Change Point Models for Real-time Cyber Attack Detection in Connected Vehicle Environment

Gurcan Comert, Mizanur Rahman, *Member, IEEE*, Mhafuzul Islam, and Mashrur Chowdhury, *Senior Member, IEEE*

Connected vehicle (CV) systems are cognizant of potential cyber attacks because of increasing connectivity between its different components such as vehicles, roadside infrastructure, and traffic management centers. However, it is a challenge to detect security threats in real-time (i.e., less than 0.1 second) and develop appropriate or effective countermeasures for a CV system because of the dynamic behavior of such attacks, high computational power requirement, and a historical data requirement for training detection models. To address these challenges, statistical models, especially change point models, have potentials for real-time anomaly detection. Thus, the objective of this study is to investigate the efficacy of two change point models, Expectation Maximization (EM) and two forms of Cumulative Summation (CUSUM) algorithms (i.e., typical and adaptive), for real-time vehicle-to-infrastructure (V2I) cyber attack detection in a CV Environment. To prove the efficacy of these models, we evaluated these two models for three different type of cyber attack, denial of service (DOS), impersonation, and false information, using basic safety messages (BSMs) generated from CVs through simulation. Results from numerical analysis revealed that EM, CUSUM, and adaptive CUSUM (aCUSUM) could detect these cyberattacks, such as DOS, impersonation, and false information with low false positives.

Index Terms—**Cyber Attack Detection, Connected Vehicles, Expectation Maximization, CUSUM, Roadside Equipment.**

I. Introduction

THe driving force behind the US economic engine is the surface transportation system, which enables reliable and efficient transportation of passengers and goods [1]. However, human errors (e.g., poor judgment, fatigue) are the leading causes of more than 94% of US highway fatalities [2]. To reduce these fatalities and associated societal costs by reducing or eliminating the influence of the human errors, the US Department of Transportation (USDOT) has been promoting connected and automated vehicles (CAV) [3], [4]. From recent reports of National Highway Traffic Safety Administration [5], [6], several benefits are foreseen with this CAV technologies, such as up to 80% reduction in fatalities from multi-vehicle crashes and preventing the majority of human error related incidents. In such CAV systems, massive amounts of data will be produced and exchanged between different components through different data communication medium, such Dedicated Short Range Communication (DSRC), WiFi, 5G and Long Term Evolution (LTE) [7], [8]. These data can be processed in a cloud, or in an edge computing device at the roadside (i.e., roadside transportation infrastructure) based on different CAV application requirements [8], [9]. Communication technologies supporting data exchange must also be secured to support CAV operations with specific requirements (e.g., delay, bandwidth and communication range). With the increase of connectivity in transportation networks, this CAV systems is cognizant of potential cyber attacks [10], [11]. In one of the recent review papers, Hahn et al. discussed current challenges as scalability when large data is available and delay sensitivity [12]. In addition, privacy preserved framework was introduced by perturbation and compression in [13].

As cybersecurity attacks are dynamic, it is a challenge to detect security threats in real-time and develop appropriate or effective countermeasures for connected transportation system [14]. To increase security and resiliency due to possible attacks or benign system errors by different events, research is needed to investigate detection techniques for different attack types, such as denial of service (DOS), impersonation, false information [15], [16]. Anomaly detection techniques are well-studied in different areas. Specifically, the cybersecurity of firmware updates, cybersecurity on heavy vehicles, vehicle-to-vehicle (V2V) communication interfaces, and trusted vehicle-to-everything (V2X) communications [17].

Different type of anomaly detection models exist in literature, such as rule-based, machine learning (ML) and data mining (DM) (including expert systems)-based, and statistical inference-based models. These can be listed as K-means, random forest, Bayesian networks, Gaussian processes, decision trees, neural networks, support vector machines, and hypothesis testing and point estimation based process control models respectively. Recent survey studies related to anomaly detection are summarized a comprehensive review of machine learning and rule (signature)-based methods, and their applications to intrusion detection systems (IDS) [18], [19]. Rule-based attack detection models, originated from cryptography, are abundant especially for their efficiency and computationally light-weight [20]. However, rule-based models require a detailed understanding of the data generation process and adaptivity or customization based on their respective environment to develop the model. On the other hand, both ML and DM-based attack detection models are adaptable to different attack types both known and unknown patterns [21]. However, major concerns are computational complexity for real-time application, training the model with different cyber attack scenarios, unavailability of cyberattack data in the transportation domain, and determination of update

Manuscript received January 31, 2020; Corresponding author: G. Comert (email: gurcan.comert@benedict.edu). G. Comert is with the Department of Computer Science, Physics, and Engineering, Benedict College, Columbia, SC 29204, USA and Information Trust Institute, University of Illinois Urbana-Champaign, 1308 West Main St., Urbana, IL 61801 USA. M. Rahman is with Department of Civil, Construction, and Environmental Engineering, University of Alabama, Tuscaloosa, AL 35487, USA, M. Islam, and M. Chowdhury are with the Glenn Department of Civil Engineering, Clemson University, Clemson, SC 29634, USA

or retraining window. To address these problems, statistical models, specially the change point models, are applicable because of the following advantages: (1) do not require fitting or training; (2) adaptive to different attack data (do not use rules); (3) perform with low data sample sizes; and (4) computationally efficient for real-time applications. Thus, the objective of this study is to investigate the efficacy of two change point models, Expectation Maximization (EM) and Cumulative Sum (CUSUM), for real-time V2I cyberattack detection in a connected vehicle (CV) Environment. To prove the efficacy of these models, we implemented three different type of cyber attacks (i.e., denial of service (DOS), impersonation, and false information) [22], using BSMs generated from CVs through simulation. Expectation Maximization's (EM) utilization for anomaly detection and adaptive CUSUM (aCUSUM) approach, algorithms' computational capability for under 0.1 second (s) intervals, and their comparison under connected vehicle framework are unique to this study. Connected and autonomous vehicles present different challenges, as datasets are not available, the attacks on calibrated microsimulation networks are utilized in our study.

A connected vehicle broadcasts basic safety messages (BSMs) at a frequency of 10 hertz (Hz) to its nearby vehicles. A BSM contains several message elements, such as location, speed, heading direction, and vehicle unique identifier. Among these elements, most important message elements are location, speed, and vehicle identifier, as these elements are related to the safety critical operation of a connected vehicle application. Based on the message elements or feature set we have chosen three types of attacks: (i) denial of service (flooding the network with unnecessary messages) (ii) impersonation (impersonating the vehicle unique identifier), and (iii) false information (broadcasting false speed and location information). Furthermore, getting a BSM within the required maximum allowable latency (0.1 s or 100 milliseconds ms) is also critical for the timely safety operation of the connected vehicles. Each vehicle on the roadway considered in this study was assumed to have a DSRC technology-enabled wireless communication radio. It was also assumed that a DSRC-enabled radio in a connected vehicle has the capability to broadcast BSMs, which can be received by roadside equipment (RSE) if a vehicle is within the DSRC coverage area of an RSE. Due to the limitation of Simulation of Urban Mobility (SUMO) traffic simulator, it is not possible to model RSE in SUMO. Therefore, we assumed that the data generated from each connected vehicle (i.e., BSMs) were received by the RSE within a vehicle's DSRC communication range. Note that we also assumed no communication latency and assumed perfect communication (i.e., no data loss and communication delay) among connected vehicles and the associated RSE.

The paper is organized as follows. Section II presents the previous research and the literature on the anomaly detection models. Section III describes EM and CUSUM algorithms for V2I cyber-attack detection. Section IV presents the data generation process and evaluation of EM and CUSUM models through numerical analysis and results. Finally, section V summarizes findings and possible future research directions.

II. Related Work

In this section, we describe past research on statistical models for anomaly detection and cyber attacks in a vehicle-to-infrastructure (V2I) environment.

A. Statistical Models for Cyber Attack Detection

Statistical and inference based models in cyber attack or in general detection problem provide adaptability and transferability to different settings and attack types with low computational costs [18], [23]. In a very basic approach, detection on process controls using quality control models based on change point algorithms such as CUSUM, and exponentially weighted moving average are utilized [24] intrusion monitoring. For detail characteristics of attack models using honeypot-captured cyber attacks are modeled with several time series models [25]. Reliability models are also studied for vulnerabilities based on good and bad states simply via nodes' deviations [26]. They consider persistent, random, and insidious attacks of sensor-actuator nodes with simple sensing, actuating, and networking models. Moreover, model-based attacks usually for power grids are investigated by researchers [27]. Attack (intrusion) models for different control systems and proper modeling for moving systems as in vehicular or mobile ad hoc network (VANET/MANET) cases are well reviewed in [28]–[31] where reputation management in vehicular networks are suggested. Possible revoking or blacklisting the information contributors are also recognized in similar survey study specifically on cooperative intelligent transportation systems [22]. Privacy of the drivers and safety critical applications are also started to be investigated by the researchers ([32]).

First proposed by Page [33], CUSUM is a classical statistical quality and process control method for industrial applications, which is then utilized by many fields such as computer network security particularly for DOS or flooding attacks [34], sensor networks, signals and control systems, pipeline break detection to neuronal spike detection [35], [36]. However, it is also heavily employed in intrusion or anomaly detection for cyber attacks [23] for its high true positive rate and low computational cost. In connected vehicles, a recent patented implementation utilizes CUSUM on for vehicle intrusion detection on electronic control units [37]. Without accuracy reporting, CUSUM was used for DOS attack detection in [38]. On the other hand, EM is used for anomaly detection as its classical meaning of parameter estimation in an analytical attack modeling on power systems [39]. In this study, both EM and CUSUM are selected as detection algorithms for their online applicability (linear in computational complexity) also observed in [18]. Both algorithms are adopted to the anomaly detection problem as sequential implementation, compared, and detailed attack data are simulated which are novel in the intrusion detection literature. Both methods contain only low level parameters such as initial underlying distributions parameters (e.g., Normal in this paper) as well as design parameters for CUSUM. Detailed recalibration or update intervals for such parameters are not investigated in this study.

