ARTICLE INFO

Article ID: 12-05-03-0020
© 2022 SAE International
doi:10.4271/12-05-03-0020

Physics-Based Misbehavior Detection System for V2X Communications

Alejandro Antonio Andrade Salazar,[1] Patrick Drew McDaniel,[1] Ryan Sheatsley,[1] and Jonathan Petit[2]

[1]The Pennsylvania State University, USA
[2]Qualcomm Technologies Inc., USA

Abstract

Vehicle to Everything (V2X) allows vehicles, pedestrians, and infrastructure to share information for the purpose of enhancing road safety, improving traffic conditions, and lowering transporation costs. Although V2X messages are authenticated, their content is not validated. Sensor errors or adversarial attacks can cause messages to be perturbed increasing the likelihood of traffic jams, compromising the decision process of other vehicles, and provoking fatal crashes. In this article, we introduce V2X Core Anomaly Detection System (VCADS), a system based on the theory presented in [1] and built for the fields provided in the periodic messages shared across vehicles (i.e., Basic Safety Messages, BSMs). VCADS uses physics-based models to constrain the values in each field and detect anomalies by finding the numerical difference between a field and and its derivation using orthogonal values. VCADS evaluation is performed with four real V2X field testing datasets and a suite of attack simulations. The results show that VCADS can constrain a variety of real-world network environments and is able to detect ~85% to ~95% of attacks coming from an adversary capable of perturbing one or more data fields.

History

Received: 01 Mar 2021
Revised: 05 Jul 2021
Accepted: 14 Feb 2022
e-Available: 04 Mar 2022

Keywords

Misbehavior detection, BSM, V2X, USDOT Connected Vehicle Pilot Deployment

Citation

Andrade Salazar, A., McDaniel, P., Sheatsley, R., and Petit, J., "Physics-Based Misbehavior Detection System for V2X Communications," *SAE Int. J. of CAV* 5(3):2022, doi:10.4271/12-05-03-0020.

ISSN: 2574-0741
e-ISSN: 2574-075X

I. Introduction

Vehicle to Everything (V2X) [2] is a wireless technology that allows vehicles, pedestrians, and infrastructure to share information. Specifically, V2X operates on messages that encompass the current state of the vehicle transmitter (e.g., location, motion, and trajectory). V2X information increases transportation awareness, coordination, efficiency, and safety and improves the decision process of the algorithms in autonomous vehicles. V2X is critical for the overall development of self-driving cars and smart cities.

In the current Institute of Electrical and Electronics Engineers (IEEE) 1609.2 standard, the security for V2X messages is primarily focused on ensuring the authenticity and pseudonymity of vehicles in the network. Although V2X security assumes the network data is correct, there are several known attacks (explored in [3, 4, 5, 6, 7, 8]) that can affect other vehicles in the network. For instance, sensors can be fooled to measure false values, internal vehicular networks can be subverted to change data measurements, and the onboard V2X system can be compromised to transmit erroneous data.

Basic Safety Messages (BSMs) allow vehicles to periodically share their location and status. Vehicles with V2X capabilities have safety applications that allow them to process BSMs through a variety of algorithms that alert the driver of possible dangers on the road. As a result of sensor errors or malicious data perturbations, false warnings can be triggered by the receiving safety applications. This outcome can be observed in the following scenarios:

- A vehicle falsely reports its location to create traffic jams, limit road resources, or reroute other vehicles.
- A vehicle falsely reports a collision or hazard ahead prompting other vehicles to change lanes, slow down, or come to a full stop.
- A vehicle falsely reports its location, speed, and/or acceleration values to trigger side or forward collision warnings causing other vehicles to slow down, hard brake, or potentially collide with each other.

Similar scenarios are explored in [1] for Cooperative Awareness Messages (CAMs), the European equivalent to BSMs. [1] suggested thresholds for location, speed, and acceleration and implemented a Kalman Filter to find inconsistencies in consecutive messages. In this article we use the theory in [1] and propose a configurable approach to constrain speed, acceleration, location, and five other fields (i.e., location accuracy, yaw rate, steering wheel angle, and vehicular dimensions; see Section II-C). Our contributions are threefold:

- Design and implement a misbehavior detection system named *V2X Core Anomaly Detection System* (VCADS). VCADS leverages physics-based models of mechanics and kinematics that relate and limit several vehicular attributes, e.g., structure, turn ratios, displacement, velocity, and acceleration, in order to detect anomalies and adversarial attacks.
- Develop a suite of attacks that encompass the common V2X Safety Application scenarios and evaluate VCADS under adversaries with different capabilities.
- Evaluate VCADS with four V2X datasets taken from real field testing data (22.5 million BSMs) and simulate our developed suite of attacks. The results show that VCADS can detect anomalies, as well as attacks, that naturally occur in V2X communications with a success rate between 85% and 95%.

The remainder of the article is organized as follows: Section II provides background, security, and applications of V2X as a system. Section III describes the attacker model, the suite of attacks used in this article, and how VCADS is used to detect the anomalies that span from these attacks. Section IV explains the evaluation process and shows the results and effectiveness of VCADS. Section V highlights related work in V2X anomaly detection, and Section VI conveys the key takeaways of this research.

II. Background

A. V2X Infrastructure Overview

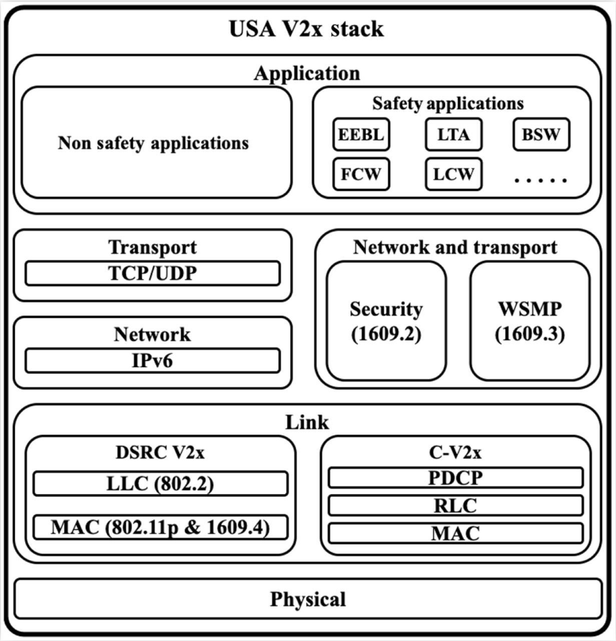
Sensors and electronic control units (ECUs) were developed to improve the overall efficiency, safety, and driving experience of vehicles. These technologies became increasingly sophisticated, resulting in the development of internal networks to connect them. The innovation process of these systems and the demand for information that understands the surrounding interactions and behavior of vehicles inspired the first external network specifications (i.e., car to car [9]). Ultimately, standardized V2X protocols emerged [2].

Figure 1 shows the components of the IEEE 1609 network stack. The main focus of this article is the safety applications and BSMs [10]. BSMs are the decoded payload of a WSM (WAVE Short Message, where WAVE stands for Wireless Access in Vehicular Environments) and are detailed in Section II-C.

B. V2X Security Overview and Limitations

V2X security provides data transport confidentiality, integrity, and availability through public-key cryptography [11], which prevents unauthorized communication and allows V2X messages to be encrypted or signed. V2X communications use long-term and short-lived certificates. Long-term certificates allow vehicles to communicate with PKI authorities, while short-lived certificates are used primarily for V2V communication [11, 12].

FIGURE 1 IEEE 1609 network stack. The link layer shows the difference between Dedicated Short Range Communications (DSRC) and Cellular V2X (C-V2X). This article focuses on a detection system for the application layer.



© SAE International

While the source that transmits the signals carrying BSMs can be validated, the accuracy of the field values within the BSMs cannot [13] as sensors may fail or become subject to adversarial attacks [5, 14]. An adversary may also subvert a node (e.g., vehicle or infrastructure) by attacking its internal network or V2X transceiver module and perturbing the data that will be sent to other nodes.

The attacks considered in this article compute malicious message perturbations to trigger misleading warnings in receiving vehicles. Traffic jams, collisions, and subverted road resources are the adversarial goal of these attacks. Vehicles that are found to be compromised due to consistent misbehavior are reported and their certificates revoked.

C. Safety Applications, Core Data Fields, and BSMs Overview

V2X allows vehicles to broadcast information and coordinate actions using WSMs [10]. The payload of these messages can be, but is not limited to, a BSM in the United States of America (USA) [15] or CAM in Europe [16]. BSMs and CAMs contain data field values that represent the state of a vehicle, measurements of its trajectory and motion (e.g., speed and acceleration), and location (latitude, longitude, elevation). The minimum required fields to transmit a BSM are referred to as Core Data Fields.

The SAE J2945/1 standard requires vehicles to transmit 10 BSMs per second [12]. Every vehicle is equipped with a set of algorithms known as Safety Applications. These algorithms process BSMs in order to assess possible risks and collisions that may occur with other vehicles. Safety Applications trigger warnings to the driver or self-driving algorithm to prevent such scenarios. Vehicles that transmit BSMs are referred to as Remote Vehicles (RV), whereas the receiver of BSMs is known as the Host Vehicle (HV). All HVs also transmit BSMs and act as RVs for other vehicles. The HV and the RVs have their own Safety Applications that run locally to enhance road safety by using all incoming messages. [12]. From the Safety Application descriptions outlined in [12] and [16], this article focuses on the following:

- **Emergency Electronic Brake Lights (EEBL)**: Alert caused by a hard brake from an RV located in front and in the same lane or adjacent lanes with respect to the HV.
- **Forward Collision Warning (FCW)**: Warning that is calculated when the HV is likely to have a collision and rear-end an RV.
- **Blind Spot Warning (BSW) and Lane Change Warning (LCW)**: Warning triggered due to an HV trying to change lanes when an RV is in the path or heading towards the lane change path of the HV.
- **Intersection Movement Assist (IMA) and Intersection Collision Warning (ICW)**: Warning caused when an HV may collide with other RVs as it enters an intersection.
- **Left Turn Assist (LTA)**: Alert created when the HV approaches an intersection and seeks to turn left, invading the path of an incoming RV.

In order to trigger warnings, the safety applications must process the Core Data Fields found in the BSMs. Table 1 describes each Core Data Field. There are a total of fourteen required fields to create a BSM [15]. Some of these fields are complex and are split into sub-field values. While BSMs may also load additional optional information, our research focuses on the Core Data Fields.

III. Approach

This section describes the threat and trust models considered in this article and introduces the attacks for the following scenarios: FCW, EEBL, LTA, and ICW. These attacks show how one or more adversaries are capable of perturbing certain data fields and trigger false warnings in a given HV. The end of this section outlines the Field Validation and Cross-validation components in VCADS, which are created to constrain and prevent false alerts from happening.

A. Trust and Threat Model

The trust model assumes that the security in the physical and transport layers of the vehicles in the V2X network have been

TABLE 1 BSM core data fields description and representation per the protocol given in [15].

Field	Description
Message Count	Message index between 0 and 127
Temporary ID	Pseudo anonymous vehicle ID
DSecond	Message creation time from minute interval
Latitude	Angular distance with respect to Earth's south and north poles
Longitude	Angular distance with respect to the Greenwich meridian
Elevation	Distance with respect to Earth's sea level
Positional Accuracy	Semi-minor/Semi-major and orientation of the Global Positioning System (GPS) positioning ellipsoid
Semi-major axis accuracy	Expected accuracy of semi-major ellipsoid
Semi-minor axis accuracy	Expected accuracy of semi-minor ellipsoid
Semi-major axis orientation	Semi-major orientation with respect to the true north
Transmission State	Neutral/Park/Forward/Reverse
Speed	Positional change of over a given period in time
Heading	Trajectory direction with respect to the true north
Steering Wheel Angle	The turn angle of the wheels with respect to the vehicle's front face
AccelerationSet4Way	Longitudinal/Lateral/Vertical/Yaw rate
Break System Status	Brakes/Traction/ABS/SCS/Brake boost/Aux. brakes
Wheel Brakes	Brake application in each tire
Traction	Traction control system status
ABS	Anti-lock system status
SCS	Stability control status
Brake Boost	Brake boost system status
Auxiliary Brakes	Auxiliary brakes system status
Vehicle Size	Vehicle's length and width

properly implemented. The messages from the RVs to the HV are signed to preserve integrity and authentication. Replay attacks, Man-in-the-Middle attacks, and data alterations are prevented by the standardized security protocols. We also assume there is full accountability for message transmissions through certificates (non-repudiation). In short, the communication between vehicles, as well as the network stack in the HV, is secure. This means that the HV's sensors are working properly and its Safety Applications have precise internal measurements and processing capabilities that yield the expected alerts or warnings. In terms of the incoming BSMs from the RVs, our trust model is straightforward: **Trust, but verify**. We trust the data that RVs transmit to the HV, as long as it is consistent.

1. *Adversarial Goal*: The adversary's goal is to subvert the Safety Application system of a given HV and trigger false warnings. The adversary exploits known vulnerabilities to compromise an RV and transmit perturbed messages in order to mislead one or more HVs. This, in turn, leads to shifted traffic flows, traffic jams, misuse of road and highway resources, unexpected defensive behavior from drivers or self-driving vehicles, and collisions.
2. *Adversarial Capabilities*: We assume that the adversary is able to alter one or more data field measurements in a compromised RV. The adversary is able to achieve this goal by subverting the RV's sensors to yield the desired measurements [5, 17], attack the internal network of a vehicle [18], or subvert a V2X transceiver module with valid certificates [19].

