the general population. Specifically, suppose the percentage of riders boarding in hour i is d_i^+ and the percentage of riders alighting in hour i is d_i^-. Denote the set of rider types that board in hour i by Λ_i^+ and those that alight in hour i by Λ_i^-. Then we would like to compute a fine-grained ridership distribution $\mathbf{p}$ to match the hourly boarding and alighting percentages, i.e., to find a point within the following convex region Ω,

$$\Omega = \{\mathbf{p}|\mathbf{p} \succeq \mathbf{0} \wedge \sum_{\lambda \in \Lambda_i^+} p_\lambda = d_i^+ \wedge \sum_{\lambda \in \Lambda_i^-} p_\lambda = d_i^-, \forall i\}.$$

We estimate the fare evader distribution by finding the analytic center of Ω, i.e., $\mathbf{p}^* = \arg\min_{\mathbf{p} \in \Omega} \sum_{\lambda \in \Lambda} - \log(p_\lambda)$, which is efficiently computable.

The inspection effectiveness f_e of an edge is assigned based on the assumption that 10 passengers can be inspected per minute. f_e is capped at 0.5 to capture the fact that the inspector cannot switch between cars while the train is moving. (Trains contain at least two cars.) The ticket fare was set to $1.5 (the actual current value) while the fine was set to $100. (Fare evaders in Los Angeles can be fined $200, but they also may be issued warnings.) If we could increase the fine dramatically the riders would have much less incentive for fare evasion, and we could achieve better revenue. However a larger fine is infeasible legally. Table 1 summarizes the detailed statistics for the Metro lines.

Experimental Results

Throughout our experiments, we fixed γ to 1 (and thus $\alpha = 1$). In our first set of experiments, we fixed penalty β to 0 (no penalty for using patrol paths with more switches), and

Line	Stops	Trains	Daily Riders	Types
Red	16	433	149991.5	26033
Blue	22	287	76906.2	46630
Gold	19	280	30940.0	41910
Green	14	217	38442.6	19559

Table 1: Statistics of Los Angeles Metro lines.

varied the maximum number of hours that an inspector can patrol from 4 to 7 hours. To create the HDT graph, we took one starting time point every hour (i.e., $\delta = 1$ hour).

Figure 8(a) shows the expected revenue per rider of the mixed patrol strategy we generated, which is the total revenue divided by the number of daily riders. Since the LP only returns an upper bound of the attainable revenue, the true expected revenue of the mixed patrol strategy was computed by evaluating the riders' best responses for all rider types. A rider can always pay the ticket price for $1.5 and will only evade the ticket when the expected fine is lower. Hence the theoretical maximum achievable value is $1.5, which is achieved when every rider purchases a ticket. As we can see, the per-rider revenue increases as the number of patrol hours increases, almost converging to the theoretical upper bound of $1.5 for the Gold and Green line. Specifically, a 4-hour patrol strategy already provides reasonably good expected value: 1.31 for the Blue line (87.4% of the maximum), 1.45 for the Gold line (97.0%), 1.48 for the Green line (98.8%), and 1.22 for the Red line (81.3%). Among the four lines, the Red line has the lowest revenue per rider. This is because the effectiveness of fare inspection decreases as the volume of daily riders increases, and the Red line has significantly higher number of daily riders than the other lines.

We depict in Figure 8(b) the percentage of the true expected revenue vs. the theoretical upper bound returned by the LP. Strategies generated by our method are near optimal; for example, our 4-hour strategies for the Blue, Gold, Green, and Red lines provided expected revenues of 96.5%, 98.5%, 99.5%, and 97.0% of the upper bound (and thus at least as much of the optimum), respectively.

