# When Security Games Hit Traffic: Optimal Traffic Enforcement Under One Sided Uncertainty


Ariel Rosenfeld
Bar-Ilan University
arielros1@gmail.com

Oleg Maksimov
Bar-Ilan University
oleg@maksimov.co.il

Sarit Kraus
Bar-Ilan University
sarit@cs.biu.ac.il



## ABSTRACT

Efficient traffic enforcement is an essential, yet complex, component in preventing road accidents. In this paper, we present a novel model and an optimizing algorithm for mitigating some of the computational challenges of real-world traffic enforcement allocation in large road networks. Our approach allows for scalable, coupled and non-Markovian optimization of multiple police units and guarantees optimality. In an extensive empirical evaluation we show that our approach favorably compares to several baseline solutions achieving a significant speed-up, using both synthetic and real-world road networks.


## 1. INTRODUCTION

About 1.25 million people die worldwide each year as a result of road accidents and many more are injured [19]. An essential component in mitigating serious traffic accidents (accidents that cause death or injury) is efficient traffic enforcement, which is based on giving drivers the feeling that they are likely to be caught and sanctioned when breaking the law. Unfortunately, traffic police cannot cover the entire road network given its limited number of police cars and officers [6].

Within the Security Games (SG) field, optimal police allocation mechanisms for mitigating various types of crimes have been developed. The generic Security Games framework consists of a defender (traffic police) which has a limited number of resources to protect a large set of targets (road segments) from an adversary (reckless drivers) [17]. To the best of our knowledge, there is only a single work in the scope of SG which addresses traffic enforcement [2]. The authors model the problem as a Stackelberg Security Game (SSG) where traffic police seeks to apprehend reckless drivers who in turn seek to avoid apprehension. In the SSG, the traffic police commits to a mixed strategy that drivers are able to first observe and then best respond. In practice, traffic enforcement seeks to reduce traffic accidents (and not necessarily to apprehend reckless drivers) [11]. Furthermore, due to the dynamic environment factors which influence driving behavior (weather, holidays, road accidents, traffic jams, etc.), drivers have been shown to act in a less strategical manner, responding to changes in their environment, including the observed police presence in current and past rounds [9]. Therefore, SSGs seem unsuitable to the task of preventing serious road accidents.

Non-strategical adversaries in SG settings have been recently modeled as opportunistic criminals which choose where and when to commit a crime in real-time based on police presence and the attractiveness of the potential targets [22]. Opportunistic criminals are *reactive* to police actions and do not consider their behaviors' effect on future police actions. We adopt this approach in this study, modeling the drivers, and thereby accidents, as reactive to police allocation. However, unlike [22], drivers may react to the *both present and past police enforcement allocations*, making the authors' Markovian assumption ineligible. For example, it has been shown that drivers continue to react to police presence long after the enforcement operation has ceased [9]. Basilico et al. [1] have investigated non-Markovian strategies for robotic patrols. However, the authors assume that the attacker is strategic, and therefore the proposed approach is inapplicable. To the best of our knowledge, no work has efficiently addressed the non-Markovian property in SG.

Most allocation mechanisms in SG simplify the computational task by assuming that planning for each police unit *separately* will bring about a (near-)optimum solution [7]. However, this is not the case in traffic enforcement. For example, experts from the Israeli Traffic Police (ITP) claim that if police cars are stationed at the same place and time, their effectiveness in reducing traffic accidents cannot be assumed greater than the effectiveness of a single police car at the same point and time, *a fact we leverage in this work*. Furthermore, significant benefits may accrue from coordination across multiple police units, e.g., allocating two police cars in *adjacent* road segments could have a stronger impact than just allocating a single police car. This notion relates to the coordinated actions notion in [8] which captures the combined effects of multiple defenders guarding the same target simultaneously.

As a result of the above, the computational task of deriving optimal traffic enforcement allocations in order to prevent serious road accidents is both *coupled* and *non-Markovian*, which makes it computationally intractable. Namely, the optimal allocation of traffic enforcement at time $t$ could depend on the trajectories of *all police cars* (i.e., coupled) up to time $t$ (i.e., non-Markovian).

To address these shortcomings, in this paper, we first formulate the TRAFFIC ENFORCEMENT ALLOCATION PROBLEM (TEAP). We prove that deriving or approximating the optimal solution to a TEAP is hard and remedy this hardness by introducing an optimal novel algorithm called the RELAXED OPTIMIZATION SOLVER ENHANCER, or ROSE for short. ROSE uses a master/slave optimization approach, aimed at reducing the computational burden of directly solving the TEAP, and leverages common characteristics of TEAPs that have not been investigated in previous works. In an extensive empirical evaluation, we show that ROSE favorably compares to several baseline approaches, achieving a few orders of magnitude speed-up, using both synthetic and real-world road networks.

The TEAP formulation is based on the prediction of accident risk at different road segments and time as well as the effectiveness of varying police enforcement allocations. We model the former using a newly developed state-of-the-art accident prediction model, based on a large set of features (110) and 11 years of col-

Figure 1: Traffic Police Enforcement in Israel.

lected accident reports. Both models are available online at `http://www.biu-ai.com/trafficPolice` in order to encourage other researchers to tackle the important and challenging task of preventing serious traffic accidents.

Our proposed solution, based on the ROSE algorithm and our prediction model, is currently being implemented by the Israeli Traffic Police (ITP) in field trials.

