

Optimal Allocation Strategy Based on Stackelberg Game for Inspecting Drunk Driving on Traffic Network

Yingmo Jie^{1,2}, Mingchu Li^{3,4}, Tingting Tang⁵ and Cheng Guo^{6,7}

¹School of Mathematical Sciences, Dalian University of Technology
Dalian, 116024-P.R.China

²School of Software Technology, Dalian University of Technology
Dalian, 116620-P.R.China
[e-mail: jymsf2015@mail.dlut.edu.cn]

³School of Software Technology, Dalian University of Technology
Dalian, 116620-P.R.China

⁴Key Laboratory for Ubiquitous Network and Service Software of Liaoning Province
Tuqiang Street, Dalian 116620-P.R.China
[e-mail: mingchul@dlut.edu.cn]

⁵School of Software Technology, Dalian University of Technology
Dalian, 116620-P.R.China
[e-mail: 1366604595@qq.com]

⁶School of Software Technology, Dalian University of Technology
Dalian, 116620-P.R.China

⁷Key Laboratory for Ubiquitous Network and Service Software of Liaoning Province
Tuqiang Street, Dalian 116620-P.R.China
[e-mail: guocheng@dlut.edu.cn]

*Corresponding author: Yingmo Jie

*Received April 4, 2017; revised July 9, 2017; accepted August 18, 2017;
published December 31, 2017*

Abstract

As the main means to cope with the stubborn problem of drunk driving, the inspection of drunk driving has already been paid more attention and thus reinforced. In this paper, we model this scenario as a Stackelberg game, where the police department (called defender) allocates resources dynamically in terms of the traffic situation on the traffic network to arrest drunk drivers and drivers who drink (called attacker), whether choosing drunk driving or designated driving service, expect to minimize their cost for given travel routes. However, with the number of resources are limited, our goal is to calculate the optimal resource allocation strategy for the defender. Therefore, first, we provide an effective approach (named OISDD) to fulfill our goal, i.e., generate the optimal strategy to inspect drunk driving. Second, we apply OISDD to directed graphs (which are abstracted from Dalian traffic network) to analyze and test its correctness and rationality. The experimental results show that OISDD is feasible and efficient.

Keywords: Stackelberg game, resource allocation, security, optimization, drunk driving

1. Introduction

With the rapid development of economy and improvement of public living standard, the demands for automobile has been increasing dramatically as it has become a common tool of driving. However, the phenomenon of drunk/drunken driving has become more and more seriously, and the consequent result of traffic accident usually causes a fatal car crash. More generally, drunk driving has become a global issue and captured extensive attentions. Many countries have invested a lot of financial resources to cope with the problem of drunk driving. Especially in China, as the main method to cope with drunk driving, the inspection of drunk driving has been reinforced during recent years, as shown in Fig. 1. Although the punishment of drunk driving enacted by the Traffic Management Bureau of China is quite severe, the case has not been abated or even become worse, and thus caused countless traffic tragedies. According to the statistical data provided by the Ministry of Public Security Traffic Management Bureau of the People's Republic of China (the Traffic Management Bureau of China), in recent years the traffic accidents caused by drunk driving were more than 25 ten thousand, and gave rise to the death of about 5 ten thousand as well as the direct economic losses of 1.3 billion in China. Therefore, how to effectively alleviate and even eliminate drunk driving problem (DDP) has become the key hot topic in China. In this paper, we will explore the problem. Considering the actual situation, wine, as the table manner of China, it is difficult to avoid drinking. Thus, as a fledging industry, the designated driving service can resolve this problem and get huge attention. Beijing, which registered 4.7 million motor vehicles by the end of 2010, has a demand for about 3000 designated drivers to serve drunk people a day, according to the estimates of business insiders. However, a lot of drunk people drive their cars to go home in order to save their cost. Thus, it is very important to inspect these illegal (or danger) actions, and so there exists a game between drunk people and police department.



Fig. 1. Inspection of drunk driving

During recent years, game theory has been widely used to protect large scale critical infrastructure such as airport train stations, ports, forests etc. Generally speaking, we allocate few security resources to the specified targets to ensure the safety. However, it is impossible to cover all targets with limited security resources at all time. Thus, how to allocate the limited security resources to these targets and obtain the optimal results is a quite important research

problem, which has been recently studied by using Stackelberg games [1]. For the inspection of drunk driving, the defender puts the security resources include policemen and their tools at the roads on the traffic network to prevent attackers from drunk driving and maintain traffic safety.

In this paper, the main contributions are as follows: (a) Firstly, we propose an approach named OISDD (Optimal Inspection Strategy for Drunk Driving) with two phases, which concludes (i) we propose an algorithm called OA to obtain an optimal strategy (a mixed strategy, i.e., a probability distribution over all pure strategies) for the defender whose objective is to maximize his/her utility, against different types of attackers, who have different places of departure and destination. For the allocation of security resources, we adhere to the rationality and availability in terms of some actual situations on the traffic network and the optimal strategy. Furthermore, in order to avoid some problems, such as the waste of resources, the heavy traffic causing by inspecting drunk driving, and drivers' dissatisfaction, we propose another algorithm called IOA to solve them and improve OA. (ii) As the payoffs of the roads may change and the inspection can be exposed over time, we need to relocate the security resources as quickly as possible with spending much time on inspecting but not travelling. Thus, we propose an algorithm named LTT to obtain the optimal relocation strategy; (b) Secondly, we demonstrate the performance of our approach based on the directed graphs abstracted from the traffic networks of Dalian city. The experimental results show the feasibility and validity of our approach and the safety of life and property with limited security resources.

In Section 2, we introduce some previous related work of optimal inspection problem (OIP), which are applied to different security scenarios. Section 3 introduces some related knowledge and notations about our paper. Section 4 proposes an approach named OISDD to inspect drunk driving. Section 5 conducts experiments to test the correctness and rationality of OISDD. Section 6 briefly narrates our work related to inspect drunk driving about this paper.

2. Related Work

Optimal inspection problem (OIP) is such one that how to make sufficient use of limited security resources to inspect various illegal (or danger) actions or attackers, protect public properties and maximize the utilities of inspectors (i.e., defenders). Many experts are interested in it. For example, Pita et al. developed an Assistant for Randomized Monitoring over Routes (ARMOR) system to optimize the setting of checkpoints and the patrol scheduling of police dogs in LAX [2]. Tsai et al. presented an Intelligent Randomization In Scheduling (IRIS) system to the Federal Air Marshals (FAMs) that provide law enforcement aboard U.S. commercial flights to prevent the potential attacks [3]. TEAMCORE group also provided a Port Resilience Operational/Tactical Enforcement to Combat Terrorism (PROTECT) system which has been deployed by the United States Coast Guard (USCG) to plan effective randomized patrols in many ports nationwide [4-5]. A new predictive anti-poaching tool, Comprehensive Anti-Poaching tool with Temporal and observation Uncertainty REasoning (CAPTURE) was built by Nguyen et al. to prevent wildlife poaching [6-7]. Yin et al. presented a Tactical Randomization for Urban Security in Transit System (TRUSTS) to schedule randomized patrols for fare inspection in transit system. The Los Angeles Sheriffs Department (LASD) is currently carrying out trials of TRUSTS on the LA Metro [8]. Matthew et al. proposed Strategic Randomization with Exploration and Exploitation in Traffic patrol Schedules (STREETS) for traffic patrolling to prevent mitigate accidents. In order to control the unpredictability in travel time through road networks, the

model weighted exploitation and exploration by solving a biobjective optimization problem was modelled by Markov Decision Process [9].

