Impacts of Constrained Sensing and Communication based Attacks on Vehicular Platoons

Mingshun Sun*, Ali Al-Hashimi*, Ming Li, Ryan Gerdes

***Abstract*—Vehicular platooning promises to bring a faster, safer, and more efficient transportation. Automated platooned vehicles will rely on information obtained from inter-vehicle communication channels and on-board sensors to make driving decisions and achieve platooning. However, such reliance creates an opportunity for safety violating attacks intended to disrupt the platoon formation and cause accidents. In this work, we investigate more realistic attacks mounted against the communication and sensing functionalities of platooned vehicles. More specifically, we are interested in approximating the set of final unsafe states, that can be reached by mounting realistically constrained attacks capable of introducing delay and injecting false-data against the aforementioned functionalities. For that purpose, we will use reachability analysis which enables us to realize whether it is possible to drive the platoon from initial to final states given performance and physical bounds. Our results suggest that these two types of attack are able to steer the platoon towards dangerous states and generate impacts on passengers' safety by causing crashes at high speeds.**

***Index Terms*—vehicular platooning, security of vehicular platoons, reachability analysis.**

I. Introduction

Vehicular platooning is a cyber-physical system (CPS) that employs automation, communication, sensing, and decision-making capabilities. The objective of such systems is to combine multiple automated vehicles to follow each other, regulate their movements, and maintain predefined inter-vehicle distances and relative speeds. Vehicular platoons are gaining a rapid interest and development, both academically and commercially, since they have shown numerous benefits such as providing a safe and comfortable environment for the passengers allows them to focus on tasks other than driving [1], reducing traffic congestion on highways which leads to a more efficient usage of roads [2], and improving fuel consumption [3].

To achieve the aforementioned objectives, each platooned vehicle implements a properly designed controller that determines the appropriate acceleration commands [4] by using information collected from local sensors and from other vehicles through inter-vehicle communication [5] or from external networks [6]. Adaptive Cruise Control (ACC) and Cooperative Adaptive Cruise Control (CACC) are the most well-known control strategies used to form and maintain platoons. To generate acceleration commands, ACC operation uses the range (relative distance) and range-rate (relative speed) of neighboring vehicles gathered from on-board sensors (e.g., RADAR, or cameras) [7] while CACC, an extension of ACC, incorporates vehicle-to-vehicle (V2V) communication so that vehicles may exchange state information (e.g., alerting other vehicles to changes in acceleration) [5]. Despite the ability of both strategies to achieve platooning, from a security point of view it has been shown that ACC and CACC based platoons are vulnerable to threats (attacks) against the sensing [8], [9] and communication [10], [11] functionalities, which are essential for implementing both strategies. Since we are interested in the minimum attack surface necessary to examine those threats, in this work we adopt both ACC and CACC to form platoons and demonstrate the impacts of such relevant threats against sensors and communication channels, respectively.

It has been verified [12], [13], [14] that vehicular platoons have a potential attack surface that can be exploited by malicious parties (attackers) and produce a disruptive behavior in the platoon. Some studies define insider attacks where the attacker is controlling a vehicle inside the platoon and, for instance, is able to modify the prevailing control law to destabilize a vehicular platoon [15]. Other studies define outsider attacks where the attack is conducted from outside the platoon, such as by employing a drone to induce jamming [11]. As a result, the security of vehicular platoons is still widely researched with the goal of defining possible vulnerabilities that can be exploited by attackers and understanding the attack-induced impacts which could include oscillations in vehicles' movements causing passengers' discomfort, increased fuel consumption, or fatal collisions at a high relative speeds.

In this work, we are concerned with two existing attacks where each attack mechanism targets a specific functionality of the automation system employed in platooned vehicles and, hence, compromises the safety of the attacked vehicular platoon. More specifically, we are interested in defining the set of final states that the platoon can reach as a result of experiencing an attack against the on-board sensors (physical state), referred to as a False-Data Injection (FDI) attack, and against the inter-vehicle communication channels (cyber state), referred to as a Message Delay (MD) attack. It has been shown that FDI attacks are possible to implement against ultrasonic, RADAR, LIDAR, and cameras [16], [17], [18], [19], which are the mostly employed sensors for vehicular platooning, and jamming/spoofing can be induced [11]. Furthermore, MD attack against V2V communications is easily achievable, either

Mingshun Sun and Ming Li are with the Dept. of Electrical and Computer Engineering, The University of Arizona, Tucson, AZ 85721. {mingshunsun,lim}@email.arizona.edu

Ali Al-Hashimi is with the Dept. of Electrical and Computer Engineering, Utah State University, Logan, UT 84321. ali.eng1@outlook.com

Ryan Gerdes is with the Dept. of Electrical and Computer Engineering, Virginia Tech, Arlington, VA 22203. rgerdes@vt.edu

* : Equally contributed authors. Manuscript received XX,2019

This work was partly supported by NSF grants CNS-1410000 and CNS-1801402.

via message collision or jamming and replay, which does not tamper with V2V messages and can be very stealthy. The realizations of both attacks require less capable outsider attackers compared with the ones assumed in existing literature [20], [9], and thus are more realistic threats. Although previous works show that the FDI and jamming attacks could potentially lead to crashes and impair the safety of vehicular platoons, only specific attack vectors are demonstrated and the realistic FDI and MD attacks' impacts have not been systematically characterized. In this work, we use reachability analysis to comprehensively investigate the extent of FDI and MD attacker's ability to induce harmful impacts.

Reachability analysis defines the reachable set of a dynamic system, which is the set of all system states that can be attained within a finite time. This analysis can be applied in real-world applications where safety needs to be determined such as collision avoidance problems in airplanes [21], or designing controllers for the platooning of unmanned aerial vehicles (UAV) [22]. We will use the reachability analysis to determine what final states can the attacked platoon reach as a result of undergoing an FDI or MD attack and how serious those states are, i.e. are collisions reachable and at what relative speeds. The results obtained from conducting such analysis will, in turn, demonstrate the attacker capability to influence the movement of the attacked vehicles. To summarize, we make the following contributions:

- We adopt two reachability analysis methods in order to comprehensively illustrate and evaluate the impacts induced by two attacks, FDI and MD, on the safety of a vehicular platoon. For each attack, we consider the system physical bounds and performance constraints such as maximum and minimum speed or acceleration, limits of the sensors measurements, or the magnitude of the attack vector sequence. Considering such constraints in our analysis creates a more realistic attack scenario, and specific threat models are formulated for each attack.
- For the FDI attack, the optimal control based reachability method [23] is used to analyze the attack-induced impacts. For this analysis, we consider the following constraints: a finite discrete attack sequence where each entry agrees with a resolution of the attacked sensor(s), spoofed measurements result in bounded acceleration commands, and the physical limits of the attacked sensor(s). For this attack analysis, our results indicate severe collisions are possible for the targeted vehicles and even for another random (non-attacked) vehicle in the platoon.
- For MD attack, this paper adopts the HJ reachability analysis to show whether the collision can happen and how severe it is for a set of control parameters during a given period of time regarding all possible input profiles. The results are more comprehensive than previous methods and bear more importance because the required attacker capability is not as strong as previously assumed. Besides, our paper is the first to model the time delay into the control input deviation and accordingly analyze its impact under the worst-case situation. Simulation results show that this attack can lead to severe crash especially when platoon vehicles suffer a longer actuation delay and larger jerk [24]. Finally, we also present some possible countermeasures, such as detecting the MD attacks.

A. Paper Organization

In Section II, related works are discussed and compared to emphasize the novelty of our paper. In Sections III and IV, the system model, CACC design and the attack model are presented for two attacks. The reachability approach is discussed in Section V. In Section VI, simulation is presented on the impact of two attacks. Also, the defense mechanism is discussed. We end with a conclusion and an overview of future work.

II. Related works

In this section, we describe three areas of related work: vehicular sensor attacks, communication channel attacks, and methods for reachability analysis.

1) Sensor Attacks: Existing work has demonstrated the possibility of manipulating vehicular sensors. For instance, the experimental results presented in [16] show that jamming and spoofing attacks can be carried out against ultrasonic sensors and cameras. Furthermore, falsifying the readings of a vehicle's RADAR, LIDAR, or cameras was achieved in [18] and [17]. On the other hand, the work of [8] and [9] show results for analyzing the FDI attacks against CACC-based platoons. In both works, the threat model assumes the presence of an attacker-controlled vehicle in the platoon (insider attack) which is capable of feeding false constant relative distance or speed measurements, with respect to the neighboring vehicle, or transmitting false constant acceleration data. In this work, we continue to further analyze the impacts of FDI attacks on vehicular platoons. For that purpose, we make the following assumptions: First, we focus on FDI attacks against ACC-based platoons. Second, an outsider FDI attack against the on-board sensors of platooned vehicles is present. Third, we assume the presence of a general attack sequence (vector), i.e. it is not a constant value. Finally, our analysis involves physical and performance constraints, such as the resolution and limits of the attacked sensor(s). The last two assumptions are included to help create a more comprehensive analysis of the impacts induced by the FDI attacks on ACC-based platoons. Our analysis results indicate the possibility of collisions and at different relative speeds. The results also show the possibility of causing collisions at random non-attacked vehicles in the platoons by launching FDI attacks on the on-board sensors of another vehicle.

2) Communication Channel Attacks: Previous MD attack literature mainly focuses on the delay's impact on string stability [25], [26], [10], [27], [28]. Researchers study the impact of time delay in the leader state reception (leader-to-all communication) [10] or in different flows (one-by-one communication) [27] on the string stability. [28] proves the string stability always holds for CACC without delay. Their results only the amount of delay that needed to break string stability under a certain set of control parameters and leading vehicle acceleration profile. But string instability may not

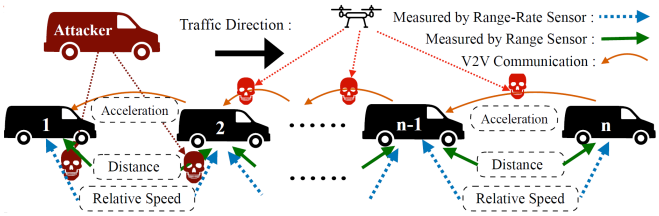


Fig. 1: A vehicular platoon with potential threats against equipped sensors and V2V communication channels

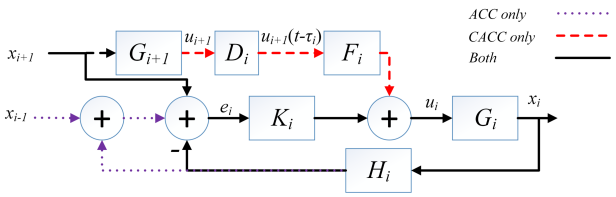


Fig. 2: General control structure

guarantee the terrible collision in a short period. Usually, the string instability can cause collisions at the end of a long platoon string after a certain period of time when the lead vehicle gets disturbed [28]. Alipour et. al. consider the jamming attack under channel fading and packet loss when they evaluate the delay's impact [11]. The jamming attack is easily detected once a vehicle discovers consecutive packet loss or expiration. Kafash et.al [29] analyze the attack impacts when subject to the physical limits of the actuators by finding the reachable sets. But they require a strong attacker who can gain access to all CACC commands and injects false data to cause a abrupt brake, which isn't an easily achievable attack compared with the MD attack. In conclusion, above works are not realistic and comprehensive enough to assess the danger of collision. From the designer's point of view, there lacks literature that evaluates the potential crash between vehicles by injecting delay into the channel during some time span.

3) Computation of reachable sets: Various methods have been proposed for obtaining the reachable sets. In [30], ellipsoidal techniques are used to calculates outer elliptical bounds around the reachable set. This method has been applied in problems such as collision avoidance in UAVs [31]. Another method is generally known as Hamilton-Jacobi (HJ) reachability [21] and has been used to solve problems such as path planning for UAVs [32]. One more method is based on using optimal control theory where the final states are included in an optimization problem whose solution determines the appropriate control sequence to drive the system towards those final states [23]. This method has been applied in problems such as determining an alternate trajectory for vehicles to be tracked and thus avoid colliding with other vehicles [33].

