Timing and Security Analysis of VANET-based Intelligent Transportation Systems

(Invited Paper)

Bowen Zheng
University of California, Riverside
Riverside, California
bzheng@ece.ucr.edu

Muhammed O. Sayin*
University of Illinois at Urbana-Champaign
Urbana, Illinois
sayin2@illinois.edu

Chung-Wei Lin
Toyota InfoTechnology Center
Mountain View, California
cwlin@us.toyota-itc.com

Shinichi Shiraishi
Toyota InfoTechnology Center
Mountain View, California
sshiraishi@us.toyota-itc.com

Qi Zhu
University of California, Riverside
Riverside, California
qzhu@ece.ucr.edu

ABSTRACT

With the fast development of autonomous driving and vehicular communication technologies, intelligent transportation systems that are based on VANET (Vehicular Ad-Hoc Network) have shown great promise. For instance, through V2V (Vehicle-to-Vehicle) and V2I (Vehicle-to-Infrastructure) communication, intelligent intersections allow more fine-grained control of vehicle crossings and significantly enhance traffic efficiency. However, the performance and safety of these VANET-based systems could be seriously impaired by communication delays and packet losses, which may be caused by network congestion or by malicious attacks that target communication timing behavior. In this paper, we quantitatively model and analyze some of the timing and security issues in transportation networks with VANET-based intelligent intersections. In particular, we demonstrate how communication delays may affect the performance and safety of a single intersection and of multiple interconnected intersections, and present our delay-tolerant intersection management protocols. We also discuss the issues of such protocols when the vehicles are non-cooperative and how they may be addressed with game theory.

CCS CONCEPTS

• **Security and privacy → Systems security**; • **Computer systems organization → Embedded and cyber-physical systems**; **Real-time systems**;

KEYWORDS

Intelligent Transportation Systems, Security, Timing, VANET

1 INTRODUCTION

VANET (Vehicular Ad-Hoc Network) is a mobile ad-hoc network aiming to provide fast and efficient communication among vehicles and roadside infrastructures [17, 26]. It has shown great promise for facilitating intelligent transportation as vehicles and infrastructures may share information such as speed, location, acceleration and traffic condition to enhance transportation safety and efficiency.

One VANET-based intelligent transportation application is intersection management. The idea is to replace traffic signals with communication messages among vehicles and infrastructures to coordinate vehicles' crossing of the intersection. Such VANET-based intelligent intersection management can be either centralized or distributed [11]. In centralized intersection management, an intersection manager accepts the requests from approaching vehicles and decides the order for vehicles to cross the intersection. In distributed intersection management, vehicles negotiate among themselves and reach consensus of the order to cross the intersection. Both systems have shown better performance compared with traditional traffic signals, under the assumption that vehicular communication is instantaneous and there are no insider or outsider attacks [3, 5–7, 11, 12, 16, 19, 20, 29].

DSRC (Dedicated Short-Range Communication) [18] is a candidate standard for the realization of VANET. Although DSRC provides security mechanisms for information confidentiality (such as encryption and authentication standards) and congestion control mechanisms, the availability issue is difficult to eliminate as wireless communication is susceptible to environment change and interference. Packet delays and losses can be significant in dense traffic and are shown to have negative impact on safety applications [10, 14, 25]. Furthermore, attackers may perform *timing attacks*, by flooding or jamming the communication channels to increase packet delays and losses [9, 28], which could lead to catastrophic outcomes for delay-sensitive applications such as intersection management.

In our previous work [27], we proposed a delay-tolerant intersection management protocol to guarantee the safety and liveness properties for four-way *single-lane* intersections. We also observed that the intersection performance, measured by the average latency for vehicle crossings, may significantly deteriorate under increasing communication delays.

In this paper, we broaden the scope to four-way *multi-lane* intelligent intersections, extend the delay-tolerant protocol, and consider both a single intersection and multiple interconnected intersections (i.e., a small transportation network). We model, analyze and simulate the timing and security issues of such intersections, in particular the relation between communication delays and the performance deterioration. The simulations are based on SUMO [1], a

*The author was also with Toyota InfoTechnology Center.

978-1-5386-3093-8/17/$31.00 ©2017 IEEE

widely used traffic simulation suite, with our extension for modeling packet transmission and delay.

In addition to timing attacks that increase communication delays, we also discuss another security issue for intelligent intersections: a self-centered or even malicious vehicle could try to mislead the intersection manager to take actions in favor of the vehicle, e.g., by providing false time tags to let it cross the intersection prior to others. From a game theoretical perspective, such vehicles that operate solely based on their own preferences could be considered as players of a non-cooperative game and thus managed with the concept of Nash equilibrium [8]. We envision the usage of game theory to formally analyze strategic level security of VANET-based applications, and discuss how to guarantee that individual vehicles will not have incentives to strategically exploit the system.

The rest of the paper is organized as follows. In Section 2, we introduce some of the related work on intelligent intersection management, including our own work on delay-tolerant protocol for four-way single-lane intersections. In Section 3, we introduce our model for a single intersection with multiple lanes, and discuss the impact of communication delays and timing attacks. In Section 4, we present our model and analysis for multiple intersections. In Section 5, we discuss how game theory may be applied to address strategic level security of intelligent intersections. Section 6 presents the experimental results and Section 7 concludes the paper.

2 RELATED WORK

2.1 VANET-based Intersection Management

In the literature, researchers have proposed various protocols and strategies for both centralized and distributed intelligent intersection management [3, 5–7, 12, 16, 19, 20, 29]. However, these works lack sufficient consideration of communication delays and losses.