Resource allocation is the scheduling of activities and the resources required by those activities while taking into consideration both the resource availability and the project time. It is widely used for various networks [10-11]. In this paper, we propose an OISDD approach to prevent drunk people from driving on traffic network. Considering some actual details, such as the cost and negative impact of security resources on the traffic networks, the severity of leaking inspection messages, and the influence of traffic flows on travel time, we divide the approach into two steps, whose aims are maximizing the total utilities of the defender and minimizing the travel time of relocating security resources to generate the optimal schedule of inspections.

3. Preliminaries

3.1 Stackelberg game

Game theory is the study of mathematical models of conflict and cooperation between intelligent rational decision-makers (players). Each player has his/her own utility function and aims to maximize it. Therefore, the solution concept of a game is a set of equilibrium strategies for each player such that, when these strategies are employed, no player can profit by unilaterally deviating from their strategies.

For Stackelberg game, there exists two players, which has been widely used as models to represent the patrol schedules for security of critical infrastructure. Particularly, the defender (leader) first commits to a mixed strategy, which can be observed by the attacker (follower) over time, then based on the observation, the attacker chooses a best response [12]. The combination between the optimal mixed strategy for the defender and the attacker's best response forms a Stackelberg equilibrium of the game.

3.2 Problem setting

Akin to the previous work, we use Stackelberg game as a basic model to simulate DDP. For the players, the Traffic Management Bureau acts the defender, whose inspection strategy is mixed and the drunk drivers act as the attackers to choose best responses according to the corresponding observations. Compared to the whole traffic network, the security resources are limited. Therefore, the defender's strategy is how to allocate the security resources to the network in order to maximize own utilities (profits), while the attackers have observation abilities and then decide whether choosing designated driving services or not after drinking to maximize own utilities (i.e., minimize own cost). In this paper, considering some actual situations, especially in China, we regard fining drunk drivers as an effective method to prevent the deaths and injuries caused by drunk driving. For the utilities of players, we propose the utility matrix of the defender and attacker in **Table 1**. B denotes the big punishment cost of drunk driving, H denotes the fee of designated driving services, note that, compared with H , B is much greater.

Table 1. Utility matrix for both players

	Inspected	Uninspected
Drunk driving	$(-B, B)$	$(0, 0)$
Not drunk driving	$(B - H, H - B)$	$(-H, H)$

We model this scenario as a zero-sum game, where the utility gained by the defender always equals to the utility lost by the attacker. Here we will explain by means of the utility matrix.

1).For the case of the action pair (Drunk driving, Inspected), the attacker is fined B due to drunk driving and being arrested by the defender. Hence, the corresponding utility pair is $(-B, B)$;

2).For the case of the action pair (Drunk driving, Uninspected), the attacker escapes from being fined due to the uninspected strategy of the defender. Hence, both of them have no payoffs, then the corresponding utility pair is $(0,0)$.

3).For the case of the action pair (Not drunk driving, Inspected), on one hand, the attacker chooses designated driving services and spends H avoiding being fined B by the defender, then the utility is $B - H$. On the other hand, the duty of the defender is to prevent the dangerous action-drunk driving. Hence, although the defender does not obtain the payoff B , the reward R of protecting the traffic safety should be considered, then the utility of the defender is $-B + R$. Here, to reduce the complexity of this model, we assume that R equals to H . Therefore, the relevant utility pair is $(B - H, H - B)$.

4).For the case of the auction pair (Not drunk driving, Uninspected), the attacker spends H while the defender does not apply the inspected strategy. Akin to the above-mentioned, as to ensure the traffic safety, the defender can obtain a reward R which equals to H . Hence, the relevant utility pair is $(-H, H)$.

The common notations can be shown in **Table 2**. Note that the superscript t for any notations denotes the period t throughout this paper.

Table 2. Notations

m	Security resources
n	Intersections of traffic network
T	Set of work hours
λ	One type of attacker
Λ	Set of all types of attackers, $ \Lambda = k$
\mathbf{p}^t	Probability distributions of all types of attackers
\mathbf{H}^t	Fee of designated driving services of all types
\mathbf{C}^t	Routes of all types
\mathbf{f}^t	Inspection ratio of all intersections
\mathbf{X}^t	Strategies of the defender for all types
\mathbf{u}^t	Expected utilities of the defender against all types
\mathbf{D}_e	Distance of all roads e on the network G
\mathbf{N}_e^t	Traffic flows of all roads e

3.3 Inspection

There exist some limited security resources used to inspect a traffic network with many intersections. For each security resource, it needs to work for some hours. Here we separate

these hours into several periods $T = \{1, 2, \dots\}$. First of all, for each period $t \in T$, we calculate the optimal strategy for the allocation of security sources, which is subject to the maximum of the defender's total profits. Then, with time past, in order to make the best use of limited resources, for any two consecutive periods, we calculate the optimal strategy for the relocation of resources by minimizing the total travel time on the network. At last, the schedule of inspecting drunk driving is generalized by these optimal strategies for all periods.

3.4 Traffic network

We address the problem of generating optimal strategies for the defender to inspect the traffic network of the development area of Dalian. The specific traffic situation has been shown in Fig. 2. We can abstract the traffic network into a directed graph $G = \langle V, E \rangle$ constituted by a set of vertices V and a set of directed edges E , which can be looked as intersections and roads respectively. In order to demonstrate the vehicle flows clearly, we add to the graph a dummy vertex \hat{v} with dummy edges \hat{e} to some vertices in V , where $\hat{e} \notin E$ and $\hat{v} \notin V$. We assign that these additional dummy edges have no effectiveness and just use for formulating the vehicle flows. For each edge e , the traffic flows from its start vertex ϕ_e to its end vertex φ_e , hence we can formulate e as $e = (\phi_e, \varphi_e)$, then \hat{E} can be formulated as $\hat{E} = \{\hat{e} | \phi_{\hat{e}} = \hat{v}\}$. Note that, there exists no self-loops, i.e., $\phi_e \neq \varphi_e$ for each edge $e \in E$. For each edge $e \in E$, suppose $E_e^- = \{e' | \varphi_{e'} = \phi_e\}$ be the set of upstream edges of e , and $E_e^+ = \{e' | \phi_{e'} = \varphi_e\}$ be the set of downstream edges of e . We also denote $E_v = \{e | v = \varphi_e, e \in E \cup \hat{E}, v \in V\}$ as the set of edges e , whose end vertex is v .



Fig. 2. The development area of Dalian map

3.5 Vehicle flow

Let D^t denote the total vehicle flows entering the graph by the vertex \hat{v} during period t . For each edge e , N_e^t represents the number of vehicle flows of e entering by the vertex ϕ_e , then we have $D^t = \sum_{e \in \hat{E}} N_e^t$, where $N_e^t \geq 0$. There exists a turning matrix $R \in \mathbb{R}_+^{(E \cup \hat{E}) \times (E \cup \hat{E})}$ associated

with the tendency of traffic flows among consecutive edges whose entry $R_{ee'}$ represents the fraction of vehicles flowing out of edge e and turning to edge e' . In the terms of rationality, we should note that $\sum_{e' \in E_e^+} R_{ee'} = 1$ for each edge e . Moreover, $R_{ee'} = 0$ if $\varphi_e \neq \phi_{e'}$ should be considered as the natural topological constraints. As the above-mentioned, the number of vehicle flows of edge e can be obtained as follow

$$N_e^t = \sum_{e' \in E_e^- \cup \hat{E}} N_{e'}^t R_{e'e}, \quad \forall e \in E. \quad (1)$$

Corollary 1[13]. Given an initial vector $\hat{\mathbf{N}}^t = \{N_{\hat{e}}^t, \hat{e} \in \hat{E}\}$ and a turning matrix R , there exists a unique vehicle flow $\mathbf{N}^t = \{N_e^t, e \in E\}$ satisfying (1).