B. V2X Attack Scenarios

We introduce a subset of four common scenarios by updating the safety application figures in [12]: EEBL, FCW, ICW, and LTA. Other scenarios are not explored, but they are similar. The difference between them lies on which fields are perturbed and what false warnings are triggered in the HV.

In Figure 2, an RV seeks to trigger a false FCW or EEBL in an HV. The attacker causes FCW alerts by a given factor that

FIGURE 2 A vehicle in front of the HV alters motion data (e.g., Speed or Acceleration) triggering an FCW and causing the HV to slow down or hard brake.

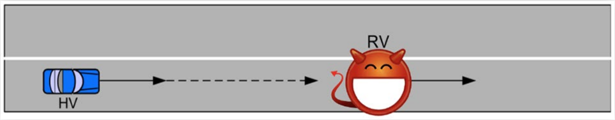
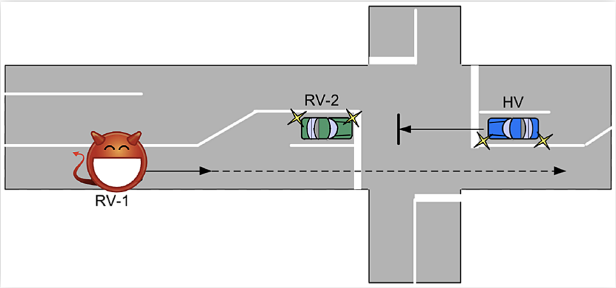


FIGURE 3 The HV's line of sight is diminished by RV-2. RV-1 misreports motion information and triggers an LTA alert in the HV. This prevents the HV from turning, and traffic builds up in the intersection.



reduces the motion fields in the RV. The RV will appear closer from the HV than it actually is, and when the Time to Collision (TTC) is reduced, the false FCW is triggered. In contrast, the EEBL alert is triggered only when negative accelerations of $-3.92\ \text{m/s}^2$ or less are received in the HV. In both scenarios, after the false warning is achieved, the HV will then slow down or hard brake to prevent a collision. As a result of this misled decision taken by the HV, traffic builds up and collisions can occur.

In Figure 3, the HV seeks to turn left in a given intersection. RV-2 prevents the HV from detecting incoming parallel traffic and RV-1 perturbs its fields to trigger a false LTA alert in the HV. The false LTA alert will be enough to prevent the HV from turning left causing traffic build-ups behind the HV. Similar to LTA, in the ICW attack, the adversary manipulates different fields to prevent an HV from crossing the intersection. In this case, the HV is perpendicular to RV-1 and there is no RV-2.

Table 2 summarizes our security model and gives a general overview of our attack scenarios, including other scenarios explored in [20, 21, 22, 23, 24, 25]. The definition of each attack scenario is based on how the adversaries leverage their capabilities to trigger false warnings in an HV. The detection method column relates the attacks to the mechanisms presented in VCADS.

C. VCADS Overview

VCADS detection mechanisms are divided into two components: Field Validation and Cross-Validation. Field Validation detects single-field anomalies by creating configurable constraints based on physics models derived from kinematics relations and limitations from vehicular mechanics. Field Cross-Validation analyzes the consistency and accuracy of a given field by relating it to other fields as different messages arrive over time. Five kinematics equations, one explored in [1], are

TABLE 2 Safety applications attacker model.

Attack	Capabilities	Threat	Definition	Detection method
Forward Collision Warning (FCW)	1. Speed 2. Location 3. Acceleration 4. Brake status 5. Transmission status	1. Sensor fooling 2. Internal network 3. V2X transceiver	Attacker reduces capability field(s) in an RV located in front of the HV. An FCW triggers in the HV and causes it to slow down or hard brake.	Cross-Validation of capability field(s) against field(s) that the attacker does not control. Available field(s) derive the same field measurement(s) as the capability field(s). The difference between these measurements are flagged by VCADS once they surpass a sensitivity threshold.
Emergency Electronic Brake Lights (EEBL)			Attacker decelerates to show a hard brake in an RV located in front of the HV. An EEBL warning gets triggered in the HV and causes it to hard brake.	
Intersection Collision Warning (ICW)			Attacker perturbs capability field(s) to position an RV in the intersection that the HV is approaching. This causes the HV to hard brake and stop.	
Left Turn Assist (LTA)			Attacker modifies capability field(s) to position an RV in the intersection that the HV seeks to turn left. This prevents the HV from turning left.	
Lane Change Warning (LCW), Blind Spot Warning (BSW)			Attacker modifies capability field(s) to position an RV in the HV's blind spot or lane change trajectory. This triggers a BSW or LCW in the HV and prevents it from changing lanes.	Cross-Validation of location and motion fields. Available field(s) derive the same field measurement(s) as the capability field(s). The difference between these measurements are flagged by VCADS once they surpass a sensitivity threshold.
Event	1. Location 2. Event (e.g., traction control loss, airbag deployment)	1. Internal network 2. V2X transceiver	Attackev modifies capability field(s) to transmit an event from an RV (e.g., airbag deployment, traction control loss). This causes receiving vehicles to slow down or reroute.	Not explored in this article (see [21, 22, 23, 24, 25, 26])

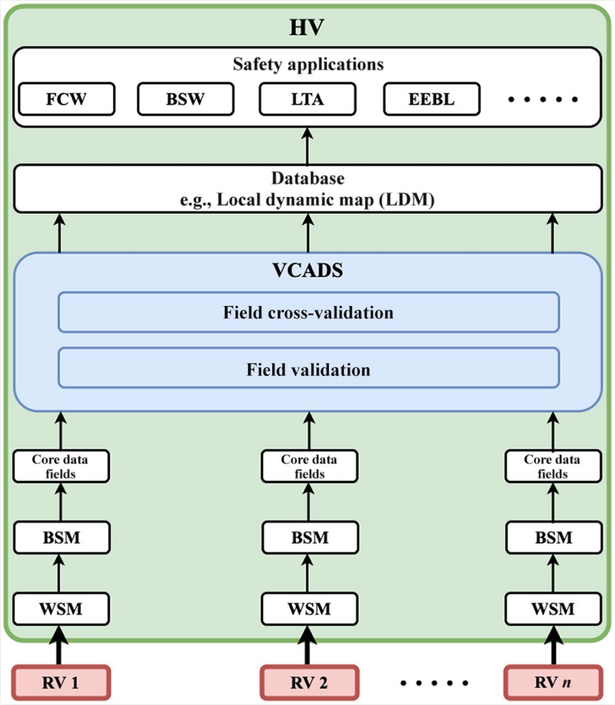
used to combine independent fields and derive a measurement from a field that is also reported in the message. The derived and reported fields are compared and the numerical difference between them is used to detect anomalies. A configurable threshold, known as sensitivity, is used to show the allowed variation between these fields; any difference above this sensitivity is considered anomalous. In this article, the sensitivity is varied in order to show the overall detection range of the system; however, a commercial implementation of VCADS could optimize this value to achieve better detection results using a variety of techniques, such as machine learning.

VCADS is located in the Application layer, and it is the first component to receive a BSM: the decoded payload of a WSM. The Field Validation and Cross-Validation components detect anomalies in all incoming BSMs' Core Data Fields and protect the Safety Applications from receiving erroneous or malicious information. Specifically, messages are validated before they are stored in the Safety Applications' database, formally known as LDM. As a result, VCADS seeks to prevent false warnings from triggering in an HV. Figure 4 shows VCADS location and interactions with other components in the V2X stack.

In order to show how fields relate and are used in VCADS, we separate the fourteen Core Data Fields into classes according to the information they represent:

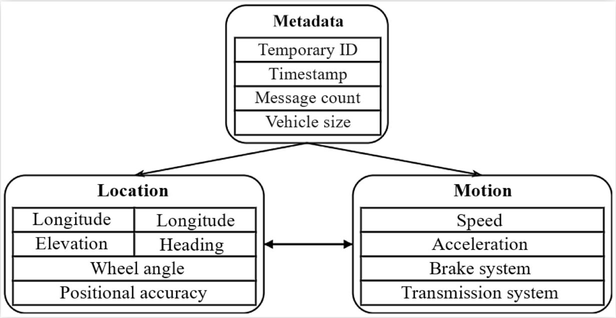
- **Metadata**: ID, Timestamp (DSecond), Msg Count (Message Count), and Vehicle Size

FIGURE 4 Data flow in an HV. Several RVs send secure data to the HV, the data is decoded into BSMs, and Core Data Fields are taken as the input of the Application layer. VCADS flags any anomalous fields and sends validated messages into the LDM.



© SAE International

FIGURE 5 V2X Core Data Fields divided into three classes based on the type of data that they hold. Metadata gives context to the data, location values give a sense of the position of the vehicle, and motion values allow us to understand the kinematics and dynamics of the vehicle.



© SAE International

- **Location**: Latitude, Longitude, Elevation, Positional Accuracy, Heading, and Steering Wheel Angle
- **Motion**: Speed, Acceleration Set, Heading, Transmission State, and Brake State

Figure 5 shows the classes and how they interact. The metadata fields give context to the motion and location fields. The Temporary ID shows which RV sent the BSM, DSecond when the field measurements were taken, and Message Count the sequence in which BSMs were created.

The location class represents GPS-related measurements. These fields allow us to approximate the exact positioning of a vehicle and its trajectory. When these values are combined with the metadata values (e.g., DSecond, Temporary ID), they derive motion fields.

Motion fields represent all the measurements from internal sensors that relate to the vehicle's kinematics. The combination of motion, metadata, and the previous location yields the following location measurement of a vehicle. Thus previous BSMs can relate their fields with the incoming BSMs. We use this principle as the theory behind the Field Cross-Validation component. For the remainder of this article, we assume HVs are augmented with a VCADS implementation to detect anomalies of several incoming messages from different RVs.

D. Field Validation

Field Validation finds constraints by applying vehicular, structural, and mechanical limitations, as well as physics equations and models. These constraints derive lower and upper bounds that narrow the allowable measurements of a data field. Field values that surpass these bounds are flagged as anomalous. It is up to the Safety Applications to filter the flagged fields or decide to discard the entire BSM.

BSMs with anomalous fields may trigger false warnings or unexpected behavior in the HV. Thus it is important to take certain actions in order to prevent error propagation into the databases, Safety Applications, or other Application layer components. Fields that are within the limits of these constraints are assumed to be properly measured and safe to process by the Field Cross-Validation component.

***Latitude Constraint*:** Latitude measurements range from −90° to 90°. We narrow this by geofencing (virtual perimeter of a geographic area) inside the USA, a given communication radius, or a specific perimeter of interest. Geofencing can be further narrowed to target RVs that have high probability of interacting with the HV. This includes the division of different transportation locations and motion patterns (e.g., finding perimeters that divide streets and highways). Any latitude that does not conform to the geofence perimeter is flagged. The resulting constraint becomes

$$Latitude_{RV} \in S_{Lat} \qquad \text{Eq. (1)}$$

where $Latitude_{RV}$ is the latitude reported by the RV, ∈ is the logic symbol exists and S_{Lat} is the set of all possible latitude values inside a determined geofence.

When a circular geofence is used, this constraint can be simplified by using a radius as the maximum allowed distance between the HV and surrounding RVs:

$$\left|Latitude_{RV-HV}\right| \le R \qquad \text{Eq. (2)}$$

where $Latitude_{RV\text{-}HV}$ is the difference (in meters) between the HV and the RV latitude, and R is a geofence radius. VCADS defaults this radius to 300 m (as per protocol [12]). However, this parameter can be adjusted depending on the desired detection sensitivity.

***Longitude Constraint*:** Longitude measurements range from −180° to 180°, and they follow the same geofence constraints as the latitude field. Longitude values are constraint by the following equation:

$$Longitude_{RV} \in S_{Long} \qquad \text{Eq. (3)}$$

where S_{Long} is the set of all the possible longitude values inside a determined geofence. Similarly, when a geofence is defined by a radius, the resulting equation becomes

$$\left|Longitude_{RV-HV}\right| \le R \qquad \text{Eq. (4)}$$

where $Longitude_{RV\text{-}HV}$ is the difference (in meters) between the RV and HV longitude values, and R is the geofence radius.

***Elevation Constraint*:** Although most roads are above sea level, V2X protocol elevations can range from −409.5 m to 6143.9 m [15]. Just like latitude and longitude, we constrain elevation values according to the geofence created around the HV and translate it to the elevation axis:

$$Elevation_{RV} \in S_{Ele} \qquad \text{Eq. (5)}$$

where $Elevation_{RV}$ is the elevation reported by the RV and S_{Ele} is the set of all the possible elevation values that an RV can report inside a geofence. When the geofence is determined by a radial distance from the HV, we use a reference angle α to translate the radius into the elevation axis. Our reference angle α is 25°, based in [26, 27]. Nonetheless, this angle can be modified to fit exclusive geographic areas, where α can be less than 25°.

$$R_{Elevation} = \sin\alpha \times R \qquad \text{Eq. (6)}$$

where $R_{Elevation}$ is the projection of R in the elevation axis. For example, when using R as 300 m and α as 25°, the lower and upper bounds for Elevation become

$$R_{Elevation} = \sin 25° \times 300\ \ \text{m} \approx 127\ \ \text{m}$$

$$\left|Elevation_{RV-HV}\right| \le R_{Elevation} \qquad \text{Eq. (7)}$$

where $Elevation_{RV\text{-}HV}$ is the elevation difference between the RV and HV.