# 2. TRAFFIC ENFORCEMENT ALLOCATION

We model the interaction between drivers and police as a repeated game over $T(<\infty)$ rounds, which takes place on a road network, represented as a graph $G=\langle V,E\rangle$ where $V=\{v\}$ is the set of intersections and $E=\{e=(u,v)\}$ is the set of road segments. We assume no accidents occur off-road, and therefore $E$ is the set of enforcement targets in this work[1]. Without losing generality, we assume that the time it takes to travel through each road segment is 1 round; this assumption can be relaxed by including dummy vertices.

The traffic police has $k(<<|E|)$ police cars at its disposal. At each round $t$, the police places enforcement on a subset of size $k$ from $E$, which we refer to as the allocation at round $t$ denoted $a_t$, such that the allocation respects the graph's connectivity constraints and no more than a single police car is assigned to any edge. Namely, at round $t$, each police car can either stay in its current road segment (enforcing for a longer period of time) or move to an adjacent edge given $a_{t-1}$. $a_1$ can assume any subset of size $k$ of $E$. We denote the traffic police *allocation history* at round $t$ as $H_t=\langle a_2,\ldots,a_t\rangle$. We use the notation $H[e_t]$ as an indicator of whether a police car is assigned to road $e$ at round $t$, $e_t$ for short. Simultaneously, drivers choose whether to obey the law (driver safely) or not at each road segment $e\in E$.

We assume that drivers' actions at round $t$ are visible to the police. For example, the ITP uses anonymous cellular reports provided by commercial companies to evaluate the distribution of speeds on each road in real time. Other technological aids such as speed and security cameras are also in use. On the other hand, drivers are only exposed to a noisy signal regarding the police allocation. For example, common applications such as WAZE and other technological instruments such as police scanners and radars allow drivers to have an indicator of police presence at $e_t$. However, these indicators are not completely accurate (police presence in a road segment is not instantaneously reported in WAZE, an indicator of police presence may not be up-to-date, a police car may be covert, etc.). As a result, the game is conducted *under one-sided uncertainty*. Due to this uncertainty, the drivers base their actions at $e_t$ according to $a_t$ (although not completely visible) and the defender's past allocations ($H_{t-1}$), which together constitute $H_t$.

[1]Intersection $v$ is considered part of the road segments that share $v$. Thus no need to consider $v$ as a different target.

| notation | meaning |
|---|---|
| $t\le T$ | Game round index. |
| $e_t$ | Road segment $e$ at round $t$. |
| $a_t$ | Defender's allocation in round $t$. |
| $H_t$ | Defender's allocation history at round $t$. |
| $H[e_t]$ | Indicator whether police is present at $e_t$. |
| $\texttt{risk}(e_t)$ | Likelihood of a car accident occurring at $e_t$ in the absence of police enforcement. |
| $\texttt{eff}(e_t,H_t)$ | The effectiveness of police enforcement on $e_t$. |

Table 1: Summary of key notations.

Following recent advancements in predictive policing, including the prediction model constructed in this paper (Section 4), and in the same spirit as done in previous works such as [16], we define the *risk of accidents* occurring at $e_t$, as $\texttt{risk}(e_t)$. The risk function measures how likely is a serious traffic accident likely to occur at $e_t$ *in the absence of police enforcement* (in the [0,1] range). We further define the *effectiveness of enforcement* as $\texttt{eff}(H_t,e_t)$. $\texttt{eff}$ measures the effect that the police allocation history has on the risk of accidents occurring at $e_t$.

The traffic police is interested in minimizing the total expected number of accidents occurring throughout the game. Formally, it seeks to minimize the following objective of the optimization problem we denote as the TRAFFIC ENFORCEMENT ALLOCATION PROBLEM (TEAP):

$$min_{H_T}\sum_{t=1,\ldots,T}\sum_{e\in E}\texttt{risk}(e_t)(1-\texttt{eff}(e_t,H_t)) \qquad (1)$$

$\texttt{risk}(e_t)$ cannot be influenced by police *enforcement* but rather through modification of the characteristics of the road (e.g., number of lanes), traffic (e.g., reducing speed-limit), etc. On the other hand, $\texttt{eff}$ heavily depends on police enforcement, $H_t$. Both $\texttt{risk}(e_t)$ and $\texttt{eff}(e_t,H_t)$ are known to the police and can be computed in polynomial time.

A summary of the notations used in this paper is available in Table 1.

## 2.1 Complexity Analysis

The solution to Eq. (1) prescribes a pure strategy for the traffic police. The police could optimize over all rounds simultaneously, however, this approach is computationally expensive; it needs to solve a possibly non-convex optimization problem as the police must consider drivers' responses (modeled within $\texttt{eff}$). Unfortunately, approximating the optimal solution to a TEAP, within any constant factor, is hard even for a single driver and a single police car.

**Theorem 1.** TEAP *cannot be approximated within a any factor of $c\ge 1$ in polynomial time, unless $P=NP$.*

PROOF. To prove the theorem we give a reduction from SAT to TEAP with one driver and one police car: On input $\Phi(x_1,\ldots,x_n)$, construct $n+1$ nodes $V=\{v_i\}$ $i=1,\ldots,n+1$. Then connect node $i$ with node $i+1$ ($i=1,\ldots,n$) using 2 directed edges, one for $x_i=True$ and one for $x_i=False$ and a single directed edges from $v_{n+1}$ to $v_1$ representing $Satisfiable$ (S). Consider the resulting graph $G = (V, E)$ as the road network for a TEAP with $T = n + 1$. A single police car starts at $v_0$. Let `risk` assume 0 for all edges at all rounds except for edge $S$ at round $t + 1$, which assumes the value of 1. Let `eff` assume 0 for all edges, rounds and allocation histories except for $\texttt{eff}(S, H_t)$, which assumes the value of 1 if the police trajectory ($H_t$) corresponds to a satisfying assignment for $\Phi(x_1, \ldots, x_n)$ and 0 otherwise. Clearly, the driver's action (causing an accident at edge S or not) can be decided in polynomial time.