Proof. Adding the set of dummy edges \hat{E} to this directed graph, then we can obtain a new one $G' = \langle V, E_{\cup} \rangle$, where $E_{\cup} = \hat{E} \cup E$. Here the edges belong to \hat{E} have no the start vertices. (1) can be rewritten as follows:

$$N_e^t = N_{\hat{e}}^t, \quad \forall e \in \hat{E}, \quad (2)$$

$$N_e^t = \sum_{e' \in E_e^-} N_{e'}^t R_{e'e}, \quad \forall e \in E_{\cup} \setminus \hat{E}. \quad (3)$$

By Varaiya [13], we may also get the unique flow $\mathbf{N}^t = \{N_e^t, e \in E\}$ satisfying (1).

Similarly, for each vertices $v \in V$, we denote \mathfrak{N}_v^t as the number of vehicle flows passing the intersection v , where

$$\mathfrak{N}_v^t = \sum_{e \in E_v} N_e^t, \quad \forall v \in V. \quad (4)$$

Note that, as the demand of relocating security resources, we need to obtain the vehicle flow of each period. For simplicity, before considering the travel of vehicles during period $t+1$, we can ignore the number of vehicles existing on the traffic network during the last period t .

3.6 Attackers

As the attackers have the ability of observation and prediction, hence they are assumed to be rational, risk-neutral economic actors [14], who make this choice in order to maximize expected utilities. For this assumption, we only consider the drivers, who decide their choices before drinking or are basically sober-minded based on the effect of alcohol on human [15-16], although alcohol has been ingested. However, we do not consider that a drunk driver, who has drunk a lot of alcohol and lost the ability of driving as well as thinking, still drives by himself (herself). Any two different vertices in the graph, which can be looked as the origin and destination of a route, are denoted as an attacker's type λ . In terms of reality and rationality, we assume that all types of attackers choose the shortest paths as their own travel routes.

Hence, the traffic network with n intersections has $O(n^2)$ types.

However, there may exist one actual case, that is, if someone who just passes by somewhere and observes that an inspection is happened at there, then he (she) may leak this message by the private communication or the public platform, such as microblog, twitter and so on. In order to avoid the inspection, some drunk drivers will adjust the original routes based on the obtained message. Generally, for the inspected road, the corresponding edge in a graph can be removed to obtain a subgraph, then the attackers will reselect the shortest paths in this subgraph. To some extent, the efficacy of this inspection has been reduced largely, hence, we may consider the relocation of resources over a period of time.

3.7 Objective

Based on the different types of attackers, the defender's utility can be divided into utilities from bilateral interactions with each individual type, moreover the utility of each type does not directly affect around each other. Hence, the game can be equivalently formulated as Bayesian Stackelberg game with two players, in which there is just one type taking on the role of one of the types. The specific types are known to the attackers but not to the defender, and is drawn from a probability distribution such that the probability p_λ of the attacker's type $\lambda \in \Lambda$ is proportional to its total volume.

Furthermore, although in general solving Bayesian Stackelberg games is NP-complete [17], the utility functions of this game satisfy the zero sum property, i.e., the utility gained by the defender always equals to the utility lost by the attacker. For such zero-sum Bayesian games, the Stackelberg equilibrium is equivalent to the maximin solution, which are solvable by linear programming formulations [18-19]. For Bayesian Stackelberg games, there exist some kinds of basic algorithms to solve Stackelberg equilibrium. MultiLPs [12] proposed by Conitzer et al. is a polynomial time algorithm; DOBSS [20] and ERASER [21] algorithms release the computational complexity of MultiLPs based on the mutual independence of attackers.

4. Proposed approach: OISDD

In this section, we will propose an approach to inspect drunk driving, called OISDD, which can be divided into two phases:

Phase 1. Optimal strategy for allocating resources: For each period, we first compute the Stackelberg equilibrium to maximize the total profits of the defender, and then the allocation of resources is decided on account of the rationality and reality.

Phase 2. Optimal strategy for relocating resources: In terms of the allocation of phase 1 and the criterion that spending much time on inspecting but not travelling, we relocate the resources for any two adjacent periods to generate the schedule of inspecting drunk driving.

4.1 Computing the allocation of resources

Before computing the optimal mixed strategy of security resources allocation, we need to consider the formulation of attackers' types. Just as above-mentioned, for any attacker, he/she will choose the shortest path as his/her travel route. Hence, we will apply Dijkstra algorithm [22] to calculate the shortest paths between any two different vertices in the graph, which can be formulated as the following functions:

$$\text{LP1: min} \quad d'_\lambda \quad (5)$$

$$s.t. \quad d_{\lambda}^t = \sum_{e=(i,j) \in E} d_{i,j} c_{i,j}^t, \quad (6)$$

$$c_{i,j}^t \in \{0,1\}, i, j \in V. \quad (7)$$

$$\text{LP2: min} \quad d_{\lambda}^t \quad (8)$$

$$s.t. \quad d_{\lambda}^t = \sum_{e=(i,j) \in E} d_{i,j} c_{i,j}^t, \quad (9)$$

$$c_{i,j}^t \in \{0,1\}, i, j \in V, \quad (10)$$

$$d_{i,j} = +\infty, \exists e_{i,j}^t = 1. \quad (11)$$

LP1 represents the travel route of the type λ , who has no message about any inspections, and LP2 means the type λ has been told that there exists an inspection at one road $e = (i, j)$, where $i, j \in V$. The objective (5) and (8) are to minimize the travel route of the type λ . Constraint (6), (7) and (9), (10) give out the formulations of specific travel routes, where $d_{i,j}$ indicates the distance of road $e = (i, j)$ and $c_{i,j}^t$ is a binary variable, that if set to 1 indicates that the road $e = (i, j)$ will exist in the route of the type λ . Constraint (11) indicates that the type λ has known that there exists an inspection at the road $e = (i, j)$, then he/she would neglect this road and reselect his/her shortest travel route.

Algorithm 1 OA (Optimal Allocation)

Input:

$$\mathbf{p}^t = (p_{\lambda_1}^t, p_{\lambda_2}^t, \dots, p_{\lambda_k}^t), \mathbf{H}^t = (h_{\lambda_1}^t, h_{\lambda_2}^t, \dots, h_{\lambda_k}^t), \mathbf{C}^t = (\mathbf{c}_{\lambda_1}^t, \mathbf{c}_{\lambda_2}^t, \dots, \mathbf{c}_{\lambda_k}^t), \mathbf{f}^t = (f_1^t, f_2^t, \dots, f_n^t)$$

Output:

$$\mathbf{X}^t = (\mathbf{x}_{\lambda_1}^t, \mathbf{x}_{\lambda_2}^t, \dots, \mathbf{x}_{\lambda_k}^t), \mathbf{u}^t = (u_{\lambda_1}^t, u_{\lambda_2}^t, \dots, u_{\lambda_k}^t)$$

Procedure:

for i from 1 to k , step 1, do

$$\mathbf{x}_{\lambda_i}^t, u_{\lambda_i}^t \leftarrow \text{MODELLP3}(p_{\lambda_i}^t, h_{\lambda_i}^t, B, \mathbf{c}_{\lambda_i}^t, \mathbf{f}^t)$$

end for

Return $\mathbf{X}^t, \mathbf{u}^t$

Note that the fee of designated driving services of all types can be written as

$$h_{\lambda_i}^t = r^t \times \left\lceil \frac{d_{\lambda_i}^t - d_0}{d_0} \right\rceil + b_0,$$

where r^t is the per unit distance price, $d_{\lambda_i}^t$ is the travel distance of the type λ_i , b_0 and d_0 are initial charge and the distance covered by initial charge respectively, which are the same for all periods; $\mathbf{c}_{\lambda_i}^t$ is a binary vector to indicate whether the road belongs to the travel route of the type λ_i or not; $\mathbf{x}_{\lambda_i}^t = (x_{1,i}^t, x_{2,i}^t, \dots, x_{n,i}^t)$, is the mixed strategy of the defender aiming at the type λ_i .