***Location Constraint*:** When all coordinates (latitude, longitude, and elevation) are combined, we can derive the overall location distance that an RV can take with respect to an HV:

$$\sqrt{Lat^2_{RV-HV} + Long^2_{RV-HV} + Ele^2_{RV-HV}} \le R \qquad \text{Eq. (8)}$$

where latitude, longitude, and elevation have been simplified to *Lat*, *Long*, and *Ele*, and R is the geofence radius. R can be changed based on the desired geofencing model and is defined as the maximum distance that an RV can report with respect to the HV. In our case, the total magnitude of all vectors combined has to be less than or equal to a given radius.

***Speed Constraint*:** Speed is not a field that can be constrained by any physics equation in such a way that can be used for anomaly detection in vehicles. Since the highest speed limit in the USA varies by state, we have chosen to constrain the maximum value of speed with the highest allowable speed limit nationwide: 85 mph (38 m/s) [28]. Although speed is not validated using physics equations, it is largely validated by the Field Cross-Validation component (described in Section III-E). The following equation shows a generic way for constraining speed, and we suggest possible approaches to find the upper and lower bounds for speed:

$$\upsilon_{\min} \le \upsilon_{\text{RV}} \le \upsilon_{\max} \qquad \text{Eq. (9)}$$

where $\upsilon_{\min}$ is the lower bound and $\upsilon_{\max}$ is the upper bound for speed. In our Field Validation component, we chose $\upsilon_{\min}$ as 0 (since speed is a magnitude and cannot be negative) and $\upsilon_{\max}$ as 42 m/s (4 m/s more than the highest posted speed for the USA).

From the BSMs analyzed in the IV dataset, we suggest that a more aggressive and realistic constraining model can be used to calculate Min_{Speed} and Max_{Speed}. This model can be based on the HV's measured speed or an offset of said speed, given that vehicles in similar roads will follow similar patterns.

***Accuracy Constraint*:** In our model, the positional accuracy constraint is based on the standard structure of vehicles, specifically the vehicles' width (2.6 m [29]). Inaccuracies that yield an uncertainty higher than 2.6 m are flagged due to the increasing error, nondeterministic behavior, and possible false warnings that can be triggered in the Safety Applications. RVs with these inaccuracies cannot be pinned to a specific lane in order to calculate possible collisions with the HV. Therefore, we derive the overall accuracy needed for these fields in the following equations:

$$SemiMinor \le L \qquad \text{Eq. (10)}$$

$$SemiMajor \le L \qquad \text{Eq. (11)}$$

$$\sqrt{SemiMinor^2 + SemiMajor^2} \le L \qquad \text{Eq. (12)}$$

where *SemiMinor* and *SemiMajor* are BSM data fields. *L* is the maximum allowable uncertainty and the overall magnitude. When combining *SemiMinor* and *SemiMajor* fields, their magnitude must be less than the *L* parameter. *L* defaults to 2.6 m in our model.

***Steering Wheel Angle Constraint*:** Ackermann's steering geometry predicts the maximum steering angle of vehicles based on their turn structure [30, 31]:

$$\alpha_{Steering} \le 65° \qquad \text{Eq. (13)}$$

where $\alpha_{Steering}$ is the RV's steering wheel angle in degrees.

***Longitudinal Acceleration Constraint*:** The longitudinal acceleration spans from the vehicle's center of mass and goes through its plane of symmetry [32]. It represents the acceleration of the vehicle's heading and is driven by the friction coefficient μ of the vehicle tires with respect to the surface, and the acceleration due to gravity g:

$$Longitudinal_{Acc} = \mu \times g$$

μ can be configured to find the maximum and minimum longitudinal accelerations of the vehicle:

$$MinLong_{Acc} \le Longitudinal_{Acc} \le MaxLong_{Acc} \qquad \text{Eq. (14)}$$

where $MinLong_{Acc}$ is the lower bound and $MaxLong_{Acc}$ is the upper bound of the longitudinal acceleration values that can be reported by an RV.

A suggested value for μ is 1.0; however, the bounds chosen for this field used a data-driven approach to find the best possible bounds for the current consumer vehicles. The model takes the acceleration developments of passenger vehicles over time. Every decade, the maximum average acceleration of vehicles decrease by 2 s while reaching speeds from 0 km/h to 96.6 km/h (26.83 m/s) [33]. Currently, passenger vehicles can achieve a speed of 26.83 m/s in 6 s; this translates to an acceleration of 4.47 m/s^2. In addition, the fastest passenger vehicles' times are between 2.3 s to 2.8 s. For example, the Tesla Model S [34] can reach 26.83 m/s in 3.1 s, and when ideal conditions are met, it can reach this speed in 2.3 s. We can use the previous information to find the upper and lower bounds for the acceleration fields:

$$Max_{Acc} = \frac{Speed_{Top}}{Time_{Elapsed}} = \frac{26.83 \text{ m/s}}{2.65 \text{ s}} = 10.12 \text{ m/s}^2 \qquad \text{Eq. (15)}$$

where $Speed_{Top}$ is the speed reference and $Time_{Elapsed}$ is the time that a vehicle takes to reach that speed. If g is considered as 9.8 m/s^2, the friction coefficient μ is calculated as 1.03.

For negative longitudinal accelerations (i.e., braking), the model has been also simplified to use Max_{Acc} as the limiting value. Both positive and negative accelerations depend on the road friction coefficient and weather conditions. Increasingly complex models can be built to take these factors into account and vary μ as the vehicle enters different environments.

***Lateral Acceleration Constraint*:** Lateral acceleration of a vehicle is driven by the same dynamics of longitudinal acceleration. Both longitudinal and lateral accelerations are parallel to the surface and they only differ in the direction from the center of mass. For this reason, the bounds for lateral acceleration have been simplified to reflect the same bounds as longitudinal acceleration.

$$MinLat_{Acc} \le Longitudinal_{Acc} \le MaxLat_{Acc} \qquad \text{Eq. (16)}$$

where $MinLat_{Acc}$ is the lower bound and $MaxLat_{Acc}$ is the upper bound of the lateral acceleration values that can be reported by an RV, and they are based on the data-driven value Max_{Acc}.

***Vertical Acceleration Constraint*:** For vertical accelerations, we use a road slope and translate Max_{Acc} to the vertical axis. Due to gravity, the negative vertical accelerations differ from the positive ones. We define Up_{Acc} as the positive upward acceleration and $Down_{Acc}$ as the negative downward acceleration. These two bounds are combined to yield the resulting vertical acceleration ($Vert_{Acc}$) constraint.

$$Down_{Acc} \le Vert_{Acc} \le Up_{Acc} \qquad \text{Eq. (17)}$$

The effects of gravity on the vehicle motion depend on the slope of the road that is mathematically defined as α. We calculate all the vector forces acting on a vehicle, which yield the total acceleration vector; this vector is then projected to the vertical axis. Since the vehicle's weight is multiplied in all accelerations, this parameter is simplified. The following equation shows how to calculate the total acceleration:

$$a = Max_{Acc} - \sin\alpha \times g - \cos\alpha \times \mu \times g \qquad \text{Eq. (18)}$$

where a is the resulting vector acceleration, g is gravity, and μ is the friction coefficient. Similar to other fields, α, and μ can be varied depending on the geographic area and the desired sensitivity of the model. Note that as α increases, so does the effect of the gravitational force in the vehicle. Similarly, the acceleration due to the friction force is always opposing the Max_{Acc} acceleration. The resulting acceleration a is then translated to the vertical axis. For example, when α is 25°, Max_{Acc} is 10.12 m/s^2 and μ is 0; the resulting Up_{Acc} and $Down_{Acc}$ are

$$Up_{Acc} = \sin\alpha \times (10.12 - \sin\alpha \times 9.80) = 2.52 \text{ m/s}^2$$

$$Down_{Acc} = \sin\alpha \times (-10.12 - \sin\alpha \times 9.80) = -6.03 \text{ m/s}^2$$

Given that gravity is always acting on the vehicle, the RV will always report the vertical acceleration with g. If we take g as −9.8 m/s^2, both constraints will be updated as $Up_{Acc} = -7.28$ m/s^2 and $Down_{Acc} = -15.83$ m/s^2.

It is important to note that these constraints do not consider uneven surfaces, which can trigger high spikes in vertical acceleration beyond these bounds. This can happen when a vehicle cruises through a pothole or a speed bump. The instantaneous speed at which the vehicle encounters the uneven surface will become a factor in the magnitude of the acceleration spike.

***Yaw Rate Constraint*:** Similar to the steering wheel angle, the bounds for Yaw Rate can be calculated using [31]. Another important parameter is the maximum velocity that a vehicle can experience while turning. This is calculated by relating the centripetal force experienced by a vehicle while turning in addition to the friction force between the tires and the road.

$$\frac{m\times\upsilon^2}{R} = \mu\times g\times m$$

$$\upsilon = \sqrt{\mu\times g\times R}$$

where υ is the maximum allowed turning speed given a friction coefficient μ, gravity acceleration g, vehicle's mass m, and radius R.

For example, if the friction coefficient μ is 0.72 and R is 7.62 m:

$$\upsilon = \sqrt{0.72\times 9.80\times 7.62} = 7.37 \ \text{m/s}$$

The yaw rate sign is based on the turning direction of a given vehicle; therefore, the lower and upper bounds are equivalent in magnitude. Using v as shown above and $Turn_{Radius,}$ the equation for yaw rate becomes

$$\left|Min_{Yaw}\right| = \left|Max_{Yaw}\right| = \frac{\upsilon}{Turn_{Radius}} \qquad \text{Eq. (19)}$$

where Min_{Yaw} is the lower yaw rate bound, Max_{Yaw} the upper yaw rate bound, and $Turn_{Radius}$) is the steering wheel angle of the vehicle. For example, assuming a vehicle's length is 5.18 m, θ is 44.33°, and υ is 7.37m/s; the maximum yaw rate is

$$Turn_{Radius} = \frac{Vehicle_{Length}}{\cos\theta} = 7.24 \ \text{m}$$

$$Max_{Yaw} = \frac{7.37}{7.24} = 1.01 \ \text{rad/s} = 57.86 \ \text{deg/s}$$

Note that the turning speed calculated with the above equation is for nearly level surfaces present in our datasets. In [35], the above equation is extended by using the superelevation of the road to scale the value of the turning speed. In order to achieve higher precision in the speed calculation, other factors can be analyzed, as well as understanding in what situations is convenient to use spin analysis.

***Vehicle Size Constraint*:** The Federal-Aid Highway Act of 1976, and subsequent amends, require vehicles to be at most 2.6 m wide, excluding mirrors. The maximum allowable length of a vehicle varies upon several factors that depend on the state, load, and its configuration [29]. Vehicles follow this structural constraints in order to drive across all US roads and around the world:

$$Vehicle_{Width} \le L_{Width} \qquad \text{Eq. (20)}$$

$$Vehicle_{Length} \le L_{Length} \qquad \text{Eq. (21)}$$

where L_{Width} is the configurable parameter for maximum allowable width and L_{Length} for length. The default configuration of our model sets L_{Width} to 2.6 m and L_{Width} to 16.15 m. L_{Width} is based on the most common trailer load. Note that these constraints should be applied to the BSMs before other constraints. The rationale for this order is that other field constraints such as yaw rate and positional accuracy use these fields as parameters to find the lower and upper bounds.

In conclusion, each constraint has certain parameters that can be updated to relax or tighten the bounds for allowable values (e.g., 3.10 s instead of 2.65 s in order to calculate maximum acceleration will result in 8.65 m/s^2 instead of 10.12 m/s^2).

E. Field Cross-Validation

The Cross-Validation component relates field values by taking a given reported value from a BSM and using other independent fields to derive the same measurement. Subsequently, we can assess the accuracy of a given field measurement, the consistency between fields, and show the numerical difference of a given measurement with respect to its derivations. If an adversary manages to perturb one or more data field values, this behavior will be evident in the Cross-Validation component.

If a BSM is cross-validated and no anomalies are detected, VCADS does not flag the message. From VCADS point of view, the RV that sent the BSM is not misbehaving or having sensor failures, and it can be propagated up the stack. If the variations on the BSMs field measurements and their derivations are beyond certain configurable sensitivity, the BSM is flagged as anomalous. The sensitivity can be modified according to the desired error variations.

Just like in Field Validation, the anomalies detected in this component do not determine intent. Field measurements might be perturbed due to sensor error or malicious behavior, and they are also dependent on the sensitivity to be deemed as anomalous. In our evaluation, it is evident that sensor errors portray smaller deviations between the derived and the reported measurements than the deviations from attack perturbations.

In order to relate field measurements with its derivations, the Field Cross-Validation component uses several equations that will be introduced in the following subsection.

1. Kinematics Equations

$$s_{Final} = s_{Initial} + \frac{\upsilon_{Final} + \upsilon_{Initial}}{2}\times\Delta t \qquad \text{Eq. (22)}$$

$$s_{Final} = s_{Initial} + \upsilon_{Initial}\times\Delta t + \frac{a_{Avg}\times\Delta t^2}{2} \qquad \text{Eq. (23)}$$

$$s_{Final} = s_{Initial} + \upsilon_{Final} \times \Delta t - \frac{a_{a=Avg} \times \Delta t^2}{2} \quad \text{Eq. (24)}$$

$$\upsilon_{Final} = \upsilon_{Initial} + a_{Avg} \times \Delta t \quad \text{Eq. (25)}$$

$$\upsilon_{Final}^2 = \upsilon_{Initial}^2 + 2 \times a_{Avg} \times \Delta s \quad \text{Eq. (26)}$$

In the above equations, *s* is displacement, *υ* velocity, *a* acceleration, *t* time, Δ variation, and *avg* average [36].