For centralized intersections management, a common idea is to discretize an intersection as grids and assign the grids to crossing vehicles at different time slots [6, 7, 12, 16, 19, 29]. For instance, the fine-grained approach in [12, 16] allows vehicles to enter the intersection as long as their routes can be separated based on time slots. Such approach requires the intersection manager to accurately estimate vehicle movements and every vehicle to follow the assigned time slots precisely in time. This assumption could be susceptible to timing attacks that target communication delays. The authors in [6, 7, 19] formally prove properties such as safety, liveness and deadlock free, however do not consider communication delays and losses. In [29], the intersection management is abstracted as a linear programming problem with flow and conflicting points as inputs, but again without consideration of communication delays.

For distributed intersection management, the negotiation among vehicles could also be complicated by packet delays and losses. For instance, in [5], every vehicle broadcasts enter, cross and exit packets with trajectory cell list, and therefore the collision detection algorithm depends on reliable and in-time communication among vehicles. In [3, 20], Timed Petri Nets are used to prove deadlock-free, without consideration of potential timing attacks.

Delay-tolerant protocol: In our previous work [27], we propose a delay-tolerant protocol for four-way intersections with a single lane on each direction. We adopt the centralized structure and use a less aggressive approach, i.e., our protocol does not allow vehicles with conflicting (intersecting) paths to enter the intersection at the same time.

In our protocol, three types of messages are defined: *Requet*, *Confirm* and *Cancel*. Every message contains a time tag that indicates the sending time. The *Request* message is sent from a vehicle to the intersection manager to acquire permission for entering the intersection. Without permission, the vehicle should stop before the intersection. An estimated arriving time and vehicle destination are included in *Request* to help the intersection manager schedule vehicles. The *Confirm* message is sent from the intersection manager to give permission to a vehicle. Inside *Confirm*, a time window $[T_L, T_H]$ is allocated to the corresponding vehicle. If the vehicle can reach the intersection within the time window, it may enter with its safety guaranteed. Otherwise, it should not enter the intersection and may send an optional *Cancel* message.

In order to address communication delays, we define three types of timeouts. 1) timeout for message transmission, i.e., a message becomes invalid after this amount of time, 2) timeout for resend, i.e., a vehicle will resend *Request* if *Confirm* is not received within this amount of time, and 3) timeout for wait, i.e., the intersection manager will wait for a vehicle that has been sent the *Confirm* message to take action for this amount of time.

With messages and timeouts as defined above, we model the behavior of vehicles and intersection manager as finite state machines. The properties of deadlock-free, safety (vehicles with conflicting routes will not enter the intersection at the same time), and liveness (every vehicle will cross the intersection as long as the communication delays are bounded) are then proved by translating the state machines to timed automata models and using the formal verification tool UPPAAL [2]. We also model and simulate our protocol in SUMO, and observe that the intersection performance may deteriorate significantly with the increase of communication delays.

In this paper, we further extend the model from single-lane single-intersection to multi-lane single-intersection and multi-lane multi-intersection, analyze the impact of communication delays, and discuss related security issues.

2.2 Game Theory

With V2V and V2I communication, vehicles and roadside units form a multi-agent system, in which they may communicate with each other to improve traffic safety and efficiency. However, self-centered or even malicious agents may have incentive not to cooperate due to their own distinct objectives. In that case, game theory, which studies the interaction among intelligent rational decision makers [8], can provide formal tools for analyzing the strategic behavior of the agents and for designing strategy-proof control mechanisms [24] that are robust against such issues.

More specifically, in a non-cooperative multi-agent system, agents may have different preferences over the system outcomes and then take actions accordingly. From a game theoretical perspective and in a non-trivial game formulation, the players can have conflicting preferences over the system outcomes but a player cannot take his/her best action independent of other players' actions, i.e., the actions are coupled even though the players are not cooperating with each other.

Nash equilibrium is a solution concept over the actions of self-centered players such that no player has any incentive to change his/her action unilaterally since any other actions cannot lead to a more preferred outcome [8]. However, such Nash equilibrium may not exist in a game (e.g., in game of matching pennies) or may not be unique (e.g., in battle of sexes) [8]. A fundamental result in game theory shows that in finite games, where players have finite number of actions, there always exists a mixed-strategy Nash equilibrium, where players choose their actions from their action sets with respect to certain probability distribution.

In addition to analyzing the equilibrium solutions of a given game, we can design the game structure such that even the strategic players take actions in a designed way. This reverse engineering approach is studied in the field of mechanism design in game theory [24]. In particular, there exists a player (called the principal) controlling the game structure (e.g., designing the game outcome given other players' actions), while the other players having access to certain private information can take their actions strategically. Another fundamental result in game theory shows that in a Vickrey-Clarke-Groves mechanism [24], players do not have any incentive to be strategic irrespective of other players' behaviors since the players are charged by a monetary amount based on their impact on other agents' game outcomes (by being present in the game).

We emphasize that game theory is a very rich field of analytically powerful tools to analyze and design behavior of strategic agents alike the aforementioned fundamental ones. We may use these tools to formally analyze connected vehicle applications with self-centered or malicious vehicles (agents), instead of simply assuming absolute cooperation or relying on heuristics. As an example, in [13], the authors have proposed an intersection management approach inspired by the chicken-game, where two drivers drive towards each other and either one of them will swerve or there will be an accident [15]. The proposed game includes two players, where the players aim to minimize their delay while also avoiding any collision; and the intersection manager controls their actions to achieve a Nash equilibrium of the game through signaling as in correlated equilibrium [4]. In [23], the authors propose a combinatorial auction based approach, in which the intersection is discretized into partitions and allocated to the agents who value the most by bidding highest for their desired partitions. However, combinatorial auctions are computationally challenging, i.e., NP-hard in general [24], and therefore the authors in [23] propose an approximated algorithm.