In the context of vehicular platoons security, reachability analysis was employed to quantify the impacts of induced attacks. For instance, reachable sets were determined for a CACC-based platoon experiencing jamming attacks on its V2V channels [29] and also for an ACC-based platoon where the attacker controls one of the platooned vehicles [34]. In both cases above, resulting sets exhibited possibility for collisions. Similarly, we continue to investigate the reachability of vehicular platoons while operating in an adversarial environment. More specifically, we determine reachable sets of both a CACC-based platoon, similar to [29], while experiencing an MD attack on the V2V communication channels and an ACC-based platoon, similar to [34], while undergoing an FDI attack on the on-board sensors.

III. System Model

The modeling of platooned vehicles as well as the control strategies, to achieve platooning, are discussed in this section.

A. Vehicle Model

We consider a homogeneous platoon with n vehicles where all vehicles share the same dynamics, controller design, and performance characteristics. In general, each platooned vehicle's dynamics are described as

$$\dot{\boldsymbol{x}}(t) = f(\boldsymbol{x}(t), \boldsymbol{u}(t)) \tag{1}$$

where $\boldsymbol{x}$ and $\boldsymbol{u}$ are the state and input vectors, respectively. Vehicle's states evolution over time is described as

$$\begin{aligned} \dot{x}_i(t) &= v_i(t) \\ \dot{v}_i(t) &= a_i(t) \\ \dot{a}_i(t) &= \frac{1}{\eta_i} u_i(t) - \frac{1}{\eta_i} a_i(t), \text{ for } i = 1, \ldots, n \end{aligned} \tag{2}$$

where x_i, v_i, a_i, u_i, and η_i refer to the i^{th} vehicle's absolute position, absolute velocity, actual acceleration, acceleration command, and actuator's delay, respectively. The last equation 2 describes the relationship between the commanded acceleration and actual acceleration [35]. A larger η leads to a smaller evolution rate of the actual acceleration. To reduce the complexity associated with determining the reachable sets, we will assume that there is no actuation delay for the FDI attack related analyses. Therefore, dynamics of the i^{th} platooned vehicle experiencing an FDI attack are described as

$$\begin{aligned} \dot{x}_i(t) &= v_i(t) \\ \dot{v}_i(t) &= u_i(t), \text{ for } i = 1, \ldots, n \end{aligned} \tag{3}$$

B. Platoon Model

The platooning of n vehicles is accomplished by determining the input vector $\boldsymbol{u}$, from (2) and (3), using either ACC or CACC control strategies. Each vehicle is equipped with front and back range and range-rate sensors (shown in blue and green arrows, respectively, in Fig 1). Also, each vehicle implements an upper-level controller, which determines the commanded acceleration, and a lower-level controller, which uses the commanded acceleration to produce throttle and brake commands. In this work, we will focus on the upper-level controller since the attacker can easily affect it.

Fig.2 refers to the general upper-level controller that we use where G, K, F, H, and D represent the vehicle dynamic, feedback PD controller, feed-forward controller(for CACC only), headway policy, and injected channel delay respectively.

The solid line represents the control structure that is shared by both ACC and CACC, which consists of the feedback control loop that requires the information of the front vehicle. The red dashed line represents the feed-forward controller that requires V2V communication, which is unique in CACC. The purple dotted line is the information from the following vehicle, which is necessary in bi-directional control for ACC.

1) ACC Platoon Model: For each platooned vehicle equipped with an ACC control structure, the error coordinates are defined as follows

$$\begin{aligned} e_{xi}(t) &= x_{i+1}(t) - x_i(t) - x_d \\ e_{vi}(t) &= v_{i+1}(t) - v_i(t) \end{aligned} \tag{4}$$

where e_{xi} and e_{vi} refer to the i^{th} vehicle's position and speed error, respectively, and x_d is a constant denoting inter-vehicle desired separation. It should be noted that error states are fully measured using the on-board range and range-rate sensors. The evolution of error states over time can be described as follows

$$\begin{aligned} \dot{e}_{xi}(t) &= v_{i+1}(t) - v_i(t) \\ \dot{e}_{vi}(t) &= u_{i+1}(t) - u_i(t) \end{aligned} \tag{5}$$

which can be rewritten in the following state-space representation

$$\dot{e}(t) = A_1 e(t) + B_1 u(t) \tag{6}$$

where

$$\begin{aligned} e(t) &= \begin{bmatrix} e_{x1}(t) & \dots & e_{xn}(t) & e_{v1}(t) & \dots & e_{vn}(t) \end{bmatrix}^T \\ u(t) &= \begin{bmatrix} u_1(t) & \dots & u_n(t) \end{bmatrix}^T \end{aligned}$$

Matrices A_1 and B_1 are described in Appendix A. Each platooned vehicle uses a bidirectional control law to determine its commanded acceleration [36]. Bidirectional control is able to guarantee platoon string stability, which maintains proper traffic flow [4], [36], and it does not need any (V2V) transmitted information to generate driving decisions. Each vehicle's commanded acceleration is calculated according to its position in the platoon. For the last vehicle in a given platoon, we have

$$u_1(t) = k_p e_{x1}(t) + k_d e_{v1}(t), \tag{7}$$

where k_p and k_d are the controller's proportional and derivative gains, respectively. For the rest of the vehicles in the platoon, we have

$$\begin{aligned} u_i(t) = k_p\big(e_{xi}(t) - e_{xi-1}(t)\big) + k_d\big(e_{vi}(t) - e_{vi-1}(t)\big), \\ \text{for } i = 2, \dots, n \end{aligned} \tag{8}$$

Commanded acceleration of all vehicle can be combined in the following state-space representation

$$u(t) = A_2 e(t) \tag{9}$$

matrix A_2 is described in Appendix A.

2) CACC Platoon Model: In CACC, dedicated Short Range Communication (DSRC) provides foundations for V2V communications. Unlike ACC, the uni-directional control is more popular in CACC where each vehicle in the platoon receives messages from its preceding vehicle and sends messages to its following vehicle respectively.

In this paper, we adopt the velocity-dependent space policy used in [28]. The error coordinates is defined as follows

$$\begin{aligned} e_{xi}(t) &= x_{i+1}(t) - x_i(t) - h \cdot v_i(t) \\ e_{vi}(t) &= v_{i+1}(t) - v_i(t) \\ e_{ai}(t) &= a_{i+1}(t) - a_i(t) \end{aligned} \tag{10}$$

where h is the constant headway time. The constant headway-time policy automatically achieves string stability if the V2V message is not delayed. As stated earlier, the actuation dynamics are also reflected in CACC platoon.

Therefore, error states evolution is similarly defined as:

$$\begin{aligned} \dot{e}_{xi}(t) &= v_{i+1}(t) - v_i(t) - h a_i(t) = e_{vi}(t) - h a_i(t) \\ \dot{e}_{vi}(t) &= a_{i+1}(t) - a_i(t) = e_{ai}(t) \\ \dot{e}_{ai}(t) &= \frac{1}{\eta} u_{i+1}(t) - \frac{1}{\eta} a_{i+1}(t) - \left(\frac{1}{\eta} u_i(t) - \frac{1}{\eta} a_i(t) \right) \\ &= -\frac{1}{\eta} e_{ai}(t) + \frac{1}{\eta} \left(u_{i+1}(t) - u_i(t) \right) \end{aligned} \tag{11}$$

The state space representation for CACC is similarly obtained:

$$\dot{e}_{cacc}(t) = A_{cacc} e_{cacc}(t) + M_{cacc} a(t) + B_{cacc} u_{cacc}(t) \tag{12}$$

Similarly, the leading vehicle is not specifically controlled by anyone. All the following vehicle are controlled by the uni-directional control law as follows

$$\begin{aligned} u_i(t) = k_p e_{xi}(t) + k_d e_{vi}(t) + k_a u_{i+1}(t), \\ \text{for } i = 1, \dots, n-1 \end{aligned} \tag{13}$$

The uni-directional control is more popular in CACC because the CACC provides much smaller inter-vehicle distance and therefore vehicles put much more focus on the front vehicle rather than the following ones in order to avoid collision.

IV. Threat Model

As shown in Figure 1, attackers can send malicious vehicles, flying drones or even malicious roadside units to contaminate the sensor measurement at any vehicle as well as to deteriorate the channel stability on any inter-vehicle links during some time without breaking into any platoon vehicles. In this section, in order to investigate the impact of both attacks, we define two different attack models.

A. FDI Attack

FDI attacks against vehicular sensors aim to generate harmful impacts in the platoon by injecting false-data into the attacked sensor(s) to perturb their measurements. Existing work has demonstrated that the most used sensors in automated vehicles, such as LIDAR or cameras, can be jammed or spoofed and that such attacks can be accomplished at a distance [18], [17], [16], [37]. For the purpose of demonstrating FDI attack impacts in our study, we assume the following: First, the

attacker is informed of the platoon model, which includes the controller design and type of sensors used. Second, the attacker has the capability to compromise the reading of one or multiple sensors equipped on one more platooned vehicles using drones, units installed on the road for that purpose, or by an attacker-controller vehicle driving alongside the platoon. The second assumption only needs the devices outside the platoon and also not requires breaking into the platoon vehicles, which still demonstrates an outside attacker. Finally, the attack sequence (vector) can only assume discrete values such that once injected it does not violate the resolution of the attacked sensor(s). The last assumption helps create realistic attack scenarios. It also helps distinguish feasible attacks for non feasible ones.

1) Attacking Range Sensors: In this case, the commanded acceleration becomes as follows

$$\begin{aligned} u_1(t) &= k_p\big(e_{x1}(t)+\delta_{x1}(t)\big)+k_d e_{v1}(t)\\ &\vdots\\ u_n(t) &= k_p\big(\big(e_{xn}(t)+\delta_{xn}(t)\big)-e_{xn-1}(t)\big)\\ &\quad + k_d\big(e_{vn}(t)-e_{vn-1}(t)\big)\end{aligned} \tag{14}$$

where δ_{xi} is the amount of false-data injected against the i^{th} vehicle's range sensor. (14) can be rewritten as follows

$$\begin{aligned} u(t) &= A_2 e(t)+B_{2,x}\delta(t)\\ \delta(t) &= \begin{bmatrix}\delta_{x1}(t) & \dots & \delta_{xn}(t)\end{bmatrix}^T\end{aligned} \tag{15}$$

2) Attacking Range-rate Sensors: In this case, the commanded acceleration becomes as follows

$$\begin{aligned} u_1(t) &= k_p e_{x1}(t)+k_d\big(e_{v1}(t)+\delta_{v1}(t)\big)\\ &\vdots\\ u_n(t) &= k_p\big(e_{xn}(t)-e_{xn-1}(t)\big)\\ &\quad + k_d\big(\big(e_{vn}(t)+\delta_{vn}(t)\big)-e_{vn-1}(t)\big)\end{aligned} \tag{16}$$

where δ_{vi} is the amount of false-data injected against the i^{th} vehicle's range-rate sensor. (16) can be rewritten as follows

$$\begin{aligned} u(t) &= A_2 e(t)+B_{2,v}\delta(t)\\ \delta(t) &= \begin{bmatrix}\delta_{v1}(t) & \dots & \delta_{vn}(t)\end{bmatrix}^T\end{aligned} \tag{17}$$

3) Attacking Both Range and Range-rate Sensors: In this case, the commanded acceleration becomes as follows

$$\begin{aligned} u_1(t) &= k_p\big(e_{x1}(t)+\delta_{x1}(t)\big)+k_d\big(e_{v1}(t)+\delta_{v1}(t)\big)\\ &\vdots\\ u_n(t) &= k_p\big(\big(e_{xn}(t)+\delta_{xn}(t)\big)-e_{xn-1}(t)\big)\\ &\quad + k_d\big(\big(e_{vn}(t)+\delta_{vn}(t)\big)-e_{vn-1}(t)\big)\end{aligned} \tag{18}$$

which can be rewritten as follows

$$\begin{aligned} u(t) &= A_2 e(t)+B_{2,xv}\delta(t)\\ \delta(t) &= \begin{bmatrix}\delta_{x1}(t) & \dots & \delta_{xn}(t) & \delta_{v1}(t) & \dots & \delta_{vn}(t)\end{bmatrix}^T\end{aligned} \tag{19}$$

Matrices $B_{2,x}$, $B_{2,v}$, and $B_{2,xv}$ are given in Appendix A. Considering the presence of attack vectors, acceleration commands, given in (9), become as follows

$$\begin{aligned} u(t) &= A_2 e(t)+B_a\delta(t)\\ B_a &\in \big\{B_{2,x},B_{2,v},B_{2,xv}\big\}\end{aligned} \tag{20}$$

by substituting (20) into (6), we get

$$\begin{aligned} \dot{e}(t) &= A_c e(t)+B_c\delta(t)\\ A_c &= A_1+B_1A_2\\ B_c &= B_1B_a\end{aligned} \tag{21}$$

Reachable sets are approximated by solving an optimization problem which determines an appropriate attack vector, as will be explained later. For that purpose, we need the discrete-time representation of the error coordinates. Using the forward difference approximation [38], (21) can be described as

$$\begin{aligned} e(k+1) &= Ae(k)+B\delta(k)\\ A &= I+T_sA_c\\ B &= T_sB_c\end{aligned} \tag{22}$$

where I is the identity matrix of proper dimensions, and T_s is the sampling time and $k=0,1,\dots,m$ is the time sample index. In this work, time horizon is defined as $t\in[t_0,t_f]$, where $(t_0=t(k=0))$ and $(t_f=t(k=m))$ correspond to initial and final time samples, respectively.