2. Haversine Formula The Haversine Formula is a mathematical model that approximates the distance between two coordinates [37]. The approximation error using this model for the distances that correspond to V2X is virtually zero and negligible. For more precision and coordinates that are further apart from V2X transmission radius, the approximation error of haversine becomes ~1% and increases as the coordinates are further apart. Such distances can be better approximated with models that have higher fidelity with respect to Earth's ellipsoid [38]. The following equation shows how to calculate the distance between two coordinates:

$$a = \sqrt{\sin^2 \frac{\phi_2 - \phi_1}{2} + \cos\phi_1 \times \cos\phi_2 \times \sin^2 \frac{\lambda_2 - \lambda_1}{2}}$$

$$distance = 2 \times r \times \arcsin a \quad \text{Eq. (27)}$$

In this equation, *r* is the Earth's radius, ϕ is latitude, and λ longitude.

We can relate the different data fields using Equations 25 and 26. These relationships are two way, meaning one combination of fields can validate the other and vice versa. We employ metadata fields (DSecond, Temporary ID, and Message Count) to relate location (i.e., latitude, longitude, elevation, and heading) with motion fields (speed, acceleration, yaw rate, brake system, and transmission system).

Within the motion fields, we cross-validate acceleration, speed, and bra and transmission systems. Acceleration is further validated with the steering wheel angle and heading. Yaw rate is also cross-validated with location coordinates, the vehicle size, and heading. Finally, two different location coordinates are cross-validated with the heading and steering wheel angle.

For instance, two consecutive BSMs from the same RV must match the RV's speed, acceleration, and heading. Likewise, speed is related to acceleration, and acceleration relates to the vehicle's brake and transmission systems. We can cross-validate these data fields and find anomalies from misbehavior and erroneous or malicious RVs.

Similar to the reported measurements, the derived measurements can also be Field Validated (III-D). These measurements can then be flagged based on single-field lower and upper bounds. Not only do we cross-validate a given field measurement and its derivations but also validate the field derivations themselves. The variations between a reported measurement and its derivations are calculated with the following equations:

$$\frac{|Actual - Derived|}{Actual} \times 100\% \quad \text{Eq. (28)}$$

When calibrating the Field Cross-Validation component, the allowed sensitivity can be chosen instead of the error percentage between the reported and derived measurements. Any inconsistency above the sensitivity value is flagged as anomalous. Note that high sensitivity values allow more variation between the actual and derived fields, and the likelihood to flag anomalous BSMs is reduced In contrast, low sensitivity values result in a higher detection rate with the downside that correct values could be flagged as anomalous. Using several data points allows us to choose the best sensitivity that will not flag expected error variations but will flag anomalous behavior.

3. Cross-Validation Constraint:

$$|Actual - Derived| \leq Sensitivity \quad \text{Eq. (29)}$$

where Sensitivity is the threshold allowed between the difference of the field measurement and the derived measurement from other fields. Table 3 summarizes all the constraints used in VCADS.

IV. Evaluation

This evaluation answers two fundamental questions:

1. Is VCADS able to model and constrain a wide variety of real-life driving environments?
2. Is VCADS an effective mechanism for detecting anomalies and data field perturbation attacks?

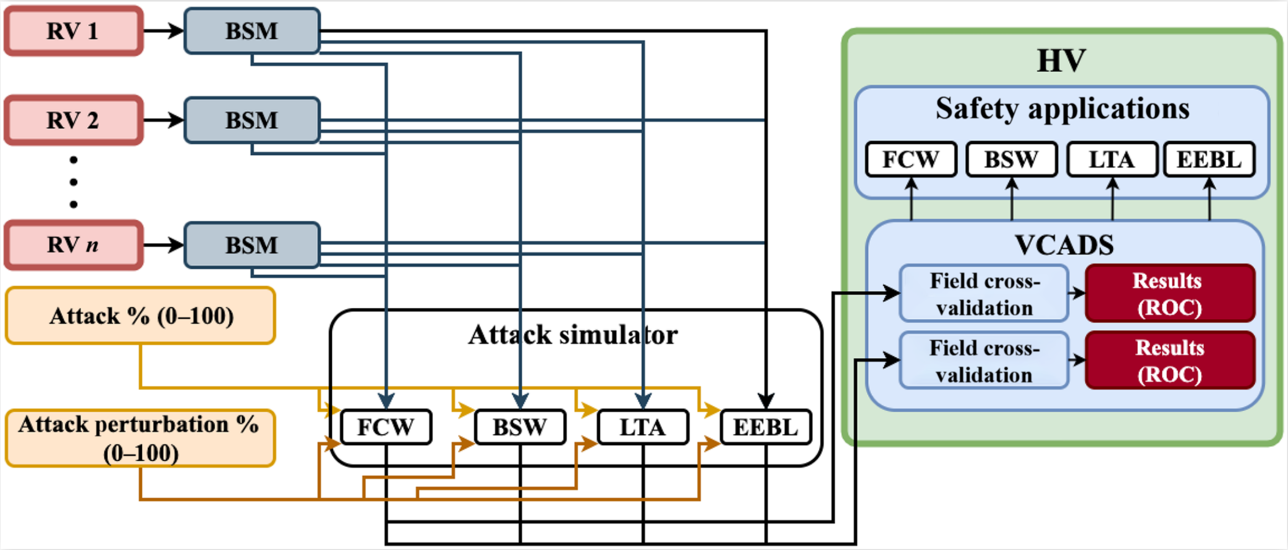
The above questions were answered using four real V2X USDOT field testing datasets and attack simulations for EEBL, FCW, LTA, and ICW. Figure 6 shows how the evaluation setup and the attack simulator interacts with the datasets and VCADS. The simulations were developed using a baseline with no attacks and, afterwards, we applied several kinds of perturbations to BSMs to trigger alerts in the HV. The diverse characteristics portrayed in the field testing datasets allow the evaluation of VCADS' ability to model and constrain different environments. In addition, the data accounts for the natural errors that occur when using hardware in V2X testing. The attack simulations help understand the detection effectiveness of VCADS on a wide range of attacks.

Through this evaluation, the Field Validation and Cross-Validation components were able to model the behavior of vehicles in different environments and driving patterns. On the other hand, these components also detected anomalies from sensor errors and data perturbation attacks. The lack of errors found in the speed and acceleration fields when subject to the Field Validation component suggests that these constraining bounds should be tightened. However, when

TABLE 3 VCADS constraints summarized by the Core Data Field.

Field	Constraint
Latitude	$\lvert Latitude_{RV-HV}\rvert \le R$ Constraints the RV's latitude at a radial distance R from the HV
Longitude	$\lvert Longitude_{RV-HV}\rvert \le R$ Constraints the RV's longitude at a radial distance R from the HV
Elevation	$\lvert Elevation_{RV-HV}\rvert \le R \times \sin\alpha$ Constraints the RV's elevation at a radial distance R from the HV
Location	$\sqrt{Lat^2_{RV-HV} + Long^2_{RV-HV} + Ele^2_{RV-HV}} \le R$ Constraints the overall magnitude of the location fields to a radial distance R from the HV
Speed	$V_{\min} \le \upsilon_{RV} \le \upsilon_{\max}$ Generic constraint bounds; field limits are not based on physics models
Accuracy	SemiMinor ≤ 2.6 m; $SemiMajor \le 2.6$ m; $\sqrt{SemiMinor^2 + SemiMajor^2} \le 2.6$m Constraints the location ellipsoid accuracy based on the width of vehicles
Steering Wheel Angle	$\alpha_{Steering} \le 65°$ Constraints the steering wheel angle based on the structure and turning mechanics of a vehicle
Longitudinal Acceleration	$\mathrm{MinLong}_{Acc} \le Longitudinal_{Acc} \le MaxLong_{Acc}$ Constraints the acceleration based on friction coefficients and top acceleration of vehicles
Lateral Acceleration	$\mathrm{MinLat}_{Acc} \le Lateral_{Acc} \le MaxLat_{Acc}$ Constraints the acceleration based on friction coefficients and top acceleration of vehicles
Vertical Acceleration	$\mathrm{Down}_{Acc} \le Vert_{Acc} \le Up_{Acc}$ Constraints the acceleration based on friction coefficients and top acceleration of vehicles
Yaw Rate	$\lvert Min_{Yaw}\rvert = \lvert Max_{Yaw}\rvert = \dfrac{\upsilon}{Turn_{Radius}}$ Constraints the yaw rate based on turning mechanics and top acceleration of vehicles
Vehicle Width	$\mathrm{Vehicle}_{Width} \le 2.6$ m Constraint based on US regulations for vehicular width design
Vehicle Length	$\mathrm{Vehicle}_{Length} \le 16.15$ m Constraint based on US regulations for vehicular length design
Cross-Validation	$\lvert Measurement - Measurement_{Derived}\rvert \le Sensitivity$ Constraint based on a field measurement and the same derived measurement from other fields

FIGURE 6 Simulation data flowchart: interaction between USDOT datasets, attack simulator, and VCADS.



subject to cross-validation, these fields were largely flagged. The cross-validation variations from the field and the derived measurements from V2X hardware errors were lower than the variations that occur from attacks. As a result, the Field Cross-Validation component was able to detect attack anomalies with high precision in the motion data fields, including speed and acceleration.

Moreover, both components detect ~85% to ~95% of attacks with no more than ~20% false positive rate in EEBL, ~2% in ICW and LTA, and ~10% in FCW. Although different attack scenarios and adversaries with different capabilities were considered, VCADS was able to consistently perform with these high detection rates and low false positive compromises.

For research replication purposes, the preprocessing, validation of the datasets, and attack simulations were developed using an 8 core, 2.3 GHz processor, 16 GB of memory, and 100 GB of storage.

A. Dataset

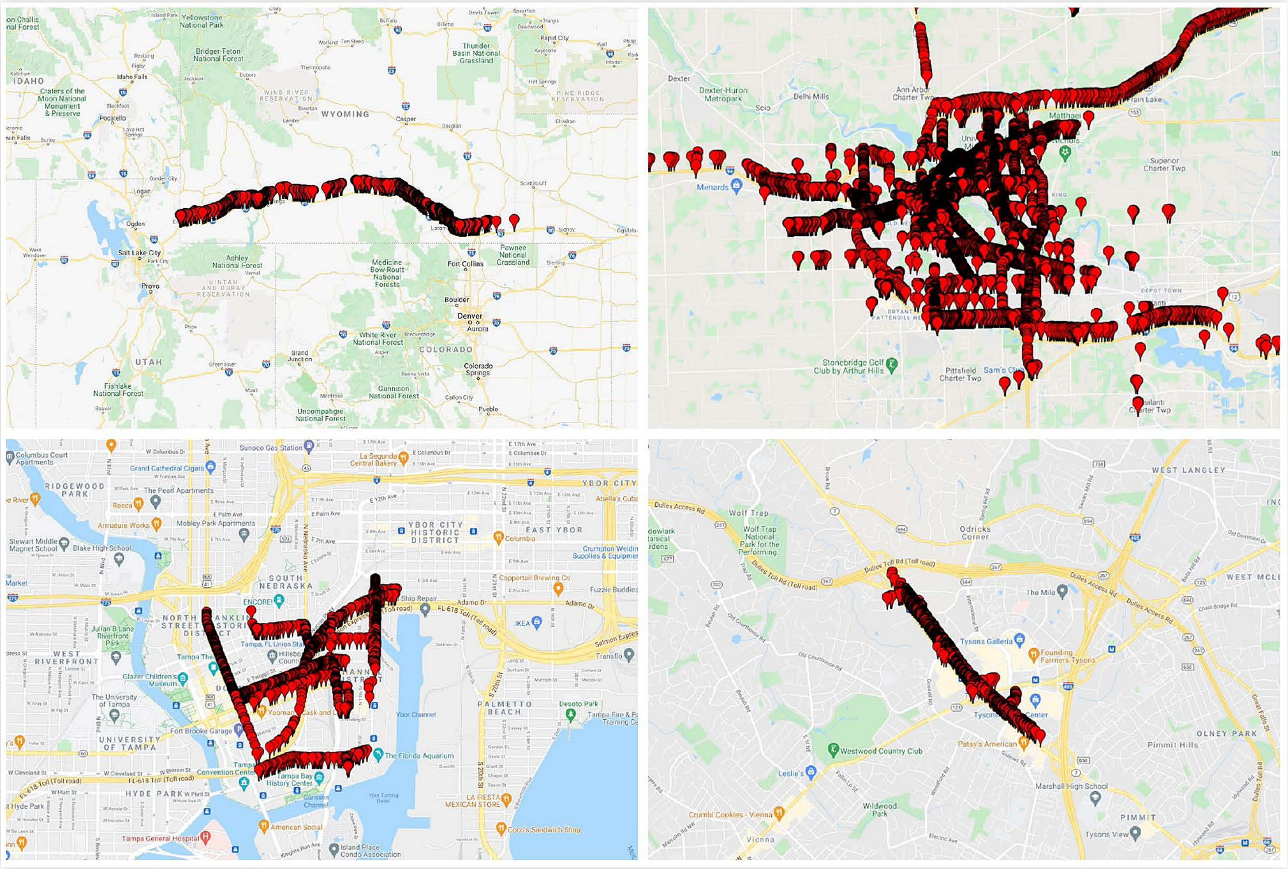
The four datasets from the V2X pilot field testing are shown in Figure 7. The BSMs in these datasets are the input of the Field Validation and Cross-Validation components. We named the datasets after the locations where the V2X pilot tests were performed: Wyoming, Ann Arbor, Tampa, and Arlington.