3 MULTI-LANE SINGLE INTERSECTION

We will first present our model for a single intersection with multiple lanes from each direction, and then discuss the challenging issues of communication delays and timing attacks. To address these challenges, we have extended our delay-tolerant protocol in [27], which only considers a single lane from each direction and assumes a much simpler model.

3.1 Model

There are some fundamental assumptions in our system:

- There is an intersection manager in the intersection. It receives requests from vehicles, schedules them, and sends confirmations to vehicles.
- All vehicles and the intersection manager are connected (if some vehicles are non-connected, roadside sensors and lane-specific traffic lights are required).
- All vehicles should follow instructions from the intersection manager, regardless of whether a vehicle is controlled autonomously or by a human driver.
- All vehicles have basic safety systems such as pre-collision systems or lane departure alerts as the final safety features. However, the intersection manager should still schedule vehicles to avoid collisions for better safety and efficiency.

Then, we define the system model for single-intersection multi-lane systems as follows:

- An intersection has a set of ways, $\mathcal{W} = \{\omega_1, \omega_2, \ldots, \omega_{|\mathcal{W}|}\}$ and a set of feasible paths, $\mathcal{P} = \{\pi_1, \pi_2, \ldots, \pi_{|\mathcal{P}|}\}$.
- Each way ω_i has a set of lanes, $\mathcal{L}_i = \{\lambda_{i1}, \lambda_{i2}, \ldots, \lambda_{i|\mathcal{L}_i|}\}$.
- Each feasible path π_k is an ordered pair of lanes $(\lambda_{ij}, \lambda_{i'j'})$.
- Each pair of feasible paths is either non-conflicted or conflicted, which is pre-defined based on the physical design of the intersection.
- Each request of a vehicle includes its current lane, a set of destination lanes, and other information that may include estimated arrival time, earliest arrival time, desired arrival time, and whether the vehicle is the first vehicle in its current lane.
- The manager will process a set of requests, send a set of confirmations, and keep unprocessed requests for the next time.
- Each confirmation is associated with a request. It includes a set of feasible paths and a time window for the corresponding vehicle to enter the intersection.

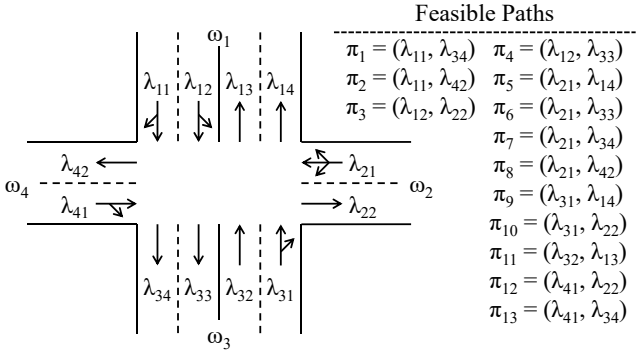

Figure 1: An example showing the intersection model.

An example is shown in Figure 1 where π_3 and π_{13} are non-conflicted and π_3 and π_{10} are conflicted. There are some hard constraints:

- The destination lane of each feasible path in a confirmation must be in the set of destination lanes in the corresponding request.
- The time windows in a confirmation must be after the earliest arrival time in the corresponding request.

- Each pair of confirmations must satisfy at least one of the following conditions: (1) the time window of one confirmation does not overlap with the time window of the other confirmation; (2) all of the feasible paths of one confirmation are non-conflicted with all feasible paths of the other confirmation.

Note that the last constraint is to provide either temporal or spatial separation for safety. As we stated before, here we use a less aggressive approach under the consideration of communication delays and losses (unlike previous methods where vehicles with conflicting paths may enter the intersection at the same time and only get separated with fine-grained scheduling that is vulnerable to timing attacks). We also make this assumption based on practical consideration of vehicle passengers' mental acceptance. Furthermore, there are other constraints that can be added:

- The time windows in a confirmation must be after the estimated arrival time in the corresponding request.
- If there are multiple feasible paths in a confirmation, the corresponding vehicle can decide and select one of them. If vehicles do not have this functionality, the manager can send a confirmation with only one feasible path.

3.2 Challenges from Communication Delays and Timing Attacks

There are many potential security risks to vehicles, and protection mechanisms at different levels have been discussed, such as

- *external interface* with secure communication protocols integrated with existing standards and protocols,
- *gateways* with intrusion detection systems (IDSs) and firewalls,
- *in-vehicular networks* with lightweight message authentication and encryption, and
- *components* with hardware security modules (HSMs), secure boot, and secret key management.

However, these protections do not cover *timing attacks*, where the attackers may try to compromise/disrupt the operation (most likely communication) timing. In many cases, a timing attack can be easily performed without the need to completely jam the communication channels. Below we discuss some of the major challenges in communication delays and timing attacks.

Delaying a request: The intersection manager can only schedule a vehicle after the corresponding request is received. Therefore, delaying a request may lead to (but is not limited to) the following scenarios. 1) Vehicle A arrives at the intersection earlier, however, due to the delay of the request, another request from vehicle B is delivered earlier. Therefore, the intersection manager has scheduled vehicle B prior to vehicle A. No matter the delay of the request from vehicle A is short or long, performance will decrease. If the delay is short, the intersection manager has to consider the request during the next round of scheduling. If the delay is long, the vehicle has to resend a request after a preset timeout. 2) For vehicles aligned in a line, it is possible that the request from vehicle A, which is physically in front of vehicle B, arrives later than the request of vehicle B. In this case, the intersection manager may send confirmation to vehicle B without sending confirmation to vehicle A. Since vehicle A is physically ahead of vehicle B, vehicle B is not able to cross the intersection even with the confirmation. As a result, the intersection manager has to wait for the time window for vehicle B to expire before it schedules other vehicles with conflicting routes. Consider a scenario that seven vehicles are aligned in a line, and only the request of the first vehicle is significantly delayed. Therefore, all the vehicles except for the first vehicle obtain the confirmation, but no vehicle can enter the intersection. The intersection manager has to wait for the time windows of all the six vehicles behind the first one to expire, which can greatly decrease performance.