Due to the mutual independence of all types, we use OA to call the MODEL3 function for each type and aim to maximize the total profits of the defender. For attackers' types, we may refer to LP1 and LP2.

$$\text{MODEL3}(p_{\lambda}^t, h_{\lambda}^t, B, \mathbf{c}_{\lambda}^t, \mathbf{f}^t) : \max_{\mathbf{x}^t, \mathbf{u}^t} \sum_{\lambda \in \Lambda} p_{\lambda}^t u_{\lambda}^t \quad (12)$$

$$s.t. \quad u_{\lambda}^t \leq \min\{H, B \sum_{i \in N} x_i^t c_{i,j}^t f_i^t\}, \quad (13)$$

$$\sum_{i \in N} x_i^t \leq m, \quad (14)$$

$$x_i^t \in [0, 1], \quad (15)$$

$$c_{i,j}^t \in \{0, 1\}, \quad (16)$$

$$f_i^t \in [0, 1]. \quad (17)$$

Given a period t , the function MODEL3 computes the Stackelberg equilibrium for each type λ . The objective of this algorithm is to compute the allocation vectors corresponding to each type that will maximize the total profits of the defender, while each attacker will decide whether choosing designated driving services or not after drinking that will minimize his/her cost for given the travel route. $c_{i,j}^t$ is a binary variable, that if set to 1 indicates that the road $e = (i, j)$ will exist in the route of attacker λ , where $i, j \in V$; x_i^t is a continuous variable, which indicates the desired allocation of intersection i ; f_i^t is a continuous variable, which represents the inspection ratio corresponding to intersection i .

The objective (12) is to maximize the total profits obtained by the defender. Constraint (13) ensures that the type λ plays a best response to the allocation vector employed by the defender. Constraint (15) limits the allocation for an intersection to less than 1 and (14) constrains that the number of intersections inspected is bounded by the number of security resources available. Constraint (16) ensures a feasible route of one type. Constraint (17) demonstrates that it is not necessary to inspect all vehicles for an intersection in order to release the traffic jam causing by inspecting drunk driving.

However, there are some deficiencies about OA. First, the cost of resources is not taken into consideration but actually the relevant cost is not cheap for the real life, such as the wages of policemen, the cost of gas and so on. Second, in general, the more resources we use, the greater profits we get, but too many resources will lead to such problems: the waste of resources, heavy traffic, and drivers' dissatisfaction. Thus, we need to explore these problems. In this paper, we propose a novel algorithm called IOA to solve them.

Algorithm 2 IOA (Improved Optimal Allocation)

Input:

$$\mathbf{p}^t = (p_{\lambda_1}^t, p_{\lambda_2}^t, \dots, p_{\lambda_k}^t), \mathbf{H}^t = (h_{\lambda_1}^t, h_{\lambda_2}^t, \dots, h_{\lambda_k}^t), \mathbf{C}^t = (\mathbf{c}_{\lambda_1}^t, \mathbf{c}_{\lambda_2}^t, \dots, \mathbf{c}_{\lambda_k}^t), \mathbf{f}^t = (f_1^t, f_2^t, \dots, f_n^t), \\ B, c, \alpha$$

Output:

$$\mathbf{X}^t = (\mathbf{x}_{\lambda_1}^t, \mathbf{x}_{\lambda_2}^t, \dots, \mathbf{x}_{\lambda_k}^t), \mathbf{u}^t = (u_{\lambda_1}^t, u_{\lambda_2}^t, \dots, u_{\lambda_k}^t)$$

Procedure:

for i from 1 to k , step 1, do

$\mathbf{x}_{\lambda_i}^t, u_{\lambda_i}^t \leftarrow \text{MODELLP4}(p_{\lambda_i}^t, h_{\lambda_i}^t, B, \mathbf{c}_{\lambda_i}^t, \mathbf{f}^t, B, c, \alpha)$

end for

Return $\mathbf{X}^t, \mathbf{u}^t$

Note that, $c > 0$ is the cost of each resource, $\alpha > 0$ is the parameter used for converting m into monetary cost, which can be regarded as the influence of the number of resources on the total profits of the defender.

Similar to OA, we apply IOA to call the MODELLP4 function for each type of attackers.

$$\text{MODELLP4}(p_{\lambda_i}^t, h_{\lambda_i}^t, B, \mathbf{c}_{\lambda_i}^t, \mathbf{f}^t, B, c, \alpha) : \max_{\mathbf{x}^t, \mathbf{u}^t} \sum_{\lambda \in \Lambda} p_{\lambda}^t u_{\lambda}^t - cm - e^{\alpha m} \quad (18)$$

$$s.t. \sum_{c_{i,j}^t \in \mathbf{c}_{\lambda}^t} x_i^t c_{i,j}^t f_i^t \leq 1, \quad (19)$$

$$(13) - (17).$$

The objective (18) is also to maximize the total profits obtained by the defender but takes the cost of resources and the influences of resources that impact on the traffic network into account. Constraint (19) limits the total probability of inspection for each type to less than 1.

After computing the above-mentioned function, we can obtain the optimal mixed strategy $\tilde{\mathbf{X}}^t = (\tilde{x}_1, \tilde{x}_2, \dots, \tilde{x}_n)$, committed by the defender. However, we cannot neglect some actual details for the allocation of resources. If we set the inspections on the intersections, there may exist some drawbacks, on the one hand, it may cause the traffic jams with the processing of inspecting drunk driving; on the other hand, it is much easier to observe the inspections and give rise to leaking the inspection messages. Hence, in order to take the actual details into full consideration and cope with these shortcomings, we will set the resources to the roads, i.e., denote the roads as the final targets, based on (20)

$$\varepsilon_{i,j}^t = \frac{1}{c_i} \cdot \frac{\tilde{x}_i^t \mathfrak{N}_i^t + \sum_{z \in V_i} \delta_{z,i}^t \tilde{x}_z^t \mathfrak{N}_z^t}{\sum_{i \in V} [\tilde{x}_i^t \mathfrak{N}_i^t + \sum_{z \in V_i} \delta_{z,i}^t \tilde{x}_z^t \mathfrak{N}_z^t]} + \frac{1}{c_j} \cdot \frac{\tilde{x}_j^t \mathfrak{N}_j^t + \sum_{z \in V_j} \delta_{z,j}^t \tilde{x}_z^t \mathfrak{N}_z^t}{\sum_{j \in V} [\tilde{x}_j^t \mathfrak{N}_j^t + \sum_{z \in V_j} \delta_{z,j}^t \tilde{x}_z^t \mathfrak{N}_z^t]}, \quad (20)$$

where $\varepsilon_{i,j}^t$ denotes the probability of setting security resources at the road $e = (i, j)$ and satisfies $\sum_{e \in E} \varepsilon_{i,j}^t = 1$, which also can be indicated as the final mixed strategy of the defender;

V_i denotes the set of intersections associating with the intersection i ; c_i denotes the degree of i , i.e., the number of forks of the intersection i ; $\delta_{z,i}^t \geq 0$ denotes the impact factor of intersection z , that acts on the intersection i , and satisfies $\sum_{i \in N_z} \delta_{z,i}^t = 1$.

4.2 Computing the relocation of resources

With the time past, on the one hand, the payoffs of targets deserving to inspect will have the corresponding change. For different places, the corresponding flourishing degrees are also different, thus the volumes of vehicle flows are quite distinct. For example, the vehicle flows of commercial district are far greater than some common areas during some certain time; on the other hand, related to the leakage of message to attackers, there exists a setting where the longer a particular location is the site of an inspection, the larger population that is aware of that inspection point is, leading to the efficacy of this inspection has been reduced largely. Hence, we need to relocate the policy resource for any two adjacent periods to generate the schedule of inspections.