In the literature, there are extensive applications of deep learning (DL) and versions of decision tree (DT) algorithms. In one of the recent studies [40], an intrusion detection system

(IDS) was introduced using deep belief networks along with other classification and clustering algorithms reporting more than 99% detection rate with about 1% false positives and 1.5% false negatives. However, using simulated data, after 40 nodes, the proposed algorithms are causing an average delay of more than 0.1 s [40]. In [41], tree-based machine learning methods were used for IDS for detecting various attacks. Versions of decision tree algorithms were found to be more accurate providing more than 99% accuracy where ensemble methods showed 100% accuracy. Although the study showed low computational costs, it is hard to judge if real time implementation would be possible from cumulative run times of some reaching to an hour [41]. Moreover, machine learning algorithms were implemented for malware detection. Authors in [42] developed an improved feature selection algorithm resulting in 93% detection accuracy with less than 0.1 s execution time [42]. IDS on vehicle platoons was used for detecting spoofing and jamming attacks with random forests and k-nearest neighbors (KNN), in which authors reported about 90% detection accuracy [43]. In a real vehicle controller, Tariq et al. [44] modeled IDS for detecting DOS, replay, and other types of attacks through long-short term memory, and they reported 100% accuracy and less than 0.1 s time delay running onboard units of a vehicle. In the physics-based model, Wang et al. used a version of Kalman filtering and DTs to detect anomalies with over 90% accuracy [45]. False location information was aimed to be detected using deep learning. Although accurately detected with deep learning, false positives were also reported [46]. Anomaly detection using machine learning was also used in [47], where versions of random forest and support vector machines (SVM) reported providing over 90% accuracy with up to 4% false negatives [47]. Versions of DTs for stealthy attack detection on smart grids. Authors report over 97% detection accuracy with 1 minute model training time [48]. A good comparison of machine learning methods for attack detection on mobile networks. Authors reported at least 91% detection accuracy with DL where they were able to achieve similar values with logistic regression as well as SVM and decision trees [49]. Using a restricted Boltzmann machine, the study in [50] presented an anomaly detection algorithm with 99% accuracy alone with false negatives from their algorithm. Although clear accuracy metrics were not provided, a reinforcement learning exploration-exploitation algorithm of multi-armed bandit was used to detect an injection attack [51]. In another study, authors utilized KNN and SVM for intrusion detection for vehicle systems where the reported accuracy is above 96% accuracy [52]. The study in [53] listed one of the gaps of recent taxonomy of connected vehicle security as considering both in-vehicle and vehicular network security together. In our study, detection algorithms can be applied from both angles. Vehicles' false information, impersonation, and DOS attack to or from a vehicle can be detected.

In sum, researchers mainly explored logically evolving DL and versions of DTs as detection methods. These methods are able to include features and have more rule-based (rather flexible thresholds) detection for different attack types. However, as machine learning methods, all of these methods require training. In this paper, we used both approaches and used EM online with a low number of iterations (e.g., 10) and samples (e.g., 10) in order to detect abnormal and normal behaviors.

B. V2I Cyber-attacks in a CV Environment

In the cyber-physical systems (CPS) security literature, recent studies [17], [28], [54]–[57], list possible cyber-attacks and discuss their detection and mitigation techniques. In these studies, abstract cyber-physical models for smart cars are also presented. Possible attacks are criminal, privacy, tracking, profiling, political threats with different structures replay, command (message) injection, false information, impersonation, eavesdropping, and denial of service [56]. For this study, we consider denial of service (DOS), impersonation, and false information attack to evaluate efficacy for EM and CUSUM models. DOS attack in the literature defined as disordering, delaying, or periodically dropping packets to decrease network performance. It consists of flooding (similar to jamming-occupying channel by outsiders) and exhausting the network resources such as bandwidth and computational power. In this study, it is dramatically increasing number of messages so that the roadside equipment (RSE or roadside unit (RSU)) or onboard equipment (OBE or onboard unit (OBU)) are not able to process and overall communication delays increase or become not available. Impersonation (node impersonation or identity theft) attack can be defined as a vehicle can pretend as if it has more than one identity unable to distinguish one or more vehicles by aiming to shape the network, manipulating other vehicle behaviors, incorrect position information etc., hard to detect-network/vehicle ID credentials management. False information attack: aims to manipulate other vehicles with selfish/malicious intent can highly impact and high detection likelihood [58]. Previous research on the vehicular communications discuss possible attacks and their mitigation methods [17], [22]. ITS applications require protocols that conflicts with anonymity and privacy requirements and report on quantifying such risks and traffic control under either lost communications based on correct or faulty communication errors. In sum, studies on quick detection of such cases and possible redundant data resources for cost effective control are needed for resiliency on transportation networks.

III. Change Point Models

In this study, we investigate statistical change point models, Expectation Maximization (EM) and Cumulative Summation (CUSUM), to detect cyber attacks in a V2I environment. We describe these models in the following sub-sections.

A. Expectation Maximization Algorithm

The Expectation Maximization (EM) algorithm is often used to estimate the parameters of mixture models or models with latent variables [59], [60]. In this research, EM algorithm is utilized for detecting cyber attacks via changes in the process mean. Given N sample points from a mixture of two Normal distributions as in Eq. (1), the EM algorithm can be applied to determine the parameters of these two distributions $\boldsymbol{\theta}$=[$\theta_1 = (\mu_1, \sigma_1)$, $\theta_2 = (\mu_2, \sigma_2)$, π] of normal and attack states, respectively. The first step of the EM algorithm specifies initial values for the parameters. In the expectation step, the

algorithm computes the responsibilities γ_i (i.e., the probability of an observation belonging to Y_2, i.e., attack state) for each data point. Using the calculated responsibilities, it then computes the five parameters in the maximization step. The iterations continue until the likelihood function convergences. The convergence of a basic EM algorithm is slow. Simple equations pertaining to the EM are given below. First, the probability density of Y is written as a mixture:

$$Y = (1 - \Delta)Y_1 + \Delta Y_2 \tag{1}$$

where $Y_1 \sim \mathcal{N}(\mu_1, \sigma_1^2)$, $Y_2 \sim \mathcal{N}(\mu_2, \sigma_2^2)$, and $\Delta \in 0, 1$ with abnormal data proportion of $\mathrm{P}(\Delta = 1) = \pi$.

$$g_Y(y) = (1 - \pi)\phi_{\theta_1}(y) + \pi\phi_{\theta_2}(y) \tag{2}$$

where $\phi_\theta(x)$ denotes normal density. For a data set of N points the loglikelihood function can be written as follows:

$$l(\theta, Z) = \sum_{i=1}^{N} ln[(1 - \pi)\phi_{\theta_1}(y_i) + \pi\phi_{\theta_2}(y_i)] \tag{3}$$

where $\boldsymbol{\theta}$= $[\theta_1 = (\mu_1, \sigma_1)$, $\theta_2 = (\mu_2, \sigma_2)$, and $\pi]$ and Z represent the data points. Analytical maximization of Eq. (3) is difficult, however, if the observation is known to belong to Y_2 (i.e., with latent variable $\Delta_i = 1$, otherwise $\Delta_i = 0$), the loglikelihood can be written as in Eq. (5) and $\Delta_i = 1$s can be estimated by Eq. (5).

$$l(\theta; \Delta, Z) = \sum_{i=1}^{N} [(1 - \Delta_i) ln[(1 - \pi)\phi_{\theta_1}(y_i)] + \Delta_i ln[\pi\phi_{\theta_2}(y_i)]] \tag{4}$$

$$\gamma_i(\theta) = E(\Delta_i \mid \theta, Z) = P(\Delta_i = 1 \mid \theta, Z) \tag{5}$$

In sum, given N data points that are assumed to be generated by mixture of two Normal distributions (i.e., normal and abnormal messages per vehicle per second (MVS), messages per vehicle (MVT), and distance), the EM algorithm is applied to determine the distribution parameters and responsibilities. Number of mixtures could be varied for various levels of attacks and impacts. N data points constitute the main input to the algorithm. To see the impact of sample size, prediction performances of EM algorithm with various N values can be checked. The EM algorithm provides the real-time estimation of the process parameters at each time point as well as conditional probabilities of a data point comes from a certain attack or no attack condition which is subsequently used for detection.

B. CUSUM Algorithm

The CUSUM chart or algorithm is commonly used for quality control purposes to detect possible shifts in the mean level of a process. In cyber attack setting, changes within expected level of deduced measures (MVS, MVT, and distance) are targeted. This paper uses tabular version or algorithmic version of the CUSUM rather than control chart. Assume that $X_i \sim$ identical independently distributed (i.i.d) with known (μ_1, σ^2) where a new process mean is observed μ_2 after a possible change. Based on statistical hypothesis testing, the log-likelihood ratio is written $s(i)$=$ln(p_{\mu_2}(X_i)/p_{\mu_1}(X_i))$ for $S_t = \sum_{i=1}^{t} s_i$ for sample size of n, the decision rule d is given by

$$d = \begin{cases} 0\ ,\ C_t < H\ ;\ H_0\ no\ change \\ 1,\ C_t \geq H;\ H_1\ change \end{cases} \tag{6}$$

where $C_t = S_t - m_t$ and $m_t = [S_i]^-_{1 \leq i \leq t}$ [61], [62].

1) Typical Form

Basic applications of this algorithm assume that the observations collected before and after the change in the mean level are i.i.d. To detect both positive and negative shifts, the two-sided version of the CUSUM algorithm was used. The algorithm works by accumulating positive and negative deviations from a certain target mean, which is commonly taken to be zero. The positive deviations (values above the target) are indicated with C_t^+, and those that are below the target are indicated with C_t^-. The statistics C_t^+ and C_t^- are referred to as one-sided upper and lower CUSUMs, respectively [63]. It is shown that the use of the two-sided CUSUM algorithm is equivalent to monitoring the following two sums for a zero-mean process:

$$\begin{aligned} C_t^+ &= [0, C_{t-1}^+ + X_t - \mu_2 - K]^+ \\ C_t^- &= [0, -C_{t-1}^- - X_t + \mu_2 - K]^+ \end{aligned} \tag{7}$$

where $C_0^+ = 0$, $C_0^- = 0$, is the residual or deviation from the mean at time t. A shift detection is issued whenever $(C_t^+ \vee C_t^-) > H$. Typical CUSUM is applied for persistent shifts or attacks. With $-C_{t-1}^-$ in Eq. (8), the algorithm behaves like one-sided and reduces false alarm rate almost 100%. Moreover, in order to employ CUSUM in real-time, once an alarm is issued by the CUSUM algorithm, the mean or intercept of the attack time series observations is estimated and updated with Eq. (8) and C_t^+, C_t^- values set to zero after every detection.

$$\mu_2 = \begin{cases} \mu_1 + K + \frac{C_t^+}{N^+}, C_t^+ > H \\ \mu_1 - K - \frac{C_t^-}{N^-}, C_t^- > H \end{cases} \tag{8}$$

The CUSUM algorithm are designed by choosing the values of K and H. The constant K is called the reference value and H is the decision interval or the threshold. The parameter K is a function of the shift in mean level to be detected by the CUSUM algorithm. The value of H is selected to give the largest in-control average run length (ARL) consistent with an adequately small out-of-control ARL. These two parameters control the ARL, a standard performance measure for online change-detection algorithms. ARL is the average number of data points that have been observed before an out-of-control signal or alarm is generated. There have been many analytical studies on investigating CUSUM's ARL performance. For example, the conventional CUSUM with $K = \delta\sigma/2$ is optimal in detecting a shift of $\delta\sigma$ from target mean. Based on past studies, Montgomery [63] suggests that selecting $K = \delta\sigma/2 = \sigma/2$ for $\delta = 1$ and $H = 5\sigma$ provides a CUSUM algorithm that has good ARL properties against small shifts in the process mean [63].

The CUSUM algorithm described previously is applied to the change point detection of the time series within basic safety

messages. The CUSUM parameters were selected as suggested in the literature: $K = \delta\sigma/2$ and $H = 5\sigma$ and $\delta = 1.0$ which represents midpoint between normal and abnormal process means.