Delaying a confirmation: A vehicle can only enter the intersection after receiving a confirmation and reaching the intersection within the valid time window. There are scenarios in which delaying a confirmation may lead to congestion: 1) The confirmation is significantly delayed and delivered after the time window has expired. In this case, the vehicle has to resend a request and wait for another round of scheduling. 2) For vehicles aligned in a line, delaying the confirmation to the first vehicle can block its following vehicles from entering the intersection even with the confirmation. In this case, all the vehicles have to wait for the confirmation to expire and resend the requests. Sending a cancellation message can also be tricky in this case. If the cancellation is sent from vehicles to the intersection manager, it is difficult for a vehicle to decide when a cancel is needed, as the vehicles behind the first vehicle dose not know whether the first vehicle successfully obtains the confirmation, and its time to arrive at the intersection highly depends on its proceeding vehicles. If the cancellation is sent from the intersection manager, it can even be dangerous, as the cancellation itself can also be delayed or lost. For example, the intersection manager has scheduled another vehicle B after sending cancellation to a previous confirmed vehicle A, however, the former delayed confirmation for A arrives but the cancellation has not yet arrived. As a result, two conflicting vehicles enters the intersection.

In the above scenarios, if the intersection management is not designed well, timing attacks can create significant traffic delays and congestions. In the worst case, no vehicle is able to go through the intersection. We have extended our delay-tolerant protocol in [27] to address these challenges. Due to space limitation, we will not present the detailed new protocol here, but some of the main design guidelines are:

- **Delay estimation**: Accurate estimations of communication delays can greatly facilitate the system operation. For example, if the delay is large, the time window inside the confirmation should also correspondingly be elongated to give the vehicle more time to arrive. In principle, all the time related parameters should be set according to the delay estimation. A good design should guarantee both safety and performance with accurate delay estimations, and at least safety under inaccurate delay estimations.
- **Using timeout to resend request**: Resending a request is necessary when the request or the corresponding confirmation is delayed or lost. It is the only way to reconnect with the intersection manager no matter whether the communication was successful. The difficulty lies in setting the value of the timeout. If the value is too small, the condition of the channel being attacked can be even more deteriorated. If the value is too large, the performance of the system will decrease.

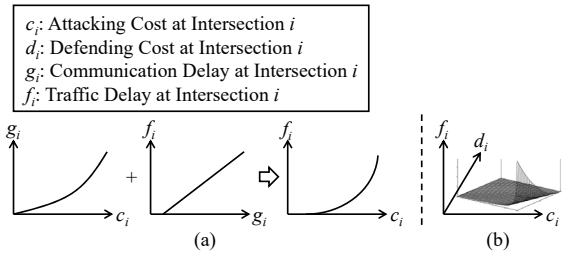

Figure 2: The traffic delay as (a) a function of attacking cost and (b) a function of attacking cost and defending cost.

- **Not using cancellation for confirmation**: Sending cancellation can be dangerous, as the cancellation itself can also be delayed or lost. The unsuccessful delivery of a cancellation can lead to the corresponding vehicle entering the intersection without knowing its time window has become invalid and not safe.

4 MULTIPLE INTERSECTIONS

We then consider a small traffic network with multiple interconnected intersections. These intersections can be modeled as a graph and defined as follows:

- Each system of multiple intersections is defined by a graph $(\mathcal{I}, \mathcal{E})$, where $\mathcal{I}$ is the vertex set and $\mathcal{E}$ is the edge set.
- Each vertex in $\mathcal{I}$ is an intersection.
- Each edge in $\mathcal{E}$ is between two intersections.

Based on the proposed analysis introduced above, we can define traffic delay as a function of attacking cost. Usually, it is a composition of a function from attacking cost to communication delay and a function from communication delay to traffic delay, as shown in Figure 2 (a). If defending cost is considered, the traffic delay can be defined as a function of attacking cost and defending cost, as shown in Figure 2 (b). Some definitions are as follows:

- c_i: the attacking cost (expense) at Intersection i.
- d_i: the defending cost (expense) at Intersection i.
- f_i: the traffic delay at Intersection i.
- w_i: the weight of Intersection i.
- C: the total resource (capacity) of attackers.
- D: the total resource (capacity) of defenders.

f_i should be increasing to c_i and decreasing to d_i. The weights can be used to consider the whole traffic network or just some part (like a path) of the traffic network. If the system and its protection are pre-defined (therefore d_i is assumed to be given and f_i is assumed to be independent from d_i), the optimization problem from the perspective of attackers is as follows:

$$\max_{\mathbf{c}} \quad \sum_{i=1}^{n} w_i f_i(c_i), \tag{1}$$

$$\text{such that} \quad \sum_{i=1}^{n} c_i \le C. \tag{2}$$

On the other hand, if the system and its protection are not predefined, the optimization problem from the perspective of defenders is as follows:

$$\min_{\mathbf{d}} \max_{\mathbf{c}} \quad \sum_{i=1}^{n} w_i f_i(c_i, d_i), \tag{3}$$

$$\text{such that} \quad \sum_{i=1}^{n} c_i \le C \text{ and } \sum_{i=1}^{n} d_i \le D, \tag{4}$$

and the optimization problem from the perspective of attackers is as follows:

$$\max_{\mathbf{c}} \min_{\mathbf{d}} \quad \sum_{i=1}^{n} w_i f_i(c_i, d_i), \tag{5}$$

$$\text{such that} \quad \sum_{i=1}^{n} c_i \le C \text{ and } \sum_{i=1}^{n} d_i \le D. \tag{6}$$

These two problems could lead to a strategic game between defenders and attackers. We have not solved this problem but have preliminary studies using game theory. Details are described in the next section.