Before the work of relocating sources, we need to find out the candidates for relocation. Given two sets of inspected targets for any two adjacent periods, if there exist targets, which are both belong to the two sets, then the resources at these targets will stay there, not to move. Here we can formulate as follows:

$$E'' = E^t - E^t \cap E^{t+1}, E''' = E^{t+1} - E^t \cap E^{t+1}.$$

Note that, E^t and E^{t+1} are the sets of inspected targets during period t and $t+1$, and satisfy $|E^t| = |E^{t+1}| = m$. E'' and E''' are the sets of candidates for relocation, i.e., obtained by deleting the inspected targets belong to the collection of E^t and E^{t+1} , where $|E''| = |E'''| = m', m' \leq m$.

After choosing the candidates, we will propose the specific algorithm of resources relocation, which satisfies two basic rules.

Rule 1. The measurement criteria of relocating resources: Considering the actual situation of traffic network, we set the time of relocation as the standard to measure the relocations and aim to minimize it.

Rule 2. The choice of relocating routes: It is well known that the inspections may cause the heavy traffic, hence, in order to save the time spending on the travel, when choosing the relocating routes, we need to keep away from the road, which was just set as an inspected target during the last period.

We divide the relocation approach into two steps:

- 1) We propose Algorithm 3 named LTT to calculate the least travel time $\tau_{i,j}^{t+1}$ from any road $i \in E''$ to any road $j \in E'''$;
- 2) Based on Hungarian algorithm [23] and given $\tau_{i,j}^{t+1}$, for all $i \in E''$ and $j \in E'''$, we minimize the total travel time to obtain the optimal strategy for relocating resources.

Least travel time of relocation route

Algorithm 3 LTT (Least Travel Time)

Input: $\mathbf{D}^t = \{d_e | e \in E\}$, $\mathbf{N}_e^{t+1} = \{N_e^{t+1} | e \in E\}$, E'' , E''' , E^t , N_{\max} , g_f

Output: $\Gamma_{E'', E'''}^{t+1} = \{\tau_{i,j}^{t+1} | i \in E'', j \in E'''\}$

Procedure:

for i from 1 to m' , step 1, do
for j from 1 to m' , step 1, do
 $\tau_{i,j}^{t+1} \leftarrow \text{MODELLP5}(\mathbf{D}^t, \mathbf{N}_e^{t+1}, i, j, E^t, N_{\max}, g_f)$
end for
Return $\Gamma_{E^t, E^{t+1}}^{t+1}$

Note that N_{\max} is the maximum capacity of road network; g_f is the free-flow speed obtained when there is no vehicle on road; $\Gamma_{E^t, E^{t+1}}^{t+1}$ represents the least travel time from any road $i \in E^t$ to any road $j \in E^{t+1}$.

LTT employs the MODELLP5 function to compute the least travel time from any road $i \in E^t$ to any road $j \in E^{t+1}$.

$$\text{MODELLP5}(\mathbf{D}^t, \mathbf{N}_e^{t+1}, i, j, E^t, N_{\max}, g_f): \min \tau_{i,j}^{t+1} \quad (21)$$

$$s.t. \tau_{i,j}^{t+1} = \sum_{e \in E} w_e^{t+1} < +\infty, \quad (22)$$

$$\sum_{e \in E} w_e^{t+1} = +\infty, \text{ if } e \in E^t, \quad (23)$$

$$w_e^{t+1} = \frac{d_e}{g_e^{t+1}}, \quad (24)$$

$$g_e^{t+1} = g_f \cdot \frac{N_{\max} - N_e^{t+1} + 1}{N_{\max}}. \quad (25)$$

The travel speed of each road e is formulated as (25), which is inspired by Smith et al. [24]. For any road e , when $N_{\max} = N_e^{t+1}$, then the travel speed g_f/N_{\max} is close to zero, and the traffic may travel in a “stop-and-go” manner, rather than complete gridlock; when $N_e^{t+1} = 0$, the travel speed $g_f + 1/N_{\max}$ is close to g_f , and the traffic may travel in an “unblock” manner. Constraint (24) represents the travel time of each road e . Given the transportation network G with the travel time of each road, (22) shows the feasible routes from road i to road j . Constraint (23) limits that if there exists an inspection at the road e during period t , we delete this road from the network in order to satisfy rule 2.

Optimal strategy of relocation

To consider this complexity, for any two adjacent periods, we get the optimal relocation strategy based on the assignment problem, whose computational complexity is $O(m^4)$. Here we employ Hungarian algorithm to call LP6 function to compute the optimal relocation strategy during the two adjacent periods t and $t+1$.

$$\text{LP6: } \min_{\mathbf{A}} \sum_{i=1}^{m'} \sum_{j=1}^{m'} \tau_{i,j}^{t+1} \cdot a_{i,j} \tag{26}$$

$$\text{s.t. } \sum_{j=1}^{m'} a_{i,j} = 1, \forall i \in E'' , \tag{27}$$

$$\sum_{i=1}^{m'} a_{i,j} = 1, \forall j \in E''^{t+1} , \tag{28}$$

$$a_{i,j} \in \{0,1\} . \tag{29}$$

The purpose of this function is to compute the matrix of security relocation that will minimize the total time spending on the travel, where the rows and columns are indexed by the resources of periods t and $t + 1$ respectively. $a_{i,j}$ is a binary variable, that if set to 1 indicates that the relocation of one security resource is relocated from i to j .

The objective (26) is to minimize the total travel time of all resources. Constraint (27) and (28) limit the uniqueness of resource relocation from one target to another, from the view of formulation, for the matrix \mathbf{A} , there has one and only one 1 in each row (column). Constraint (29) ensures a feasible relocation of one security resource.

Given the optimal allocation strategy for each period and the optimal relocation strategy for any two adjacent periods, we generate an optimal schedule of inspections with all security resources.

Example

There are 9 intersections (vertices) and 13 roads (edges) in Fig. 3(a). The working hours are divided into three periods, in order to narrate simplicity, we demonstrate the approach based on period t and $t + 1$. For given 3 security resources, we calculate the final mixed strategies based on IOA and (20) during period t and $t + 1$ respectively. The optimal allocation of resources is $e_{(4,5)}$, $e_{(3,7)}$ and $e_{(8,9)}$ for the period t while that of period $t + 1$ is $e_{(4,5)}$, $e_{(6,9)}$ and $e_{(1,3)}$. Next, we need to consider the relocation of resources for periods t and $t + 1$, which has been shown in Fig. 3(b). Based on LTT and LP6, we obtain the optimal relocation of resources, where the green arrow and red arrow represent the moving and not moving of the corresponding resources respectively.

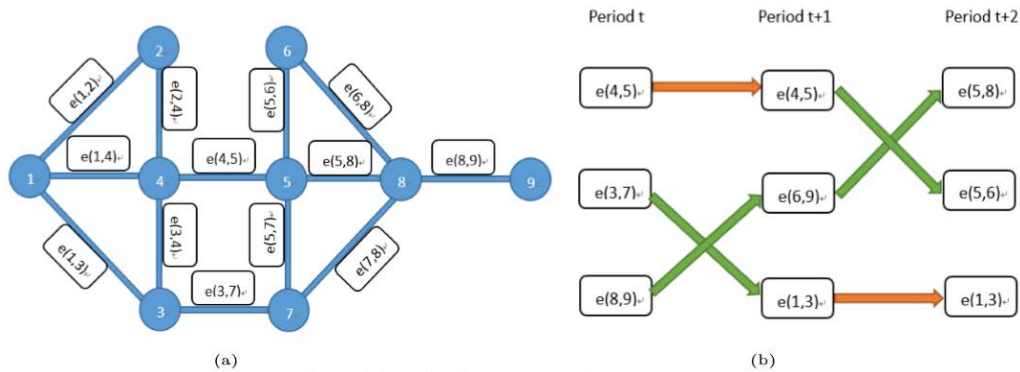


Fig. 3. Example of 13 targets and 3 resources

5. Experimental Classification Results and Analysis

To evaluate OISDD, we present our simulations in context of three directed graphs, which are abstracted from Dalian traffic network. All computations are run on a machine with 2.3GHz Intel Core i5 with 4 GB memory.