2) Adaptive Form

Adaptive version, denoted as aCUSUM, is actually adopted from [62] revised to perform for other than zero mean processes, lower false positives, and single weight parameter (α). Table III shows only initial mean values are different which could be used as simple as 1^{st} value observed in the process. It is applied to $\tilde{X}_t = X_t - \bar{\mu}_{t-1}$.

$$C_t^+ = [0, C_{t-1}^+ + \frac{\alpha D_t}{\sigma^2}[X_t - D_t - \alpha D_t/2]]^+ \quad (9)$$
$$C_t^- = [0, C_{t-1}^- - \frac{\alpha D_t}{\sigma^2}[X_t + D_t + \alpha D_t/2]]^+$$

where $D_t = (\bar{\mu}_t - \mu_1)$ and $\bar{\mu}_t = \alpha\bar{\mu}_{t-1} + (1-\alpha)X_t$. This adaptive form of CUSUM algorithm is not very sensitive to $K = \delta\sigma/2$ and $\delta = 1.0$. As in the typical algorithm, for less false positive detection H is set to 5σ.

IV. Numerical Experiments

This section presents the data generation to evaluate the methods for different vehicle-to-infrastructure (V2I) attacks and gives numerical results for performance of the proposed detection models.

A. Attack Model

We have created three attacks: (i) denial of service (DoS) attack, (ii) false information attack, and (iii) impersonation attack. We assumed an attack can be carried out in three different ways: (1) an attacker can connect to the OBU through the Ethernet locally and then alter the code in OBU, and create and send false messages to generate false location of a vehicle and/or create false vehicle identity, (2) an attacker can remotely compromise an OBU of a CV or an RSE through unauthorized access to generate false location information, and/or false vehicle identity, and/or flood the communication channel with unnecessary data to cause a CV application to be unavailable to other CVs and RSEs, and (3) an attacker can intercept the data flow in a communication channel and alter the data packets with false location information of a vehicle, and/or false vehicle identity through man-in-the-middle (MITM) attacks.

An attacker's capabilities also depend on the configuration of communication radios. In this study, we consider DSRC communication radios. DSRC has seven communication channels using different frequencies ranging from 5.85 GHz to 5.925 GHz. These seven channels are divided into two categories: Control Channel (CCH) and Service Channel (SCH). In this study, we consider Channel number 178 is assigned for CCH, and the remaining channels from 172 to182 are assigned as SCH. After the initial authentication and key exchange, the RSE and vehicle OBE agree to communicate on a single service channel with a fixed frequency.

An attacker CV can launch the DoS attack by flooding the communication channel to cause a service to be unavailable to other CVs. An attacker uses its maximum transmission capacity to flood the network and disrupts the V2I communication by transmitting more data than the receiver's (e.g., RSE) maximum receiving capacity. In the fake (or false) information attack, false GPS location information (i.e., longitude and latitude) of a vehicle is generated using a random variable generation approach. We have crafted the attack to create a random location within a given geo-fenced region so that it seems normal geo-location to humans. This false information is also broadcasted by the attacker vehicle at 10 Hz or 10 BSM packets/s. A false identification (ID) for a vehicle instead of its original ID is used for modeling an impersonation attack. Two different GPS locations and speeds for a vehicle have been used for this purpose in this study.

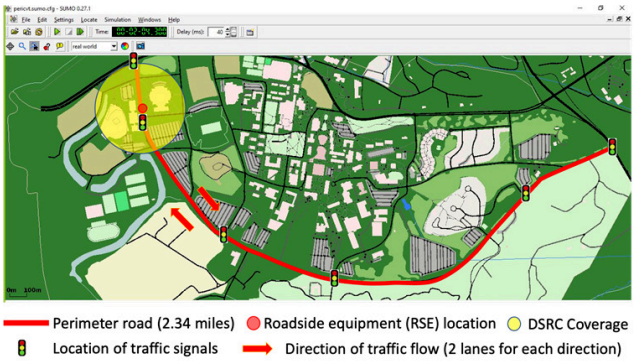


Fig. 1. Route configuration

B. Data Generation for V2I Cyber-Attacks

In this subsection, data generation process for different type of V2I cyber-attacks using microscopic traffic simulator is presented. In order to generate the realistic roadway traffic behavior, a microscopic simulation software, Simulation of Urban Mobility (SUMO) is utilized [64]. To mimic real-world vehicular movement in a connected vehicle environment, a roadside equipment (RSE) is assumed to be placed at the Jervy Gym location of Perimeter Road in Clemson, South Carolina (SC), USA [65]. The length of the roadway network is 2.34 $miles$; the total number of intersections is five; and we have considered unidirectional traffic flow. We have used a single volume input, i.e., 200 vehicles per per hour per lane (see Fig. 1).

In our study, we have used the Intelligent Driver Model (IDM) as the car-following model for connected vehicles, such that all the simulated vehicles mimic the driving behavior of a human driver. Moreover, we assume that all vehicles are wirelessly connected, and each vehicle broadcasts basic safety messages (BSMs), which contain latitude- longitude, timestamps, and speed [66].

In detail, the simulation network was a calibrated roadway network used in one of our previous studies [67]. We used a random number generator (RNG) function in SUMO to generate different seed numbers and added stochasticity to our simulation [68], [69]. We also used a speed attribute (i.e., speedFactor) in SUMO that allows the specification of the parameters of a Normal distribution with optional cutoffs. In this way, a random value was selected from the Normal

distribution for each vehicle at the time of its generation and considers heterogeneous mix of vehicle speeds in the simulation. However, it is necessary to select a performance metric (e.g., travel time and traffic volume) for calibrating the simulated network so that it can represent a real-world scenario. We selected travel time as a performance metric for calibrating our simulation model in SUMO and used the following equation to calculate the optimal number of simulation-runs, N_{TT}, with different seed numbers. It is a trial-and-error approach. For example, after selecting a certain number of seeds, we need to run the simulation and collect average travel time, and it is necessary to calculate different parameter values of Eq. (10) and determine the number of simulation-runs. If the calculated number of simulation runs is higher than the previously selected number of simulation-runs, it is necessary to run the simulation again for the calculated number of simulations, N_{TT}, with different seed numbers. After that, we need to collect the average travel time for each seed and calculate the number of simulations runs again. One needs to follow this procedure until the calculated N_{TT} based on the new simulation run is less than the required number of simulation runs as estimated in the previous step.

$$SE_{TT} = z_{score} \times \frac{\sigma_{TT}}{\sqrt{N_{TT}}} \tag{10}$$

where, SE_{TT} is allowable error,which is a fraction of the travel time, N_{TT} is the number of simulation runs for the travel time performance measure, z_{score}=z is statistic value for a given confidence level of the Normally distributed performance measure-i.e.,travel time, and σ_{TT} is estimated standard deviation of the performance measure -i.e.,travel time.

Table I presents the parameters for identifying the required number of simulation-runs within the 95% confidence interval. Please note that we do not conduct any other sensitivity test as the focus of our paper is evaluating the performance of cyber-attack detection models.

TABLE I. Parameters for calibrating simulation model

Name of the Parameters	Parameter value
Allowable error, SE_{TT}	5% of the average travel time
Z statistic value, z_{score}	1.96 (for 95% confidence interval)
Estimated standard deviation, σ_{TT}	3.66 (for the given seed numbers)
Seed number for the random number generator (RNG) function in SUMO	100, 150, 200, 250, and 300

Each vehicle on this roadway are DSRC communication-enabled and can broadcast a part of BSMs (e.g., time stamp, car ID, latitude, longitude, and speed) every one-tenth of a second to the RSE. RSE is a static node on the side of a road with a defined communication range (i.e., 300 m), whereas vehicles containing the OBE are moving nodes on a roadway and having a defined communication range (i.e., 300 m). Due to the limitations of the Simulation of Urban Mobility (SUMO) traffic simulator, it is not possible to model roadside equipment (RSE) in the simulator. Thus, we only generate vehicles' movement using SUMO and collect the mobility information of the vehicles using a trace file, which contains vehicle's location and speed at each timestep in a JSON format. Then, using a python script, we specify the RSE location, and filter out the location and speed of the vehicles from the trace file using the location of each timestamp within the dedicated short-range communication (DSRC) range (300m) of the RSE. We separated data through this post-processing step, and it means that we have assumed no communication latency between connected vehicles and RSE. The simulation is comprised of 200 vehicles per hour per lane on the Perimeter Road, a four-lane arterial roadway (two lanes each direction) with 56 kilometers per hour (kph) (or 35 miles per hour) speed limit.

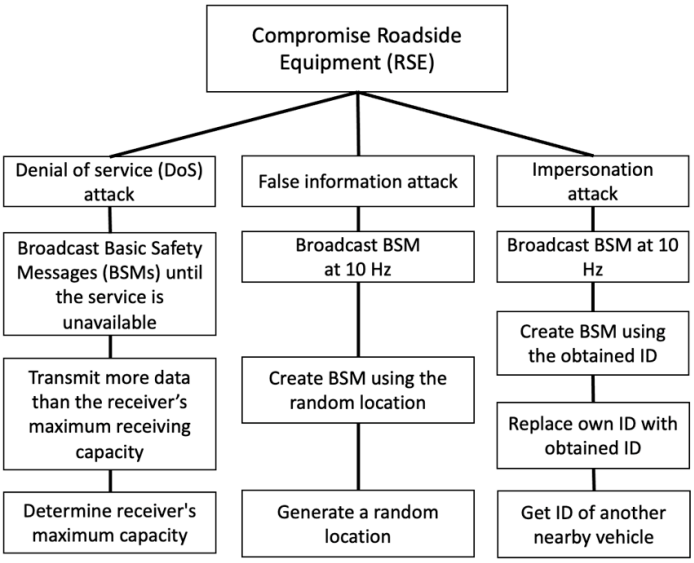
Using the generated trace file from the SUMO simulation, three different cyber-attack scenarios are generated (see Fig. 2a):

(i) After the initial authentication and key exchange, the RSE and OBE of a vehicle agrees to communicate on a single service channel or a fixed frequency. Then, a vehicle can launch the DOS attack by flooding the communication channel in order to cause the service to be unavailable to other vehicles. Typically, an attacker uses its maximum transmission capacity to flood the network. In order to create a breakdown of V2I communication, attackers need to transmit more data than the receiver's (e.g., RSE) maximum receiving capacity. For generating DOS attack data in our experiment, vehicle number 6 (ID6) is flooding at 1000 Hz while other vehicles are sharing data at 10 Hz in a CV environment where each CV is broadcasted BSMs every one-tenth of a second. The total simulation time is 200 seconds (s) for generating the attack data.