The above construction takes polynomial time. Assume to the contrary that such an approximation polynomial time algorithm $App(G)$ exists. If there is no satisfying assignment to $\Phi$, then every trajectory the police car may take will bring about an objective value of 1, thus $App(G) \geq c$. If there is a satisfying assignment, then the defender can take the respective trajectory and receive a value of 0, hence $App(G) = 0$.

Two key computational challenges arise from the TEAP formulation. First, the arbitrary `risk` and `eff` functions, which can take any polynomial time computable form and depend on an unbounded history of police actions (`eff`), pose a significant optimization challenge. Second, the space of possible police strategies (joint schedules for all police cars) grows exponential in the number of resources and the number of time steps which make the computation even more challenging.

# 3. OPTIMIZING POLICE STRATEGY

In this work we derive optimal *pure strategy* for traffic enforcement for $T$ steps. Our goal is to find the pure strategy that would minimize the total expected number of serious accidents. In our framing, any randomized mixed-strategy, which is the combination of pure strategies, results in a greater number of crimes than the optimal pure strategy as in [22].

Given Theorem 1, we resort to remedying the hardness of solving the TEAP by introducing an optimal novel algorithm called the RELAXED OPTIMIZATION SOLVER ENHANCER, or ROSE for short. ROSE uses a master/slave optimization approach, aimed at reducing the computational burden of directly solving the TEAP. It exploits the fact that no two police cars are allowed to enforce the same road segment at the same time. ROSE is guaranteed to return an *optimal solution*, hence, in the worst case, ROSE will run in exponential time. Nevertheless, experimental results (see Section 5) on road networks of varying characteristics show that ROSE is able to derive an *optimal solution* significantly faster than competing approaches.

Before introducing ROSE, we first cast the TEAP as a graph flow problem and present an exponential sized Binary Integer Program (BIP) for solving it.

## 3.1 TEAP as Graph Flow

We model the TEAP using a *transition graph* [21]. The transition graph is a compact representation which captures the spatio-temporal structure of the road network and allows us to handle the exponential strategy space by avoiding the enumeration of all pure strategies. Technically, given a road network $G$, we transform it into a $T$ time-expanded graph $G_T$ such that each vertex $v$ (edge $e$) is replicated $T$ times, one for each round, denoted $v_t$ ($e_t$).

Each $v_t$ in the transition graph is associated with the number of police cars that start their trajectories in it minus the number of police cars that end their trajectory in it, denoted $b_{v_t}$. $b_{v_t}$ is assumed to be known in advance and cannot be changed by the police.[2] The resulting flow problem can be formulated as the following mathematical program:

$$\min_{H_T} \sum_t \sum_{e_t} \texttt{risk}(e_t) \cdot (1 - \texttt{eff}(e_t, H_t)) \tag{2}$$

$$\text{s.t} \sum_{v'_{t-1}} H_t[(v'_{t-1}, v_t)_{t-1}] - \sum_{v'_{t+1}} H_{t+1}[(v_t, v'_{t+1})_{t+1}] = b_{v_t} \quad \forall v_t \in G_T \tag{3}$$

$$H_T[e_t] \in \{0, 1\} \quad \forall e, t \tag{4}$$

Constraints (3) and (4) are standard binary flow constraints. Let $Sol = \{e_t | H_T[e_t] = 1\}$ denote the set of $e_t$s that were assigned a unit of flow (a police car) in the optimal assignment.

We transform the above mathematical program into a 0-1 integer linear program (or Binary Integer Problem, BIP for short), of exponential size, using the following procedure: $\texttt{risk}(e_t)$ and $\texttt{eff}(e_t, H_t)$ are enumerable; for every $e_t$ and possible $H_T$ (which is bounded in size by $2^{|V||E||T|}$) one can *conceptually* calculate the value of $\texttt{risk}(e_t) \cdot (1 - \texttt{eff}(e_t, H_t))$ offline and store it in a table. For every entry $i$ in the table, which assumes a possible allocation history $H_t^i$, we denote $Visit_i = \{e_t | H_t^i[e_t] = 1\}$ as the set of $e_t$s that assumed the value of 1 under $H_t^i$. Let $Value_i$ denote the value of $\texttt{risk}(e_t) \cdot (1 - \texttt{eff}(e_t, H_t))$ for entry $i$. For every entry $i$ we create a new binary variable $p_i$ which takes the value of 1 if $Sol \cap Visited_i = Visited_i$. To that end, we add the constraint:

$$p_i = \prod_{e_t \in \{Visited_i\}} H_T[e_t] \tag{5}$$

Equation (5) might seem non-linear at first. However, it is rather easy to linearize it using a fix-sized set of linear constraints that will force the indicator $p_i$ to assume the correct value[3].

Let $Pow(i)$ be the set of all *strict* (proper) subsets of $Visited_i$. We then modify the optimization objective (2) using the inclusion-exclusion principle:

$$min_{H_T} \sum_t \sum_{e_t} \sum_i p_i (Value_i + \sum_{Visited_j \in Pow(i)} (-1)^{|Visited_j \cap Visited_i| + 1} Value_j) \tag{6}$$

Intuitively, for a given $e_t$ and $i$, we shall refer to the summed term as *penalty* if the summed term is *positive*, and *reward* otherwise.