To study riders' responses to the computed strategy, we partitioned the entire population of riders into three groups depending on their expected fine if fare-evading: riders who prefer purchasing tickets (expected fine is greater than 1.7—13.3% above the ticket price), riders who prefer fare evasion (expected fine is less than 1.3—13.3% below the ticket price), and indifferent riders (expected fine is between 1.3 and 1.7). In Figure 8(c), we show the distribution of the three groups against the strategies computed for the Red line. The three dashed lines inside the region of indifferent riders represent, from top to bottom, the percentages of riders whose expected fine is less than 1.6, 1.5, and 1.4, respectively. As the number of patrol hours increases from 4 to 7, the percentage of riders who prefer fare evasion decreases from 38% to 7%, the percentage of riders who prefer purchasing tickets increases from 17% to 43%, and the percentage of indifferent riders remains stable between 45% and 50%.

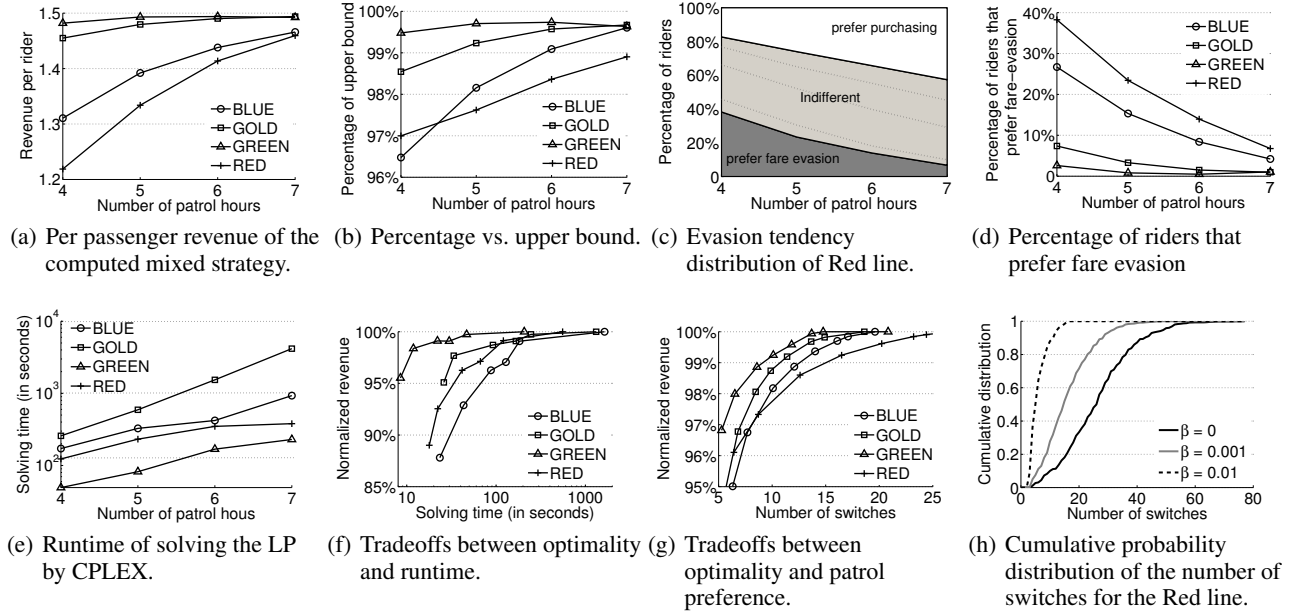Zooming in on the fare evasion, Figure 8(d) shows the

(a) Per passenger revenue of the computed mixed strategy.

(b) Percentage vs. upper bound.

(c) Evasion tendency distribution of Red line.

(d) Percentage of riders that prefer fare evasion

(e) Runtime of solving the LP by CPLEX.

(f) Tradeoffs between optimality and runtime.

(g) Tradeoffs between optimality and patrol preference.

(h) Cumulative probability distribution of the number of switches for the Red line.

Figure 8: Experimental results.

percentage of riders who preferred fare evasion against the patrol strategies computed. As we can see, this percentage decreased almost linearly in the number of additional patrol hours beyond 4. Our 7-hour patrol strategy lowered this percentage to 4.2% for the Blue line, 0.01% for the Gold line, 0.01% for the Green line, and 6.8% for the Red line. Again, due to having the highest daily volume, the Red line had the highest percentage of riders who preferred fare evasion.