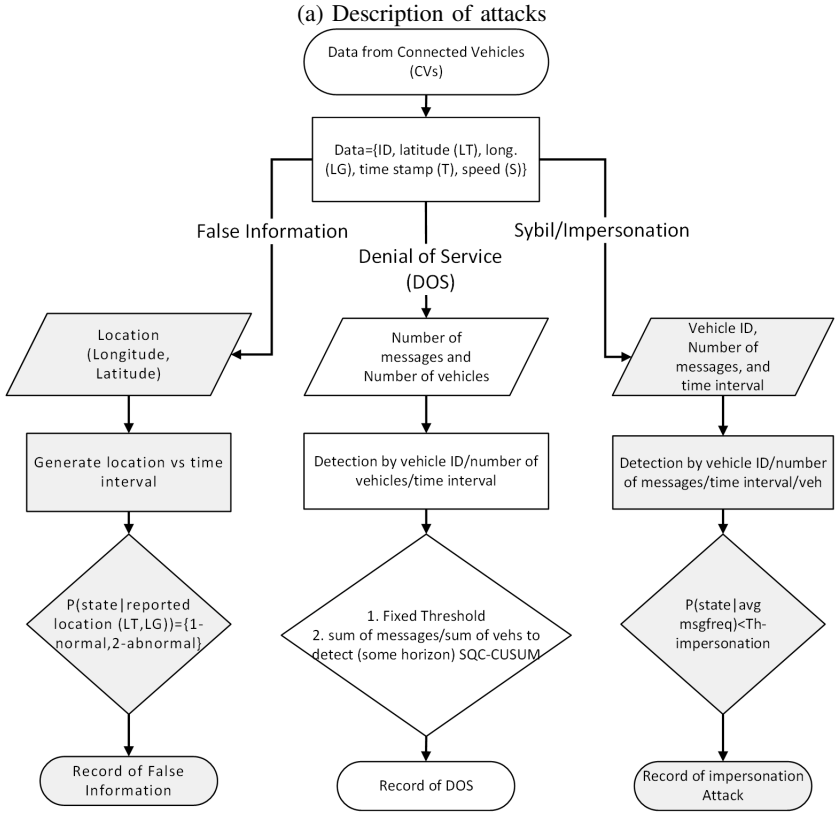
(ii) False information attack: For fake (or false) information attack, false GPS location information (i.e., longitude and latitude) of vehicle number 2 (ID2) are generated using random variable generation library from python. We have crafted the attack in such a way that it generates random location within a given geo-fenced region so that it seems normal geo-location to humans. This false information is also broadcasted by the attacker vehicle at 10 Hz or 10 packets/s. The total simulation time is 200 s for false information attack.

(iii) Impersonation attack: To emulate the data for impersonation attack, a false identification (ID) for vehicle number 3 is used as vehicle number 2 (ID2). Two different GPS location and speed information for the vehicle ID 2 are simultaneously generated. In the trace file, the vehicle ID of vehicle 3 was replaced by the vehicle ID 2 to craft an impersonation attack, where we assume that both of the vehicle 2 and vehicle 3 are in the same region. Thus, two different GPS locations and speeds are being broadcasted containing the same vehicle ID simultaneously. Both of the vehicles are broadcasting the data at 10 packets/s, and simulation was run for 200 s.

Examples of generated attack data are given in Table II. Evident from the table, multilevel attack monitoring could be designed by vehicle ID and timestamps as micro level tracing (0.1 s) of such values. However, this approach considerably slows detection capability within time interval of 0.1 s which is critical for safety applications. Therefore, this study tracks



(a) Description of attacks



(b) Attack detection approach in V2I connected vehicle environment

Fig. 2. Data generation steps and attack detection approach

aggregate measures such as average message frequency per vehicle per second (MVS), average message frequency per vehicle per time interval (MVT), distances, and/or track of vehicle speeds within time series framework and detects changes. Detailed vehicle information are not tagged, however, signature is present in the historical data can be traced back for mitigation efforts.

C. Attack Detection Framework

Fig. 2b depicts the approach of attack detection using EM and CUSUM. In order to implement change point detection methods, first step is to identify the processing time window in which information need to track, and how to convert such information in time series behavior to detect shifts due to malicious attacks and/or benign system malfunctions. Such changes result in switching system dynamics and alter critical communications in ITS applications, such

TABLE II. Examples of attack data generated on RSE

Type	TS(s)	ID	Lat.	Long.	Speed(m/s)	Pos.(m)	MsgRate
DOS	5.10	1	-82.85	34.68	9.94	0.08	10.00
	5.10	2	-82.85	34.68	8.22	0.52	10.00
	5.10	3	-82.85	34.68	6.21	0.74	10.00
	5.10	5	-82.84	34.68	2.32	0.14	10.00
	5.10	6	-82.84	34.68	0.00	0.00	10.12
	5.10	6	-82.84	34.68	0.00	0.00	10.23
	5.10	6	-82.84	34.68	0.00	0.00	10.35
IMP	1.30	1	-82.85	34.68	2.65	0.00	1.00
	1.30	2	-82.85	34.68	0.51	0.00	1.00
	1.40	1	-82.85	34.68	2.87	0.00	1.00
	1.40	2	-82.85	34.68	0.75	0.00	1.00
	1.40	2	-82.85	34.68	1.00	0.00	2.00
	1.50	1	-82.85	34.68	3.19	0.00	1.00
	1.50	2	-82.85	34.68	1.24	0.00	1.00
FAL	2.00	1	-82.85	34.68	4.15	0.00	1.00
	2.00	2	-82.85	34.68	2.26	0.00	1.00
	2.00	3	-82.04	34.16	0.00	72.32	1.00
	2.10	1	-82.85	34.68	4.31	0.00	1.00
	2.10	2	-82.85	34.68	2.48	0.00	1.00
	2.10	3	-82.81	34.30	0.26	71.20	1.00
	2.20	1	-82.85	34.68	4.57	0.00	1.00
	2.20	2	-82.85	34.68	2.62	0.00	1.00

as cooperative adaptive cruise control (CACC) and signal control algorithms. In DOS or flooding attacks, vehicles are expected to send more messages than the designed frequency $parameter$ (MVS). Therefore, tracking messages per vehicle and estimating MVS can be used as indicator for cyber-attack detection. For impersonation attack, multiple messages in unit time interval (0.1 s) are sent and by monitoring MVT, this type of attack is detected. Lastly, false information attack can be defined as any type of irregularity in the collected messages, such as high or low speed compared to rest of the traffic (inherent) at a roadway segment or an unrealistic gap between any two adjacent vehicles within a certain time frame. CUSUM algorithms monitor deviation from process mean and identify violations. On the other hand, EM calculates conditional probabilities of $P(DOSattack|MVS) > 0.001$, where $P(impersonation|MVT)$ and $P(attackstate|distance)$ is given. If the likelihoods at any time is > 0.001, then an attack is detected.

D. Description of EM and CUSUM Parameters

Parameters for EM and CUSUMs are set as provided in Table III. Initialization parameters of EM algorithm are θ_1, θ_2, π, $N = 10$ random variates 7 normal 3 abnormal, and 10 iterations per time interval or new observation received. For CUSUMs, design parameters as well as initial mean and standard deviations are given in the Table III below. Overall aim here is to give models normal and/or abnormal observations. For instance, in case of DOS attack, 10 messages per second per vehicle is expected with low or no variations, thus, initial parameters are set to $\mathcal{N}(\mu_1 = 10, \sigma^2 = 10^{-6})$ for both methods. Moreover, from Table IV, very small *normal* distance values are calculated from latitude and longitude values (i.e., $\mu_1 = 0.05$) and false information is calculated to be considerably high so initialized from $\mathcal{N}(\mu_2 = 50, \sigma_2^2 = 25)$.

E. Analysis and Results

In this section, the effectiveness of attack detection using EM and CUSUM are discussed. Both methods are evaluated using datasets as described in 'Data Generation for V2I Cyber-Attacks' subsection. Table IV provides an example of the generated data from the simulation, attack and detection results.

TABLE III. Selected model parameters for numerical experiments

Type	EM	CUSUM	aCUSUM
DOS	$\theta_1 = (10, 10^{-4}), \theta_2 = (15, 5), \pi = 0.75)$	$\mu_1 = 10.00$	$\mu_1 = 10.00$
	$Y_{1:7} \sim \mathcal{N}(10, 10^{-6})$,$Y_{8:10} \sim \mathcal{N}(15, 10^2)$	$\sigma = 0.0001$	$\sigma = \sqrt{5.10^{-3}\mu_1}$
IMP	$\theta_1 = (1, 10^{-3}), \theta_2 = (2, 0.5), \pi = 0.99)$	$\mu_1 = 1.00$	$\mu_1 = 1.00$
	$Y_{1:7} \sim \mathcal{N}(0.05, 10^{-2})$,$Y_{8:10} \sim \mathcal{N}(15, 10^2)$	$\sigma = 0.01$	$\sigma = \sqrt{5.10^{-3}\mu_1}$
FAL	$\theta_1 = (0.05, 10^{-2}), \theta_2 = (50, 5), \pi = 0.99)$	$\mu_1 = 0.05$	$\mu_1 = 1.00$
	$Y_{1:7} \sim \mathcal{N}(1, 10^{-6})$,$Y_{8:10} \sim \mathcal{N}(2, 0.5^2)$	$\sigma = 0.1$	$\sigma = \sqrt{5.10^{-3}\mu_1}$
All	$N = 10$ and $iteration = 10$	$H = 5\sigma$	$H = 5\sigma, K = \delta\sigma/2$
		$K = \delta\sigma/2$	$\delta = 1.00$
		$\delta = 1.00$	α=0.025