5 GAME THEORY ANALYSIS

As stated at the beginning of Section 3.2, there are different layers in addressing automotive security concerns. Those security protocols, mechanisms and modules can only protect against outsider attackers, as shown in Figure 3. They are not sufficient for insider attackers and application-layer security vulnerability that is due to the *strategic behavior of the self-centered or malicious agents*. Therefore, security should also be considered from the perspective of the application layer such that intelligent agents do not have any incentive to exploit the system strategically.

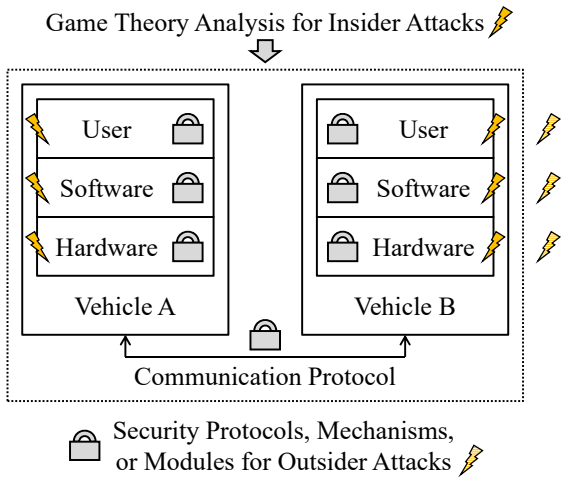

Figure 3: The comparison between insider attacks and outsider attacks.

5.1 Examples of Applying Game Theory for Automotive Security

In the following, we discuss two examples about the intelligent intersection management in non-cooperative environments:

A Non-Cooperative Intersection Game Between Two Intelligent Agents: [22] introduces an intersection game formulation. Different from the chicken game, they consider an intersection management scenario, where two non-cooperative autonomous agents (vehicles) seek to use a single intersection resource at a *specific* time they desire. However, there is a certain amount of time that the agents will need while using the intersection resource, and there can be a conflict between their desired intersection usage. Particularly, the agents disclose their desired passing times through the intersection to the intersection roadside units, and the roadside units schedule the temporal intersection usage based on the reported information by the agents. For such a scenario, the authors analyze strategic behaviors of the agents and formulate Nash equilibria. Importantly, in general, the agents play strategically in case of conflicting interests by misreporting their desired passing times. Furthermore, there exist multiple Nash equilibria. Therefore, the authors identify the socially optimal equilibrium, with respect to certain social objective, and correspondingly propose a strategy-proof intersection mechanism in which agents disclose their private information truthfully.

Information-Driven Intersection Management with Accuracy Guarantee: [21] proposes an information-driven intersection management mechanism through prioritization among the vehicles in intersection usage based on the information provided by them, while ensuring truthful information disclosure via a payment system. The authors introduce a time-token accounting system, where each driver (or vehicle) has a time-token balance that he/she can use while passing through intersections. They propose a payment system based on the VCG mechanism, which theoretically ensures that truthful information disclosure is the dominant strategy for all the agents. In particular, each vehicle in close proximity of an intersection sends a reservation request signal including basic safety information (which is essential for the reservation of the intersection safely and cannot be misreported strategically) and a driver-exclusive utility function (which specifies the preferences of the driver over different reservations). That driver-exclusive content of the signal is used for prioritization of the requests yet can be misreported strategically. Therefore, the manager charges each driver (or vehicle) with certain amount of time-tokens, which depends on the impact of their reported utility functions on other drivers' reservations so that the drivers have no incentive to misreport that information. The authors also propose a base time-token payment from the intersection roadside unit to the drivers (or vehicles) such that the unit neither loses nor earns time-tokens over time.

5.2 Preliminary Directions

We note that these two examples are mainly related to autonomous intersections with managers, however, rich analytical tools of game theory could also be used in various other parts of connected vehicle networks and systems. We are exploring a few possible research directions as follows:

- characterizing equilibrium for autonomous intersections without managers,
- analyzing the impact of a malicious agent on other agents' quality of transportation for security guarantees,
- formulating traffic usage mechanisms via pricing in order to incentivize drivers to carpool, and
- "strategic" real-time traffic reporting in order to prevent possible future congestions due to the drivers aiming to minimize their traveling time solely.

Finally, each driver may train his/her autonomous vehicle to learn the environment by identifying the encountered objects and a pricing mechanism can incentivize the drivers to identify the objects accurately for efficient trainings.

6 SIMULATION

In this section, we use simulations to demonstrate the impact of communication delays and timing attacks on the performance of intelligent intersection management. The simulations are based on our extension of the SUMO tool. We add explicit modeling of communication delays by defining message classes and channel classes with the methods to send and receive the messages. In case of timing attack, the channel will add a delay to the message according to the level of the attack. The behavior of vehicles and intersection manager is implemented in the vehicle class and the intersection manager class, respectively. By leveraging the TraCI API from SUMO, both classes are able to control the movement of the vehicles. By calling the communication interface we have defined, sending and receiving messages are possible. Simulations in SUMO are based on time steps, therefore the TraCI API is able to obtain information, such as speed, acceleration and location, and change the values of speed and location at each time step. The SUMO simulation environment is shown in Figure 4.