Finally, Figure 8(e) shows the runtime required by CPLEX to solve the LPs we created. As we can see, the runtime increased as the number of patrol hours increased for all the metro lines. This is because the size of the HDT graph constructed is roughly proportional to the maximum length of the patrols, and a larger HDT graph requires an LP with more variables and constraints. Among the four lines, the Red and the Green lines have significantly fewer types, and are thus easier to solve than the other two lines.

In our second experiment, we varied the interval δ of taking starting time points, trading off solution quality for efficiency. We fixed the patrol length κ to 4 hours and penalty parameter β to 0. For each line, we tested 6 interval (δ) settings ranging from 0.5 hour to 4 hours. In Figure 8(f), the x-axis is the runtime (in log-scale) and the y-axis is the normalized revenue against the expected revenue of $\delta = 0.5$ within each line. For each line, a data point from left to right corresponds to $\delta = 4, 3, 2, 1.5, 1,$ and 0.5 respectively. Increasing the runtime (by decreasing δ) always led to a better solution; however, the quality gain diminished. For example, for the Blue line, it took 20 seconds of additional runtime to increase the solution quality from 87.9% ($\delta = 4$ hours) to 92.9% ($\delta = 3$ hours), whereas it took 1456 seconds of additional runtime to increase the solution quality from 99.1% ($\delta = 1$ hour) to 100% ($\delta = 0.5$ hour).

In the final experiment, we varied the penalty β, trading off between the solution quality and the average number of

switches. We fixed the patrol length κ to 4 hours and starting time interval δ to one hour. For each line, we tested 7 penalty settings from $\beta = 0$ to $\beta = 0.01$. Figure 8(g) plots the average number of switches against the normalized revenue against the expected revenue of $\beta = 0$ within each line. For all lines, higher β values led to both lower solution quality and fewer number of switches. For example, the average number of switches in the solution of the highest revenue ($\beta = 0$) ranged from 18.6 (Gold line) to 26.7 (Red line). However, by allowing 3% quality loss, this number could be lowered to less than 10 for all the four lines.

To further understand the patrol paths returned in these solutions, we show, in Figure 8(h), the cumulative probability distributions of the number of switches for the Red line given 3 settings of β: 0, 0.001, and 0.01. Choosing a lower β tended to lead to more complex patrol paths. For example, the solution of $\beta = 0$ used patrol paths whose number of switches is greater than 20 with 68.9% probability; the solution of $\beta = 0.001$ (99.7% of the optimum) only used such paths with 31.2% probability. And the solution of $\beta = 0.01$ (97.0% of the optimum) never used patrol paths that had more than 20 switches.

LASD Evaluation of TRUSTS

Since Januaray 2012, LASD has been testing our generated patrol strategies on the Red Line. For example, in initial test runs on Thursday, Jan 4 and Friday, Jan 5, 2012, one patrol unit conducted a 4-hour fare-inspection patrol on each day. A total of 851 fare checks were made, with 41 fare evaders cited and 3 felons arrested. The patrols implemented in the two days had 4 and 5 switches, respectively, and the officers were able to make the switches we requested. The LASD officers have given us valuable feedback and we have incorporated their suggestions, including inserting breaks in the middle of patrol shifts. Dozens of test patrol runs have been

Figure 9: Fare check at a station exit.

carried out so far, and more tests are scheduled in the future to provide a more thorough evaluation of the effectiveness of our strategies. Figure 9 shows two officers doing fare checks at a station exit during one of the test patrols.

Summary

In this paper we presented TRUSTS, a novel application for fare-evasion deterrence in urban transit systems. Our development of TRUSTS opens the door to applying game-theoretical randomization beyond previous applications of counterterrorism, to a much broader setting in which common individuals and daily routines are involved. We modeled the domain as a Stackelberg game, providing a novel compact representation of the leader's mixed strategies as flows in the history-duplicate transition graph. We found in our simulations that our method computed close-to-optimal strategies, which effectively deterred fare evasion and ensured high levels of revenue with few patrol hours. We are currently evaluating TRUSTS within the LA Metro system in collaboration with LASD.