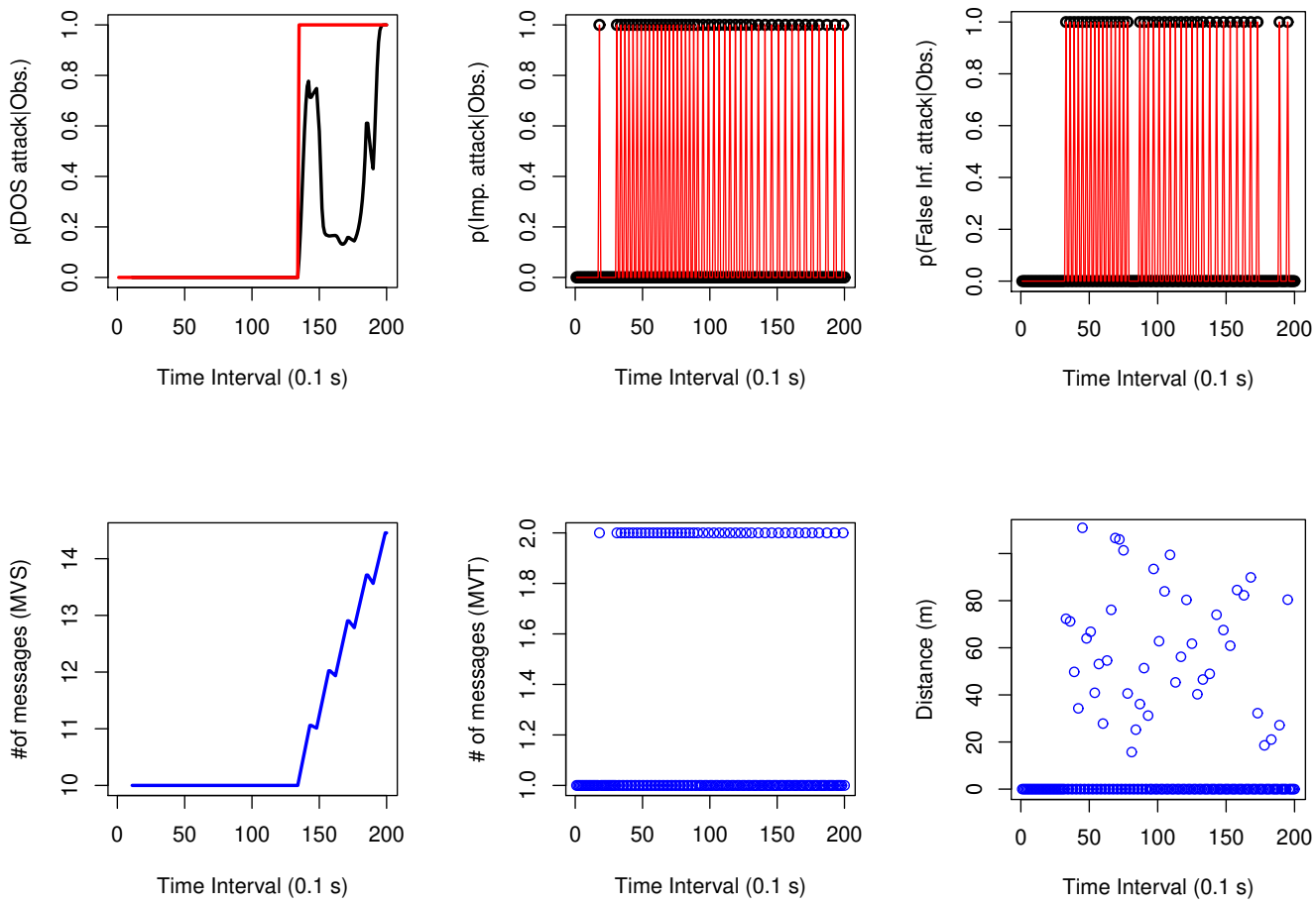
Performances are given as true positive (attack, detected), true negative (no attack, not detected), false positive (no attack, detected) and false negative (attack, not detected) are denoted by TP, TN, FP, and FN, respectively. For running the algorithms, we used a PC with 8GB of memory, Pentium I5 Quad-Core CPU. We observed from the table that abnormal behavior is detected accurately by both methods. Since cyber attacks are persistent and CUSUM is based on cumulative differences, shifts are reflected after the detection with rest of observations. Therefore, detection is continuous. This is also evident from Fig. 4a-4f. For EM, as a classification algorithm, the detection is based on conditional state probability calculations given the observation and past updated parameters and EM is able to flag normal and abnormal observations (also see Figs.3a-3c). Detection alarms are set $P(attackstate|observation) > 0.001$ and $H = 5\sigma$ for EM and CUSUM respectively.

TABLE IV. Examples of attack data on RSE and detection by EM and CUSUM

Type	Freq.	TS(s)	ID	Spd(m/s)	Pos.(m)	Msgs.	$P(D\|Y_t)$	EM	(C^+, C^-)	CUS
DOS	127	5.00	1	9.73	0.08	10.00	0.00	TN	(0,0)	TN
	128	5.00	2	8.08	0.52	10.00	0.00	TN	(0,0)	TN
	129	5.00	3	5.97	0.74	10.00	0.00	TN	(0,0)	TN
	130	5.00	5	2.14	0.40	10.00	0.00	TN	(0,0)	TN
	131	5.10	1	9.94	0.08	10.00	0.00	TN	(0,0)	TN
	132	5.10	2	8.22	0.52	10.00	0.00	TN	(0,0)	TN
	133	5.10	3	6.21	0.74	10.00	0.00	TN	(0,0)	TN
	134	5.10	5	2.32	0.14	10.00	0.00	TN	(0,0)	TN
	135	5.10	6	0.00	0.00	10.12	0.02	TP	(0.12,0)	TP
	136	5.10	6	0.00	0.00	10.23	0.05	TP	(0.12,0)	TP
	137	5.10	6	0.00	0.00	10.35	0.09	TP	(0.29,0)	TP
IMP	13	1.20	1	2.44	0.00	1.00	0.00	TN	(0,0)	TN
	14	1.20	2	0.26	0.00	1.00	0.00	TN	(0,0)	TN
	15	1.30	1	2.65	0.00	1.00	0.00	TN	(0,0)	TN
	16	1.30	2	0.51	0.00	1.00	0.00	TN	(0,0)	TN
	17	1.40	1	2.87	0.00	1.00	0.00	TN	(0,0)	TN
	18	1.40	2	0.75	0.00	1.00	0.00	TN	(0,0)	TN
	19	1.40	2	1.00	0.00	2.00	0.67	TP	0.99,0	TP
	20	1.50	1	3.19	0.00	1.00	0.00	TN	0,0.99	FP
	21	1.50	2	1.24	0.00	1.00	0.00	TN	0.99,0	FP
FAL	31	2.00	1	4.15	0.00	1.00	0.00	TN	(0,0)	TN
	32	2.00	2	2.26	0.00	1.00	0.00	TN	(0,0)	TN
	33	2.00	3	0.00	72.32	1.00	1.00	TP	(72.2,0)	TP
	34	2.10	1	4.31	0.00	1.00	0.00	TN	(0,72.3)	FP
	35	2.10	2	2.48	0.00	1.00	0.00	TN	(72.2,0)	FP
	36	2.10	3	0.26	71.20	1.00	1.00	TP	(34.9,0)	TP
	37	2.20	1	4.57	0.00	1.00	0.00	TN	(0,11.7)	FP
	38	2.20	2	2.62	0.00	1.00	0.00	TN	(5.8,0)	FP
	39	2.20	3	0.52	49.76	1.00	1.00	TP	(48.2,0)	TP
	40	2.30	1	4.81	0.00	1.00	0.00	TN	(0,9.7)	FP
	41	2.30	2	2.83	0.00	1.00	0.00	TN	(3.2,0)	FP
	42	2.30	3	0.75	34.29	1.00	1.00	TP	(33.6,0)	TP
	43	2.40	1	4.95	0.00	1.00	0.00	TN	(0,4.9)	FP

true positive (TP), true negative (TN), false positive (FP), false negative (FN)

In Table IV, position column is calculated in meters (m) from two consecutive latitude and longitude values by using the generic formula: $Pos = 1242sin^{-1}(\sqrt{a})$ where $a = 0.5 - cos((x_2 - x_1)p)/2 + cos(px_1)cos(px_2)(1 - cos((y_2 - y_1)p))/2$ and $p = \pi/180$. As discussed above, MST and MSV measures are deduced from time and ID columns for every time interval of 0.1 s and time series are generated for statistical detection. It should also be noted that for the DOS attack vehicle number 6 is not sending speed and location correctly. Attack detection using the change of speed and distance would be trivial. At-



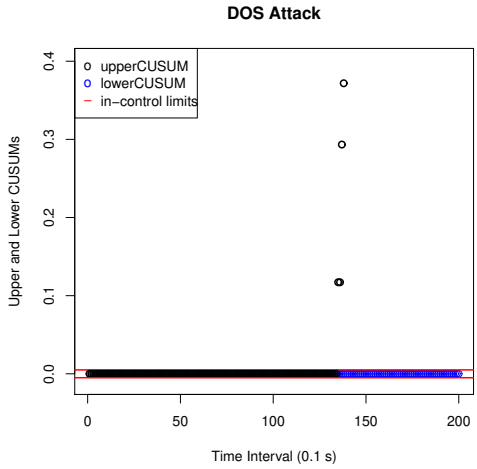
(a) DOS attack (b) Impersonation attack (c) False information attack

Fig. 3. Attack detection by EM algorithm

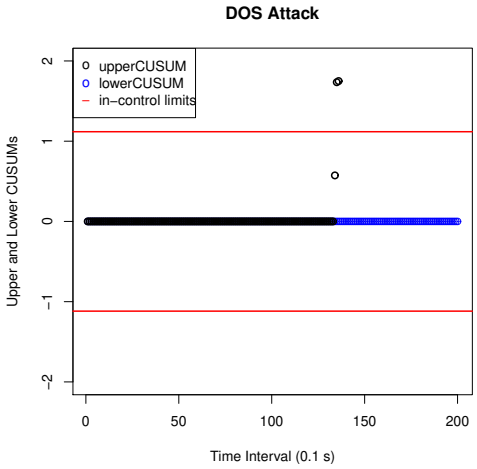
tacker would also replicate reasonable values. So, detection is carried out using message frequency in MSV. From the table, EM's $P(attack|observation)$ is denoted as $P(D|Y_t) > 0.001$ resulting as detection, otherwise no detection. Similarly, for CUSUM (C^+, C^-) values are given. Based on these values, when $(C^+ \vee C^-) > 5\sigma$ a detection is observed, otherwise ND is issued. Persistent attacks are easily detected by CUSUM and EM. CUSUM continues to detect normal observations as attacks as an out-of-control process and generates false positive errors. This can be fixed in CUSUM with a slight revision in C^- values mimicking one-sided control. However, in this study, the performance of a typical CUSUM has been investigated without any modifications. EM's performance on false positive errors is promising. Detailed detection performance metrics are presented in Fig. 5.

Figs. 3a-3c depict performance of EM algorithm for detecting different attacks. From the figures, we observe that in DOS attacks we are monitoring the average message received frequency per vehicle. As soon as we see an increase in the average message frequency EM is able to detect via an increase in the likelihood of this observation coming from the abnormal distribution. A threshold of 0.001 is enough to monitor DOS attack with EM. For impersonation and false information attacks, we are monitoring vehicles' information. In impersonation attack detection, we are looking at the vehicle ID per message, we can see that as soon as we see 2 vehicle ID from impersonating vehicle, EM likelihood completely switches as the difference between 1 and 2 is very different considering the long series of observations. Similarly, false location information sent is different resulting 0-1 switches as soon as we see an attacker vehicle's location information.