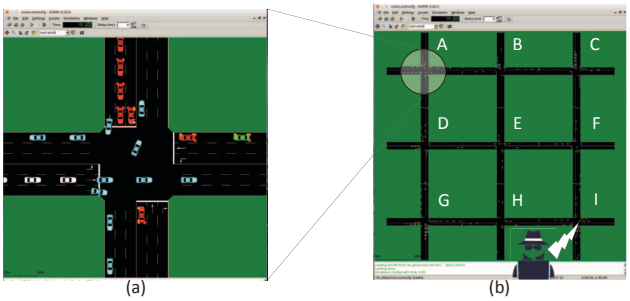

Figure 4: Simulations based on SUMO.

6.1 Single Intersection with Multiple Lanes

In this experiment, we study the communication delay and its impact on the performance of a single intersection with multiple lanes. The system setup is shown in Figure 5. The intersection has four ways ω_1, ω_2, ω_3 and ω_4, and each direction is associated with three lanes, namely, one lane for left turn, one lane for going straight and one lane for right turn. Therefore, the feasible paths for the system are listed as follows: $\pi_1 = (\lambda_{11}, \lambda_{46})$, $\pi_2 = (\lambda_{12}, \lambda_{35})$, $\pi_3 = (\lambda_{13}, \lambda_{24})$, $\pi_4 = (\lambda_{21}, \lambda_{16})$, $\pi_5 = (\lambda_{22}, \lambda_{45})$, $\pi_6 = (\lambda_{23}, \lambda_{34})$, $\pi_7 = (\lambda_{31}, \lambda_{26})$, $\pi_8 = (\lambda_{32}, \lambda_{15})$, $\pi_9 = (\lambda_{33}, \lambda_{44})$, $\pi_{10} = (\lambda_{41}, \lambda_{36})$, $\pi_{11} = (\lambda_{42}, \lambda_{25})$, $\pi_{12} = (\lambda_{43}, \lambda_{14})$. The length of each lane is 100 meters. In this simulation, vehicles are assumed to have a length

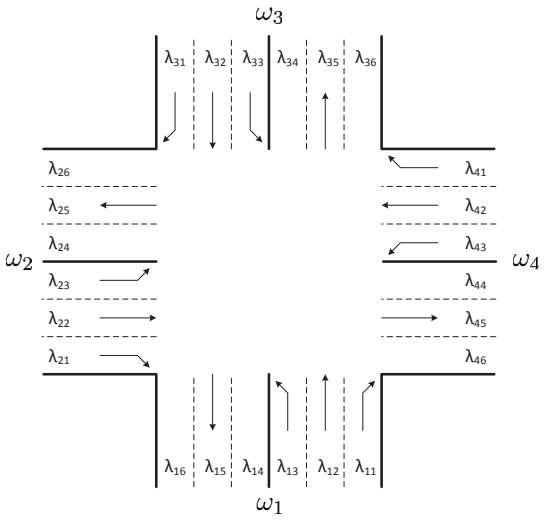

Figure 5: Setup for single-intersection simulations.

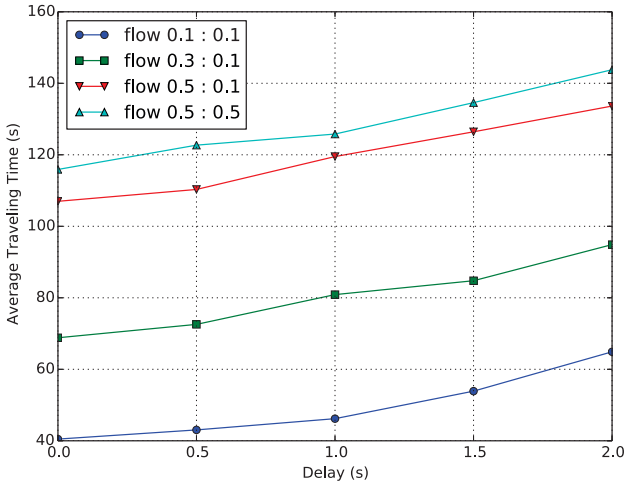

Figure 6: Performance of a single intersection under different communication delays.

of 5 meters, with maximum acceleration 0.8 m/s^2 and deceleration 4.5 m/s^2. The speed limit of the vehicle is 10 m/s. The routes of the vehicles are randomly generated, with the probability ratio of left turn, going straight and right turn set as 0.25:0.5:0.25. The arriving time of the vehicles follows Poisson distribution with an arriving rate denoting how many vehicles will arrive per second in average. In our experiment, the arriving rate ranges from 0.1 vehicle/s to 0.5 vehicle/s. The number of total vehicles entering the intersection is set as 300. The level of timing attack is represented by the delay added to the messages. The performance of the intersection is evaluated as the average traveling time of all the vehicles aiming to cross the intersection.

The simulation results are shown in Figure 6. The x-axis denotes the communication delay caused by the timing attack, and the y-axis denotes the average traveling time of the vehicles as performance. We define traffic patterns in our simulation as the flow ratio of the vehicles arriving from north-south directions and the vehicles arriving from west-east directions. For example, traffic pattern "flow 0.5 : 0.1" denotes the average traffic flow from north-south directions is 0.5 vehicle/s and the average traffic flow from west-east directions is 0.1 vehicle/s. The figure shows that for each traffic pattern, the performance significantly decreases as the communication delay increases. For each specific delay, no matter symmetric or asymmetric traffic pattern, the trend of performance deterioration is similar.