B. MD Attack

MD attack aims at delaying the message time of arrival to damage the message timeliness by message interference or jamming and message replay. The DSRC requires message re-transmission when the message does not arrive on time. Attackers can jam the channel and replay the message after a short period of time. MD attack does not require any sophisticated operation from malicious attacker such as breaking the authentication protocol or hijacking platoon vehicles, which is more achievable and realistic. Recent work reveals the existence of such attacks initiated by malicious flying drones [11] or neighboring vehicles [14], [39].

To demonstrate the impact of a very basic delay attack, we assume the attacker has the following capabilities. First, we assume a mobile outside attacker who can only delay the message arrival in the inter-vehicle link to some extent. Second, attackers cannot control the acceleration input profile of the leading vehicle.

Specifically, attackers can change the link delay $\tau_i(t)$ for $i=1,2,\dots,n-1$ at any time. The delay is not necessary to be the same in each channel. But the maximum delay is bounded by the τ_{max}, which means $\tau_i(t)\in[0,\tau_{max}]$. Based on the research in DSRC protocol [40], the allowable latency (largest lifetime) of the V2V message is around $500\,\mathrm{ms}$. Additionally, in the worst case, τ_{max} can be infinity when no message is received. However, the platoon can notice this worst case and accordingly, switches to the ACC control if necessary. Besides, we assume that the message integrity and authenticity are protected and validated via traditional cryptography methods. The attack variable paves its way into CACC platoon model as follows

$$\begin{aligned} u_i(t) &= k_p e_{xi}(t)+k_d e_{vi}(t)+k_a u_{r,i}(t,\tau_i)\\ &\text{for } i=1,\dots,n-1\end{aligned} \tag{23}$$

Where $u_{r,i}(t,\tau_i)$ is the received vehicle $i+1^{th}$ commanded acceleration at the i^{th} vehicle's receiver. However, we find that it is difficult to quantify the impact of delay τ_i on the system

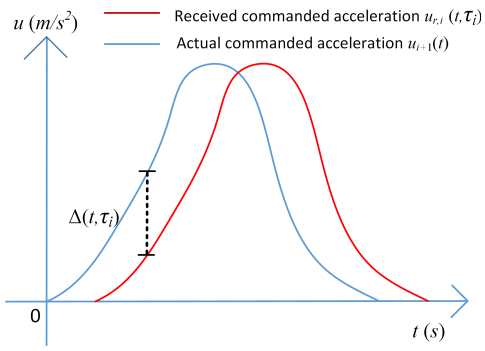


Fig. 3: Impact of delay on acceleration

equation in the above form. We need to note that the delay attack is of no use if the vehicle platoon is in the steady state. In other words, the attacker will delay messages only when it observes the leading vehicle's acceleration(deceleration).

Then we quantify the impact of delay based on fig 3. We observe that there will be a discrepancy between the actual commanded acceleration of the preceding vehicle and received acceleration at the following adjacent vehicle. This discrepancy is defined as $\Delta_a(t,\tau_i)$ for the vehicle i. Then we re-express the delayed acceleration as follows

$$u_{r,i}(t,\tau_i) = u_{i+1}(t) + \Delta_a(t,\tau_i) \tag{24}$$

where $\Delta(\tau_i)$ represents the pre-defined discrepancy. The value of this discrepancy highly depends on the input profile of the preceding vehicle as well as the delay time.

Considering that the attacker has no idea of the control profile of the leading vehicle, we evaluate the discrepancy by combining it with the vehicle's moving ability to generalize the results for all possible input patterns.

Therefore, we introduce the definition of jerk j as follows

$$j = \frac{da}{dt} \tag{25}$$

The jerk is the changing rate of acceleration.

We can therefore conclude that the discrepancy $\Delta_a(t,\tau_i)$ is bounded by both jerk and the τ_{max}. Specifically speaking,

$$|\Delta_a(t,\tau_i)| \leq j_u \cdot \tau_{max} \tag{26}$$

And we define j_u is the upper bound of the jerk j where $-j_u \leq j \leq j_u$ And also homogeneous vehicle string shares the same j_u. So the system equation in equation 11 becomes:

$$\begin{aligned}
\dot{e}_{xi}(t) &= v_{i+1}(t) - v_i(t) - h \cdot a_i(t) \\
\dot{e}_{vi}(t) &= u_{i+1}(t) - u_i(t) \\
\dot{e}_{ai}(t) &= -\frac{1}{\eta} e_{ai}(t) + \frac{1}{\eta}\left(u_{i+1}(t) - u_i(t)\right) \\
&= -\frac{1}{\eta} e_{ai}(t) + \frac{1}{\eta}\big[u_{i+1}(t) - [k_p e_{xi}(t) \\
&\quad + k_d e_{vi}(t) + k_a \cdot \left(u_{i+1}(t) + \Delta_a(t,\tau_i)\right)]\big]
\end{aligned} \tag{27}$$

The discrepancy bound is adopted as the bound of the attacker capability range in the reachability analysis, which is analyzed in the next section.

V. Reachability Analysis of Constrained Attacks on Vehicle Platoons

Generally, reachability analysis is a mathematical tool that provides information about the evolution of a dynamical system states over time considering that the system may have physical constraints on the control inputs and the states. In this work, we will use this analysis to answer the following question: "Given the attacker capability to manipulate one or more functionalities of the vehicle's automation system, is it possible to drive the vehicular platoon to unsafe state (collisions between two or more vehicles within the platoon)? If so, what is the speed of impact (collision)?"

A. Optimal Control Based Reachability

We use the optimal control based reachability method in order to compute the reachable set of a platoon undergoing an FDI attack. Using this method, the error state space is divided into a number of equidistant target points e_s and for each one of them an optimal control problem is solved to determine whether a feasible trajectory exists between initial states e_0 and the target states e_s. Mathematically, we seek a solution to the following optimization problem

$$\underset{\delta(.)}{\text{minimize}}\ J = \frac{1}{2}||Ce(m) - e_s||_2^2 \tag{28}$$

subject to

- initial error states.
- dynamics of the platoon, which are the error states, acceleration (control commands), and the FDI vector.
- constraints on the state, input, targeted sensors, and FDI vector.
- the FDI attack vector is determined such that the increment/decrement of the spoofed measurements is according to the attacked sensor(s) resolution.

The matrix C defines the target vehicle, by selecting its position and velocity errors from the state vector e. The target vehicle is where the attacker intends to cause a collision while the attacked vehicle is where the attacker injects the FDI attack vector. If a solution can be found for (28), then there is an attack sequence $\delta(.)$ which is able to minimize the distance between the final state of the platoon $e(m)$ and e_s, which means the attacker can cause the platoon to steer towards e_s. If, on the other hand, a solution does not exist, then the attacker cannot drive the platoon to the candidate states e_s.

Since we are mainly concerned to determine the safety of the vehicular platoon while experiencing an FDI attack, we will define e_s as only the unsafe points in the error state space, that is the points where the position error is equal to $-x_d$ (for collisions) and for different velocity errors (speed of impact). In order to solve the problem in (28) numerically, we need the following formulations

1) Evolution of Errors State Vector: For an initial state vector $e(0)$, the error coordinates of the platoon , given in (22), will develop over time for $k = 0, 1, \ldots, m$ as follows

$$\begin{aligned} e(1) &= Ae(0) + B\delta(0) \\ e(2) &= Ae(1) + B\delta(1) = A^2e(0) + AB\delta(0) + B\delta(1) \\ &\vdots \\ e(m) &= A^me(0) + A^{m-1}B\delta(0) + \cdots + B\delta(m-1) \end{aligned} \tag{29}$$

final error state vector can be rewritten as

$$e(m) = \bar{A}e(0) + \bar{B}\boldsymbol{\delta} \tag{30}$$

where

$$\begin{aligned} \bar{A} &= A^m \\ \bar{B} &= \begin{bmatrix} A^{m-1}B & A^{m-2}B & \dots & B \end{bmatrix} \\ \boldsymbol{\delta} &= \begin{bmatrix} \delta(0) & \delta(1) & \dots & \delta(m-1) \end{bmatrix}^T \end{aligned}$$

2) Initial Conditions and Constraints: We assume that the FDI attack begins once the platoon is at the steady-state, which means both desired separation and relative speed are achieved for all vehicles. Mathematically, the steady-state of the platoon is equivalent to zero position and velocity errors for all vehicles. Besides, in order to create realistic scenarios for the FDI attacks, we define the following constraints

- At any time sample, the attack vector must take a value between a predefined minimum $\delta_{\min}$ and maximum $\delta_{\max}$ values, as shown below

  $$\delta(k) \leq \delta_{\max} \text{ and } \delta(k) \geq \delta_{\min}$$

  which can be rewritten as follows

  $$\boldsymbol{\delta} \leq \boldsymbol{\delta}_{\max} \text{ and } \boldsymbol{\delta} \geq \boldsymbol{\delta}_{\min} \tag{31}$$

  where

  $$\begin{aligned} \boldsymbol{\delta}_{\max} &= \begin{bmatrix} \delta_{\max} & \dots & \delta_{\max} \end{bmatrix}^T \\ \boldsymbol{\delta}_{\min} &= \begin{bmatrix} \delta_{\min} & \dots & \delta_{\min} \end{bmatrix}^T \end{aligned}$$

- As shown in section IV-A, the attack vector has an effect on the calculation of commanded acceleration. Furthermore, each vehicle has physical acceleration limits. For those two reasons, the attack vector must not result in acceleration commands violates a predefined minimum $u_{\min}$ and maximum $u_{\max}$ limits once injected into the attack sensors, as shown below

  $$\begin{aligned} A_2e(k) + B_2\delta(k) &\leq u_{\max} \\ A_2e(k) + B_2\delta(k) &\geq u_{\min} \end{aligned}$$

  using (29), this constraint can be rewritten as follows

  $$\begin{aligned} K_1e(0) + K_2\boldsymbol{\delta} &\leq \boldsymbol{u}_{\max} \\ K_1e(0) + K_2\boldsymbol{\delta} &\geq \boldsymbol{u}_{\min} \end{aligned} \tag{32}$$

  where

  $$\begin{aligned} \boldsymbol{u}_{\max} &= \begin{bmatrix} u_{\max} & \dots & u_{\max} \end{bmatrix}^T \\ \boldsymbol{u}_{\min} &= \begin{bmatrix} u_{\min} & \dots & u_{\min} \end{bmatrix}^T \end{aligned}$$