The Wyoming dataset is recorded on an interstate with almost straight motion patterns, which implies low field value variations. In comparison, Arlington was recorded in an interstate environment and has higher motion variations than Wyoming. Ann Arbor is a city bounded dataset that covers different driving patterns in streets. Tampa is similar to the Ann Arbor environment, although it accounts for narrower streets and more concentrated areas surrounding downtown.

Additionally, the size and amount of BSMs in each dataset varies. The total amount of BSMs per dataset is as follows: 13, 085, 109 in Ann Arbor; 5, 612, 741 in Arlington; 3, 800, 001 in Wyoming; and 34, 750 in Tampa. For these datasets, the Core Data Fields in a BSM were not reported in their entirety. Arlington has the most fields, followed by Tampa, Wyoming, and Ann Arbor (see Figure 10). However, none of the fundamental location, motion, and metadata fields used in VCADS validation were missing. Therefore, all datasets were used to perform VCADS Field Validation and Cross-Validation.

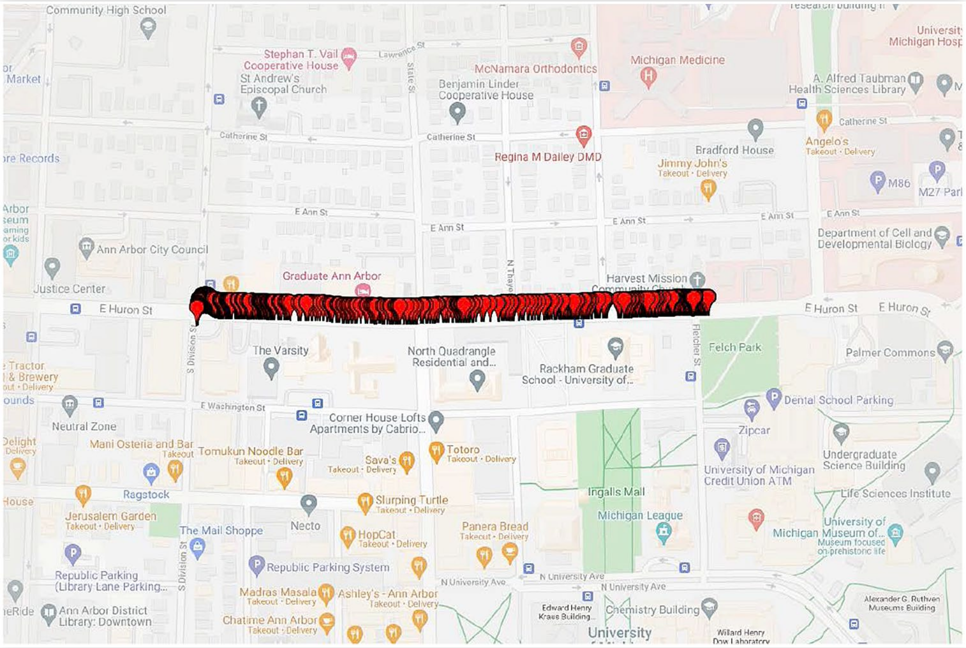
For the attacks simulations, we took a subset of 545 BSMs from the Ann Arbor dataset (as shown in Figure 8) and perturbed different values, following our threat model. Ann

FIGURE 7 BSM location of the different USDOT CVPD datasets used for the VCADS evaluation. Upper left: Wyoming state. Upper right: Ann Arbor, MI. Lower left: Tampa, FL. Lower right: Arlington, VA.



Map data © 2021 Google

FIGURE 8 Subset of Ann Arbor dataset used to test our attacks.



Map data © 2021 Google

Arbor was chosen for the simulations because it (1) had the least errors after the Field Validation and (2) has intersections needed to simulate the LTA and ICW attack. The BSMs in Wyoming did not have a Timestamp; thus, they could not be sequenced. Additionally, the interstate highway environment in Wyoming lacks the intersections needed to simulate the ICW and LTA attacks. Similarly, Arlington is a state highway and does not have the intersections needed to simulate the attacks. On the other hand, Tampa's environment can simulate all attacks, though the dataset had several missing fields and anomalies after the Field Validation. Filling those fields with the right values would have removed the real aspect of the V2X dataset field testing. Overall, Ann Arbor was a reliable dataset and had enough data fields to simulate the attacks in a subset of 545 BSMs across an avenue with intersections. For the ICW and LTA attacks, we select the intersection of East Huron Street and South State Street as a pivot point. We use this pivot point to modify BSMs and make the RV appear to reach this intersection earlier than it actually is. For the FCW and EEBL attacks, we modified the BSMs to make the RV hard brake or appear closer to an HV than it actually is.

B. Simulation Setup

All BSMs were decoded and mapped to the International System of Units (IS). The IS mapping varied depending on the dataset and the field units encoding scheme. We found the following encoding: [15] encoding, nonstandard encoding with metadata files that described the units, and imperial system units. Once all the BSMs were mapped to the IS, each BSM was taken as input to the Field Validation. Subsequently, the dataset with Timestamps and Temporary IDs were used in the Field Cross-Validation.

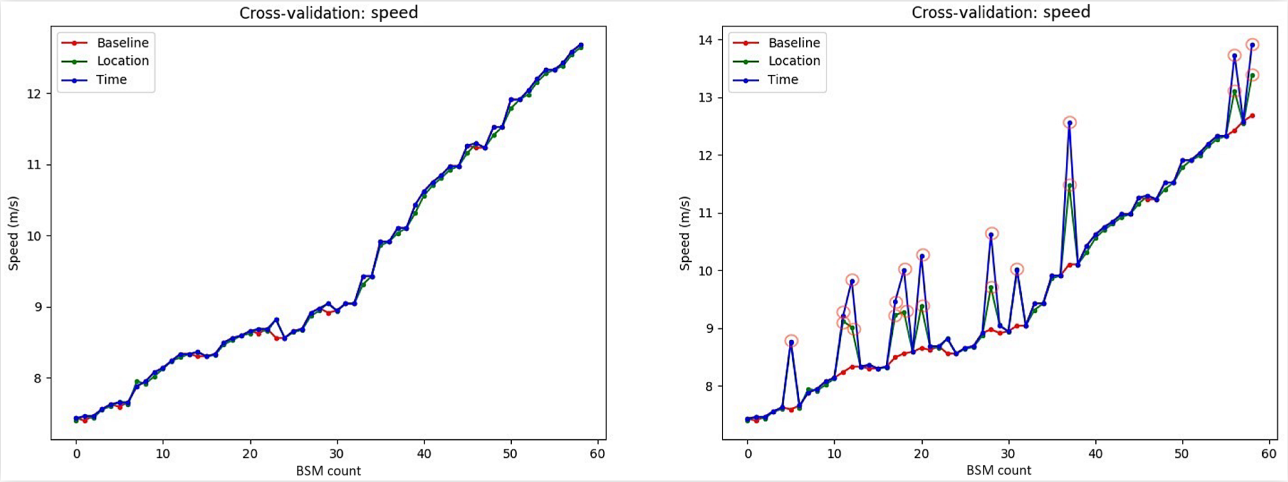
The Timestamp and Temporary ID allow the Field Cross-Validation component to understand and relate consecutive BSMs. In a real-time detection environment, the Timestamp can be derived by combining the DSecond and Message Count fields. The Temporary ID is used to link BSMs to a specific vehicle. All datasets had BSMs with Temporary IDs. However, only Ann Arbor, Arlington, and Tampa had a Timestamp in the BSMs required for the Field Cross-Validation component.

For the attack simulations, Figure 9 shows the initial state of the simulation and how it changes once an adversary introduces perturbations in certain field values. The initial state shows an average difference between the field value and its derivations of less than 0.1 m/s. In the other hand, the attacks show an average difference of 2.4 m/s in the time derivation (kinematics Equation 25) and 1.1 m/s in the location derivation (kinematics Equation 26). Although the perturbations vary according to the attack scenario, the difference between slight inconsistencies in the datasets when no attacks are present versus when they are is evident.

In the EEBL attack, the adversary manipulates the overall acceleration by perturbing the individual vector fields at different ranges. These ranges are negative accelerations that are meant to trick an HV into thinking that an RV (in front) made a hard brake (deceleration with a magnitude of $3.92\ \text{m/s}^2$ or more [12]). In our attack simulation, this overall acceleration parameter varies from $-4\ \text{m/s}^2$ to $-13\ \text{m/s}^2$, and we use VCADS to detect this anomalous behavior.

Similar to the EEBL attack, the FCW attack modifies acceleration fields to reduce the distance between the RV and

FIGURE 9 The speed and derivations of an RV in the Ann Arbor dataset. The left graph displays the initial state of an RV in a one-minute interval. The right graph shows the same RV when 11 EEBL attacks are introduced by an adversary. We have circled the derivations that start to diverge from the reported speed when the attacks are present.



© SAE International

the HV. However, no hard brakes are needed and other motion fields can be modified. The motion field values are reduced to the point that Safety Applications sense a possible collision between the RV and HV, and a false FCW is triggered.

In our FCW attack simulation, we use a reduction factor parameter that ranges between 0.0 and 1.0. A complete reduction in all motion fields is represented by 0.0. This is equivalent to an RV making a full stop. Contrarily, 1.0 represents no reduction performed. In this attack, the adversary is able to reduce Speed and/or Acceleration motion field(s) that report false locations closer to the HV.

For the LTA attack, the attacker perturbs the Speed and/or Acceleration field(s) to make an RV appear as if it is approaching an intersection faster than it actually is. This will trigger an LTA alert in the HV, which seeks to turn left while parallel and in opposing direction with respect to the RV. In our simulation, we use a distance parameter from the intersection point and a given BSM. The subsequent BSM field values are perturbed according to this distance parameter and the RV will appear as if it is reaching the intersection. Evidently, if an RV is closer to the intersection, the perturbations of the BSM location fields are going to be less than if it were further from the intersection.

Similar to LTA, the ICW attack uses a distance parameter from an intersection location in order to calculate how the motion fields are perturbed. However, in this scenario the HV approaches the intersection perpendicular to the RV heading.

C. Results

1. Field Validation The default values calculated in Section III-D were used as the boundaries of this detection component. Tighter bounds can be achieved by modifying the configurable variables of the equations in order to create higher sensitivity and flag more values of a field. Figure 10 shows the results of the Field Validation detection component in the 22, 532, 601 BSMs from all datasets.

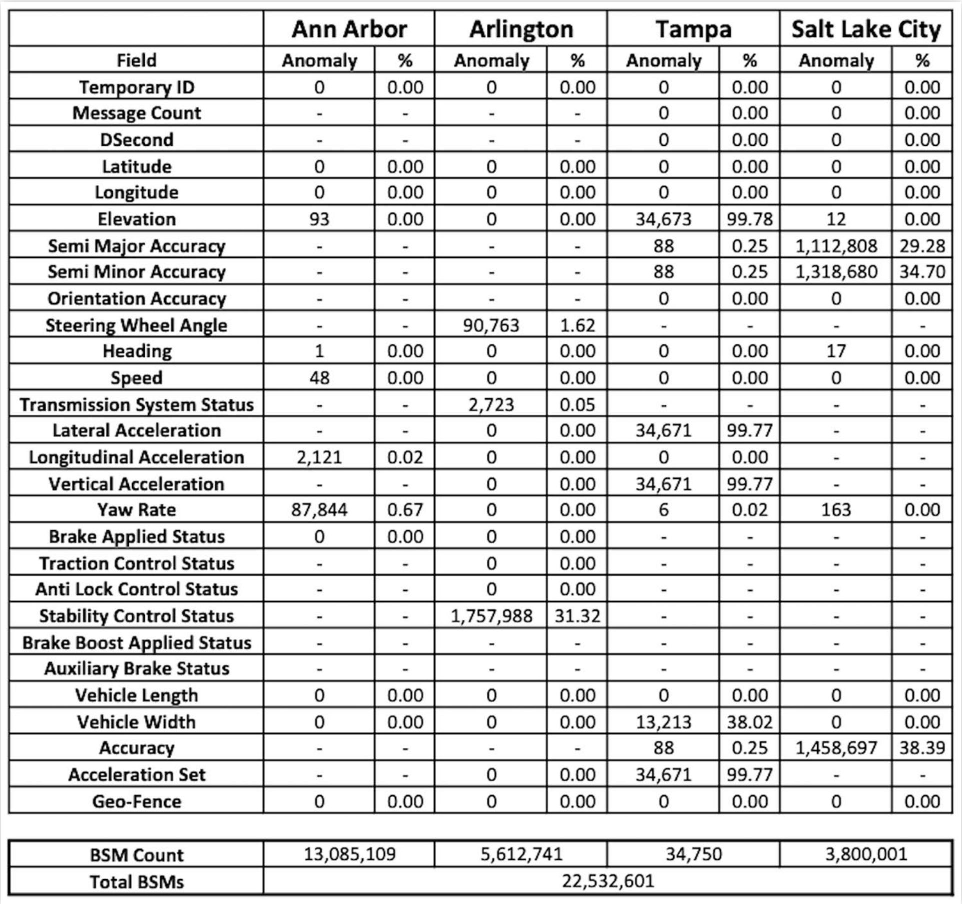
From the 28 Core Data Field values the validation was as follows: 25 Core Data Fields, including the 3 added fields explained above, where Ann Arbor had 12, Arlington 20, Tampa 20, and Wyoming 16. The results show that all datasets are incomplete. However, by analyzing all datasets we can have a full coverage of all field values in a BSM, except for Brake Boost Applied Status and Auxiliary Brake Status. Both of the latter fields measure systems are not common in most passenger vehicles at this time. The remainder of this section will expand on the specific results of each dataset.