Clearly, the result is a BIP. Furthermore, the resulting BIP is *insensitive to the number of police cars*. The correctness of the above procedure follows that of the inclusion-exclusion principle. In order to understand the procedure better, consider the following example:

**Example 1.** *Assume a time-expanded graph with 2 vertices $(u, v)$ expanded over 3 time steps $(v_1, u_1, v_2, u_2, v_3, u_3)$ such that $v_1$ and $u_1$ are connected to $v_2$ and $u_2$, and $v_2$ and $u_2$ are connected to $v_3$ and $u_3$. There are 2 guards, starting at nodes $v_1$ and $u_1$, and they finish their trajectories at $v_3$ and $u_3$. Overall, the problem induces 8 binary decision variables, written for short as $I_{v_1,v_2}, I_{v_1,u_2}, I_{u_1,v_2}$, $I_{u_1,u_2}$, etc.* `risk` *is set to 1 for all edges.* `eff` *is set to 1 for all edges and strategies except for $(v_1, v_2)$ which is set to 0.6 if $I_{v_1,v_2} = 1$, 0.8 if $I_{u_1,v_2} = 1$ and to 0.5 if both $I_{v_1,v_2} = 1$*

[2]This formulation allows police cars to start and finish their paths at different times and locations.

[3]A short explanation of the procedure is available at `http://www.leandro-coelho.com/linearization-product-variables/`

*and $I_{u_1,v_2} = 1$. We introduce three new variables $p_1, p_2$ and $p_3$, and three new constraints: $p_1 = I_{u_1,v_2}$, $p_2 = I_{u_1,v_2}$ and $p_3 = I_{u_1,v_2} \cdot I_{u_1,v_2}$. Thus, the optimization objective is:* $\min_I 8 + (0.6-1)p_1 + (0.8-1)p_2 + (0.5-(0.6+0.8)+1)p_3$. *Note that the terms associated with $p_1$ and $p_2$ are* ***rewards*** *(they intuitively help the optimizer lower the objective) and the term associated with $p_3$ is a* ***penalty*** *(it obstructs the optimizer from lowering the objective).*

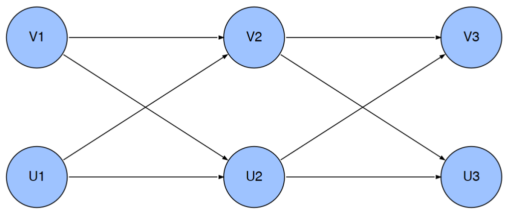



Figure 2: Time-Expanded Graph (Example 1)

## 3.2 Linear Optimization Using ROSE

The resulting BIP of the procedure above cannot scale up due to the exponential number of variables and constraints (see empirical evaluation in Section 5). To overcome this limitation we introduce a novel master/slave-based optimization algorithm, ROSE. The Master program consists of two-levels: At the high-level the Master program maintains a subset of penalty terms, denoted $P$. At the low-level a BIP solver is used to solve a **relaxed BIP** in which only a subset of penalty terms are introduced along with their associated binary variables, $p_i$. At the beginning of the execution, $P$ contains all penalty terms of Eq. (6) and the low-level solver generates a solution, $Sol$, while contemplating only $p_i$ variables associated with reward terms. Given $Sol$, the Slave program is executed to examine whether any penalty term $p \in P$ is *triggered*, that is, the Slave program checks whether any binary variable $p_i$ associated with a penalty term in $P$ should assume the value of 1 given $Sol$. If no penalty terms from $P$ are triggered, the Slave returns an empty set, indicating that an optimal solution has been found and ROSE terminates; otherwise, a set of penalty terms $P' \subseteq P$ *containing at least one triggered penalty from* $P$ is returned. The returned $P'$ is injected into the relaxed BIP and removed from $P$ by the Master.

**Algorithm 1** ROSE

**Require:** Time-expanded graph $G_T$, BIP Solver $Solver$.
**function** MASTER
  $BIP \leftarrow$ Initialize BIP with reward terms
  $P \leftarrow$ Penalty terms
  $Sol \leftarrow \emptyset$
  **repeat**
    $Sol \leftarrow Solver(BIP)$
    $P' \leftarrow Slave(Sol, P)$
    **if** $P' = \emptyset$ **then**
      **return** $Sol$
    $P = P \setminus P'$
    Introduce $P'$ into $BIP$
**function** SLAVE(Sol,P)
  $P' = \{p | p \in P \wedge$ p is triggered by Sol$\}$
  **return** $P'$

The Slave program can return any subset $P' \subseteq P$ as long as it obeys the following two rules: 1) $P' = \emptyset$ if no penalty terms from $P$ are triggered under $Sol$, 2) $P'$ contains at least one triggered penalty term (if such exists). We use an elementary implementation of the Slave program, returning all triggered penalty terms from $P$. The investigation of more elaborate Slave programs which *predict* which penalty terms are most beneficial to introduce, in terms of minimizing ROSE 's run-time, is left for future work.

**Proposition 2.** ROSE *always terminates and returns an optimal solution.*

PROOF. The Slave program introduces at least one penalty term to the relaxed BIP at each non-terminal iteration. Due to the finite number of penalty terms, ROSE terminates after a finite number of steps. At each iteration, the value of each feasible solution cannot decrease as ROSE only introduces penalty terms to the objective function. When ROSE terminates, all penalty terms triggered by $Sol$ have been injected into the relaxed BIP, therefore, the relaxed BIP's objective value under $Sol$ is the optimal value under both the relaxed BIP and the original BIP, and its objective value would not change if any additional penalties from $P$ were to be added to the objective function.