- Each sensor has physical limits, that is the reading is always between a minimum $s_{\min}$ and a maximum $s_{\max}$ values. That means, once injected, the attack vector will not result in a spoofed measurement outside the attacked sensor limits, as shown below

  $$\begin{aligned} k_3e(k) + \delta(k) &\leq s_{\max} \\ k_3e(k) + \delta(k) &\geq s_{\min} \end{aligned}$$

  where k_3 is a row vector specifies error states corresponding to the attacked sensors. Using (29), this constraint can be rewritten as follows

  $$\begin{aligned} K_3e(0) + K_4\boldsymbol{\delta} &\leq \boldsymbol{s}_{\max} \\ K_3e(0) + K_4\boldsymbol{\delta} &\geq \boldsymbol{s}_{\min} \end{aligned} \tag{33}$$

  where

  $$\begin{aligned} \boldsymbol{s}_{\max} &= \begin{bmatrix} s_{\max} & \dots & s_{\max} \end{bmatrix}^T \\ \boldsymbol{s}_{\min} &= \begin{bmatrix} s_{\min} & \dots & s_{\min} \end{bmatrix}^T \end{aligned}$$

- No collision should be induced in the platoon before reaching the end of attack window (time), as shown below

  $$k_5e(k) + \delta(k) \leq \psi$$

  where k_5 is a row vector specifies the position errors in the state vector and ψ is the collision threshold, which is equal to $-x_d$ in our case. Using (29), this constraint is rewritten as follows

  $$K_5e(0) + K_6\boldsymbol{\delta} \leq \Psi \tag{34}$$

  where

  $$\Psi = \begin{bmatrix} \psi & \dots & \psi \end{bmatrix}^T$$

- Increment/decrement of the FDI attack vector is predefined using a certain resolution. For that reason, the range of possible values for $\delta(k)$ is also predefined and the solution of the problem in (28) is set to integers.

All constraints given in (31)-(34) can be combined in the following compact form

$$A_{ineq}\boldsymbol{\delta} \leq b_{ineq} \tag{35}$$

Definitions of K_1, K_2, K_3, K_4, K_5, K_6, A_{ineq}, and b_{ineq} are given in Appendix A.

3) Computation of FDI Reachable Sets: The cost function of the problem in (28) can be rewritten as

$$\begin{aligned} J &= \frac{1}{2}\left[(Ce(m) - e_s)^T(Ce(m) - e_s)\right] \\ &= \frac{1}{2}\left[e^T(m)C^TCe(m) - 2e_s^TCe(m) + e_s^Te_s\right] \end{aligned} \tag{36}$$

by substituting (30) into (36), we get

$$J = M_1\boldsymbol{\delta} + \boldsymbol{\delta}^TM_2\boldsymbol{\delta} + \text{other terms} \tag{37}$$

where

$$\begin{aligned} M_1 &= e^T(0)\bar{A}^TC^TC\bar{B} - e_s^TC\bar{B} \\ M_2 &= \bar{B}^TC^TC\bar{B} \end{aligned}$$

It should be noted that the "other terms" in (37) do not include any attack vector sequence and, hence, will be omitted since they do not affect the minimization of J. In summary, for

each one of the target states e_s of interest, the reachable set is determined by solving the following

$$\begin{aligned} \min_{\boldsymbol{\delta}} \quad & M_1\boldsymbol{\delta} + \boldsymbol{\delta}^T M_2 \boldsymbol{\delta} \\ \text{s.t.} \quad & A_{ineq}\boldsymbol{\delta} \leq b_{ineq} \end{aligned} \tag{38}$$

B. Hamilton Jacobian Reachability against MD attack

The Hamilton-Jacobi (HJ) reachability analysis is another popular and effective reachability method for examining the performance and safety properties of the dynamic system. It provides a formal way of modeling the attacker's capability in the system evolution, which benefits us in demonstrating the impact of the delay. Specifically speaking, we use it to find the answer to the following question: "Considering the attacker's capability to delay messages in multiple channels, will the platoon be driven into unsafe states (collision between two adjacent vehicles)?". The backward reachable set (BRS) answer this question by finding out all the possible initial states that can find its way to the target set at any time during some pre-defined time horizon. In this work, we use it to find out the BRS that can lead to a decent vehicle collision given the bound of delay and system evolution function.

In detail, we try to solve the viscosity solution of a time-dependent Hamilton Jacobi Issac equation $\mathscr{E}$. We let a function $v(x,t)$ be the viscosity solution of $\mathscr{E}$,

$$D_t v(x,t) + min[0, H(x, D_x v(x,t))] = 0 \tag{39}$$

where

$$\begin{aligned} H(x, D_x v(x,t)) &= \min_{b\in B} D_x v(x,t)^T \cdot f(x,a,b) \\ D_x v(x,t) &= \frac{\partial v(x,t)}{\partial x_i} = p^T \end{aligned} \tag{40}$$

Then, the zero sublevel set of v describes the BRS

$$\mathscr{G}_\tau = \{x \in \mathbb{R}^n \mid v(x,t) \leq 0\} \tag{41}$$

where $H(x, D_x v(x,t))$ is the Hamiltonian function. The basic idea of the attacker is to minimize the Hamiltonian by changing the control input. As the BRS increases, the Hamiltonian also increases. When the Hamiltonian reaches 0, we will stop and therefore correctly obtain the BRS.

The main drawback of this method lies in its exponential computational complexity with respect to the number of state variables. The HJ reachability analysis usually does not make any sense when the dimension of the state vector is larger than 5. Previous works [21], [23], [33] only deal with 3D state vector, which limits their applicability to demonstrate a more complex system. In this model, we use a 5D state vector to evaluate the system safety property against the MD attack.

1) System Formulation: Due to the size limitation of the state vector, we can only consider one adjacent vehicle pair at each simulation. 3 or more vehicles system will make no sense. In this formulation, we use vehicle 2 and vehicle 1 to represent the front and back vehicle in this pair. The state vector is defined as

$$\begin{bmatrix} e_1(t) \\ e_2(t) \\ e_3(t) \\ v_1(t) \\ a_1(t) \end{bmatrix} = \begin{bmatrix} x_2(t) - x_1(t) - h \cdot v_1 \\ v_2(t) - v_1(t) \\ a_2(t) - a_1(t) \\ v_1(t) \\ a_1(t) \end{bmatrix} \tag{42}$$

where $x_2(t)/x_1(t),\ v_2(t)/v_1(t),\ a_2(t)/a_1(t)$ represent the position, velocity and actual acceleration of the front/back vehicle respectively.

The system dynamic $f(x,\tau)$ we use is shown as

$$\begin{aligned} \dot{e_1}(t) &= e_2(t) - h a_1(t) \\ \dot{e_2}(t) &= e_3(t) \\ \dot{e_3}(t) &= -\frac{1}{\eta}\big(e_3(t)\big) + \frac{1}{\eta}\big[u_2(t) + \Delta(t,\tau_1) - \big(k_p e_1(t) \\ &\quad + k_d e_2(t) + k_a \cdot \big(u_2(t) + \Delta(t,\tau_1)\big)\big)\big] \\ \dot{v_1}(t) &= a_1(t) \\ \dot{a_1}(t) &= -\frac{1}{\eta} a_1(t) + \frac{1}{\eta}\big[k_p(e_1(t)) + k_d(e_2(t)) \\ &\quad + k_a \cdot \big(u_2(t) + \Delta(t,\tau_1)\big)\big] \end{aligned} \tag{43}$$

Where $u_2(t)/u_1(t)$ is the commanded acceleration of the preceding/following vehicle. $\Delta(t,\tau_1)$ is the acceleration discrepancy.

Then we can get the Hamiltonian in the following equation

$$\begin{aligned} &H(x,p) = \min_{\tau\in[0,\tau_{max}]} p^T \cdot f(x,\tau) \\ &= p_1\big(e_2(t) - h a_1(t)\big) + p_2 e_3(t) - p_3\eta^{-1} e_3(t) - p_3 \cdot \\ &\quad \eta^{-1}\big(k_p e_1(t) + k_d e_2(t)\big) + p_4 a_1(t) - p_5\eta^{-1} a_1(t) + \\ &\quad p_5 \cdot \eta^{-1}\big(k_p e_1(t) + k_d e_2(t)\big) + \frac{2}{\eta} \cdot k_a \left|p_3 + p_5\right| \cdot \\ &\quad \Delta(\tau_1) + \eta^{-1} \cdot \left|p_3(1-k_a) + p_5 k_a\right| u_2(t) \end{aligned} \tag{44}$$

where $p = [p_1, p_2, ..., p_5]$ is the partial derivative of the viscosity solution of the the terminal value HJI PDE w.r.t the state variables. The bound on the attacker input is $\Delta(\tau_1) \in [-j_u\tau_{max}, j_u\tau_{max}]$. The system input $u_2(t)$ is bounded by the maximum acceleration and the j_u. The attacker's objective is to find the τ_1 to minimize the Hamiltonian.

VI. Simulation Results and Analysis

In this section, we evaluate the severity of the FDI and MD attacks by analyzing the reachable sets, determined using the approaches described in Section V, for platoons undergoing the aforementioned attacks.

TABLE I: Reachable set for FDI attacks on a single range sensor

	V_1	V_2	V_3	V_4
$\delta_{x,1}$	$c_{1,2}$ (-1.995)	-	-	-
$\delta_{x,2}$	-	$c_{2,3}$ (-3.996)	-	-
$\delta_{x,3}$	-	-	$c_{3,4}$ (-4.005)	$c_{3,4}$ (-1.848) $c_{4,5}$ (-2.003)
$\delta_{x,4}$	-	-	-	$c_{4,5}$ (-5.987)

A. FDI Attacks

For FDI attacks, the approach explained in Section V-A is used to determine the reachable set of the attacked platoon. CPLEX solver was used to determine the integer solution of (38). Tables I and II show the reachable sets resulting from mounting FDI attacks against one range sensor and range-rate sensor, respectively, equipped on one vehicle in a platoon with ($n = 4$). In each table, $\delta_{x,i}$ and $\delta_{v,i}$ refer to the attacked range and range-rate sensors, respectively, equipped on the i^{th} attacked vehicle, V_i refers to the i^{th} target vehicle in the platoon, specified by C in (38), $c_{i,j}$ refers to a collision between the i^{th} and j^{th} vehicles, and the numbers shown in parenthesis are the maximum reachable speed of impact with respect to the two collided vehicles. For these results, resolution of the attacked sensor is selected as 0.5 m and 0.25 $\mathrm{m/s^2}$ for the range and range-rate sensors, respectively.

We can see in the above-mentioned tables that different impacts can be generated for different scenarios of FDI attacks. For example, attacking the range sensor of the 1^{st} vehicle in the platoon can cause the target vehicle V_1 and the preceding 2^{nd} vehicle to collide at a relative speed that could reach -1.955 $\mathrm{m/s}$. However, attacking the same sensor does not cause any accidents when the target is any vehicle other than the first one V_1 in the platoon, as shown in the first row of Table I. On the other hand, we can see in the second row of Table II that attacking the range-rate sensor of the 2^{nd} vehicle in the platoon can cause a crash at the target vehicle V_3 and, in addition, the 2^{nd} and 4^{th} vehicles although the last two were not part of the attacker's intention in the first place, which indicates the possibility of attacking one vehicle in the platoon while targeting (causing an accident) in another non-attacked vehicle in the same platoon. For these results, accidents occur when a time in the range of 90 to 120 seconds pass after the beginning of an attack.