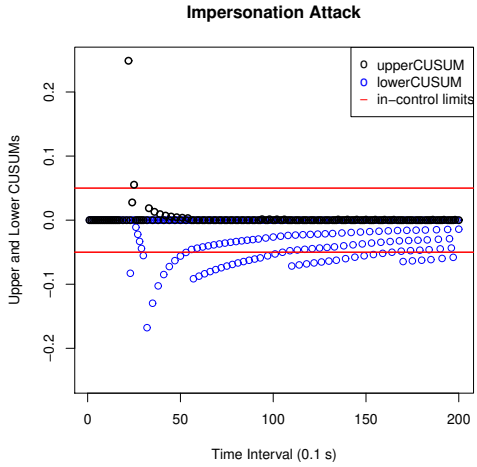
Technically, Fig. 3a shows the likelihood of an attack given 135^{th} observation that is also given in *msg* column in Table IV as $10.12 > 10.00$, i.e., $P(attack|Y_{135} = 10.12) = 0.02 > 0.0001$ (see *EM* column) a very low practical threshold. $P(attack|Y_t)$ increases as frequency values gets larger. For other type of attacks, the changes in observations are not gradual rather sudden which leads to $P(attack|Y_{19} = 2) = 0.67$ and $P(attack|Y_{33} = 72.32) = 1.00$ in impersonation and false information attacks, respectively. However, this statistical inference via EM comes with a computational cost. Especially for DOS attack where change is gradual and more messages



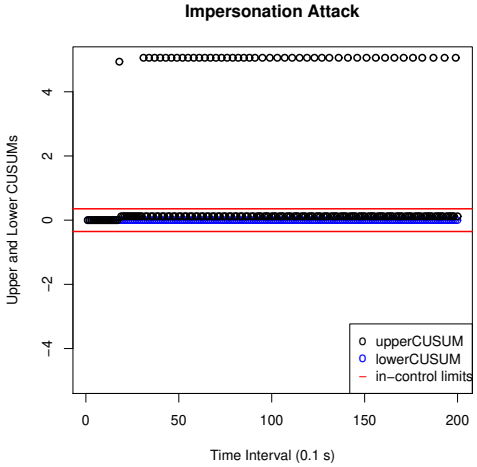
(a) DOS CUSUM



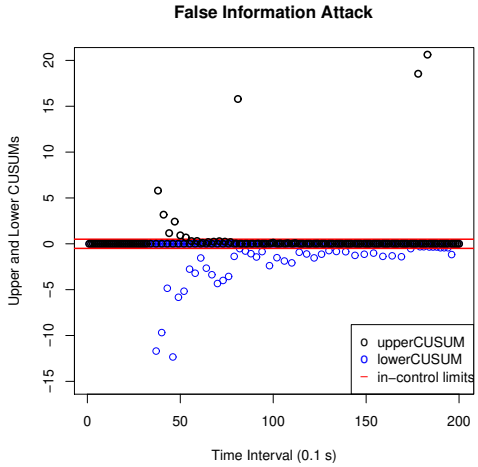
(b) DOS aCUSUM



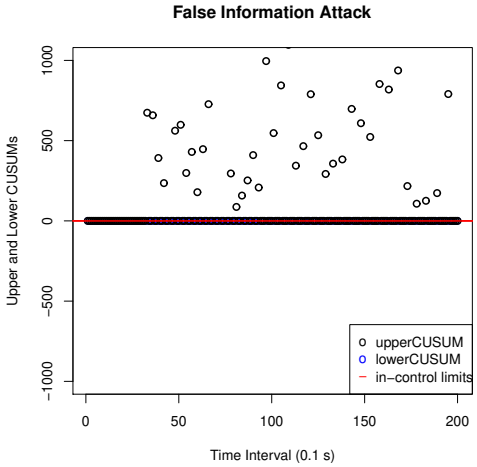
(c) Impersonation CUSUM



(d) Impersonation aCUSUM



(e) False Information CUSUM



(f) False Information aCUSUM

Fig. 4. Attack detection by CUSUM algorithms

sent per vehicle, therefore, more data points to be processed per time step ends up with higher computational time. In Table V, *Attack* column for EM and CUSUM shows computational times of 50, 200, and 4761 to process all the data points.

Thus, the feasibility of using EM for DOS attack detection using messages per vehicle per time 1 s would be second-by-second monitoring. Similarly, false information attack would also require about a second (0.53 s). Only, impersonation attack seems feasible to detect within 0.1 s. These results are consistent with the approximate computational complexity of EM being $O(nkj)$ where n is sample size or time step and $k = 2$ is the number of mixtures, and j=10 denotes the number of iterations. Similarly, it is linear for CUSUM $O(nm)$ with m being number of elementary operations within each n time interval.

Furthermore, we observe from the Table V that EM is impacted more in terms of computation time to process increased observations 4761 instead of 50. For CUSUM, the computational time reaches near 1 second for denial of service (DOS) and about 0.1 second for false information attack (FAL). So, we conclude that we may not be able to process data from approximately 4000 vehicles within 0.1 second. Note that the computations were run on a computer having Intel i5 processor with 8GB memory. Certainly, higher computational power can easily facilitate the use of CUSUMs in real-time for the three attack types considered in this study. Regardless, further study is necessary to investigate how the change point models will capture those effects and how it will satisfy the requirements of real-time CV applications.

TABLE V. Computational times in seconds experienced for EM and CUSUMs

Attack Type	EM			CUSUM			aCUSUM		
\n	50	200	4761	50	200	4761	50	200	4761
DOS	2.19	2.24	44.58	0.43	0.46	0.65	0.77	0.77	1.02
IMP	0.02	0.02	0.02	0.01	0.01	0.02	0.01	0	0.01
FAL	0.53	6.15	44.91	0.03	0.06	0.19	0.14	0.14	0.28

Fig. 4 presents detection results of the CUSUM and aCUSUM algorithms for first 200 data points with 0.1 s intervals. Shorter intervals are shown in order to provide legibility. In Figs. 4a-4b, a shift occurs at 135^{th} observation for DOS attack. CUSUMs advantage over EM is that it can be implemented for short time intervals due to less computational times. The duration for detecting DOS attack using EM is higher than 0.1 s interval. However, impersonation and false information attacks can be detected within 0.1 s (see Figs. 4c-4f). Given sufficient time window, EM algorithm would be able to adapt to detect different attack types with new set of normal data set is fed. It has less parameters to be tuned compared to CUSUMs and prone less to false positive alarms. In their simple forms, they are vulnerable to high false positive when adaptive thresholds are used. CUSUMs are very sensitive to real-time estimation or update of μ_1, μ_2, σ values. In another appropriate midterm application, an hybrid method can be developed to estimate these parameters with EM and input to CUSUMs. Because of space limitations, these experiments are left for another study.

In Fig. 5, we compared detection performances of the models. Metrics adopted from [18] are given as true positive (TP), true negative (TN), false positive (FP) and false negative (FN) are inserted in $accuracy$=$(TP + TN)/(TP + TN + FP + FN)$, $precision$=$TP/(TP + FP)$, $sensitivity$ or $detection$=$TP/(TP+FN)$ [18]. EM only contains about 2% FN for false information and 1% FN for DOS attacks where CUSUM gives 11.8% FP for false information and 2.2% for impersonation attack. For false information attack, EM gives only 83% sensitivity measure and CUSUM is low 87% in precision. After carefully tuning, aCUSUM outperforms both EM and typical CUSUM with no FP and FN for all attack types.

In sum, the initialization of algorithms was done after detailed experimentation and the three most common attack types were selected for demonstrating the capability of the EM and CUSUM models. DOS attack and impersonation cases are relatively easier as frequency levels are known. But, for a false information attack, it is needed to identify the normal behavior. For basic CUSUM, high FPs are observed and adaptive algorithm showed improved results. For EM, performance is highly correlated to the definition of normal distribution or behavior. Note that, critical point in both of the algorithms is the mean level of normal behavior. For any attack type after identifying what measure to monitor and include the normal mean level, algorithms can handle additional attack types. CUSUMs were not trained actually; however, we have estimated CUSUM parameters using the given dataset. EM was also run online with a given normal distribution. If these are carefully input, both algorithms are adaptive and can be used to monitor continuously.

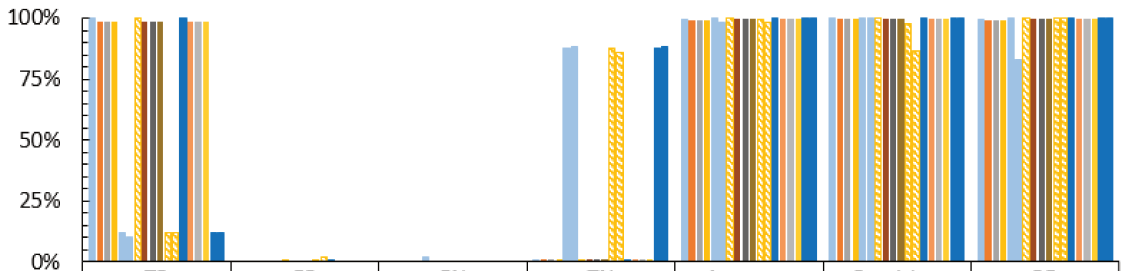
F. Impact of Noise in the Data

In order to evaluate the efficacy of the detection models on attack data with noise, we considered the false information attack. According to the white paper on dedicated short range communication [70], the GPS error with 95% confidence is 0.90731 m in an open sky environment. We generated a noisy false attack data transforming values Y_t assuming 0.0 m mean normal errors ϵ_t with a standard deviation of 0.463 m (obtained from 95% confidence interval since 1σ=0.9073/1.959) as $|Y_t + \epsilon_t|$. Specifically, we introduced this error or noise ($\mathcal{N}(0.0, 0.463^2)$) to the gap (distance) between each subject vehicle and its immediate front vehicle.

Initial parameters of the detection models are used identical as before (shown in Table III). Table VI is given to demonstrate the impact on detection performances. EM algorithm estimates the mean and variance level of the underlying process real-time, thus, expectedly it is not impacted by noise added in distance (gap) values. For CUSUMs, false positives increase with the noise in observations. Adaptive CUSUM's false positives only slightly increase (54) with noise. Note that, we introduce a significant level of noise in the data, which are approximately half of the initial mean for CUSUMs (i.e., $0.463/1.00 \approx 50\%$ coefficient of variation (CoV)). If the noises in the data are low, CUSUMs also would not be impacted. As an example, if we introduce σ=0.0463 (5% CoV), detection of aCUSUM does not produce any false positives and CUSUM produces 49 false positives, which is the same without the noisy dataset.

V. Conclusions

In this study, we investigated the efficacy of two main statistical change point models, EM and CUSUM, for real-time V2I cyber attack detection in a CV Environment. To



	TP	FP	FN	TN	Accuracy	Precision	PD
EM DOS	99.92%	0.00%	0.00%	0.08%	99%	100%	99%
EM DOS2	98.76%	0.00%	0.00%	1.24%	99%	100%	99%
EM DOS3	98.86%	0.00%	0.00%	1.14%	99%	100%	99%
EM DOS4	98.8%	0.0%	0.0%	1.2%	99%	100%	99%
EM IMP	12.1%	0.0%	0.0%	87.9%	100%	100%	100%
EM FAL	10.0%	0.0%	2.0%	88.0%	98%	100%	83%
CUSUM DOS	99.9%	0.0%	0.0%	0.1%	100%	100%	100%
CUSUM DOS2	98.8%	0.0%	0.0%	1.2%	100%	100%	100%
CUSUM DOS3	98.9%	0.0%	0.0%	1.1%	100%	100%	100%
CUSUM DOS4	98.8%	0.0%	0.0%	1.2%	100%	100%	100%
CUSUM IMP	12.1%	0.3%	0.0%	87.6%	100%	98%	100%
CUSUM FAL	11.9%	1.8%	0.0%	86.2%	98%	87%	100%
aCUSUM DOS	99.9%	0.0%	0.0%	0.1%	100%	100%	100%
aCUSUM DOS2	98.8%	0.0%	0.0%	1.2%	100%	100%	100%
aCUSUM DOS3	98.9%	0.0%	0.0%	1.1%	100%	100%	100%
aCUSUM DOS4	98.8%	0.0%	0.0%	1.2%	100%	100%	100%
aCUSUM IMP	12.1%	0.0%	0.0%	87.9%	100%	100%	100%
aCUSUM FAL	11.9%	0.0%	0.0%	88.1%	100%	100%	100%

Fig. 5. Comparison of detection performances between EM and CUSUM algorithms

TABLE VI. Impact of noise in location data on EM, CUSUM, and aCUSUM attack detection models

	EM	CUSUM	aCUSUM	EM	CUSUM	aCUSUM
		without noise			with noise $\sim \mathcal{N}(0.0, 0.463^2)$	
TP	602	602	603	602	602	603
FP	1	49	0	1	741	54
FN	0	0	0	0	0	0
TN	6888	6840	6888	6888	6148	6834
Accuracy	100%	99.3%	100%	100%	90.1%	99.3%
Precision	99.8%	92.5%	100%	99.8%	44.8%	91.8%
Detection	100%	100%	100%	100%	100%	100%

prove the efficacy of these models, we evaluated these two models for three different type of cyber attacks, denial of service (DOS), impersonation, and false information, using BSMs generated from CVs. A comprehensive attack modeling is developed for all type of cyber attacks. To generate the data for different cyber attacks, a microscopic traffic simulation software, SUMO, was used for simulating realistic traffic behavior. Instead of tracking data values such as message frequency, speed, and distance individually for each time interval and vehicle ID, aggregate measures are deduced from BSMs to be used in effective real-time detection. Based on the numerical analysis, we found that:

1) Given proper initialization, i.e., mean and variance measures of normal and abnormal behavior, and enough computational power, both algorithms can detect all three attack types accurately.
2) When the attack detection time window is critical, such as safety applications, detection time window for EM is greater than 0.1 s, whereas the detection time window for CUSUM is below 0.1 s computational times.
3) When multiple states could be observed for an attack or to classify different impacts, as well as any changes in the normal RSU communication frequencies, EM algorithm would be able to provide conditional probabilities for multiple states.