BIP solvers are sensitive to the number of constraints. Therefore, ROSE's computational performance depends on the number of penalty terms in $P$ which can be avoided in the iterative penalty generation process. Namely, the choice of Slave function has a significant effect on computation time. Similar to other iterative methods such as cutting plane and column generation (see [5]), it is hard to guarantee the computational benefit of the approach in the general case. While ROSE may be inefficient in some cases (e.g., no penalty term can be avoided regardless of Slave implementation), in several settings, including realistic and real-world traffic enforcement settings, it can bring about a significant improvement in runtime. For example, as discussed in [8], in many security settings, `eff` is *submodular*. Namely,

**Definition 3.** `eff` *is submodular if for every $e_t, e'_t$ and $H_t \subseteq H'_t$,* $\texttt{eff}(e_t, H_t \cup \{e'_t\}) - \texttt{eff}(e_t, H_t) \geq \texttt{eff}(e_t, H'_t \cup \{e'_t\}) - \texttt{eff}(e_t, H'_t)$

A submodular `eff` means that performing an additional enforcement activity (allocating a police car at $e'_t$) has diminishing gains in effectiveness. In Section 5 we show the significant runtime benefits that can be generated by ROSE when this property holds.

# 4. REALISTIC ENFORCEMENT ENVIRONMENT

For reproducibility purposes and to allow future research on traffic enforcement, we establish a realistic simulation environment which we name *SECURE*. SECURE consists of 3 components: 1) Several real-world road networks; 2) State-of-the-art prediction model for modeling `risk`; and 3) A submodular `eff` function. `risk` and `eff` are derived from 11 years of accident data, extensive literature review on accident prevention and analysis and human expert knowledge from the ITP. In this section we briefly describe the details of SECURE. For complete details and source code see `http://www.biu-ai.com/trafficPolice`.

## 4.1 `risk`

We obtained a record of 11 years of accident reports from Israeli Central Bureau of Statistics (2005-2015). By cross referencing these reports with additional sources such as GIS database and weather reports, we were able to characterize each accident with 110 features including infrastructure characteristics (e.g., number of lanes), date and time characteristics (e.g., weekend-weekday), weather (e.g., precipitation), traffic (e.g., average speed), etc. To the best of our knowledge, this is the largest set of features ever to be used to predict serious car accidents. For comparison, the Indiana traffic police use an intelligent accident prediction tool[4] which is based on approximately 90 features which we also use here. Experts in traffic enforcement claim that only the Indiana and Tennessee State traffic police use accident prediction tools but we were only able to obtain the former's features. Using more than 30,000 accident records and undersampling the "non-accident" class (see [4]), we trained a deep neural network model that, given 110 features representing $e_t$, returns a value in the [0,1] range, *acting as a proxy* to the likelihood of an accident occurring at $e_t$.[5]. We compared our prediction model to several baseline prediction models such as logistic regression, SVM and XGBoost (which is currently in use by the Indiana traffic police). Our model achieves a AUC of 0.87, which outperforms logistic regression, SVM and XGBoost that recorded 0.78, 0.77 and 0.82, respectively.

### 4.2 `eff`

We base `eff` on [18], which used a unique database to track the exact location of the Dallas Police Department's patrol cars throughout 2009 and crossed it with the car accidents of that year. To the best of our knowledge, this is the most recent investigation of the topic. The author found that if $e_t$ is enforced, `eff` should assume a value of 36%. However, enforcement effects are not restricted to the specific time and space in which the enforcement is performed. For example, *Time halo* is the time and the intensity to which the effects of enforcement on drivers' behavior continue after the enforcement operations have been concluded. It has been recorded that longer enforcement efforts cause more intense time halo effects that can last for hours and influence the next day(s) or even week(s) during the same time of day as the enforcement. *Distance halo* is defined as the distance over which the effects of an enforcement operation last after a driver passes the enforcement site. The most frequent distance halo effects are in the range of 1.5 - 3.5 kilometers from the enforcement site (see [9] for a review). In accordance with the ITP's estimations, we define time halo effects in the exponential diminishing form $\frac{36}{2^k}\%$ where $k \geq 0$ is the time-steps that have passed from the enforcement effort. To avoid negligible effects, we prune the effect at $k = 3$. The Distance halo effect is defined to be 5%, given that the two road segments are adjutant. Given the police allocation, `eff` takes a simple submodular form where `eff` takes the largest applicable effect and adds half of each of the smaller appropriate effects to it. For example, if both $e_t$ are $e_{t+1}$ are enforced (and no other time or distance halo effects are appropriate), `eff` assumes 45% $(= 36\% + \frac{18}{2}\%)$.

We are currently investigating a more data-driven approach for modeling `eff` in Israel. The above instantiation is used in our evaluation in Section 5.

## 5. EVALUATION

We evaluate ROSE, Algorithm 1, on synthetic road networks and the Israeli road network which are available on SECURE.

ROSE is compared with 4 baseline solutions: First, a **Naïve** solver which solves the entire BIP (Eq. (6)) in its general form. Second, a **Random** solver which for each police unit selects an action at random at each time step, resolving conflicts locally. Third, a **Greedy** solver, which computes a greedy path for each individual police car *iteratively* capturing a (wrongly) assumed adaptivity in individual police car gains. Greedy considers a simplified version of `eff` which only accounts for the marginal gains that an enforcement in a road segment will generate given the current allocation of other police cars. Given the calculated path, Greedy updates the simplified `eff` given the visited road segments and continues to the next police car. Last, we compare ROSE with **Domain Expert** allocations from the ITP. We could not evaluate Cartesian product solutions, which capture the joint effects of all police units, such as the ones presented in [20, 22], due to their lack of scalability in the number of road segments (we were unable to solve road networks larger than 5 road segments, which are unrealistic).

The resulting allocations are evaluated on the basis of two criteria: 1) **Quality**, the reduction in the objective value (Eq. (6)) between the no police enforcement condition and the provided solution; 2) **Runtime and Scalability** of the deployed algorithm with respect to in the number of police cars, road segments and the density of the road network.