5.1 Data set

Fig. 2 shows the map of one part of the development area of Dalian. We divide it into three parts based on the yellow line, which contain 22 (the right bottom of the map), 27 (the top of the map), 35 (the left bottom of the map) intersections respectively. Then, we abstract them into three directed graphs for experiments. For different periods, we fix different total vehicle flows to calculate the flow of each road and intersection. The turning matrix R is decided by some actual situations, for example, the vehicle flow may prefer the residential areas and the leisure centers to the central business districts during quitting time. All types of attackers will be taken into consideration to simulate, and the number of each type is random but guarantees the average number ranging from 30 to 50. For the fee of designated driving services, we set $r' = 20, d_0 = 10, b_0 = 10$ to obtain the fee of each type, and the punishment cost of drunken driving B is denoted as 500 RMB. For the inspection ratio f of different intersections, we set them reasonably based on the vehicle flows of each intersection for every period.

5.2 Results

In **Fig. 4**, we evaluate the influence on the rate of drunk driving as we test three different graphs with 22, 27, 35 intersections respectively. The x-axis is the number of security resources and y-axis is the rate of drunk driving. For each attacker, he/ she will compare the fee of designated driving services according to the corresponding travel route with the punishment cost of drunk driving by oneself, then decide whether drunk driving or not. From the results, we can observe two general trends. First, the increase of the number of resources will leads to a reduction of the rate of drunk driving on the traffic network. Second, with intersections increasing, the decreasing rate of the rate of drunk driving will be slow. It is easy to understand that the possibility of inspections becomes bigger with more resources, hence the drunk people prefer to choose designated driving services rather than driving by himself/herself. Usually, it is easier to escape from inspections on the bigger traffic network, i.e., with more intersections.

In **Fig. 5**, we evaluate the influence on the total profits of the defender as we test three different graphs with 22, 27, 35 intersections respectively. The x-axis is the number of security resources and y-axis is the total profits in the function MODEL3, i.e., the objective (12), obtained by the optimal mixed strategy of the defender. From these results, we can observe two general trends. First, increasing the number of resources leads to a growth of the total profits. Second, with intersections increasing, the growth rate of the total profits will be quick. Those make sense, the more resources we use, the more profits we get. Similarly, the more intersections the network has, the more potential attackers it exists, to some extent, the more profits it brings to the defender.

In **Fig. 6**, we evaluate the influence of the inspection ratio f on the rate of drunk driving as we test three different graphs with 22, 27, 35 intersections respectively. The x-axis is the inspection ratio f_i at the intersection i and y-axis is the rate of drunk driving. The types of attackers, whose travel routes pass through the intersection i , are all taken into considerations.

From the results, we can observe two general trends. First, increasing f_i leads to a reduction of the rate of drunk driving. Second, with intersections increasing, the decreasing rate of the inspection ratio will be slow. It is obvious to understand that drunk people are afraid of drunk driving if the policemen reinforce the power of inspections. However, with the certain inspection ratio, the rate of drunk driving on the bigger network is still higher than that on the smaller one. Hence, for different networks, we need to adopt the proper inspection ratio of drunk driving to prevent traffic jams (too high ratio) and drunk driving (too low ratio).

In Fig. 7, we evaluate the influence of the distance of road on the rate of drunk driving as we test three different graphs with 22, 27, 35 intersections respectively. The x-axis is the number of security resources and y-axis is the rate of drunk driving. Here we amplify the distance of each road to 1.5 and 2 times. From these results, we find that the decreasing rate of the rate of drunk drinking will be slow with the longer distance. It makes sense that the longer route will leads to spend much more on designated driving services, hence the drunk people will choose to drive by himself/herself.

In Fig. 8, we evaluate the influence of the parameter used for converting resources into monetary cost on the total profits of the defender as we test three different graphs with 22, 27, 35 intersections respectively. The x-axis is the number of security resources and y-axis is the total profits of the defender in the function MODEL P4, i.e., the objective (18), obtained by the optimal mixed strategy of the defender. Here we set $c = 2000$ RMB. From the results, we can observe three general trends. First, the profit discriminations between different parameters are quite little with a few resources. Second, the profit function is unimodal. Third, with α increasing, the increasing rate of the total profits will be slow, but the decreasing rate will be quick. It is reasonable for those trends. The security resources have little negative effect on the traffic network, when the number of them is small. However, considering some actual factors, such as the cost of resources, the traffic jams and the limited resources, we need to apply the appropriate number of resources to inspect drunk driving. The bigger negative influence of resources, such as the traffic jams and the dissatisfaction of the public, leads to the less profits of the defender with certain resources.

In Fig. 9, we evaluate the influence of leaking inspection message on the attackers' cost as we test three different graphs with 22, 27, 35 intersections respectively. The x-axis is the number of security resources and y-axis is the average of attackers' cost. Here we assume that the message of an inspection at one road is leaked to the attackers, whose travel routes pass through this road. We calculate the average cost of all those types, where H represents the average cost of designated driving services; B and B_1 represent the average cost of not leaking and leaking message respectively. From the results, we can observe three general trends. First, increasing the number of resources leads to a general growth of both average cost B and B_1 . This makes sense. With the growth of resources, the possibility of inspected may rise roughly, i.e., the rising of cost. Second, for the same number of resources, B is always bigger than B_1 . Third, taking Fig. 9(a) as an example, the junction of H and B is approximate to $x = 2$, while the junction of H and B_1 is approximate to $x = 4$. These mean that the leakage of inspections brings some losses to the inspections of drunk driving.

In Fig. 10, we illustrate the efficacy of our approach by comparing the solution quality obtained by our approach with other two ones. The quality of a solution x is measured by the utility of the defender $U_d(x, y(x))$, where y is the attacker best response of x . The two approaches are: i) mean1: a pure defender strategy with marginal coverage probability of each

vertex equals to $\frac{m}{|V|}$; ii) mean2: a pure defender strategy that each vertex is covered by

vehicle inspection probability, where $x_i = \frac{n_i}{|n|} = \frac{N_i^t \cdot f_i^t}{\sqrt{\sum_{i \in E} n_i^2}}$. We evaluate the influence on the

total profits of the defender as we test three different graphs with 22, 27, 35 intersections respectively. The x-axis is the number of security resources and y-axis is the total profits of the defender. From these results, we can observe that our approach mean1 is quite better than other two ones.