Results from numerical analysis also revealed that EM, CUSUM, and aCUSUM could detect these cyber attacks with an accuracy of at least 98%, 98%, and 100%, respectively. Models can be applied for real-time cyber attack detection with a one-second interval.

When the number of vehicles increases, the computation time of the Expectation Maximization (EM) based detection algorithm would be higher like many statistical/machine learning algorithms because of the increased number of observations (i.e., data). For example, processing data from 1000 vehicles is much higher than 200 vehicles, and it would be more than 100 ms, which is the time requirement for CV safety applications. The processing time could be less if we

could increase computational power and running algorithms in parallel, which is beyond the scope of this paper. However, CUSUM is a sequential testing algorithm, which would be much faster, and it is not required iterations for convergence.

One of the limitations of this study is that the model parameters are limited to only three types of cyberattacks presented in this paper (i.e., denial of service, false information and impersonation attacks). To scale our models for detecting other attacks, it is necessary to determine appropriate sets of model parameters in future studies. Moreover, the effectiveness of the models on the adversarial attacks was also not evaluated. Each vehicle on this roadway was assumed to be DSRC communication enabled and assumed to broadcast basic safety messages. Due to limitation of Simulation of Urban Mobility (SUMO) traffic simulator, it is not possible to model roadside equipment (RSE) or communication in the simulator. Thus, these were only assumed. We also assumed no communication latency and assumed perfect communication (i.e., no data loss and communication delay) among connected vehicles and RSE. Furthermore, we considered a road network with low traffic volume in our experiments. Thus, for a higher volume road network, further study is needed to evaluate whether the change-point models could handle BSMs from a much higher number of CVs in a congested road network. In sum, possible improvements to this research and future directions can be followings: (1) further research is needed to investigate factors affecting the optimal selection of such parameters with multiple data sets; (2) hybrid methods can be formulated for detection both fast and less sensitive to initialization, (3) Detecting benign abnormalities and sensor failure with additional filters, and (4) as data generation processes expected to be correlated, algorithms within state-space time series models can be utilized, and the effectiveness of the proposed methods against the adversarial attack can be evaluated.

ACKNOWLEDGMENTS

Authors would like to thank the Editors and Reviewers for their insightful comments which significantly improved the manuscript. This study is supported by the Center for Connected Multimodal Mobility (C^2M^2) (USDOT Tier 1 University Transportation Center) headquartered at Clemson University, Clemson, SC. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of C^2M^2 and the official policy or position of the USDOT/OST-R, or any State or other entity, and the U.S. Government assumes no liability for the contents or use thereof. It is also partially supported by U.S. Department of Homeland Security SRT Follow-On grant and NSF Grant Nos. 1719501 and 1954532.

REFERENCES

[1] U. DOT, "Beyond traffic 2045: Trends and choices," *US: DOT*, 2015.
[2] N. H. T. S. Administration *et al.*, "National motor vehicle crash causation survey: Report to congress," *National Highway Traffic Safety Administration Technical Report DOT HS*, vol. 811, p. 059, 2008.
[3] U. I. J. Office, "What are connected vehicles and why do we need them," $http://www.its.dot.gov/cv_basics/cv_basics_what$, 2016, accessed: 2016-11-16.
[4] F. A. V. Policy, "Accelerating the next revolution in roadway safety, nhtsa, us dept," *Transportation*, 2016.
[5] L. Kaiser, "Transportation Industrial Control System (ICS) Cybersecurity Standards Strategy 2013-2023," National Highway Traffic Safety Administration, Technical Report, 2013.
[6] NHTSA, " Cybersecurity best practices for modern vehicles," National Highway Traffic Safety Administration, USDOT, Technical Report Report No. DOT HS 812 333, 2016.
[7] M. Burt, M. Cuddy, M. Razo *et al.*, "Big data's implications for transportation operations: an exploration." U.S. Department of Transportation, Tech. Rep., 2014.
[8] CVRIA, "Connected vehicle reference implementation architecture," $http://local.iteris.com/cvria$, 2015, accessed: 2017-06-26.
[9] M. Whaiduzzaman, M. Sookhak, A. Gani, and R. Buyya, "A survey on vehicular cloud computing," *Journal of Network and Computer Applications*, vol. 40, pp. 325–344, 2014.
[10] M. Raya and J.-P. Hubaux, "Securing vehicular ad hoc networks," *Journal of computer security*, vol. 15, no. 1, pp. 39–68, 2007.
[11] U. S. G. A. Office, "Vehicle cyber security: Dot and industry have efforts under way, but dot needs to define its role in responding to a real-world attack," $http://www.gao.gov/assets/680/676064.pdf$, 2018, accessed: 2018-07-31.
[12] D. A. Hahn, A. Munir, and V. Behzadan, "Security and privacy issues in intelligent transportation systems: Classification and challenges," *IEEE Intell. Transp. Syst*, 2019.
[13] S. Ghane, A. Jolfaei, L. Kulik, K. Ramamohanarao, and D. Puthal, "Preserving privacy in the internet of connected vehicles," *IEEE Transactions on Intelligent Transportation Systems*, 2020.
[14] D. M. Nicol, "Modeling and simulation in security evaluation," *IEEE security & privacy*, vol. 3, no. 5, pp. 71–74, 2005.
[15] A. Pathre, C. Agrawal, and A. Jain, "A novel defense scheme against ddos attack in vanet," in *Wireless and Optical Communications Networks (WOCN), 2013 Tenth International Conference on*. IEEE, 2013, pp. 1–5.
[16] M. N. Mejri, J. Ben-Othman, and M. Hamdi, "Survey on vanet security challenges and possible cryptographic solutions," *Vehicular Communications*, vol. 1, no. 2, pp. 53–66, 2014.
[17] J. Petit and S. E. Shladover, "Potential cyberattacks on automated vehicles," *IEEE Transactions on Intelligent Transportation Systems*, vol. 16, no. 2, pp. 546–556, 2015.
[18] A. L. Buczak and E. Guven, "A survey of data mining and machine learning methods for cyber security intrusion detection," *IEEE Communications Surveys & Tutorials*, vol. 18, no. 2, pp. 1153–1176, 2016.
[19] A. Patcha and J.-M. Park, "An overview of anomaly detection techniques: Existing solutions and latest technological trends," *Computer networks*, vol. 51, no. 12, pp. 3448–3470, 2007.
[20] H. Sedjelmaci, S. M. Senouci, and M. A. Abu-Rgheff, "An efficient and lightweight intrusion detection mechanism for service-oriented vehicular networks," *IEEE Internet of things journal*, vol. 1, no. 6, pp. 570–577, 2014.
[21] F. van Wyk, Y. Wang, A. Khojandi, and N. Masoud, "Real-time sensor anomaly detection and identification in automated vehicles," *IEEE Transactions on Intelligent Transportation Systems*, 2019.
[22] R. W. van der Heijden, S. Dietzel, T. Leinmüller, and F. Kargl, "Survey on misbehavior detection in cooperative intelligent transportation systems," *arXiv preprint arXiv:1610.06810*, 2016.
[23] G. Carl, G. Kesidis, R. R. Brooks, and S. Rai, "Denial-of-service attack-detection techniques," *IEEE Internet computing*, vol. 10, no. 1, pp. 82–89, 2006.
[24] A. A. Cárdenas, S. Amin, Z.-S. Lin, Y.-L. Huang, C.-Y. Huang, and S. Sastry, "Attacks against process control systems: risk assessment, detection, and response," in *Proceedings of the 6th ACM symposium on information, computer and communications security*. ACM, 2011, pp. 355–366.
[25] Z. Zhan, M. Xu, and S. Xu, "Characterizing honeypot-captured cyber attacks: Statistical framework and case study," *IEEE Transactions on Information Forensics and Security*, vol. 8, no. 11, pp. 1775–1789, 2013.
[26] R. Mitchell and R. Chen, "Effect of intrusion detection and response on reliability of cyber physical systems," *IEEE Transactions on Reliability*, vol. 62, no. 1, pp. 199–210, 2013.
[27] S. Sridhar and M. Govindarasu, "Model-based attack detection and mitigation for automatic generation control," *IEEE Transactions on Smart Grid*, vol. 5, no. 2, pp. 580–591, 2014.
[28] R. Mitchell and I.-R. Chen, "A survey of intrusion detection techniques for cyber-physical systems," *ACM Computing Surveys (CSUR)*, vol. 46, no. 4, p. 55, 2014.
[29] W. Li and H. Song, "Art: An attack-resistant trust management scheme for securing vehicular ad hoc networks," *IEEE Transactions on Intelligent Transportation Systems*, vol. 17, no. 4, pp. 960–969, 2015.

[30] W. Min, M. Fan, X. Guo, and Q. Han, "A new approach to track multiple vehicles with the combination of robust detection and two classifiers," *IEEE Transactions on Intelligent Transportation Systems*, vol. 19, no. 1, pp. 174–186, 2017.

[31] J. Liang, Q. Lin, J. Chen, and Y. Zhu, "A filter model based on hidden generalized mixture transition distribution model for intrusion detection system in vehicle ad hoc networks," *IEEE Transactions on Intelligent Transportation Systems*, 2019.

[32] V. Sucasas, G. Mantas, F. B. Saghezchi, A. Radwan, and J. Rodriguez, "An autonomous privacy-preserving authentication scheme for intelligent transportation systems," *Computers & Security*, vol. 60, pp. 193–205, 2016.

[33] E. S. Page, "Continuous inspection schemes," *Biometrika*, vol. 41, pp. 100–115, 1954.

[34] A. G. Tartakovsky, B. L. Rozovskii, R. B. Blažek, and H. Kim, "Detection of intrusions in information systems by sequential change-point methods," *Statistical methodology*, vol. 3, no. 3, pp. 252–293, 2006.