Firstly, the Ann Arbor anomalies were negligible in comparison with the overall amount of BSMs. The heading only had one anomalous value of 360.01°. This is 0.01° higher than any possible measurement of heading from protocol limits. Speed and acceleration had 48 and 2, 121 anomalies respectively, where speed values exceeded 94.33 mph, and the accelerations recorded were beyond 11.2 m/s^2 in the longitude axis (higher than the fastest Tesla in all axes). Yaw rate showed anomalous values ranging from 97.4 deg /s to 326.59 deg /s, this is well beyond the bounds of any vehicle and turning motions, according to its structure.

Secondly, Arlington had the most fields available of all datasets. The resulting anomalous behavior was as follows: 1.62% of anomalies were found in the Steering Wheel Angle. The anomalies range from 65.89° (right above protocol threshold) to 188.96° (well beyond Ackermann's geometry). The Transmission Status showed the gear in neutral while in motion 0.05% of the time, and the Stability Control field had unavailable values in 31.32% of the BSMs.

Thirdly, the Tampa dataset had a wide range of anomalies. Although Tampa is close to sea level, negative elevations were reported in 99.78% of the BSMs. This figure suggests a poor calibration of the altimeter sensor. Similarly, both vertical and latitude accelerations were unavailable 99.77% of the time,

FIGURE 10 Field Validation results in V2X pilot field testing datasets from four different cities.



	Ann Arbor		Arlington		Tampa		Salt Lake City	
Field	Anomaly	%	Anomaly	%	Anomaly	%	Anomaly	%
Temporary ID	0	0.00	0	0.00	0	0.00	0	0.00
Message Count	-	-	-	-	0	0.00	0	0.00
DSecond	-	-	-	-	0	0.00	0	0.00
Latitude	0	0.00	0	0.00	0	0.00	0	0.00
Longitude	0	0.00	0	0.00	0	0.00	0	0.00
Elevation	93	0.00	0	0.00	34,673	99.78	12	0.00
Semi Major Accuracy	-	-	-	-	88	0.25	1,112,808	29.28
Semi Minor Accuracy	-	-	-	-	88	0.25	1,318,680	34.70
Orientation Accuracy	-	-	-	-	0	0.00	0	0.00
Steering Wheel Angle	-	-	90,763	1.62	-	-	-	-
Heading	1	0.00	0	0.00	0	0.00	17	0.00
Speed	48	0.00	0	0.00	0	0.00	0	0.00
Transmission System Status	-	-	2,723	0.05	-	-	-	-
Lateral Acceleration	-	-	0	0.00	34,671	99.77	-	-
Longitudinal Acceleration	2,121	0.02	0	0.00	0	0.00	-	-
Vertical Acceleration	-	-	0	0.00	34,671	99.77	-	-
Yaw Rate	87,844	0.67	0	0.00	6	0.02	163	0.00
Brake Applied Status	0	0.00	0	0.00	-	-	-	-
Traction Control Status	-	-	0	0.00	-	-	-	-
Anti Lock Control Status	-	-	0	0.00	-	-	-	-
Stability Control Status	-	-	1,757,988	31.32	-	-	-	-
Brake Boost Applied Status	-	-	-	-	-	-	-	-
Auxiliary Brake Status	-	-	-	-	-	-	-	-
Vehicle Length	0	0.00	0	0.00	0	0.00	0	0.00
Vehicle Width	0	0.00	0	0.00	13,213	38.02	0	0.00
Accuracy	-	-	-	-	88	0.25	1,458,697	38.39
Acceleration Set	-	-	0	0.00	34,671	99.77	-	-
Geo-Fence	0	0.00	0	0.00	0	0.00	0	0.00
BSM Count	13,085,109		5,612,741		34,750		3,800,001	
Total BSMs	22,532,601							

which caused the overall acceleration set to be flagged in the same way. Furthermore, the vehicle width presented anomalies in 38.02% of the BSMs. The width reported for some vehicles was 3.1 m instead of the manufacturing limit of 2.6.

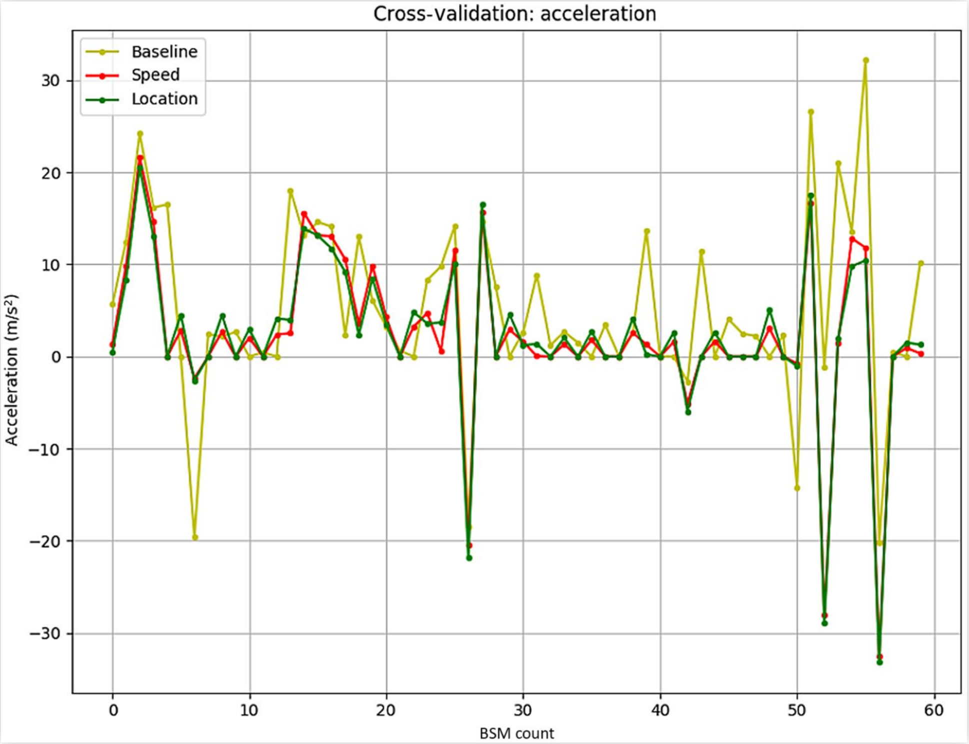
Finally, Wyoming had several anomalies in the semi major and minor accuracy fields. A total of 29.28% of the semi-major and 34.70% of the semi-minor fields had anomalies between 2.7 m and 12.65 m. The average inaccuracy was equivalent to 7.82 m in the semi-major field and 7.92 m in the semi-minor fields. The overall BSM accuracy showed anomalies 38.39% of the time, suggesting a low GPS precision that is not fit for V2X Safety Applications processing. Other field anomalies were negligible compared to the overall BSMs in the dataset. These anomalies were elevation with 12 BSMs of unavailable data, heading with 17 BSMs of values beyond protocol measurements (28, 800 rad), and yaw rate with 163 values between 84.8 deg /s and 326.59 deg /s, resulting in an average of 262.41 deg /s.

2. Field Cross-Validation The results in this component were achieved by dividing BSMs using the Temporary ID. The BSMs from each RV were sorted in chronological order. In real-time mechanisms DSecond and Message Count are used to derive the BSM Timestamp. However, in these datasets, DSecond values wrap around the 60,000 milliseconds and Message Count around 127, making it impossible to know the exact order of the BSMs. Fortunately, the datasets come with a Timestamp metadata field for each BSM. The Timestamp is created when all the Core Data Fields have been measured, right before sending the BSMs. This metadata allows us to sort the BSMs and use them as inputs for the Field Cross-Validation.

As explained in Section III, we derive a given field measurement using other independent data fields found in the same and consecutive BSMs. With regard to acceleration, Figure 11 shows three derived measurements: time, location, and speed. The fourth measurement represents the actual field value referred to as baseline. The time measurement for acceleration is based on Equation 23 or 24, speed uses Equation 25 and location Equation 26. All the variables in these equations are filled with fields in the previous and current BSMs. Additionally, acceleration is cross-validated with the transmission and brake system. These values are not plotted due to the fact that they are binary validations on the acceleration sign and magnitude.

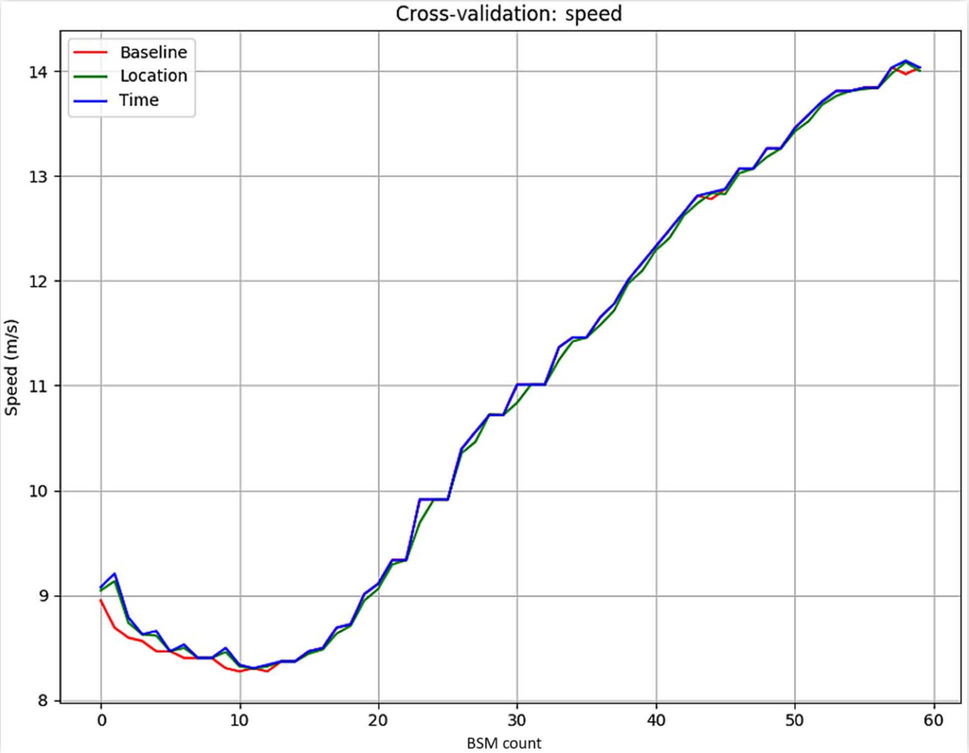
For Figure 12, we show two measurements: location and time, as the derived measurements of the reported baseline. Time uses Equation 25 and location Equation 26.

FIGURE 11 A total of 60 BSMs' acceleration and its derivations are plotted over an ~10-second period from an RV in the Arlington dataset. This figure shows an overall inconsistency with the acceleration reported and its derivations.



© SAE International

FIGURE 12 A total of 60 BSMs' speed and its derivations are plotted over an ~10-second period.



© SAE International

3. Attack Simulations For the attack simulations we isolated an RV that cruises around Ann Arbor. Its reported BSMs are perturbed according to the explored attacks in this article (i.e., FCW, ICW, LTA, EEBL). We used 545 BSMs shown in Figure 8. The Cross-Validation mechanism was used to defend against the data perturbations of these attack scenarios by detecting the anomalies that they caused.

The attack scenarios were implemented according to our attack model in terms of adversarial capabilities and specifications. The results of each attack scenario are plotted as ROC curves.

For EEBL, Figure 13 shows the ROC curves resulting from an attack at different acceleration ranges. The EEBL warning in [39] defines a hard brake (deceleration) of −3.92 m/s^2 or higher magnitude. We show the results for hard brakes ranging from −4 m/s^2 to −13 m/s^2. These values translate from a vehicle suddenly braking from an unexpected situation to a vehicle braking with its highest possible deceleration and ideal friction conditions.

The EEBL attack scenario shows that as the deceleration magnitude of the attack increases, it becomes easier to detect. An adversary that wants to be as stealthy as possible will try to trigger an alert with the smallest deceleration value (−3.92 m/s^2). The overall True Positive Rate ranged from 80% (−4 m/s^2 perturbations) to 97 % (−13 m/s^2 perturbations), which indicates a very high detection ratio.

Given the nature of the Acceleration Set fields in the BSMs, VCADS jumps above the random line (50/50 True Positive and False Positive rate) at 21%. As a result, some BSMs were incorrectly detected as anomalous. The overall compromise between True Positive and False Positive rates is justifiable and shows that VCADS is able to detect EEBL attacks with high precision and with few compromises.

The FCW attack results (in Figure 14) depict higher detection ratios and less False Positive rate compromises. The reductions done in the RV by the adversary were easier to detect when the factor approaches to 0.0, rather than staying close to 1.0. Overall, there was a 80% True Positive detection rate with False Positive rates compromises between 7% and 11%. After 11%, all reduction factors seem to converge and increase the detection ratio as the sensitivity decreases without any regard for the reduction factor.