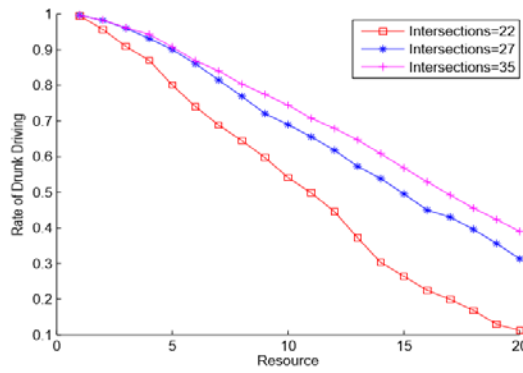


Fig. 4. Influence of intersections on rate of drunk driving

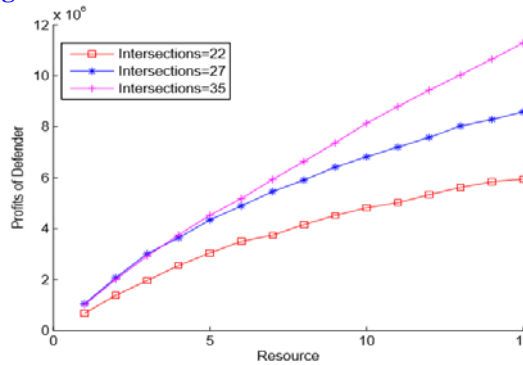


Fig. 5. Influence of intersections on profits of defender

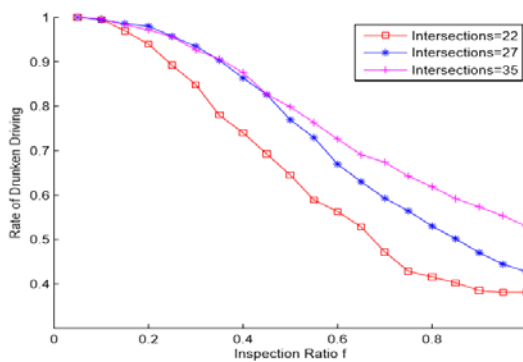


Fig. 6. Influence of f on rate of drunk driving

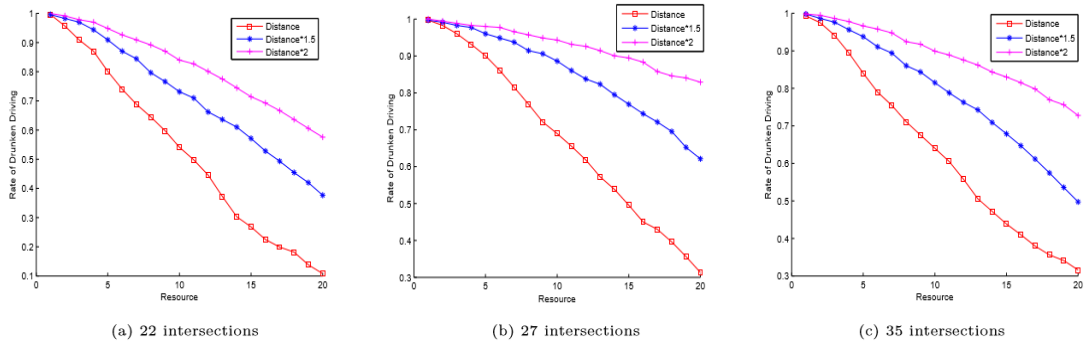


Fig. 7. Influence of distances on rate of drunk driving

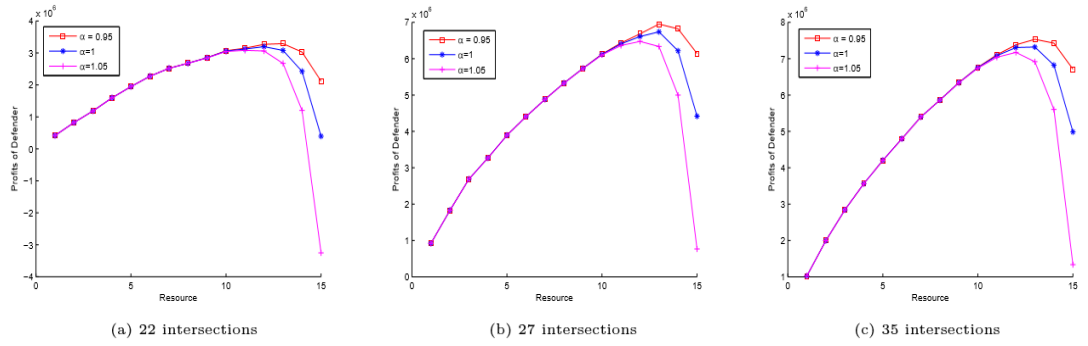


Fig. 8. Influence of α on profits of defender

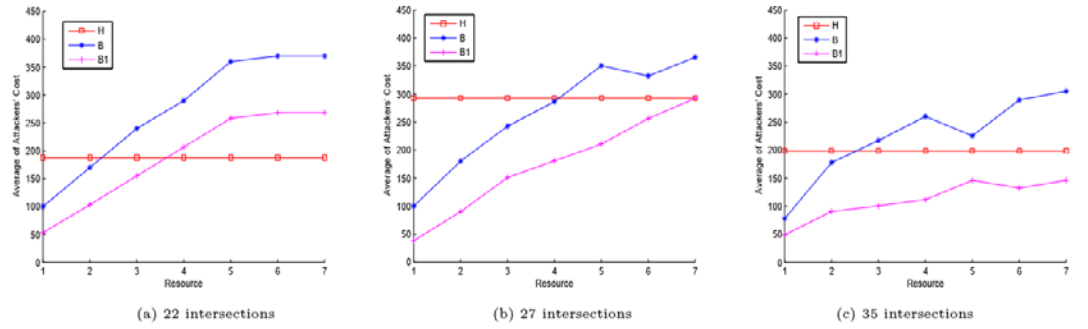


Fig. 9. Influence of leakage message on average of attackers' cost

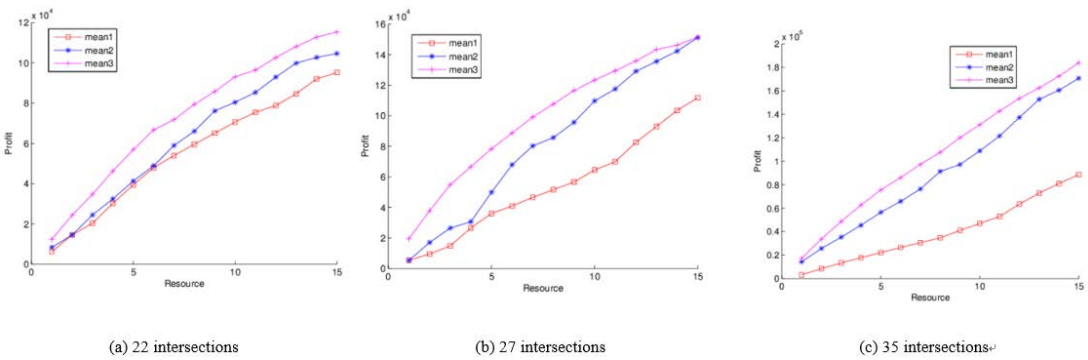


Fig. 10. Influence of solution' quality on profits of defender

6. Conclusion

In this paper, for the sake of reinforcing the power of inspecting drunk driving to crack down on such illegal actions, we propose an approach named OISDD to allocate limited resources to the traffic network and generate optimal schedules of inspecting drunk driving. We explore influence on OISDD based on the leakage of inspection messages, heavy traffic causing by inspections, waste of resources and so on. Experimental results demonstrate that OISDD is feasible and efficient.

Acknowledgement

This paper is supported by the Nature Science Foundation of China under grant No.61572095, No. 61501080 and the Fundamental Research Funds for the Central Universities' under No.D UT16QY09.