6.2 Multiple Intersection with Multiple Lanes

In this experiment, the setup is a traffic network with nine intersections as in Figure 4 (b). Each intersection has the same setup as in Figure 5 with designated lanes for left turn, going straight and right turn. The connection of adjacent intersections is to connect the corresponding lanes together, i.e., connecting left lane with left lane, middle lane with middle lane, and right lane with right lane. In this simulation, vehicles can only arrive from the twelve

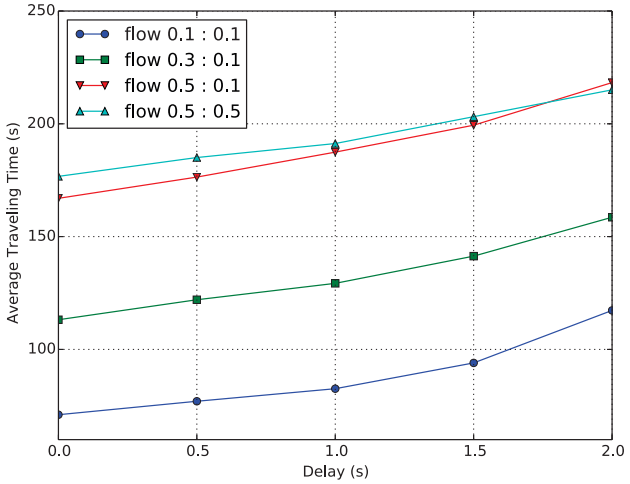

Figure 7: Performance of nine interconnected intersections under the same attack.

entrances. At each intersection, the vehicle has a 0.25 probability to turn left, a 0.5 probability to go straight, and a 0.25 probability to turn right. The total number of vehicles entering the network is set as 1200. We first assume the attacker to launch timing attack to all the intersections, and then study the influence by attacking only one intersection.

Figure 7 shows that average traveling time increases with communication delay applied to all nine intersections. In this case, the trend is similar to the single intersection with multiple lanes. For each traffic pattern, the average traveling time of vehicles increases as delay increases. Note that the performances of traffic patterns "0.5 : 0.1" and "0.5 : 0.5" are very similar and pattern "0.5 : 0.1" is even slightly better than "0.5 : 0.5" when the delay is 2.0 second.

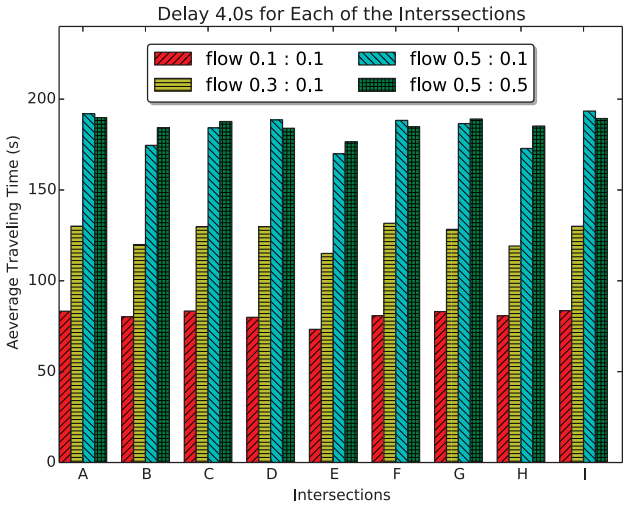

Figure 8: Performance of nine intersections (denoted by "A" to "I") if one intersection is under timing attack.

Figure 8 shows the attack to only one of the intersections with a delay of 4.0 second. Letters 'A' to 'I' represent different intersections, and their positions are shown in Figure 4 (b). The y-axis denotes the average traveling time of vehicles.

7 CONCLUSION

In this paper, we address the modeling, analysis and simulation of intelligent intersection management with the consideration of communication delays and timing attacks. We consider both single intersections with multiple lanes and multiple interconnected intersections. We also discuss how game theory may be applied to analyze the strategic level security issues in such intelligent intersections. Finally, simulation results demonstrate the significant impact of timing attacks on intersection performance.

REFERENCES

[1] 2017. SUMO. http://www.dlr.de/ts/en/desktopdefault.aspx/tabid-9883/16931_read-41000/. (2017).

[2] 2017. UPPAAL. https://www.uppaal.org/. (2017).

[3] M. Ahmane, A. Abbas-Turki, F. Perronnet, J. Wu, A. El Moudni, J. Buisson, and R. Zeo. 2013. Modeling and controlling an isolated urban intersection based on cooperative vehicles. *Transportation Research Part C: Emerging Technologies* 28 (2013), 44–62.

[4] R. Aumann. 1987. Correlated Equilibrium as an Expression of Bayesian Rationality. *Econometrica* 55, 1 (1987), 1–18.

[5] R. Azimi, G. Bhatia, R. Rajkumar, and P. Mudalige. 2012. *Intersection management using vehicular networks.* Technical Report. SAE Technical Paper.

[6] R. Azimi, G. Bhatia, R. Rajkumar, and P. Mudalige. 2014. STIP: Spatio-temporal intersection protocols for autonomous vehicles. In *ICCPS'14: ACM/IEEE 5th International Conference on Cyber-Physical Systems (with CPS Week 2014)*. IEEE Computer Society, 1–12.

[7] S.R. Azimi, G. Bhatia, R. Rajkumar, and P. Mudalige. 2013. Reliable intersection protocols using vehicular networks. In *Proceedings of the ACM/IEEE 4th International Conference on Cyber-Physical Systems*. ACM, 1–10.