Tables III and IV show the reachable set resulting from attacking two range sensors and range-rate sensors, respectively, equipped on two vehicles in the same platoon ($n = 4$). In each table, $\delta_{x,ij}$ and $\delta_{v,ij}$ refer to the attacked range and range-rate sensors, respectively, equipped on the i^{th} and j^{th} attacked vehicles. We see clearly that targeting two sensors, regardless of their types, on two different vehicles generates a bigger reachable set for the FDI attack as we can see fewer infeasible attack cases, shown with the (-) sign. Also, compared with the results of attacking one sensor, it is possible in some scenarios to cause collisions at greater speeds of impacts and more random non-attacked vehicles can be harmed. For example, by attacking range sensors of the first and second vehicles, it is possible to induce collision in the target vehicle V_4 and also in the third and fifth vehicles with a severe speed of impacts range from -3.6 to -4 $\mathrm{m/s}$, as shown in Table III. Furthermore, for these results, 40 to 60 seconds are needed to cause the first collision.

Tables V and VI show the reachable set resulting from attacking two range and range-rate sensors equipped on one or two vehicles, respectively, in the same platoon ($n = 4$).

TABLE II: Reachable set for FDI attacks on a single range-rate sensor

	V_1	V_2	V_3	V_4
$\delta_{v,1}$	$c_{1,2}$ (-1.977)	-	-	-
$\delta_{v,2}$	-	$c_{2,3}$ (-2.017)	$c_{2,3}$ (-1.959) $c_{3,4}$ (-1.992) $c_{4,5}$ (-2.210)	-
$\delta_{v,3}$	-	-	$c_{3,4}$ (-2.432)	$c_{3,4}$ (-3.715) $c_{4,5}$ (-2.003)
$\delta_{v,4}$	-	-	-	$c_{4,5}$ (-7.962)

TABLE III: Reachable set for FDI attacks on two range sensors

	V_1	V_2	V_3	V_4
$\delta_{x,12}$	$c_{1,2}$ (-2.021)	$c_{2,3}$ (-5.997)	$c_{2,3}$ (-1.873) $c_{3,4}$ (-2.004) $c_{4,5}$ (-2.130)	$c_{2,3}$ (-3.666) $c_{3,4}$ (-3.853) $c_{4,5}$ (-3.997)
$\delta_{x,13}$	$c_{1,2}$ (-4.003) $c_{3,4}$ (-3.860)	$c_{1,2}$ (-1.807) $c_{2,3}$ (-2.013)	$c_{3,4}$ (-7.984)	$c_{3,4}$ (-5.743) $c_{4,5}$ (-6.002)
$\delta_{x,14}$	$c_{1,2}$ (-3.988)	$c_{1,2}$ (-1.831) $c_{2,3}$ (-1.987) $c_{3,4}$ (-1.771)	$c_{1,2}$ (-1.468) $c_{2,3}$ (-1.508) $c_{3,4}$ (-1.982)	$c_{4,5}$ (-8.006)
$\delta_{x,23}$	-	$c_{2,3}$ (-4.003) $c_{3,4}$ (-4.008)	$c_{3,4}$ (-7.993)	$c_{3,4}$ (-5.723) $c_{4,5}$ (-6.005)
$\delta_{x,24}$	-	$c_{2,3}$ (-3.996) $c_{4,5}$ (-3.998)	$c_{2,3}$ (-1.833) $c_{3,4}$ (-1.997)	$c_{4,5}$ (-8.997)
$\delta_{x,34}$	-	-	$c_{3,4}$ (-3.566)	$c_{4,5}$ (-8.270)

TABLE IV: Reachable set for FDI attacks on two range-rate sensors

	V_1	V_2	V_3	V_4
$\delta_{v,12}$	$c_{1,2}$ (-8.004)	$c_{2,3}$ (-8.102)	$c_{3,4}$ (-4.193)	-
$\delta_{v,13}$	$c_{1,2}$ (-7.989) $c_{2,3}$ (-7.990)	$c_{2,3}$ (-7.984)	$c_{3,4}$ (-3.970)	-
$\delta_{v,14}$	$c_{1,2}$ (-6.003) $c_{3,4}$ (-5.921)	$c_{1,2}$ (-5.585) $c_{2,3}$ (-5.975) $c_{3,4}$ (-6.138)	$c_{3,4}$ (-7.981)	$c_{4,5}$ (-5.988)
$\delta_{v,23}$	-	$c_{2,3}$ (-5.967)	$c_{3,4}$ (-1.974)	$c_{3,4}$ (-1.982) $c_{4,5}$ (-2.015)
$\delta_{v,24}$	-	$c_{2,3}$ (-5.997) $c_{3,4}$ (-6.224)	$c_{3,4}$ (-7.984)	$c_{4,5}$ (-3.991)
$\delta_{v,34}$	-	-	$c_{3,4}$ (-7.977)	$c_{4,5}$ (-6.011)

TABLE V: Reachable set for FDI attacks on range & range-rate sensors

	V_1	V_2	V_3	V_4
$\delta_{x,1}$ $\delta_{v,1}$	$c_{1,2}$ (-2.010)	$c_{1,2}$ (-1.866) $c_{2,3}$ (-1.987) $c_{3,4}$ (-2.166)	-	-
$\delta_{x,2}$ $\delta_{v,2}$	-	$c_{2,3}$ (-1.991) $c_{3,4}$ (-2.593)	-	-
$\delta_{x,3}$ $\delta_{v,3}$	-	-	$c_{3,4}$ (-1.799)	$c_{4,5}$ (-1.791)
$\delta_{x,4}$ $\delta_{v,4}$	-	-	-	$c_{4,5}$ (-3.992)

TABLE VI: Reachable set for FDI attacks on two range & range-rate sensors

	V_1	V_2	V_3	V_4
$\delta_{x,12}$ $\delta_{v,12}$	$c_{1,2}$ (-7.984)	-	-	-
$\delta_{x,13}$ $\delta_{v,13}$	$c_{1,2}$ (-1.970) $c_{2,3}$ (-2.532)	-	-	-
$\delta_{x,14}$ $\delta_{v,14}$	$c_{1,2}$ (-4.027) $c_{2,3}$ (-4.460) $c_{3,4}$ (-4.701)	$c_{1,2}$ (-1.883) $c_{2,3}$ (-2.004) $c_{3,4}$ (-2.254)	$c_{3,4}$ (-6.011)	$c_{4,5}$ (-3.974)
$\delta_{x,23}$ $\delta_{v,23}$	-	$c_{2,3}$ (-2.015)	-	-
$\delta_{x,24}$ $\delta_{v,24}$	-	$c_{2,3}$ (-7.952) $c_{3,4}$ (-8.118)	$c_{3,4}$ (-3.963)	$c_{4,5}$ (-3.985)
$\delta_{x,34}$ $\delta_{v,34}$	-	-	$c_{3,4}$ (-1.983)	-

Finally, we increased the size of the platoon to ($n = 5$) and determined the reachable set resulting from attacking two range-rate sensors on two different vehicles. Table VII shows the results for these attack cases.

Similarly, we have also conducted the same analyses to determine the reachable sets of FDI attacks on a single/double range or range-rate sensors however we neglected the constraint regarding the sensor resolution, which means that $\delta(.)$ can take any continuous values between $\delta_{\min}$ and $\delta_{\max}$. The reason for that was to compare the results with that shown in Tables tables I to VI. In the case of continuous $\delta(.)$, resulting reachable set was bigger in terms of the number of induced collisions and the magnitude of the speed of impact. However, that represents an unrealistic case since it is not feasible to inject false-data that could take any arbitrary value.

In summary, whether it is an attack on a single or multiple sensors, the reachability analysis results shown in the aforementioned tables indicate that the impacts of FDI attacks on the sensors of platooned vehicles are serious and need to be considered during the design of platooning controllers, which rely on such sensors, for those vehicles. Although some attacks cases do not result in any accidents, shown with the (-) sign in the tables, FDI attacks are still able to generate severe collisions with a speed of impact between -2 to -8 $\mathrm{m/s}$. Additionally, FDI attacks can induce impacts at random vehicles in the platoon by attacking on-board sensors of other vehicles in the same platoon.

For the purpose of detecting potential attacks against on-board sensors, we suggest the following. First, a simple approach is to check whether or not the sensor's readings are reasonable. For instance, if those readings are not within the sensor's normal range or if the increments/decrements of the successive sensor measurements do not agree with sensor's frequency, then the sensor may be attacked. Second, another approach for detection is sensor redundancy. For instance, equipping multiple sensors on-board to measure the same quantity, range or range-rate, or installing units on road with

TABLE VII: Reachable set for FDI attacks on two range-rate sensors ($n = 5$)

	V_1	V_2	V_3	V_4	V_5
$\delta_{v,12}$	$c_{1,2}$ (-5.221)	$c_{2,3}$ (-8.343)	-	-	-
$\delta_{v,13}$	$c_{1,2}$ (-7.559)	$c_{2,3}$ (-7.401)	$c_{3,4}$ (-3.720)	-	-
$\delta_{v,14}$	$c_{1,2}$ (-7.022) $c_{3,4}$ (-6.631)	$c_{2,3}$ (-6.775) $c_{3,4}$ (-6.138)	$c_{3,4}$ (-6.981)	$c_{4,5}$ (-4.218)	-
$\delta_{v,15}$	$c_{1,2}$ (-5.277)	-	-	-	$c_{5,6}$ (-6.011)
$\delta_{v,23}$	-	$c_{2,3}$ (-5.967)	$c_{3,4}$ (-1.974)	-	-
$\delta_{v,24}$	-	$c_{2,3}$ (-6.127)	$c_{3,4}$ (-5.226)	$c_{4,5}$ (-4.991)	-
$\delta_{v,25}$	-	$c_{2,3}$ (-7.034)	-	-	$c_{5,6}$ (-6.501)
$\delta_{v,34}$	-	-	$c_{3,4}$ (-8.244)	$c_{4,5}$ (-7.935)	-
$\delta_{v,35}$	-	-	$c_{3,4}$ (-7.900)	-	$c_{5,6}$ (-6.113)
$\delta_{v,45}$	-	-	-	-	$c_{5,6}$ (-5.731)

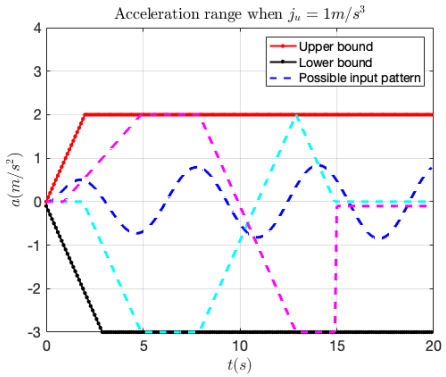


Fig. 4: Acceleration range of the front vehicle

I2V (Infrastructure-2-Vehicle) and V2I capabilities [41] where vehicle's information can be exchanged. By generating multiple readings and then comparing them, spoofed measurements will be easy to detect.

B. MD Attack

In this work, we evaluate the influence of delay by finding the BRS of this delayed system. We use the code from [42] for reference, which is famous in solving HJ reachability. Also, we adapt it to fit our own system dynamic and attacker settings. The simulation platform is MATLAB 2018b on PC.

1) System Parameters and Target Set: We assume a 10-vehicle string that is running in the steady state. Suddenly, the lead vehicle changes its speed and therefore following vehicles react based on the CACC. The attacker observes this string disturbance and simultaneously begins to delay the inter-vehicle messages.

We assume the initial vehicle platoon is in the steady state with a common speed $30\,\mathrm{m/s}$. We constrain the speed between 0 to $35\,\mathrm{m/s}$ in our simulation which helps avoid backward driving. Also, we assume the maximum acceleration ranges from $-3\,\mathrm{m/s^2}$ to $2\,\mathrm{m/s^2}$ which avoids uncomfortable brakes or pushes for passengers. We are considering a 20-second time span, which is sufficient for the following vehicles to evaluate and react to the potential danger.

The target set $\mathscr{G}_T$ is called a 'decent' collision, which is mathematically defined as

$$\begin{aligned}\mathscr{G}_T = \big[e_1 \le 0 \quad e_2 \le -v_c \quad -5 \le e_3 \le 5 \\ 0 \le v_1 \le 35 \quad -3 \le a_1 \le 2\big]^T\end{aligned} \tag{45}$$

$\mathscr{G}_T$ is an area when the inter-vehicle space is 0 or negative, the collision speed between two vehicles are larger than $v_c\mathrm{m/s}$ with $v_c \ge 0$. Other dimensions fall within its own range.