References

- [1] H. Von, *Marktform und Gleichgewicht*, 2nd Edition, Springer, New York, 1934.
[Article \(CrossRef Link\)](#).
- [2] J. Pita, M. Jain, J. Marecki, F. Ordonez, C. Portway, M. Tambe, C. Western, P. Paruchuri and S. Kraus, "Deployed Armor protection: the application of a game theoretic model for security at the Los Angeles International Airport," in *Proc. of 17th International Joint Conference on Autonomous Agents and Multiagent Systems*, pp. 125-132, May 12-16, 2008.
[Article \(CrossRef Link\)](#).
- [3] J. Tsai, S. Rathi, C. Kiekintveld, F. Ordonez and M. Tambe, "IRIS-a tool for strategic security allocation in transportation networks," in *Proc. of 18th International Joint Conference on Autonomous Agents and Multiagent Systems*, pp. 37-44, May 10-15, 2009.
[Article \(CrossRef Link\)](#).
- [4] E. Shieh, B. An, R. Yang, M. Tambe, C. Baldwin, J. DiRenzo, B. Maule and G. Meyer, "Protect: an application of computational game theory for the security of the ports of the United States," in *Proc. of 26th Association for the Advance of Artificial Intelligence*, pp. 2173-2179, July 22-26, 2012. [Article \(CrossRef Link\)](#).
- [5] B. An, F. Ordonez, M. Tambe, E. Shieh, R. Yang, C. Baldwin, J. DiRenzo, K. Moretti, B. Maule and G. Meyer, "A deployed quantal response-based patrol planning system for the U.S. coast guard," *Interfaces*, vol. 43, no. 5, pp. 400-420, 2013. [Article \(CrossRef Link\)](#).
- [6] T. Nguyen, A. Sinha, S. Gholami, A. Plumptre, L. Joppa, M. Tambe, M. Driciru, F. Wanyama, A. Rwetsiba, R. Critchlow and C. Beale, "CAPTURE: A new predictive anti-poaching tool for wildlife protection," in *Proc. of 25th International Joint Conference on Autonomous Agents and Multiagent Systems*, pp. 509-540, May 9-13, 2016. [Article \(CrossRef Link\)](#).
- [7] F. Fang, T. Nguyen, R. Pickles, W. Lam, G. Clements, B. An, A. Singh, M. Tambe and A. Lemieux, "Deploying paws: Field optimization of the protection assistant for wildlife security" in *Proc. of 28th Annual Conference on Innovative Applications of Artificial Intelligence*, pp. 3966-3973, February 12-17, 2016. [Article \(CrossRef Link\)](#).
- [8] Z. Yin, A. Jiang, M. Tambe, C. Kiekintveld, K. Leyton-Brown, T. Sandholm and J. Sullivan, "Trusts: scheduling randomized patrols for fare inspection in transit systems using game theory" in *Proc. of 24th Annual Conference on Innovative Applications of Artificial Intelligence*, pp. 2348-2355, July 22-26, 2012. [Article \(CrossRef Link\)](#).
- [9] M. Brown, S. Saisubramanian, P. Varakantham and M. Tambe, "STREETS: game-theoretic traffic patrolling with exploration and exploitation," in *Proc. of 28th Association for the Advance of Artificial Intelligence*, pp. 2966-2971, July 27-31, 2014. [Article \(CrossRef Link\)](#).

- [10] J. Niu, W. Tang and W. Guo, "Coalition formation game based relay selection and frequency sharing for cooperative relay assisted wireless D2D networks with QoS constraints," *KSII Transactions on Internet and Information Systems*, vol. 10, no. 11, pp. 5253-5270, 2016. [Article \(CrossRef Link\)](#).
- [11] Lin, M. and Chen, Y, "Performance analysis of buffer aware scheduling for video services in LTE network," *KSII Transactions on Internet and Information Systems*, vol. 9, no. 9, pp. 3594-3610, 2015. [Article \(CrossRef Link\)](#).
- [12] V. Conitzer and T. Sandholm, "Computing the optimal strategy to commit to," *EC*, pp. 82-90, 2006. [Article \(CrossRef Link\)](#).
- [13] P. Varaiya, "Max pressure control of a network of signalized intersections," *Transportation Research Part C: Emerging Technologies*, vol. 36, pp. 177-195, 2013. [Article \(CrossRef Link\)](#).
- [14] G. Becker and W. Landes, *Essays in the economics of crime and punishment*, Columbia University, 1974. [Article \(CrossRef Link\)](#).
- [15] M.T. Fillmore, J.S. Blackburn and L.R. Harrison, "Acute disinhibiting effects of alcohol as a factor in risky driving behavior," *Drug and Alcohol Dependence*, vol. 95, pp. 97-106, 2008. [Article \(CrossRef Link\)](#).
- [16] M.E. Rakauskas, N.J. Ward, E.R. Boer, E.M. Bernat, M. Cadwallader and C.J. Patrick, "Combined effects of alcohol and distraction on driving performance," *Accident Analysis and Prevention*, vol. 40, pp. 1742-1749, 2008. [Article \(CrossRef Link\)](#).
- [17] D. Korzhuk, V. Conitzer and R. Parr, "Complexity of computing optimal Stackelberg strategies in security resources allocation games," in *Proc. of 24th Association for the Advance of Artificial Intelligence*, pp. 805-810, July 11-15, 2010. [Article \(CrossRef Link\)](#).
- [18] J. Ponsard and S. Sorin, "The L.P. formulation of finite zero-sum games with incomplete information," *International Journal of Game Theory*, vol. 9, pp. 99-105, 1980. [Article \(CrossRef Link\)](#).
- [19] D. Koller, N. Megiddo and B. von Stengel, "Fast algorithms for finding randomized strategies in game trees," in *Proc. of 26th Annual ACM Symposium on Theory of Computing*, pp. 750-760, May 23-25, 1994. [Article \(CrossRef Link\)](#).
- [20] P. Paruchuri and P. Pearce, "Playing games for security: an efficient exact algorithm for solving bayesian stackelberg games," in *Proc. of 17th International Joint Conference on Autonomous Agents and Multiagent Systems*, pp. 895-902, May 12-16, 2008. [Article \(CrossRef Link\)](#).
- [21] C. Kiekintveld and M. Jain, "Computing optimal randomized resource allocations for massive security games," in *Proc. of 18th International Joint Conference on Autonomous Agents and Multiagent Systems*, pp. 689-696, May 10-15, 2009. [Article \(CrossRef Link\)](#).
- [22] E.W. Dijkstra, "A note on two problems in connexion with graphs," *Numerische Mathematik*, vol. 1, pp. 269-271, 1959. [Article \(CrossRef Link\)](#).
- [23] H.W. Kuhn, "The Hungarian method for the assignment problem," *Naval Research Logistics Quarterly*, vol. 2, pp. 83-97, 1955. [Article \(CrossRef Link\)](#).
- [24] J. Smith and F. Cruz, "State dependent travel time models," *Physica A Statistical Mechanics & Its Applications*, vol. 395, no. 4, pp. 560-579, 2014. [Article \(CrossRef Link\)](#).



Yingmo Jie received the B.S. degree in information and computing science from Tianjin University of Technology and Education in 2011. She received the M.S. degree in 2015 in applied mathematics from Civil Aviation University of China, Tianjin, China. Since 2015, she has been working for the Ph.D degree in the School of Mathematical Sciences at the Dalian University of Technology. Her current research interests include information security, resources allocation and Game Theory.



Mingchu Li received the B.S. degree in mathematics, Jiangxi Normal University and the M.S. degree in applied science, University of Science and Technology Beijing in 1983 and 1989, respectively. He worked for University of Science and Technology Beijing in the capacity of associate professor from 1989 to 1994. He received his doctorate in Mathematics, University of Toronto in 1997. He was engaged in research and development on information security at Longview Solution Inc, Compuware Inc. from 1997 to 2002. From 2002, he worked for School of Software of Tianjin University as a full professor, and from 2004 to now, he worked for School of Software Technology of Dalian University of Technology as a full Professor, Ph.D. supervisor, and vice dean. His main research interests include theoretical computer science and cryptography.



Tingting Tang received his B.E. degree in School of Software Technology from Dalian University of technology, Dalian, Liaoning, P.R. China, in June 2017. She is currently pursuing a master degree at Software Technology from Dalian University of technology. Her current research interests include game theory.



Cheng Guo received the B.S. degree in computer science from Xi'an University of Architecture and Technology in 2002. He received the M.S. degree in 2006 and his Ph.D in computer application and technology, in 2009, both from the Dalian University of Technology, Dalian, China. From July 2010 to July 2012, he was a post doc in the Department of Computer Science at the National TsingHua University, Hsinchu, Taiwan. Since 2013, he has been an associate professor in the School of Software Technology at the Dalian University of Technology. His current research interests include information security, cryptology and cloud security.