[8] T. Başar and G. J. Olsder. 1999. *Dynamic Noncooperative Game Theory.* SIAM Series in Classics in Applied Mathematics, Philadelphia.

[9] Y. O. Basciftci, F. Chen, J. Weston, R. Burton, and C. E. Koksal. 2015. How vulnerable is vehicular communication to physical layer jamming attacks?. In *2015 IEEE 82nd Vehicular Technology Conference (VTC2015-Fall)*. 1–5. https://doi.org/10.1109/VTCFall.2015.7390968

[10] S. Bastani, B. Landfeldt, and L. Libman. 2011. On the reliability of safety message broadcast in urban vehicular ad hoc networks. In *Proceedings of the 14th ACM International Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems (MSWiM '11)*. ACM, New York, NY, USA, 307–316. https://doi.org/10.1145/2068897.2068951

[11] L. Chen and C. Englund. 2016. Cooperative intersection management: a survey. *IEEE Transactions on Intelligent Transportation Systems* 17, 2 (Feb 2016), 570–586. https://doi.org/10.1109/TITS.2015.2471812

[12] K. Dresner and P. Stone. 2008. A multiagent approach to autonomous intersection management. *Journal of artificial intelligence research* 31 (2008), 591–656.

[13] M. Elhenawy, A. A. Elbery, A. A. Hassan, and H. A. Rakha. 2015. An intersection game-theory-based traffic control algorithm in a connected vehicle environment. In *IEEE International Conference on Intelligent Transportation Systems (ITSC)*. 343–347.

[14] Y. P. Fallah and M. K. Khandani. 2015. Analysis of the coupling of communication network and safety application in cooperative collision warning systems. In *Proceedings of the ACM/IEEE Sixth International Conference on Cyber-Physical Systems (ICCPS '15)*. ACM, New York, NY, USA, 228–237. https://doi.org/10.1145/2735960.2735975

[15] Z. Han, D. Niyato, W. Saad, and T. Başar. 2012. *Game Theory in Wireless and Communication Networks: Theory, Models, and Applications.* Cambridge University Press, Cambridge, UK.

[16] Q. Jin, G. Wu, K. Boriboonsomsin, and M. Barth. 2012. Advanced intersection management for connected vehicles using a multi-agent systems approach. In *Intelligent Vehicles Symposium (IV), 2012 IEEE*. IEEE, 932–937.

[17] F. K. Karnadi, Z. H. Mo, and K. c. Lan. 2007. Rapid generation of realistic mobility models for VANET. In *2007 IEEE Wireless Communications and Networking Conference*. 2506–2511. https://doi.org/10.1109/WCNC.2007.467

[18] J. B. Kenney. 2011. Dedicated short-range communications (DSRC) standards in the United States. *Proc. IEEE* 99, 7 (July 2011), 1162–1182.

[19] H. Kowshik, D. Caveney, and PR. Kumar. 2011. Provable systemwide safety in intelligent intersections. *IEEE transactions on vehicular technology* 60, 3 (2011), 804–818.

[20] R. Naumann, R. Rasche, J. Tacken, and C. Tahedi. 1997. Validation and simulation of a decentralized intersection collision avoidance algorithm. In *Intelligent Transportation System, 1997. ITSC'97., IEEE Conference on*. IEEE, 818–823.

[21] M. O. Sayin, C.-W. Lin, S. Shiraishi, and T. Başar. 2017. Information-driven intersection management: Truthfulness via Payments. *Submitted to IEEE Transactions on Intelligent Transportation Systems* (2017).

[22] M. O. Sayin, C.-W. Lin, S. Shiraishi, and T. Başar. 2017. Truthfulness of intersection management with strategic autonomous and connected agents. *Submitted to American Control Conference.*

[23] M. Vasirani and S. Ossowski. 2012. A market-inspired approach for intersection management in urban road traffic networks. *Journal of Artificial Intelligence Research* 43 (2012), 621–659.

[24] V. V. Vazirani, N. Nisan, T. Roughgarden, and É. Tardos. 2007. *Algorithmic Game Theory.* Cambridge University Press, Cambridge, UK.

[25] Y. Yao, L. Rao, X. Liu, and X. Zhou. 2013. Delay analysis and study of IEEE 802.11p based DSRC safety communication in a highway environment. In *2013 Proceedings IEEE INFOCOM*. 1591–1599. https://doi.org/10.1109/INFCOM.2013.6566955

[26] S. Yousefi, M. S. Mousavi, and M. Fathy. 2006. vehicular ad hoc networks (VANETs): challenges and perspectives. In *2006 6th International Conference on ITS Telecommunications*. 761–766. https://doi.org/10.1109/ITST.2006.289012

[27] B. Zheng, C. W. Lin, H. Liang, S. Shiraishi, W. Li, and Q. Zhu. 2017. Delay-aware design, analysis and verification of intelligent intersection management. In *2017 IEEE International Conference on Smart Computing (SMARTCOMP)*. 1–8. https://doi.org/10.1109/SMARTCOMP.2017.7946999

[28] B. Zheng, C. W. Lin, H. Yu, H. Liang, and Q. Zhu. 2016. CONVINCE: A cross-layer modeling, exploration and validation framework for next-generation connected vehicles. In *2016 IEEE/ACM International Conference on Computer-Aided Design (ICCAD)*. 1–8. https://doi.org/10.1145/2966986.2980078

[29] F. Zhu and S. V. Ukkusuri. 2015. A linear programming formulation for autonomous intersection control within a dynamic traffic assignment and connected vehicle environment. *Transportation Research Part C: Emerging Technologies* 55 (2015), 363–378.