For the LTA attack, more than one field was perturbed by the adversary. Figure 3 shows that there were high efficiency detection ratios (between 85% and ~100%) with almost no compromises (0% to 3% False Positive rate). With an adversary that can only manipulate the Acceleration Set fields, the detection of our mechanism was the highest. This was followed by an adversary that can only manipulate Speed, and finally, an adversary that is able to manipulate both the Acceleration Set and Speed fields. As the True Positive and False Positive rate increase, the benefits of an adversary that can manipulate more data fields becomes evident.

The results of the ICW attack (in Figure 15) are similar to the LTA attack. This is a consistent result with the fact that both LTA and ICW attacks are crafted based on the

FIGURE 13 ROC curve of the Cross-Validation mechanism detecting anomalies from an EEBL attack scenario at different acceleration values (ranging from −4 m/s^2 to −13 m/s^2).

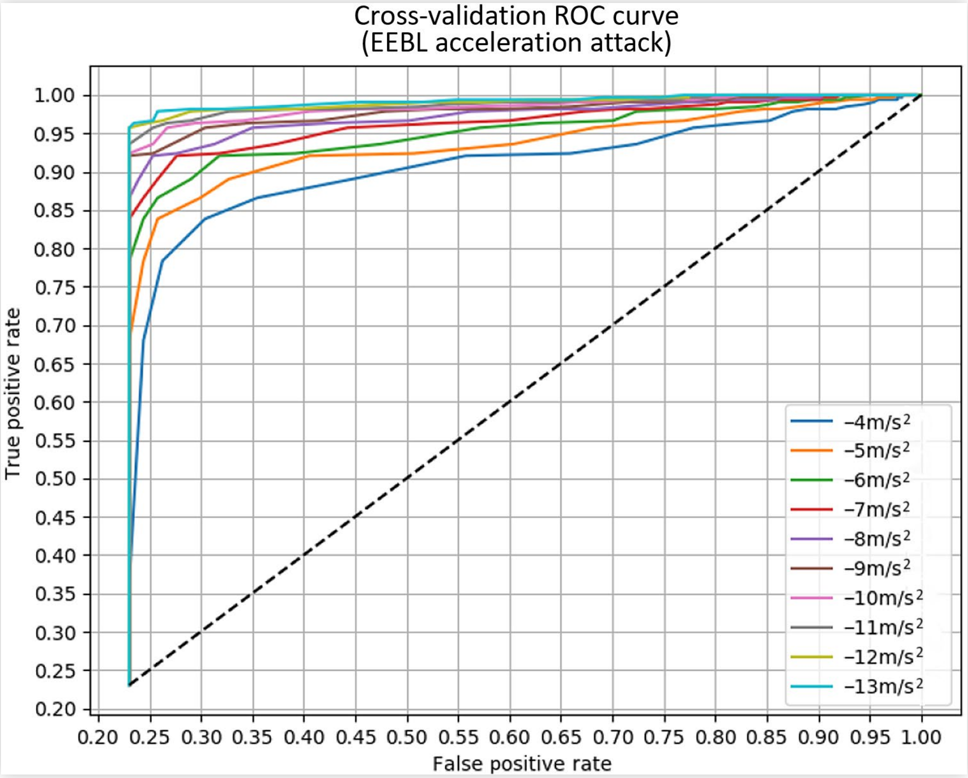
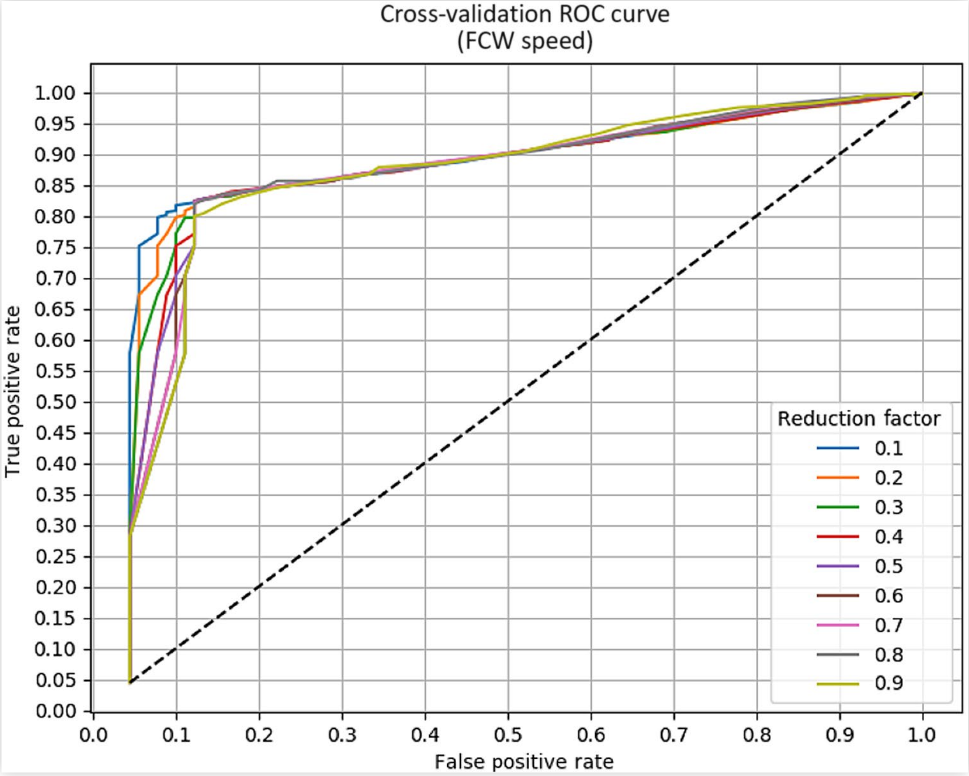
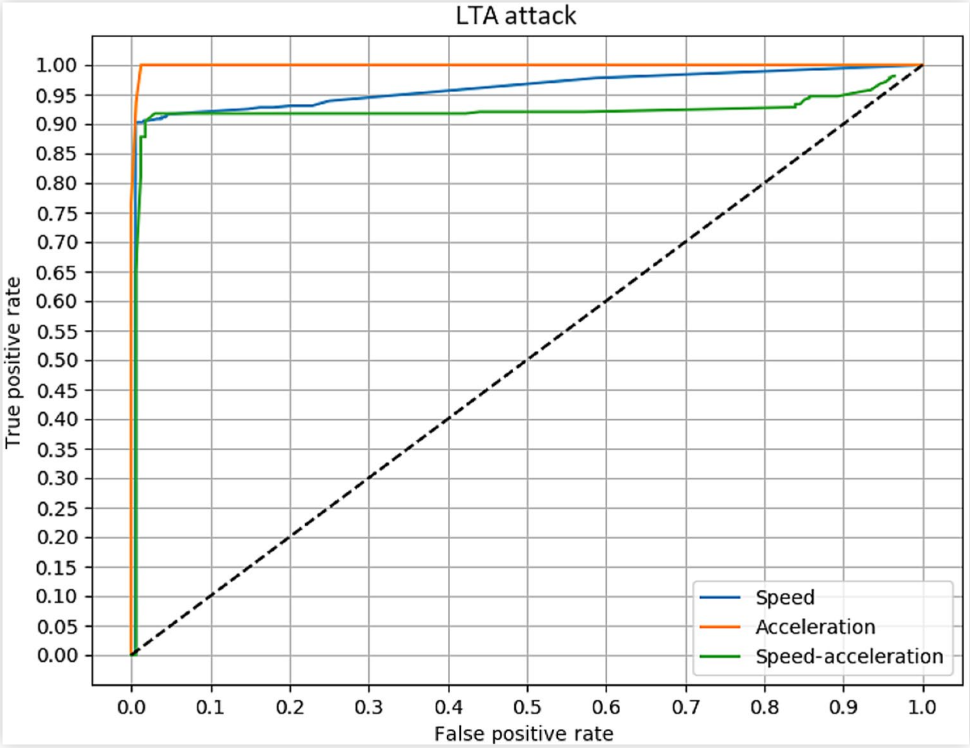


FIGURE 14 ROC curve of the Cross-Validation mechanism detecting anomalies from an FCW attack scenario. The adversary controls the speed capability and reduces it to show the RV in imminent collision with the HV.



© SAE International

FIGURE 15 ROC curve of the Cross-Validation mechanism detecting anomalies from an LTA attack scenario. The adversary controls different capabilities: Speed and/or Acceleration.



© SAE International

intersection location. Although the True Positive rate of the ICW attack was not as high as the LTA attack, which varies from 85% to 90%), it had very similar compromises in the False Positive rate, and the difference in capability fields of an adversary was evident.

Furthermore, Section III-B shows that an adversary may manipulate one or more field values to attack an HV and trigger false warnings. If an adversary is able to manipulate all fields in consecutive BSMs and synchronize them to be consistent with each other (using the Field Cross-Validation equations or other approximation mechanisms), the variation between fields and derivations will not be detected as we can see in these results. For this reason, if the adversaries are aware of the Cross-Validation mechanism, they will not be able to bypass this mechanism as long as they cannot manipulate all the vehicular fields.

V. Related Works

In this section, we present related works on data-centric trust and misbehavior detection in V2X.

So et al. [7] proposed an RSSI-based misbehavior detector and tested it on the VeReMi dataset. However, signal strength is not an stable indicator as it can be faked by increasing or decreasing the transmit power. Similar detection mechanisms were explored in [20, 21, 22, 23, 24, 25], including trust models that allow vehicles to share the detected signal strength through tables. One of the caveats in these tables is the vector attacks that span from trusting other vehicles (e.g., Sybil and Byzantine attacks). A detection system to verify transmission signals' energy was developed in [40]. The system extracts features of pre-established anomalous signals and uses it to validate future signals.

Golle et al. [41] validated V2X data traffic by finding explanations as to why data has certain values. It uses internal physical sensors, such as radar. This approach limits the amount of vehicles that can be validated, as it is dependent on the presence of specific sensors in vehicles.

Unsupervised learning model (i.e., K-means clustering for vehicular distances) was used in [42]. It uses speed and acceleration values in order to detect vehicles outside the clusters. K-means clustering optimizes the hyper-parameter K before it can be used, but here it is arbitrarily chosen by clustering vehicles into groups of 30. This rather creates an unstable model that does not necessarily fit the population of vehicles. Moreover, vehicle count is often inaccurate as vehicles are constantly changing their temporal IDs and certificates. Thus, using unsupervised approaches is challenging in highly dynamic environments, such as modeling traffic patterns.

Similarly to [42, 43] used K-means clustering for fuzzy time series to detect Sybil attacks. Unlike [42, 43] optimized the K hyper-parameter. The whole article assumed Sybil attacks are co-located but did not specify distance ranges and the ability of receivers to monitor vehicles as their IDs constantly change. Contrary to [42, 43, 44] applied supervised machine learning to extract vehicular features regarding location, speed, acceleration, and message periodicity; however, supervised learning requires previously defined training data, and the variability of vehicular environments is high enough that the training data in one environment cannot be trivially generalized to other environments.

Attacks on stationary features of a BSM (e.g., vehicle dimensions) were analyzed in [6]. Machine learning algorithms such as MLP, AdaBoost, and Random Forest were used to detect attacks with a high success rate.

Finally, [1] developed a detection mechanism similar to ours. It uses Kalman filter to predict the future behavior of vehicles by relating position, speed, and acceleration. This is the same theory used in VCADS' Field Cross-Validation component. However, redundant data is not explored in depth and models to constrain single fields were not shown.

VI. Conclusion

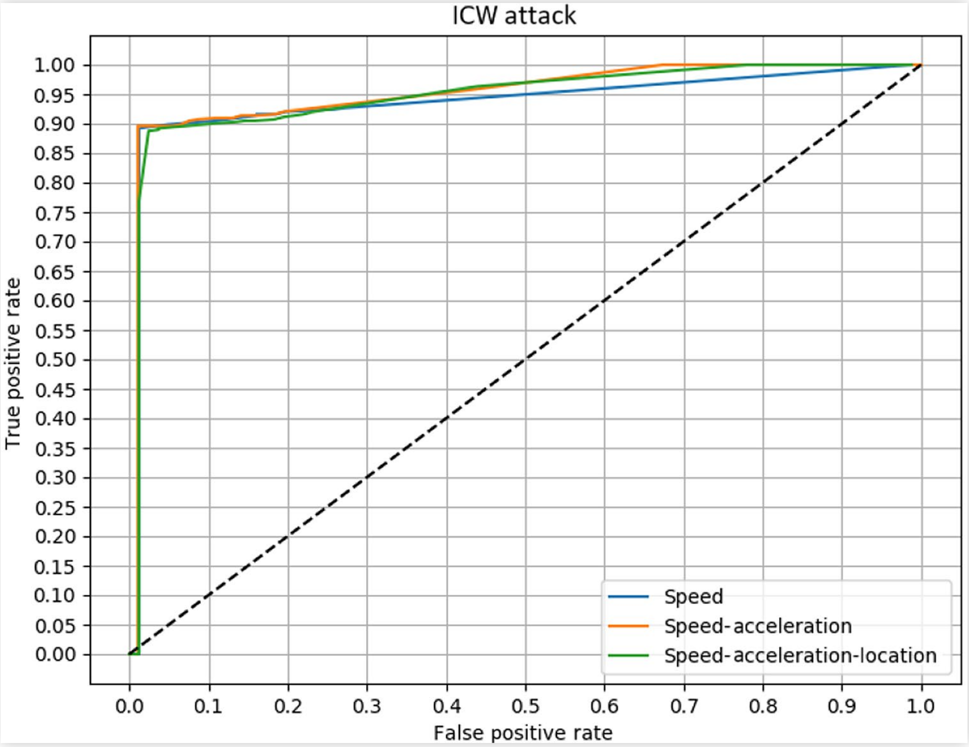
V2X is an important technology capable of synchronizing vehicles and making the transportation system safer and more efficient. Security efforts have been focused on providing reliable message transmissions and detecting signal transmission anomalies. However, detecting anomalous content from incoming BSMs is still under research and development.