2) Constraints and Initial Condition: In order to examine the impact of delay on different vehicle dynamics, we assume j_u have three choices: $1\,\mathrm{m/s^3}$,$2\,\mathrm{m/s^3}$ and $3\,\mathrm{m/s^3}$ [24].

Meanwhile, the string stability means that the inter-vehicle space error will attenuate along the vehicle string. For a homogeneous vehicle string with the same platoon parameter, we can say that the maximum and minimum space error will simultaneously occur at the first and last pair of vehicles. However, the way we define the input of the lead vehicle in

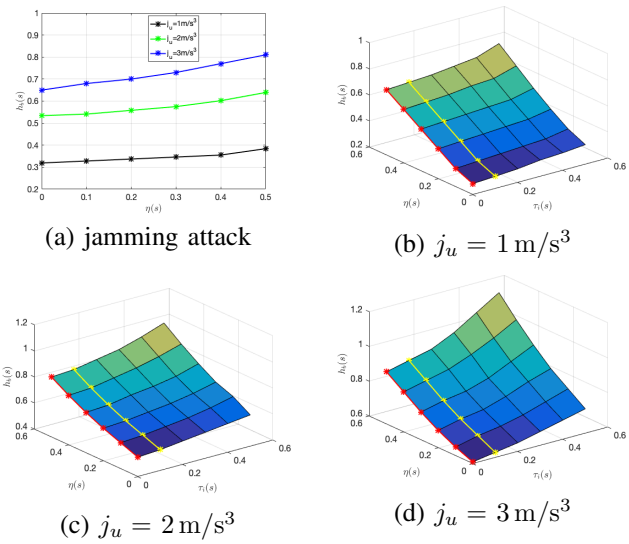
section iii has already included all possible patterns, which means the input of the last vehicle pair will fall within the range of the first pair. Therefore, we only need to show the plot the first pair to demonstrate the impact of the delay.

In Figure 4, we use $j_u = 1\,\mathrm{m/s^3}$ as an example to show the bound of u_2 by the physical limit of the vehicle, the lead vehicle input profile. In detail, the red and black solid line are the upper and lower bound of the acceleration range when $j_u = 1\,\mathrm{m/s^3}$ respectively. The colorful dashed lines represent 3 possible input patterns inside this range. In reality, the acceleration pattern can behave randomly as long as it is inside the range bounded by the red-line upper bound and black-line lower bound. The upper or lower bound will reach to the maximum/minimum value faster as the j_u becomes larger, which means a better engine. This is a more comprehensive way to model the input u_2 as a range because we consider all possible inputs rather than some specific input profile used by some previous work. Also, the way that we bound the u_2 will make the analysis realistic because we consider the physical limits of vehicles.

3) Delay Impact Evaluation: In this part, we want to demonstrate the impact of delay by answering the following question: Given the final target set (collision), a constant time headway time h, and a maximum delay bound, will the steady state be included in the initial BRS? If so, it means that, in the worst case, the attacker can always find at least a delay sequence to drive the system into the targeted collision from the steady state. Since the platoon designer can choose headway time h to represent different safety preference. Usually, a larger headway time results in a safer vehicle string setting. We will explore what is the boundary headway time h_b of the BRS. This boundary h_b tells us that, if this platoon has a $h \leq h_b$, it may face a significant collision under the worst case.

We adopt the control parameter set that we find in previous literature as $k_p = k_d^2 = 0.25$ and $k_a = 0.5$ [28]. We have two objectives in this simulation. First, we want to explain why we don't cover the periodic jamming attack, by comparing its impact with MD attack. Here, we assume that the periodic jamming attack jams every 1 out of 5 messages. Consecutive message jamming is considered more easily detectable than periodic message jamming [43], which may turn the platoon into a safer ACC mode. Similarly, a higher jamming frequency can raise the attention of platoon. Second, we want to demonstrate the effectiveness and severity of the MD attack.

Figure 5 (a) is the h_b under the aforementioned periodic jamming attack. Figure 5 (b), (c) and (d) demonstrate the h_b against different actuation delay η and delay upper bound $\tau_{1,max}$, where the red and yellow curves are the h_b under no delay and $\tau_{1,max} = 0.1$ respectively. First of all, if we compare black curve in the Figure 5 (a) with red and yellow curves in (b), we observe that the h_b of the jamming attack (black) is only slightly larger than no delay (red) case. Meanwhile, it is less than the MD attack with $\tau_{1,max} = 0.1s$ (yellow). We obtain similar results for other j_u values. In other words, the periodic jamming attack has a less severe impact on platoon safety than the $\tau_{1,max} = 0.1$ case because it needs a smaller headway time to avoid the crash. So we don't



(a) jamming attack (b) $j_u = 1\,\mathrm{m/s^3}$ (c) $j_u = 2\,\mathrm{m/s^3}$ (d) $j_u = 3\,\mathrm{m/s^3}$

Fig. 5: h_b under different j_u

further compare with the periodic jamming in our simulation. Besides, as η and $\tau_{1,max}$ increases, h_b need to increase, which means vehicles need a much more headway time to stay safe. Besides, a larger j_u enables faster changes in deceleration and accordingly it needs a larger h_b to avoid potential crash, which is demonstrated in the results. And also, η has a larger impact than the $\tau_{1,max}$. It is also reasonable since the larger actuation delay means the control system cannot react in a timely manner, leading to worse consequences than only delaying the acceleration information.

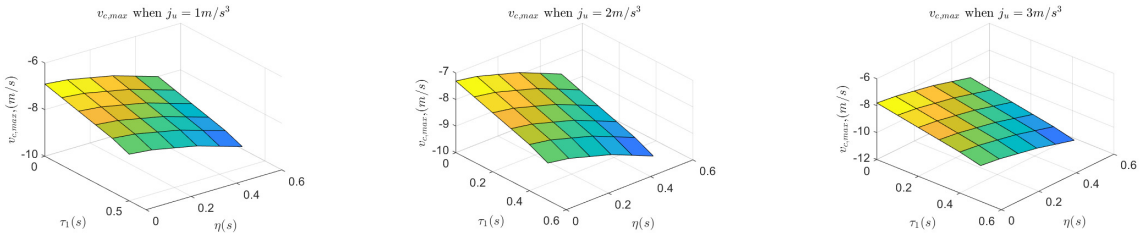
After we obtain the boundary h, we may consider another question: given all the platoon parameters, what is the most severe collision that may happen starting from some common unsteady initial state. We fix the platoon parameters and change the target set (increase the collision speed v_c). We found that the BRS will gradually shrink as we increase v_c. When it does not contain any reasonable initial states, we may say that this v_c is the most severe crash that can happen under certain initial common platoon states.

We first define the possible initial condition range (PICR).

- Initial inter-vehicle distance is between 0 and $40\,\mathrm{m}$.
- Initial relative speed ranges from -5 to $5\,\mathrm{m/s}$.
- Initial lead vehicle speed ranges from 10 to $30\,\mathrm{m/s}$.

Then we change the target set by changing the crash speed v_c and therefore find the maximum crash speed $v_{c,max}$ such that the corresponding BRS has intersection with the PICR. In other words, if $v_c > v_{c,max}$, the resulting BRS would have no intersection with the PICR. In Figure 6, we plot the $v_{c,max}$ against different η and delay bound $\tau_{1,max}$. We can see that as actuation delay and $\tau_{1,max}$ increases, the $v_{c,max}$ also increases, which means the platoon vehicles may reach worse collisions under larger delays. Besides, a larger j_u represent a harder brake ability and accordingly, it may lead to a more severe crash. The $v_{c,max}$ is at least $6.5\,\mathrm{m/s}$ for all possible cases, which represents a severe crash.

In summary, Figure 5 shows the minimum headway time needed for the given actuation delay η as well as the maximum



(a) $j_u = 1\,\mathrm{m/s^3}$ (b) $j_u = 2\,\mathrm{m/s^3}$ (c) $j_u = 3\,\mathrm{m/s^3}$

Fig. 6: $v_{c,max}$ under different j_u

delay τ_{max} to avoid severe collisions. Because the time duration is 20 seconds and minimum headway time is achieved on the boundary of the reachable set, it takes at least 20 seconds for any pair of vehicles to collide from the defined steady state under the minimum headway time h_b given η and τ_{max}. Moreover, Figure 7 shows the worst-case collision scenario when the platoon starts from a reasonable initial state. We plot two examples that the MD attack leads to collisions using the same control parameter and initial state during the 20s time span. The message delay during this period is always 0.3s, and η is 0.5s and j_u is $1\,\mathrm{m/s^3}$. The front vehicle first accelerates to the max speed and then applies a full break till the end. The headway time is $0.5\,\mathrm{s}$ and $0.6\,\mathrm{s}$, respectively. The collision happens when the curves fall below the red line. We can see that the worst-case collisions happen when the headway time is $0.6\,\mathrm{s}$, which corresponds to the value in Figure 5 (b) where the h_b is around $0.64\,\mathrm{s}$.

Several previous papers focus on the collision analysis have different system models or attacker assumptions, such as [29] and [44], whose result is not directly comparable with ours. Compared with previous works focusing on string stability under the same control law, our results show that the collision can happen when the platoon is either string stable or unstable. [28] shows the minimum headway time h_s that needed to keep the string stability. We compare the h_b and h_s under the same actuation delay η. From Fig 8 (b), there are 4 regions divided by two curves. Region 1 represents the headway time that results in collision-free and string stability. Region 4 represents the headway time that makes the platoon suffer from both collision and string instability. Region 2 means string unstable but collision-free, Region 4 represents the opposite to the region 2. We can see that string instability and collisions are

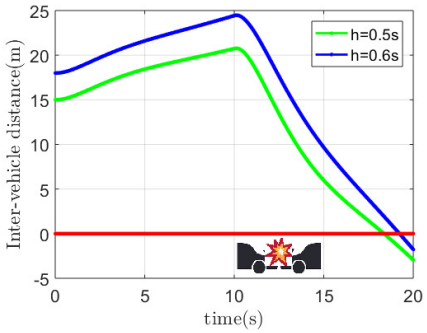


Fig. 7: Inter-vehicle distance when h =0.5 s and h =0.6 s under τ_{max} =0.3 s, η =0.5 s. j_u =1 $\mathrm{m/s^3}$

positively correlated but there is no definite causality between these two conditions. In [35], authors show that jamming is able to result in a more severe impact on string stability. However, jamming can be regarded as infinite delay attack, which is obviously stronger if jamming is persistent. Besides, it is easily detectable while MD attack is stealthy.

4) Finding the Optimal Delay Attack: HJ reachability also tells us the optimal control sequence of the attacker.

$$\begin{aligned} a_d^* =& \underset{\tau_1}{\text{argmin}} \quad p^T \cdot f(x, \tau_1) \\ =& \underset{\tau_1}{\text{argmin}} \quad c(t) + \frac{2}{\eta} \cdot k_a \left| p_3 + p_5 \right| \cdot \Delta(\tau_1) \\ s.t. \quad & 0 \le \tau_1 \le \tau_{max} \\ & |\Delta(\tau_1)| \le j_t \cdot \tau_1 \end{aligned}$$

where a_d^* is the optimal control for the attacker, $c(t)$ represents the sum of all items in the $p^T \cdot f(x, \tau_1)$ that does not contain the decision variable τ_1. It is a very simple optimization problem. The objective value will be achieved as long as the $\tau_1 = \tau_{max}$ and $\Delta(\tau_1) = j_t \cdot \tau_1$. It means that, if we start the platoon from the state inside BRS, the attacker will never miss the target set as long as it chooses the largest delay τ_{max} all the time.