[35] D. Misiunas, J. Vítkovskỳ, G. Olsson, A. Simpson, and M. Lambert, "Pipeline break detection using pressure transient monitoring," *Journal of Water Resources Planning and Management*, vol. 131, no. 4, pp. 316–325, 2005.

[36] R. Ratnam, J. B. Goense, and M. E. Nelson, "Change-point detection in neuronal spike train activity," *Neurocomputing*, vol. 52, pp. 849–855, 2003.

[37] K.-T. Cho and K. G. Shin, "Fingerprinting electronic control units for vehicle intrusion detection." in *USENIX Security Symposium*, 2016, pp. 911–927.

[38] A. Haydari and Y. Yilmaz, "Real-time detection and mitigation of ddos attacks in intelligent transportation systems," in *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*. IEEE, 2018, pp. 157–163.

[39] D. Lee and D. Kundur, "Cyber attack detection in pmu measurements via the expectation-maximization algorithm," in *Signal and Information Processing (GlobalSIP), 2014 IEEE Global Conference on*. IEEE, 2014, pp. 223–227.

[40] M. Aloqaily, S. Otoum, I. Al Ridhawi, and Y. Jararweh, "An intrusion detection system for connected vehicles in smart cities," *Ad Hoc Networks*, vol. 90, p. 101842, 2019.

[41] L. Yang, A. Moubayed, I. Hamieh, and A. Shami, "Tree-based intelligent intrusion detection system in internet of vehicles," in *2019 IEEE Global Communications Conference (GLOBECOM)*. IEEE, 2019, pp. 1–6.

[42] S. Park and J.-Y. Choi, "Malware detection in self-driving vehicles using machine learning algorithms," *Journal of advanced transportation*, vol. 2020, 2020.

[43] D. Kosmanos, A. Pappas, F. J. Aparicio-Navarro, L. Maglaras, H. Janicke, E. Boiten, and A. Argyriou, "Intrusion detection system for platooning connected autonomous vehicles," in *2019 4th South-East Europe Design Automation, Computer Engineering, Computer Networks and Social Media Conference (SEEDA-CECNSM)*. IEEE, 2019, pp. 1–9.

[44] S. Tariq, S. Lee, H. K. Kim, and S. S. Woo, "Can-adf: The controller area network attack detection framework," *Computers & Security*, p. 101857, 2020.

[45] Y. Wang, N. Masoud, and A. Khojandi, "Real-time sensor anomaly detection and recovery in connected automated vehicle sensors," *IEEE Transactions on Intelligent Transportation Systems*, 2020.

[46] X. Wang, I. Mavromatis, A. Tassi, R. Santos-Rodriguez, and R. J. Piechocki, "Location anomalies detection for connected and autonomous vehicles," in *2019 IEEE 2nd Connected and Automated Vehicles Symposium (CAVS)*. IEEE, 2019, pp. 1–5.

[47] O. Avatefipour, A. S. Al-Sumaiti, A. M. El-Sherbeeny, E. M. Awwad, M. A. Elmeligy, M. A. Mohamed, and H. Malik, "An intelligent secured framework for cyberattack detection in electric vehicles' can bus using machine learning," *IEEE Access*, vol. 7, pp. 127 580–127 592, 2019.

[48] M. R. C. Acosta, S. Ahmed, C. E. Garcia, and I. Koo, "Extremely randomized trees-based scheme for stealthy cyber-attack detection in smart grid networks," *IEEE Access*, vol. 8, pp. 19 921–19 933, 2020.

[49] K. K. Nguyen, D. T. Hoang, D. Niyato, P. Wang, D. Nguyen, and E. Dutkiewicz, "Cyberattack detection in mobile cloud computing: A deep learning approach," in *2018 IEEE Wireless Communications and Networking Conference (WCNC)*. IEEE, 2018, pp. 1–6.

[50] H. Karimipour, A. Dehghantanha, R. M. Parizi, K.-K. R. Choo, and H. Leung, "A deep and scalable unsupervised machine learning system for cyber-attack detection in large-scale smart grids," *IEEE Access*, vol. 7, pp. 80 778–80 788, 2019.

[51] A. Ferdowsi, S. Ali, W. Saad, and N. B. Mandayam, "Cyber-physical security and safety of autonomous connected vehicles: Optimal control meets multi-armed bandit learning," *IEEE Transactions on Communications*, vol. 67, no. 10, pp. 7228–7244, 2019.

[52] A. Alshammari, M. A. Zohdy, D. Debnath, and G. Corser, "Classification approach for intrusion detection in vehicle systems," *Wireless Engineering and Technology*, vol. 9, no. 4, pp. 79–94, 2018.

[53] G. K. Rajbahadur, A. J. Malton, A. Walenstein, and A. E. Hassan, "A survey of anomaly detection for connected vehicle cybersecurity and safety," in *2018 IEEE Intelligent Vehicles Symposium (IV)*. IEEE, 2018, pp. 421–426.

[54] A. Humayed, J. Lin, F. Li, and B. Luo, "Cyber-physical systems security–a survey," *arXiv preprint arXiv:1701.04525*, 2017.

[55] S.-H. Kong and S.-Y. Jun, "Cooperative positioning technique with decentralized malicious vehicle detection," *IEEE Transactions on Intelligent Transportation Systems*, vol. 19, no. 3, pp. 826–838, 2017.

[56] Z. A. Biron, S. Dey, and P. Pisu, "Real-time detection and estimation of denial of service attack in connected vehicle systems," *IEEE Transactions on Intelligent Transportation Systems*, vol. 19, no. 12, pp. 3893–3902, 2018.

[57] E. Mousavinejad, F. Yang, Q.-L. Han, X. Ge, and L. Vlacic, "Distributed cyber attacks detection and recovery mechanism for vehicle platooning," *IEEE Transactions on Intelligent Transportation Systems*, 2019.

[58] F. Sakiz and S. Sen, "A survey of attacks and detection mechanisms on intelligent transportation systems: Vanets and iov," *Ad Hoc Networks*, vol. 61, pp. 33–50, 2017.

[59] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the em algorithm," *Journal of The Royal Statistical Society, Series B*, vol. 39, no. 1, pp. 1–38, 1977.

[60] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements Statistical Learning: Data Mining, Inference, and Prediction*, 2nd ed. Springer, 2009.

[61] M. Basseville, I. V. Nikiforov *et al.*, *Detection of abrupt changes: theory and application*. prentice Hall Englewood Cliffs, 1993, vol. 104.

[62] V. A. Siris and F. Papagalou, "Application of anomaly detection algorithms for detecting syn flooding attacks," *Computer communications*, vol. 29, no. 9, pp. 1433–1442, 2006.

[63] D. C. Montgomery, *Introduction to statistical quality control*. John Wiley & Sons (New York), 2009.

[64] D. Krajzewicz and C. Rossel, "Simulation of urban mobility (sumo)," *Centre for Applied Informatics (ZAIK) and the Institute of Transport Research at the German Aerospace Centre*, 2007.

[65] M. Chowdhury, M. Rahman, A. Rayamajhi, S. M. Khan, M. Islam, Z. Khan, and J. Martin, "Lessons learned from the real-world deployment of a connected vehicle testbed," *Transportation Research Record*, vol. 2672, no. 22, pp. 10–23, 2018.

[66] L. Liu, C. Li, Y. Li, S. Peeta, and L. Lin, "Car-following behavior of connected vehicles in a mixed traffic flow: modeling and stability analysis," in *2018 IEEE 8th Annual International Conference on CYBER Technology in Automation, Control, and Intelligent Systems (CYBER)*. IEEE, 2018, pp. 1085–1088.

[67] M. Islam, M. Rahman, S. M. Khan, M. Chowdhury, and L. Deka, "Development and performance evaluation of a connected vehicle application development platform," *Transportation Research Record*, p. 0361198120917146, 2020.

[68] M. Behrisch, L. Bieker, J. Erdmann, and D. Krajzewicz, "Sumo–simulation of urban mobility: an overview," in *Proceedings of SIMUL 2011, The Third International Conference on Advances in System Simulation*. ThinkMind, 2011.

[69] C. Flitsch, K.-H. Kastner, K. Bósa, and M. Neubauer, "Calibrating traffic simulation models in sumo based upon diverse historical real-time traffic data–lessons learned in its upper austria," *EPiC Series in Engineering*, vol. 2, pp. 25–42, 2018.

[70] W. Cohda, "V2x-locate positioning system whitepaper," www.cohdawireless.com, 2017, accessed: 08-13-2020.

Gurcan Comert received the B.Sc. and M.Sc. degree in Industrial Engineering from Fatih University, Istanbul, Turkey and the Ph.D. degree in Civil Engineering from University of South Carolina, Columbia, SC, in 2003, 2005, and 2008, respectively. He is currently with Benedict College, Columbia, SC, associate director at the USDOT Center for Connected Multimodal Mobility (C^2M^2), and a researcher at Information Trust Institute, University of Illinois Urbana-Champaign. His research interests include applications of probabilistic models to transportation problems.

Mizanur Rahman is an assistant professor in the Department of Civil, Construction and Environmental Engineering at the University of Alabama, Tuscaloosa, Alabama. He received his M.Sc. and Ph.D. degrees in Civil Engineering (Transportation systems), from Clemson University, in 2013 and 2018, respectively. After his graduation in August 2018, he joined as a postdoctoral research fellow for the Center for Connected Multimodal Mobility (C^2M^2), a U.S. Department of Transportation Tier 1 University Transportation Center (ces.clenson.edu/c2m2). After that, he has also served as an Assistant Director of C2M2. He was closely involved in the development of South Carolina Connected Vehicle Testbed (SC-CVT). His research focuses on traffic flow theory, cyber security, and transportation cyber-physical systems for connected and automated vehicles and smart cities.

Mhafuzul Islam received the BS degree in computer science and engineering from the Bangladesh University of Engineering and Technology in 2014 and MS degree in Civil Engineering from Clemson University at 2018. He is currently a Ph.D. student in the Glenn Department of Civil Engineering at Clemson University. His research interests include Transportation Cyber-Physical Systems with an emphasis on Data-driven Connected Autonomous Vehicle.

Mashrur Chowdhury (SM'12) received the Ph.D. degree in civil engineering from the University of Virginia, USA in 1995. Prior to entering academia in 2000, he was a Senior ITS Systems Engineer with Iteris Inc. and a Senior Engineer with Bellomo McGee Inc., where he served as a Consultant to many state and local agencies, and the U.S. DOT on ITS related projects. He is the Eugene Douglas Mays Professor of Transportation with the Glenn Department of Civil Engineering, Clemson University, SC, USA. He is also a Professor of Automotive Engineering and Computer Science at Clemson University. He is the Director of the USDOT Center for Connected Multimodal Mobility (C^2M^2) (a Tier 1 USDOT University Transportation Center). He is Co-Director of the Complex Systems, Data Analytics and Visualization Institute (CSAVI) at Clemson University. He is also the Director of the Transportation Cyber-Physical Systems Laboratory at Clemson University. He serves as an Associate Editor for the IEEE Transactions on Intelligent Transportation Systems and Journal of Intelligent Transportation Systems. He is a Fellow of the American Society of Civil Engineers and a Senior Member of IEEE.