An anomaly detection system at the Safety Applications layer of an ITS stack ensures that data inaccuracies and inconsistencies, coming from an adversary or erroneous sensors, can be detected and filtered out. This security system is paramount to create reliable Safety Applications that can warn drivers or self-driving algorithms of real possible collisions. In this article, we proposed physics-based misbehavior detectors for BSMs. Our solution, called VCADS, was implemented and tested on real datasets provided by the USDOT Connected Vehicle Pilot Deployment and other State DOTs. We simulated a series of attacks, and demonstrated that sensor error and attack perturbations can be detected with field validation and cross-validation constraints.

The attacks on LTA and ICW were proven to be inefficient and easily detectable by VCADS. On the other hand, the EEBL and FCW false attacks were more effective and yielded higher false positive rates. In the case of the EEBL attack, higher false positive rates than expected were attributed to the noise levels in the acceleration parameter of the datasets. Real-life datasets in V2X are expected to have noise levels from the environment and hardware sensor measurements. Nevertheless, the results in Figures 13, 14, 15, and 16 show that, even with such noise levels, VCADS can be implemented and has proven to be useful at detecting anomalies that come from malicious attacks and misbehaved vehicles. We hope that standardization bodies in V2X use our constraint models to specify minimum requirements of misbehavior detection for commercial V2X platforms.

As future work, VCADS will be tested against a wider range of attacks. Indeed, such tests will allow us to formally learn what classes of misbehavior can be detected via our

FIGURE 16 ROC curve of the Cross-Validation mechanism detecting anomalies from an ICW attack scenario with an adversary controlling different capabilities: Speed, Speed and Acceleration, or Speed, Acceleration, and location altogether.



© SAE International

physics-based detectors. Moreover, VCADS will be added to the open-source simulation framework F^2MD [45] to ensure fair comparison against other detection systems.

Contact Information

Alejandro Andrade Salazar
alejandro_liga@hotmail.com

Acronym Table

BSM - Basic Safety Message

BSW - Blind Spot Warning

C-V2X - Cellular V2X

CAM - Cooperative Awareness Message

DSRC - Dedicated Short Range Communications

ECU - Electronic Control Unit

EEBL - Emergency Electronic Brake Lights

FCW - Forward Collision Warning

HV - Host Vehicle

ICW - Intersection Collision Warning

IMA - Intersection Movement Assist

LCW - Lane Change Warning

LDM - Local Dynamic Map

LLC - Logical Link Control

LTA - Left Turn Assist

PDCP - Packet Data Convergence Protocol

RLC - Radio Link Control

RV - Remote Vehicle

TTC - Time to Collision

USDOT - United States Department of Transportation

V2X - Vehicle to Everything

VCADS - V2X Core Anomaly Detection System

WAVE - Wireless Access in Vehicular Environments

WSM - WAVE Short Message

WSMP - WSM Protocol

References

1. Jaeger, A., Bimeyer, N., Stbing, H., and Huss, S., “A Novel Framework for Efficient Mobility Data Verification in Vehicular AD-HOC Networks,” *International Journal of Intelligent Transportation Systems Research* 10 (2012): 11-21.
2. IEEE, “IEEE Guide for Wireless Access in Vehicular Environments (Wave) Architecture,” IEEE Std. 1609.0-2019 (Revision of IEEE Std. 1609.0-2013), 1-106, April 2019.

3. Hadded, M., Shagdar, O., and Merdrignac, P., “Augmented Perception by v2x Cooperation (Pac-v2x): Security Issues and Misbehavior Detection Solutions,” in *2019 15th International Wireless Communications Mobile Computing Conference (IWCMC)*, Tangier, Morocco, 2019, 907-912.
4. Petit, J. and Shladover, S., “Potential Cyberattacks on Automated Vehicles,” *IEEE Transactions on Intelligent Transportation Systems* 16 (2014): 546-556.
5. Petit, J., Bas Stottelaar, M., and Kargl, F., “Remote Attacks on Automated Vehicles Sensors: Experiments on Camera and Lidar,” BlackHat Europe, 2015.
6. Monteuuis, J., Petit, J., Zhang, J., Labiod, H. et al., “My Autonomous Car Is an Elephant: A Machine Learning Based Detector for Implausible Dimension,” in *2018 Third International Conference on Security of Smart Cities, Industrial Control System and Communications (SSIC)*, Shanghai, China, October 2018, 1-8.
7. So, S., Petit, J., and Starobinski, D., “Physical Layer Plausibility Checks for Misbehavior Detection in v2x Networks,” in *Proceedings of the 12th Conference on Security and Privacy in Wireless and Mobile Networks—WiSec 19*, 2019.
8. van der Heijden R.W., Dietzel S., Leinmüller T., and Kargl F., “Survey on Misbehavior Detection in Cooperative Intelligent Transportation Systems,” arXiv:1610.06810, 2016, http://arxiv.org/abs/1610.06810.
9. Toh, P.C.K., *AD HOC Mobile Wireless Networks* (Pearson India, 2001). https://books.google.com/books?id=VkX3twEACAAJ.
10. IEEE, “IEEE Standard for Wireless Access in Vehicular Environments (Wave)—Networking Services,” IEEE Std. 1609.3-2016 (Revision of IEEE Std. 1609.3-2010), 1-160, April 2016.
11. IEEE, “IEEE Standard for Wireless Access in Vehicular Environments-Security Services for Applications and Management Messages,” IEEE Std. 1609.2-2016 (Revision of IEEE Std. 1609.2-2013), 1-240, March 2016.
12. SAE International, “On-Board System Requirements for V2V Safety Communications,” SAE Standard J2945/1_201603, March 2016, https://doi.org/10.4271/J2945/1201603.
13. Petit, J., Schaub, F., Feiri, M., and Kargl, F., “Pseudonym Schemes in Vehicular Networks: A Survey,” *IEEE Communications Surveys Tutorials* 17, no. 1 (2015): 228-255.
14. Lin, X., Lu, R., Zhang, C., Zhu, H. et al., “Security in Vehicular AD HOC Networks,” *IEEE Communications Magazine* 46, no. 4 (2008): 88-95.
15. SAE International, “Dedicated Short Range Communications (DSRC) Message Set Dictionary ASN File,” SAE Standard J2735ASN_201603, March 2016, https://doi.org/10.4271/J2735ASN201603.
16. ETSI, “Intelligent Transport Systems (Its); Vehicular Communications; Basic Set of Applications; Local Dynamic Map (LDM); Rationale for and Guidance on Standardization,” ETSI TR, vol. 102 863, no. V1.1.1, June 2011.
17. Monteuuis J.-P., “Attacker Model for Connected and Automated Vehicles,” 2018.
18. Cho K.-T. and Shin K.G., “Fingerprinting Electronic Control Units for Vehicle Intrusion Detection,” in *25th USENIX Security Symposium (USENIX Security 16)* (Austin, TX: USENIX Association, August 2016), 911-927, https://www.usenix.org/conference/usenixsecurity16/technicalsessions/presentation/cho.
19. Henniger, O., Ruddle, A., Seudi, H., Weyl, B. et al., “Securing Vehicular On-Board IT Systems: The Evita Project,” in *VDI/VW Automotive Security Conference*, 2009.
20. Schmidt, R., Leinmller, T., Schoch, E., Held, A. et al., “Vehicle Behavior Analysis to Enhance Security in Vanets,” in *Proceedings of the 4th IEEE Vehicle-to-Vehicle Communications Workshop (V2VCOM2008)*, 2020.
21. van der Heijden, R.W., Al-Momani, R.W., Kargl, F., and Abu-Sharkh, O.M.F., “Enhanced Position Verification for Vanets Using Subjective Logic,” in *2016 IEEE 84th Vehicular Technology Conference (VTC-Fall)*, Montreal, Canada, 2016, 1-7.
22. Leinmller, T., Schmidt, R., Schoch, E., Held, A. et al., “Modeling Roadside Attacker Behavior in Vanets,” in *2008 IEEE Globecom Workshops*, New Orleans, LA, 2009, 1-10.
23. Leinmller, T., Schoch, E., Kargl, F., and Maihfer, C., “Decentralized Position Verification in Geographic AD HOC Routing,” *Security and Communication Networks* 3, no. 4 (2010): 289-302, https://doi.org/10.1002/sec.56.
24. Ruj, S., Cavenaghi, M.A., Huang, Z., Nayak, A. et al., “On Data-Centric Misbehavior Detection in Vanets,” in *2011 IEEE Vehicular Technology Conference (VTC Fall)*, San Francisco, CA, 2011, 1-5.
25. Lo, N. and Tsai, N., “Illusion Attack on Vanet Applications—A Message Plausibility Problem,” in *2007 IEEE Globecom Workshops*, Washington, DC, 2007, 1-8.
26. Ansari, J.A., Sharma, S., Majumdar, A., Murthy, J.K. et al., “The Earth Ain’t Flat: Monocular Reconstruction of Vehicles on Steep and Graded Roads from a Moving Camera,” in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Madrid, Spain, October 2018, 8404-8410.
27. Misachi, J., “Steepest Streets in the World,” World Atlas, June 2020.
28. IIHS, “Speed: Maximum Posted Speed Limits by State,” https://www.iihs.org/topics/speed/speed-limit-laws.
29. US Government Printing Office, “Public Law 94-280: 94th Congress, H.R. 8235, May 5, 1976: An Act to Authorize Appropriations for the Construction of Certain Highways in Accordance with Title 23 of the United States Code and for Other Purposes,” For Sale by the Superintendent of Documents, US Government Printing Office, 1976.
30. King-Hele, D., “Erasmus Darwin’s Improved Design for Steering Carriages—And Cars,” *Notes and Records of the Royal Society of London* 56, no. 1 (2002): 41-62, https://doi.org/10.1098/rsnr.2002.0166.
31. Simionescu, P.A. and Beale, D., “Optimum Synthesis of the Four-Bar Function Generator in Its Symmetric Embodiment: The Ackermann Steering Linkage,” *Mechanism and Machine Theory* 37, no. 12 (2002): 1487-1504.

32. SAE International, "Vehicle Dynamics Terminology," SAE Standard J670_200801, January 2008, https://doi.org/10.4271/J670_200801.

33. MacKenzie, D. and Heywood, J., "Acceleration Performance Trends and Evolving Relationship between Power, Weight, and Acceleration in us Light-Duty Vehicles: Linear Regression Analysis," *Transportation Research Record* 2287, no. 1 (2012): 122-131.

34. Tesla, "Tesla Model S," https://www.tesla.com/models.

35. SAE International, "Surface Vehicle Recommended Practice, Use of the Critical Speed Formula," SAE Standard J2969_201701, 2017, https://doi.org/10.4271/J2969_201701.

36. Knight, R.D., *Physics for Scientists and Engineers: A Strategic Approach* (Boston, MA: Pearson, 2017)

37. Robusto, C.C., "The Cosine-Haversine Formula," *The American Mathematical Monthly* 64, no. 1 (1957): 38-40, https://doi.org/10.2307/2309088.

38. http://www.unoosa.org/pdf/icg/2012/template/WGS_84.pdf.

39. IEEE, "IEEE Standard for Wireless Access in Vehicular Environments (Wave)—Multi-Channel Operation," IEEE Std. 1609.4-2016 (Revision of IEEE Std. 1609.4-2010), 1-94, March 2016.

40. Eriksson, B., Barford, P., Bowden, R., Duffield, N. et al., "Basisdetect: A Model-Based Network Event Detection Framework," in *IMC '10: Proceedings of the 10th ACM SIGCOMM Conference on Internet Measurement* (New York: ACM, 2010), 451-464, https://doi.org/10.1145/1879141.1879200.

41. Golle, P., Greene, D., and Staddon, J., "Detecting and Correcting Malicious Data in VANETs," in *VANET '04: Proceedings of the 1st ACM International Workshop on Vehicular AD-HOC Networks* (New York: ACM, 2004), 29-37, https://doi.org/10.1145/1023875.1023881.

42. Balzano, W. and Vitale, F., "Rads: A Smart Road Anomalies Detection System Using Vehicle-2-Vehicle Network and Cluster Features(s)," in *DMSVIVA 2018*, 2018, 51-56.

43. Dutta, N. and Chellappan, S., "A Time-Series Clustering Approach for Sybil Attack Detection in Vehicular AD-HOC Networks," in *The Second International Conference on Advances in Vehicular Systems, Technologies and Applications*, Nice, France, 2013.

44. So, S., Sharma, P., and Petit, J., "Integrating Plausibility Checks and Machine Learning for Misbehavior Detection in Vanet," in *2018 17th IEEE International Conference on Machine Learning and Applications (ICMLA)*, Orlando, FL, December 2018, 564-571.

45. Kamel, J., Ansari, M.R., Petit, J., Kaiser, A. et al., "Simulation Framework for Misbehavior Detection in Vehicular Networks," *IEEE Transactions on Vehicular Technology* 69, no. 6 (2020): 6631-6643.

Positions and opinions advanced in this work are those of the author(s) and not necessarily those of SAE International. Responsibility for the content of the work lies solely with the author(s).