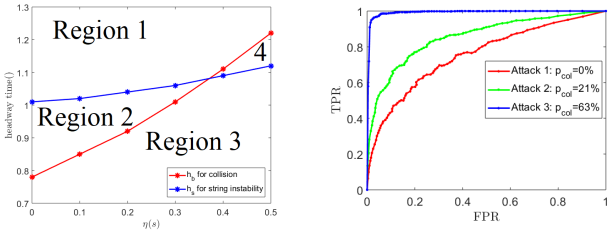
5) Defenses Against MD Attack: Here we discuss possible countermeasures to the MD attack, including attack detection and mitigation. For detection, we adopt the idea of anomaly detection by comparing measured message delay sequences with normal channel-induced delay distributions. That is, each vehicle i can locally collect a sequence of N consecutive V2V messages received from another vehicle j, and compute the delay $\{\tau_1, \tau_2, ..., \tau_N\}$ by subtracting the received time by the timestamp from the sender (since messages are timestamped before transmission in DSRC and digital signatures are used to verify the timestamps, their integrity is protected from the attacker. Of course this requires time synchronization between vehicles which can be realized by GPS signals). Then vehicle i can compute the distribution $\tilde{\mathcal{P}}_\tau$ of the sampled delay sequence. If we have the normal message delay distribution $\mathcal{P}_\tau$ induced by the communication channel, vehicle i can then compare the two distributions (e.g., calculating some distance metric such as KL-divergence [45] and compare with a threshold, or using Chi-Square test [46] to examine whether those measured samples actually belong to $\mathcal{P}_\tau$).

One major challenge is how to obtain the normal message delay distribution for the detection algorithm. Note that, normal message delay includes both physical-channel propagation delays, the delay from MAC/link layer (i.e., contention delay and re-transmissions due to collision), and processing delay

at the source/destinations. The first delay is predictable since it can be computed by dividing the inter-vehicle distance by the speed of light. The third one is usually very small which can be neglected (or can be modeled for different transceivers). The MAC delay constitutes the majority and is less predictable since it depends on many factors including: wireless channel conditions (which can change rapidly in a mobile environment), vehicle (device) density, communication/messaging rate and patterns, as well as network topology. While several existing studies [47], [48], [49] published experimental results about the message delay distribution under different real-world vehicular networking settings, they only provide general statistics for reference, which may not be applicable to the specific channel/traffic environment in the real-world.

An alternative approach is to introduce a training process to obtain an initial normal message delay distribution/profile $\tilde{\mathcal{P}}_1$ for each V2V link (can be done locally), and update the profile over time (by keeping a constantly moving window of delay measurements). The previous profile is used to authenticate the current measured delay sequence profile (e.g., comparing the distance with a threshold or Chi-Square test), and if the current profile is legitimate, it is used to authenticate the next delay sequence. This scheme can work as long as the initial training is secured (e.g., when a vehicle first starts and enters the road, assuming no attack at this point), and the channel/traffic environment does not change too abruptly. This is reasonable since DSRC specifies that V2V messaging rates shall be no less than 10/s, and the network topology is unlikely to change significantly within a second (where 10 messages can be used to build the profile). One may wonder if the attacker can gradually inject a tiny but increasing amount of delay for a long time to spoof the above profile updating process without being noticed such as in [50]. However, in order to launch this stealthy online hill-climbing-style attack, attackers have to send vehicles to physically follow the platoon for a long time, which is too costly and easy to be detected.

In addition, there is an intrinsic trade-off between the MD attack's stealthiness and effectiveness. While the attacker can try to lower the probability of detection, it also reduces the success rate (e.g., in terms of probability of causing vehicle collisions). On the one hand, the optimal MD attack is achieved when injecting a constant maximum delay (tolerated by the link layer, e.g., 0.5s) during a consecutive time period, which is quite easy to detect. On the other hand, if the attacker injects delays periodically or randomly, not all messages are heavily delayed which will be less effective. This can be seen by our simulation results in Fig. 8 (a). In this simulation, we set η=0.5 s, h =0.6 s, time duration as 20 s. The vehicle pair starts from the steady state with speed 25 m/s. The lead vehicle first accelerates to 35 m/s and then brakes ($j_u = 3\,\mathrm{m/s^3}$) until it stops. The control parameters are the same as the previous simulation. For normal delay distribution, we adopt the CDF of the message delay for single hop-broadcasting in [49], which is approximately a Gaussian distribution of mean 100 ms and STD 30 ms. We use 3 sets of delay distributions in the simulated MD attacks, whose distribution mean are 150 ms, 200 ms, and 400 ms respectively, with the same STD as 30 ms. The attacker delays every message by a value drawn



(a) Collision vs String Instability (b)Detection rate vs p_{col}

Fig. 8: Result evaluation and comparison

from the distribution. Then we run Monte-Carlo simulation to get the collision probability p_{col}. Also the Chi-Square test is used to detect the MD attack. The results show that a larger deviation (blue curve) between the normal and attack delay distributions leads to a higher vehicle collision possibility but is also much easier to be detected.

After detection, we need to mitigate the MD attack. A naive way is to lower the weight k_a in the control equation, which can lower the error caused by the MD attack. Platooned vehicles can increase the k_a when the detected message delay is small, and decrease k_a when the detected message delay is high. However, if the weight is too small, the received information will have less functionality in the control law. Besides, there are some works on how to mitigate the message delays. Besselink et.al [25] adopts a delay-based spacing policy to handle the system and message delay in the platoon control systems. Zeng et.al [51] optimizes the control system to achieve the maximum derived wireless system reliability under channel delay. In the real-world, a combination of such methods can be adopted. One can also fall back to ACC without using the V2V messages, which is less desirable. Delay-robust and efficient platooning control algorithms will be part of our future work.

VII. Conclusion

This paper adopts the reachability analysis method to demonstrate and evaluate the impact of two more realistic attacks, FDI attack, and MD attack, on the safety performance of the two platoon control techniques, ACC and CACC, respectively. The error evolution of the system state is derived, which is combined with the attacker's capability to describe the system dynamics in the reachability analysis. We also specify the reachable sets under different attack bounds to tell the collision severity of each attack. Simulation results show impacts of both attacks on the sensors or the communication channels of platooned vehicles can definitely lead to severe crashes within a specific time span. The collision may happen even before the platoon string becomes unstable compared with other works. Therefore both attacks need to be taken into account when designing the platoon upper-level controllers.

APPENDIX A
DEFINITION OF MATRICES

$$B_1 = \begin{bmatrix} & \mathbf{0}_{(n\times n)} & & & \\ -1 & 1 & 0 & \dots & 0 \\ 0 & -1 & 1 & \dots & 0 \\ \vdots & & & & \\ 0 & 0 & 0 & \dots & -1 \end{bmatrix}$$

$$A_2 = \begin{bmatrix} k_p & 0 & \dots & 0 & k_d & 0 & \dots & 0 \\ -k_p & k_p & \dots & 0 & -k_d & k_d & \dots & 0 \\ \vdots & & & & & & & \\ 0 & 0 & \dots & k_p & 0 & 0 & \dots & k_d \end{bmatrix}$$

$$B_{2,x} = \begin{bmatrix} k_p & 0 & \dots & 0 \\ 0 & k_p & \dots & 0 \\ \vdots & & & \\ 0 & 0 & \dots & k_p \end{bmatrix}, B_{2,v} = \begin{bmatrix} k_d & 0 & \dots & 0 \\ 0 & k_d & \dots & 0 \\ \vdots & & & \\ 0 & 0 & \dots & k_d \end{bmatrix}$$

$$B_{2,xv} = \begin{bmatrix} B_{2,x} & B_{2,v} \end{bmatrix}$$

$$K_1 = \begin{bmatrix} A_2 & A_2A & \dots & A_2A^{m-1} \end{bmatrix}^T$$

$$K_2 = \begin{bmatrix} B_2 & 0 & \dots & 0 \\ A_2B & B_2 & \dots & 0 \\ & \vdots & & \\ A_2A^{m-2}B & A_2A^{m-3}B & \dots & B_2 \end{bmatrix}$$

$$K_3 = \begin{bmatrix} k_3 & k_3A & \dots & k_3A^{m-1} \end{bmatrix}^T$$

$$K_4 = \begin{bmatrix} I & 0 & \dots & 0 \\ k_3B & I & \dots & 0 \\ & \vdots & & \\ k_3A^{m-2}B & k_3A^{m-3}B & \dots & I \end{bmatrix}$$

$$K_5 = \begin{bmatrix} k_5 & k_5A & \dots & k_5A^{m-1} \end{bmatrix}^T$$

$$K_6 = \begin{bmatrix} 0 & 0 & \dots & 0 \\ k_5B & 0 & \dots & 0 \\ & \vdots & & \\ k_5A^{m-2}B & k_5A^{m-3}B & \dots & 0 \end{bmatrix}$$

$$A_{ineq} = \begin{bmatrix} I & -I & K_2 & -K_2 & K_4 & -K_4 & K_6 \end{bmatrix}^T$$

$$b_{ineq} = \begin{bmatrix} \boldsymbol{\delta}_{\max} & -\boldsymbol{\delta}_{\min} & \boldsymbol{u}_{\max} - K_1e(0) & -\boldsymbol{u}_{\min} + K_1e(0) \\ \boldsymbol{s}_{\max} - K_3e(0) & -\boldsymbol{s}_{\min} + K_3e(0) & \Psi - K_3e(0) \end{bmatrix}^T$$

$$e_{cacc}(t) = \begin{bmatrix} e_{x1}(t) & \dots & e_{xn}(t) & e_{v1}(t) & \dots & e_{vn}(t) \\ e_{a1}(t) & \dots & e_{an}(t) \end{bmatrix}^T$$

$$a(t) = \begin{bmatrix} a_1(t) & a_2(t) & \dots & a_n(t) \end{bmatrix}^T$$

$$u_{cacc}(t) = \begin{bmatrix} u_1(t) & u_2(t) & \dots & u_n(t) \end{bmatrix}^T$$

$$A_{cacc} = \begin{bmatrix} \mathbf{0}_{(n\times n)} & \boldsymbol{I}_{(n\times n)} & \mathbf{0}_{(n\times n)} \\ \mathbf{0}_{(n\times n)} & \mathbf{0}_{(n\times n)} & \boldsymbol{I}_{(n\times n)} \\ \mathbf{0}_{(n\times n)} & \mathbf{0}_{(n\times n)} & -\eta^{-1}\boldsymbol{I}_{(n\times n)} \end{bmatrix}$$

$$M_{cacc} = \begin{bmatrix} -h\boldsymbol{I}_{(n\times n)} & \mathbf{0}_{(n\times n)} & \mathbf{0}_{(n\times n)} \end{bmatrix}^T$$

$$B_{cacc} = \begin{bmatrix} & \mathbf{0}_{(2n\times n)} & & & \\ -\eta^{-1} & \eta^{-1} & 0 & \dots & 0 \\ 0 & -\eta^{-1} & \eta^{-1} & \dots & 0 \\ \vdots & & & & \\ 0 & 0 & 0 & \dots & -\eta^{-1} \end{bmatrix}$$

REFERENCES

[1] E. Coelingh and S. Solyom, "All aboard the robotic road train," *IEEE Spectrum*, vol. 49, pp. 34–49, 2012.

[2] E. van Nunen, M. R. J. A. E. Kwakkernaat, J. Ploeg, and B. D. Netten, "Cooperative competition for future mobility," *IEEE Transactions on ITS*, vol. 13, no. 3, pp. 1018–1025, Sep. 2012.

[3] K.-Y. Liang, J. Mårtensson, and K. H. Johansson, "Fuel-saving potentials of platooning evaluated through sparse heavy-duty vehicle position data," *2014 IEEE Intelligent Vehicles Symposium Proceedings*, p. 1061, 2014.

[4] R. Rajamani, *Vehicle Dynamics and Control*, ser. Mechanical Engineering Series. Springer, 2011. [Online]. Available: https://books.google.com/books?id=eoy19aWAjBgC

[5] B. van Arem, C. J. G. van Driel, and R. Visser, "The impact of cooperative adaptive cruise control on traffic-flow characteristics," *IEEE Transactions on ITS*, vol. 7, no. 4, pp. 429–436, Dec 2006.