The evaluation was done on a personal computer with 16 GB RAM and a CPU with 4 cores each operating at 4 GHz and the BIP solver of choice was GUROBI [13].

### 5.1 Synthetic Road Networks

We evaluate ROSE, Naïve, Random and Greedy on a series of synthetic road networks. We used 2 sets of synthetic road networks: Small networks (each consists of between 40 and 100 road segments in intervals of 10) and realistic networks (each consists of between 200 and 400 road segments in intervals of 100). Connectivity between road segments (i.e., the network density) is randomized such that each two road segments are connected by an intersection with a probability ranging between 0.05 and 0.15 (in intervals of 0.05), allowing for different topologies. `risk` uniformly samples a value in the $[0, 1]$ interval for each road segment and round and `eff` is defined as in the SECURE simulation (Section 4). The number of police cars is set to either 5, 10 or 15 and $T$ is set for either 8, 16 or 24. Overall, 270 networks were evaluated. A 30 minute timeout was set for all conditions and networks.

#### *5.1.1 Quality*

As expected, ROSE and Naïve return optimal allocations. On average, both reduce 22.7% and 5.3% of the no-enforcement objective value in small and realistic networks, respectively. On the other hand, on average, Random and Greedy reduce 1% (in small networks) of the no-enforcement objective value in all realistic networks. In realistic networks Greedy exceeded the timeout for all networks of size 300 and 400 and thus its quality cannot be evaluated properly. In our trials, Random and Greedy did not come up with an optimal allocation in any of the cases. Figures 3a and 3e present the results.

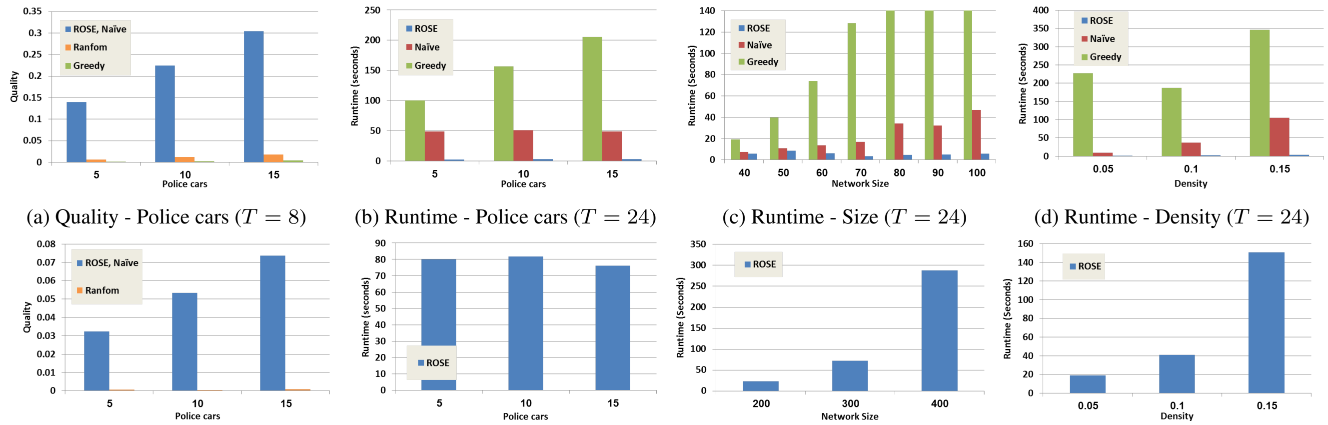
#### *5.1.2 Runtime and Scalability*

We begin by analyzing the non-optimal algorithms, aimed at reducing runtime. Random takes negligible time under all settings ($<$ 3 seconds). Greedy is linear in the number of police cars (it iteratively solves the problem for each police car separately) but exponential in the size of the network. For example, for a network of size 100 with density of 0.1, 10 police cars and $T = 16$, ROSE takes exactly 1 second to derive an optimal solution while Greedy takes 289 seconds, and produces a suboptimal solution. Greedy reached the timeout for all realistic networks.

Analyzing the Naïve and ROSE conditions head-to-head provides interesting insights. First, in *all* tested networks, ROSE performed faster than Naïve . On average, for small networks, ROSE

[4]`https://www.in.gov/isp/ispCrashApp/main.html`

[5]Serious accidents are sporadic events in both time and space. Therefore, directly estimating the probability of accidents occurring at $e_t$ is extremely challenging.




(a) Quality - Police cars ($T = 8$) (b) Runtime - Police cars ($T = 24$) (c) Runtime - Size ($T = 24$) (d) Runtime - Density ($T = 24$)

(e) Quality - Police cars ($T = 8$) (f) Runtime - Police cars ($T = 24$) (g) Runtime - Size ($T = 24$) (h) Runtime - Density ($T = 24$)

Figure 3: Synthetic road networks, results for the small networks set are depicted in Figures (a) to (d), and results for the realistic networks are depicted in Figures (e)-(h). Note that for the realistic networks, Naïve and Greedy exceeded the timeout and thus do not appear in Figures (e)-(h).

requires only 19% of the runtime needed by Naïve . We were able to manually engineer circumstances in which Naïve outperforms ROSE, mainly in very small networks (size $< 40$) or in networks with high number of police cars ($> 25$). The runtime difference increases significantly in the network's size and density but *slightly* decreases in the number of police cars and network's density. Similar to Greedy, Naïve was unable to solve most networks of size 200 and all networks of size 300 (and above) in 30 minutes time. See Figures 3b,3c, 3d, 3f, 3g and 3h.