We briefly mention a couple of future directions. One of the observations from the test runs is that LASD officers often need to deviate from the patrol schedule due to making felony arrests or being called to deal with emergencies at other parts of the train system. After dealing with the emergency, which could take a significant amount of time, it is often difficult to catch up to the original patrol schedule. We are currently investigating more efficient ways of utilizing resources in case of such emergencies. Another future direction is to investigate bounded rational models of the passengers, since they are human decision-makers and thus might not be perfect optimizers. Incorporating such behavior models of the adversary in security games (Pita et al. 2010; Yang et al. 2011) may potentially increase the robustness of our solutions. And we ultimately want to combine fare-checking with LASD's other tasks of crime prevention and counterterrorism. We plan to make use of multi-objective optimization techniques to intelligently allocate resources that makes trade-offs among these different objectives.

Acknowledgement

References

Alpern, S. 1992. Infiltration games on arbitrary graphs. *Journal of Mathematical Analysis and Applications* 163(1):286 – 288.

Becker, G., and Landes, W. 1974. *Essays in the Economics of Crime and Punishment*. Columbia University Pres.

Booz Allen Hamilton. 2007. Faregating analysis. Report commissioned by the LA Metro, `http://boardarchives.metro.net/Items/2007/11_November/20071115EMACItem27.pdf`.

Conitzer, V., and Sandholm, T. 2006. Computing the optimal strategy to commit to. In *EC*.

Dickerson, J. P.; Simari, G. I.; Subrahmanian, V. S.; and Kraus, S. 2010. A graph-theoretic approach to protect static and moving targets from adversaries. In *AAMAS*.

Gal, S. 1979. Search games with mobile and immobile hider. *SIAM Journal on Control and Optimization* 17(1):99–122.

Halvorson, E.; Conitzer, V.; and Parr, R. 2009. Multi-step multi-sensor hider-seeker games. In *IJCAI*.

Horizon Research Corporation. 2002. Metropolitan transit authority fare evasion study. `http://libraryarchives.metro.net/DPGTL/studies/2002_horizon_fare_evasion_study.pdf`.

Jain, M.; Tsai, J.; Pita, J.; Kiekintveld, C.; Rathi, S.; Tambe, M.; and Ordóñez, F. 2010. Software Assistants for Randomized Patrol Planning for the LAX Airport Police and the Federal Air Marshals Service. *Interfaces* 40:267–290.

Koller, D.; Megiddo, N.; and von Stengel, B. 1994. Fast algorithms for finding randomized strategies in game trees. In *STOC*.

Pita, J.; Jain, M.; Tambe, M.; Ordóñez, F.; and Kraus, S. 2010. Robust solutions to stackelberg games: Addressing bounded rationality and limited observations in human cognition. *AIJ* 174:1142–1171.

Pita, J.; Tambe, M.; Kiekintveld, C.; Cullen, S.; and Steigerwald, E. 2011. Guards - game theoretic security allocation on a national scale. In *AAMAS (Industry Track)*.

Ponssard, J., and Sorin, S. 1980. The L.P. formulation of finite zero-sum games with incomplete information. *Int. J. of Game Theory* 9:99–105.

Tambe, M. 2011. *Security and Game Theory: Algorithms, Deployed Systems, Lessons Learned*. Cambridge University Press.

Tsai, J.; Yin, Z.; Kwak, J.; Kempe, D.; Kiekintveld, C.; and Tambe, M. 2010. Urban security: Game-theoretic resource allocation in networked physical domains. In *AAAI*.

Wagenaar, W. A. 1972. Generation of random sequences by human subjects: A critical survey of literature. *Psychological Bulletin* 77(1):65–72.

Yang, R.; Kiekintveld, C.; Ordóñez, F.; Tambe, M.; and John, R. 2011. Improved computational models of human behavior in security games. In *AAMAS (Ext. Abstract)*.