[6] S. Oncu, J. Ploeg, N. van de Wouw, and H. Nijmeijer, "Cooperative adaptive cruise control: Network-aware analysis of string stability," *IEEE Transactions on ITS*, vol. 15, no. 4, pp. 1527–1537, Aug 2014.

[7] J. Wang and R. Rajamani, "Adaptive cruise control system design and its impact on highway traffic flow," in *Proceedings of the 2002 American Control Conference*, vol. 5, May 2002, pp. 3690–3695 vol.5.

[8] R. van der Heijden and T. Lukaseder, "Analyzing attacks on cooperative adaptive cruise control (cacc)," in *2017 IEEE VNC*, Nov 2017, p. 45.

[9] M. Jagielski, N. Jones, C.-W. Lin, C. Nita-Rotaru, and S. Shiraishi, "Threat detection for collaborative adaptive cruise control in connected cars," in *Proceedings of the 11th ACM Conference on Wisec*. New York, NY, USA: ACM, 2018, pp. 184–189. [Online]. Available: http://doi.acm.org/10.1145/3212480.3212492

[10] A. A. Peters, R. H. Middleton, and O. Mason, "Leader tracking in homogeneous vehicle platoons with broadcast delays," *Automatica*, vol. 50, no. 1, pp. 64–74, 2014.

[11] A. Alipour-Fanid, M. Dabaghchian, H. Zhang, and K. Zeng, "String stability analysis of cooperative adaptive cruise control under jamming attacks," in *2017 IEEE 18th International Symposium on High Assurance Systems Engineering (HASE)*. IEEE, 2017, pp. 157–162.

[12] R. M. Gerdes, C. Winstead, and K. Heaslip, "Cps: an efficiency-motivated attack against autonomous vehicular transportation," in *Proceedings of the 29th Annual CSAC*. ACM, 2013, pp. 99–108.

[13] D. D. Dunn, S. A. Mitchell, I. Sajjad, R. M. Gerdes, R. Sharma, and M. Li, "Regular: Attacker-induced traffic flow instability in a stream of semi-automated vehicles," in *2017 47th Annual IEEE/IFIP DSN*, June 2017, pp. 499–510.

[14] B. DeBruhl, S. Weerakkody, B. Sinopoli, and P. Tague, "Is your commute driving you crazy?: a study of misbehavior in vehicular platoons," in *Proceedings of the 8th ACM Wisec*. ACM, 2015, p. 22.

[15] S. Dadras, R. M. Gerdes, and R. Sharma, "Vehicular platooning in an adversarial environment," in *Proceedings of the 10th ACM Symposium on ICCS*. ACM, 2015, pp. 167–178.

[16] C. Yan, W. Xu, and J. Liu, "Can you trust autonomous vehicles: Contactless attacks against sensors of self-driving vehicle," *DEF CON*, vol. 24, 2016.

[17] J. Petit, B. Stottelaar, M. Feiri, and F. Kargl, "Remote attacks on automated vehicles sensors: Experiments on camera and lidar," *Black Hat Europe*, vol. 11, p. 2015, 2015.

[18] R. Chauhan, R. M. Gerdes, and K. Heaslip, "Attack against an fmcw radar," in *Proceedings of Embedded Security in Cars Conference*, 2014.

[19] Y. Man, M. Li, and R. Gerdes, "Poster: Perceived adversarial examples," in *IEEE Symposium on Security and Privacy*, no. 2019, 2019.

[20] M. Amoozadeh and A. Raghuramu, "Security vulnerabilities of connected vehicle streams and their impact on cooperative driving," *IEEE Communications Magazine*, vol. 53, no. 6, pp. 126–132, 2015.

[21] I. M. Mitchell, A. M. Bayen, and C. J. Tomlin, "A time-dependent hamilton-jacobi formulation of reachable sets for continuous dynamic games," *IEEE TAC*, vol. 50, no. 7, pp. 947–957, July 2005.

[22] M. Chen, Q. Hu, C. Mackin, J. F. Fisac, and C. J. Tomlin, "Safe platooning of unmanned aerial vehicles via reachability," *2015 54th IEEE Conference on Decision and Control (CDC)*, pp. 4695–4701, 2015.

[23] R. Baier and M. Gerdts, "A computational method for non-convex reachable sets using optimal control," in *2009 ECC*, Aug 2009, p. 97.

[24] R. W. Diller, "Apparatus and method responsive to vehicle jerk for actuating a passenger restraint system in a passenger vehicle," Dec. 4 1990, uS Patent 4,975,850.

[25] B. Besselink and K. H. Johansson, "String stability and a delay-based spacing policy for vehicle platoons subject to disturbances," *IEEE Transactions on Automatic Control*, vol. 62, no. 9, pp. 4376–4391, 2017.

[26] X. Liu, A. Goldsmith, S. S. Mahal, and J. K. Hedrick, "Effects of communication delay on string stability in vehicle platoons," in *ITSC 2001. 2001 IEEE Intelligent Transportation Systems. Proceedings (Cat. No. 01TH8585)*. IEEE, 2001, pp. 625–630.

[27] I. G. Jin and G. Orosz, "Dynamics of connected vehicle systems with delayed acceleration feedback," *Transportation Research Part C: Emerging Technologies*, vol. 46, pp. 46–64, 2014.

[28] G. J. Naus, R. P. Vugts, J. Ploeg, and M. Steinbuch, "String-stable cacc design and experimental validation: A frequency-domain approach," *IEEE TVT*, vol. 59, no. 9, pp. 4268–4279, 2010.

[29] S. H. Kafash, J. Giraldo, C. Murguia, A. A. Cardenas, and J. Ruths, "Constraining attacker capabilities through actuator saturation," in *2018 American Control Conference (ACC)*. IEEE, 2018, pp. 986–991.

[30] A. A. Kurzhanskiy and P. Varaiya, "Ellipsoidal techniques for reachability analysis of discrete-time linear systems," *IEEE Transactions on Automatic Control*, vol. 52, no. 1, pp. 26–38, Jan 2007.

[31] Y. Zhou and J. S. Baras, "Reachable set approach to collision avoidance for uavs," in *2015 54th IEEE CDC*, Dec 2015, pp. 5947–5952.

[32] M. Chen, S. Bansal, J. F. Fisac, and C. J. Tomlin, "Robust sequential trajectory planning under disturbances and adversarial intruder," *IEEE Transactions on Control Systems Technology*, pp. 1–17, 2018.

[33] M. Gerdts and I. Xausa, "Avoidance trajectories using reachable sets and parametric sensitivity analysis," in *System Modeling and Optimization*, D. Hömberg and F. Tröltzsch, Eds., 2013, pp. 491–500.

[34] S. Dadras, S. Dadras, and C. Winstead, "Reachable set analysis of vehicular platooning in adversarial environment," in *2018 Annual American Control Conference (ACC)*, June 2018, pp. 5568–5575.

[35] A. Alipour-Fanid, M. Dabaghchian, and K. Zeng, "Platoon stability and safety analysis of cooperative adaptive cruise control under wireless rician fading channels and jamming attacks," *arXiv preprint arXiv:1710.08476*, 2017.

[36] D. Yanakiev and I. Kanellakopoulos, "A simplified framework for string stability analysis in ahs," in *Proceedings of the 13th IFAC World Congress, 1996*, 1996, pp. 177–182.

[37] E. Yeh, J. Choi, N. Prelcic, and C. Bhat, "Security in automotive radar and vehicular networks," *submitted to Microwave Journal*, 2016.

[38] F. Borrelli, *Constrained optimal control of linear and hybrid systems*. Springer, 2003, vol. 290.

[39] A. Petrillo, A. Pescape, and S. Santini, "A collaborative control strategy for platoons of autonomous vehicles in the presence of message falsification attacks," in *2017 5th IEEE MT-ITS*. IEEE, 2017, pp. 110–115.

[40] Q. Xu, T. Mak, J. Ko, and R. Sengupta, "Vehicle-to-vehicle safety messaging in dsrc," in *Proceedings of the 1st ACM international workshop on Vehicular ad hoc networks*. ACM, 2004, pp. 19–28.

[41] C. Chou, C. Li, W. Chien, and K. Lan, "A feasibility study on vehicle-to-infrastructure communication: Wifi vs. wimax," in *2009 Tenth International Conference on Mobile Data Management: Systems, Services and Middleware*, 2009, pp. 397–398.

[42] I. M. Mitchell, "A toolbox of level set methods," *UBC Department of Computer Science Technical Report TR-2007-11*, 2007.

[43] N. Lyamin, A. Vinel, M. Jonsson, and J. Loo, "Real-time detection of denial-of-service attacks in ieee 802.11 p vehicular networks," *IEEE Communications letters*, vol. 18, no. 1, pp. 110–113, 2013.

[44] C. Somarakis, Y. Ghaedsharaf, and N. Motee, "Risk of collision and detachment in vehicle platooning: Time-delay-induced limitations and trade-offs," *IEEE Transactions on automatic control*, 2019.

[45] J. Goldberger, S. Gordon, and H. Greenspan, "An efficient image similarity measure based on approximations of kl-divergence between two gaussian mixtures," in *null*. IEEE, 2003, p. 487.

[46] A. Satorra and P. M. Bentler, "A scaled difference chi-square test statistic for moment structure analysis," *Psychometrika*, vol. 66, no. 4, pp. 507–514, 2001.

[47] J. Fukuyama, "A delay time analysis for multi-hop v2v communications over a linear vanet," in *2009 IEEE Vehicular Networking Conference (VNC)*. IEEE, 2009, pp. 1–7.

[48] L. Qiao, Y. Shi, and S. Chen, "Modeling and analysis of safety messages propagation in platoon-based vehicular cyber-physical systems," *Wireless Communications and Mobile Computing*, vol. 2018, 2018.

[49] A. Balador, E. Uhlemann, C. Calafate, and J.-C. Cano, "Supporting beacon and event-driven messages in vehicular platoons through token-based strategies," *Sensors*, vol. 18, no. 4, p. 955, 2018.

[50] M. Foruhandeh, Y. Man, R. Gerdes, M. Li, and T. Chantem, "Simple: single-frame based physical layer identification for intrusion detection and prevention on in-vehicle networks," in *Proceedings of the 35th ACSAC*. ACM, 2019, pp. 229–244.

[51] T. Zeng, O. Semiariy, W. Saad, and M. Bennis, "Joint communication and control for wireless autonomous vehicular platoon systems," *IEEE Transactions on Communications*, 2019.

Mingshun Sun received his B.Sc. and M.Sc. in Control Science and Engineering from the Shandong University in 2013 and 2016, respectively. He is currently a Ph.D. student in Electrical Engineering in The University of Arizona. His research interests include vehicular network security and planning, with an emphasis on secure vehicle tracking and vehicle platoon control.

Ali Al-Hashimi received his B.Sc. and M.Sc. in Electrical Engineering from the University of Basrah in 2008 and 2011, respectively. He received his Ph.D. in Electrical Engineering from Utah State University in 2020. His research interests include cyber-physical systems security, with an emphasis on designing secure and reliable control systems, and optimal control.

Ming Li (M'11, SM'17) is an Associate Professor in the Department of Electrical and Computer Engineering of University of Arizona. He was an Assistant Professor in the Computer Science Department at Utah State University from 2011 to 2015. He received his Ph.D. in ECE from Worcester Polytechnic Institute in 2011. His main research interests are wireless and cyber security, with current emphases on cross-layer optimization and machine learning in wireless networks, wireless physical layer security, privacy enhancing technologies, and cyber-physical system security. He received the NSF Early Faculty Development (CAREER) Award in 2014, and the ONR Young Investigator Program (YIP) Award in 2016. He is a senior member of IEEE, and a member of ACM.

Ryan Gerdes is an Assistant Professor in the Bradley Department of Electrical and Computer Engineering at Virginia Tech. His research interests include cyber-physical systems security, with an emphasis on the operation of autonomous systems in unknown, uncertain, and adversarial environments.