## 5.2 Real-World Road Network

We evaluate ROSE using the Israeli road network. Unlike for synthetic networks, for the Israeli road network we used the `risk` prediction model available at SECURE. $T$ was set to $8, 16$ and $24$ and the number of police cars varied between 5 and 30 (in intervals of 5) per the Israeli police abilities (36 runs in total). We also evaluate a Domain Expert condition in which we asked an experienced Israeli traffic police superintendent who specialized in traffic enforcement to provide an allocation.

The Israeli road network is much larger than the synthetic networks analyzed previously, consisting of 715 road segments, but with a very low density (on average, each intersection connects between 3 and 4 road segments). Therefore, the results display slightly different pasterns. The results show that both in terms of quality and runtime ROSE outperforms the Naïve, Greedy and Domain Expert conditions by a large margin. Naïve achieves the same solution quality as ROSE (5.5%) but requires up to **6** times longer runtime. For example, under $T = 16$ and 10 police cars ROSE requires only 45 seconds compared to almost 4 minutes required by Naïve. Greedy and Random produced extremely poor solutions across the conditions, averaging less then 0.5%. Greedy required a significantly longer runtime than ROSE and reached our timeout of 30 minutes in most cases. As expected, Random required negligible runtime under all settings ($< 2$ seconds). See Table 2 for the results.

The Domain Expert produced allocation where police cars were allocated permanently at the most risky road segments. The quality of the proposed allocation was similar to that of Greedy ( 0.5%).

# 6. DISCUSSION

When presenting a new formulation, such as the TEAP, and new solutions such as ROSE, it is worth discussing limitations.

ROSE allows us to optimally solve large TEAPs with significant runtime improvement compared to baseline approaches. This improvement is most significant for large, dense networks. However, ROSE's runtime is slightly impaired with the increase in the number of police units. The reason is simple, with the increase in the number of police cars, penalty terms are more likely to be triggered by feasible solutions. Therefore, in a "congested" TEAP (i.e., small network with many police cars), ROSE could be counterproductive. In our experiments, such road networks are solved very quickly by both ROSE and Naive.

The TEAP solution is a pure strategy for the police, which makes *predictability* an issue. While the TEAP assumes that the drivers are reactive to police presence and essentially do not learn the police's actual policy, this assumption may lead to repetitive police allocations which drivers may (eventually) understand and anticipate. A practical solution to this concern is to (periodically) define additional allocation constraints that impose or restrict the enforcement of a specific road segment, similar to the entropy-based approach suggested in [2]. Today, police forces occasionally define road segments that must or must not be visited during a shift due to special enforcement needs (e.g., road work). The injection of these constraints in the TEAP formulation is straightforward. An automatic process may randomly select which road segments must/must-not be enforced in a given allocation such that every road segment has at least a user-defined $\epsilon$ probability of being enforced at every time step.

When a police car is delayed, it might become impossible to complete the patrol schedule. An efficient way to resolve this issue is for the central command to allocate the police cars assuming perfect execution and only after a non-default transition occurs does the central command resolves the TEAP starting from the current state [7]. Given the positive runtime results of ROSE, such reallocation should not pose a significant computational concern.

# 7. RELATED WORK

It has been established that a significant reduction in the occurrence of serious traffic accidents can be achieved by efficient traffic police allocation [10]. Therefore, recently, traffic police forces

| Police Cars | T=8 | | | | T=16 | | | | T=24 | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | ROSE | Naïve | Greedy | Random | ROSE | Naïve | Greedy | Random | ROSE | Naïve | Greedy | Random |
| 5 | 5 | 33 | 277 | 0 | 31 | 153 | 1065 | 0 | 58 | 352 | N/A | 0 |
| 10 | 7 | 36 | 558 | 0 | 45 | 191 | N/A | 0 | 212 | 402 | N/A | 0 |
| 15 | 11 | 36 | 837 | 0 | 219 | 301 | N/A | 0 | 384 | 875 | N/A | 1 |
| 20 | 12 | 40 | 1109 | 0 | 119 | 263 | N/A | 0 | 471 | 695 | N/A | 1 |
| 25 | 21 | 53 | 1340 | 0 | 394 | 487 | N/A | 0 | 1432 | 1520 | N/A | 1 |
| 30 | 36 | 40 | 1350 | 0 | 479 | 520 | N/A | 0 | N/A | N/A | N/A | 1 |

Table 2: Runtime of ROSE, Naïve, Greedy and Random for the Israeli road network under varying numbers of police cars and $T$. Runtime is measured in seconds. N/A means that a timeout of 30 minutes was reached.

have begun implementing the predictive policing paradigm [15] through which police officers can identify people and locations at increased risk. From a methodological standpoint, the effort of predicting traffic accidents has mainly focused on aggregative analysis, specifically, on the prediction of the *annual* number of serious accidents per road segment using statistical methods such as Poisson or negative binomial regression models [3]. Such aggregation is limited in its use to police forces as the allocation of traffic police enforcement requires a prediction on a much more finely-grained level (say, an hourly basis). To the best of our knowledge, the state-of-the-art prediction models provide prediction for three hour time-frames. Overall, despite its promise and successful implementation, predictive policing does not provide police officers with a means to derive optimal enforcement allocations. In this study, we were able to construct a prediction model that provides beneficial predictions for *one hour time-frames* by using a unique set of features and 11 years of collected data.

The Gambler's Fallacy is the phenomenon where people tend to put ample amount of weight on previous events, believing that they influence future outcomes. This phenomenon manifests itself in the context of traffic patrol in the form of *halo effects*. For over 4 decades traffic halo effects have been validated repeatedly – showing that enforcement effects are not restricted to the specific time and space in which the enforcement is performed. Two such effects are called time-halo and distance-halo [9]. To our knowledge, this is the first work to formulate and integrate halo effects in enforcement optimization. Existing works on modeling human behavior in SG settings such as [14, 12], consider the adaptive nature of human behavior to successes and failures in past rounds. However, the integration of halo effects in such models is not straightforward.

# 8. CONCLUSIONS

This paper introduces a novel framework for designing traffic police allocation in realistic settings. First, we model the interaction between drivers and traffic police as a Traffic Allocation Enforcement Problem (TEAP) and prove that accurately solving or approximating the optimal solution of a TEAP is hard. Next, we cast the TEAP as a binary graph flow problem, which in turn is translated into a unique binary optimization problem, that we show how to solve efficiently and optimally by a new algorithm called the RELAXED OPTIMIZATION SOLVER ENHANCER, ROSE. Extensive empirical evaluation, with real and synthetic road networks, demonstrates the benefits of our approach.

We hope that this study will encourage other researchers to tackle the important and challenging task of preventing serious traffic accidents. To assist others with this challenge, we also provide a realistic simulation environment, which we name *SECURE*, that includes a state-of-the-art accident prediction model along with useful road networks and data.

# REFERENCES


[1] N. Basilico, N. Gatti, and F. Amigoni. Leader-follower strategies for robotic patrolling in environments with arbitrary topologies. In *Proceedings of The 8th International Conference on Autonomous Agents and Multiagent Systems-Volume 1*, pages 57–64. International Foundation for Autonomous Agents and Multiagent Systems, 2009.

[2] M. Brown, S. Saisubramanian, P. R. Varakantham, and M. Tambe. Streets: game-theoretic traffic patrolling with exploration and exploitation. 2014.

[3] L.-Y. Chang. Analysis of freeway accident frequencies: negative binomial regression versus artificial neural network. *Safety science*, 43(8):541–557, 2005.

[4] N. V. Chawla. Data mining for imbalanced datasets: An overview. In *Data mining and knowledge discovery handbook*, pages 853–867. Springer, 2005.

[5] D.-S. Chen, R. G. Batson, and Y. Dang. *Applied integer programming: modeling and solution*. John Wiley & Sons, 2011.

[6] G. DeAngelo and B. Hansen. Life and death in the fast lane: Police enforcement and traffic fatalities. *American Economic Journal: Economic Policy*, 6(2):231–257, 2014.

[7] F. M. Delle Fave, A. X. Jiang, Z. Yin, C. Zhang, M. Tambe, S. Kraus, and J. P. Sullivan. Game-theoretic patrolling with dynamic execution uncertainty and a case study on a real transit system. *Journal of Artificial Intelligence Research*, 50:321–367, 2014.

[8] F. M. Delle Fave, E. Shieh, M. Jain, A. X. Jiang, H. Rosoff, M. Tambe, and J. P. Sullivan. Efficient solutions for joint activity based security games: fast algorithms, results and a field experiment on a transit system. *Autonomous Agents and Multi-Agent Systems*, 29(5):787–820, 2015.

[9] M. Elliott, J. Broughton, et al. *How methods and levels of policing affect road casualty rates*. Transport Research Laboratory, 2005.

[10] R. Elvik, T. Vaa, A. Erke, and M. Sorensen. *The handbook of road safety measures*. Emerald Group Publishing, 2009.

[11] European Transport Safety Council. How traffic law enforcement can contribute to safer roads: Pin flash report 31. Technical report, June 2016.

[12] F. Fang, P. Stone, and M. Tambe. When security games go green: Designing defender strategies to prevent poaching and illegal fishing. In *International Joint Conference on Artificial Intelligence (IJCAI)*, 2015.

[13] I. Gurobi Optimization. Gurobi optimizer reference manual, 2016.

[14] D. Kar, F. Fang, F. Delle Fave, N. Sintov, and M. Tambe. A game of thrones: when human behavior models compete in

repeated stackelberg security games. In *Proceedings of the 2015 International Conference on Autonomous Agents and Multiagent Systems*, pages 1381–1390. International Foundation for Autonomous Agents and Multiagent Systems, 2015.

[15] W. L. Perry. *Predictive policing: The role of crime forecasting in law enforcement operations*. Rand Corporation, 2013.

[16] E. Shieh, B. An, R. Yang, M. Tambe, C. Baldwin, J. DiRenzo, B. Maule, and G. Meyer. Protect: A deployed game theoretic system to protect the ports of the united states. In *Proceedings of the 11th International Conference on Autonomous Agents and Multiagent Systems-Volume 1*, pages 13–20. International Foundation for Autonomous Agents and Multiagent Systems, 2012.

[17] M. Tambe. *Security and game theory: algorithms, deployed systems, lessons learned*. Cambridge University Press, 2011.

[18] S. Weisburd. Does police presence reduce car accidents? Technical report, Auguest 2016.

[19] World Health Organization. Road traffic injuries fact sheet. Technical report, November 2016.

[20] R. Yang, A. X. Jiang, M. Tambe, and F. Ordonez. Scaling-up security games with boundedly rational adversaries: A cutting-plane approach. In *IJCAI*, 2013.

[21] Z. Yin, A. X. Jiang, M. P. Johnson, C. Kiekintveld, K. Leyton-Brown, T. Sandholm, M. Tambe, and J. P. Sullivan. Trusts: Scheduling randomized patrols for fare inspection in transit systems. In *IAAI*, 2012.

[22] C. Zhang, S. Gholami, D. Kar, A. Sinha, M. Jain, R. Goyal, and M. Tambe. Keeping pace with criminals: An extended study of designing patrol allocation against adaptive opportunistic criminals. *Games*, 7(3):15, 2016.