Efficient Detection and Localization of DoS Attacks in Heterogeneous Vehicular Networks

Meenu Rani Dey, *Student Member, IEEE,* Moumita Patra, *Member, IEEE,* and Prabhat Mishra, *Fellow, IEEE*

***Abstract*—Vehicular communication has emerged as a powerful tool for providing a safe and comfortable driving experience for users. Long Term Evolution (LTE) supports and enhances the quality of vehicular communication due to its properties such as high data rate, spatial reuse, and low delay. However, high mobility of vehicles introduces a wide variety of security threats, including Denial-of-Service (DoS) attacks. In this paper, we propose effective solutions for real-time detection and localization of DoS attacks in an LTE-based vehicular network with mobile network components (e.g., vehicles, femto access points, etc.). We consider malicious data transmission by vehicles in two ways- using real identification (unintentional) and using fake identification (intentional). This paper makes three important contributions. First, we propose an efficient attack detection technique based on data packet counter and average Packet Delivery Ratio (PDR). Next, we present an improved attack detection framework using machine learning algorithms. We use some ML-based supervised classification algorithms to make detection more robust and consistent. Finally, we propose Data Packet Counter (DPC)-based, triangulation-based and measurement report based localization for both intentional and unintentional DoS attacks. We analyze the average packet delay incurred by vehicles by modelling the system as an $M/M/m$ queue. Our experimental evaluation demonstrates that our proposed technique significantly outperforms state-of-the-art techniques.**

***Index Terms*—LTE-based vehicular network, denial-of-service, intrusion detection, intrusion localization, machine learning**

I. Introduction

Vehicular communication has been widely investigated by researchers for providing users a smart, safe, and comfortable driving experience. To this direction, Long-Term Evolution (LTE) has recently emerged as a technology that can support vehicular applications due to its attractive features such as, high capacity, lower delay, and spatial reuse [2], [3]. However, features such as high mobility of vehicles and constantly changing network topology pose various Quality of Service (QoS) and security-related challenges [4].

Vehicular communication systems enable many exciting applications that will make driving safer, more efficient, and more comfortable. But there are some security issues still open for researchers such as authenticity, confidentiality, availability, non-repudiation, privacy etc. Based on these challenges several security attacks can possible such as Denial-of-Service (DoS), Distributed Denial-of-Service (DDoS), Sybil attack, Location Tracking, Malware, Man in the Middle attack, Brute force attack etc [5], In this work, we focus on DoS attack [6] and propose corresponding attack detection and localization techniques.

LTE over the years has evolved to become a highly-complex heterogeneous system in order to support large traffic demands of users. Entities such as, Macro Base Stations (MBSs) and Femtocells/Femto Access Points (FAPs) are used to provide the increasing coverage and capacity demands [4]. FAPs are low-cost, low-power cellular base stations usually deployed in homes/offices to provide improved coverage to nearby users. Recently, mobile FAPs have gained momentum for providing better capacity and coverage in highly mobile scenarios [3], [7]. A FAP is a small base station that is primarily used to improve the coverage of a given cellular network. It serves as a base station that uses a radio/air interface to connect nodes to the service provider's core network. A FAP connects to the operator's core network via the subscriber's wired/wireless backhaul connection. FAP performs the same functions as a macro base station. Any subscriber can use a FAP. The usage of FAP can be restricted to a small group of users, or by a hybrid of both, with priority given to favoured users in certain scenarios [8]. FAPs being low-power base stations are vulnerable to a wide variety of attacks including DoS attacks [9]. Additionally, mobility of FAPs further adds challenges in the detection and localization of attacks. In this work, we have considered an LTE-based Vehicular network (LTE-Vnet) with mobile FAPs. Given the high mobility of vehicles as well as FAPs, it becomes a major challenge to detect and localize DoS attacks.

Machine Learning (ML) has gained significant interest in recent years for detection of security attacks. ML models are able to detect security attacks by learning the behavior of attack scenario as well as normal scenario. ML-based classification algorithms can guarantee the detection of attacker with consistent accuracy [10]. Specifically, we utilize supervised classification algorithms to detect the attack in the LTE-Vnet environment. We also use the feature selection method to reduce the computational cost for detection.

A. Attack Model

Our threat model is based on attack in the data plane of the network via malicious vehicles. It assumes that malicious

This is an extended version of the paper that appeared in DATE 2021 [1]. This version has significant additional material including: (i) DoS attack detection using machine learning, (ii) several algorithms for attack detection and localization, and (ii) additional results and detailed comparison with related efforts.

M. Dey and M. Patra are with the Department of Computer Science and Engineering, Indian Institute of Technology Guwahati, Assam 781039, India (e-mail: rmeenu@iitg.ac.in; moumita.patra@iitg.ac.in). P. Mishra is with the Department of Computer and Information Science and Engineering, University of Florida, Gainesville, FL 32611, USA.

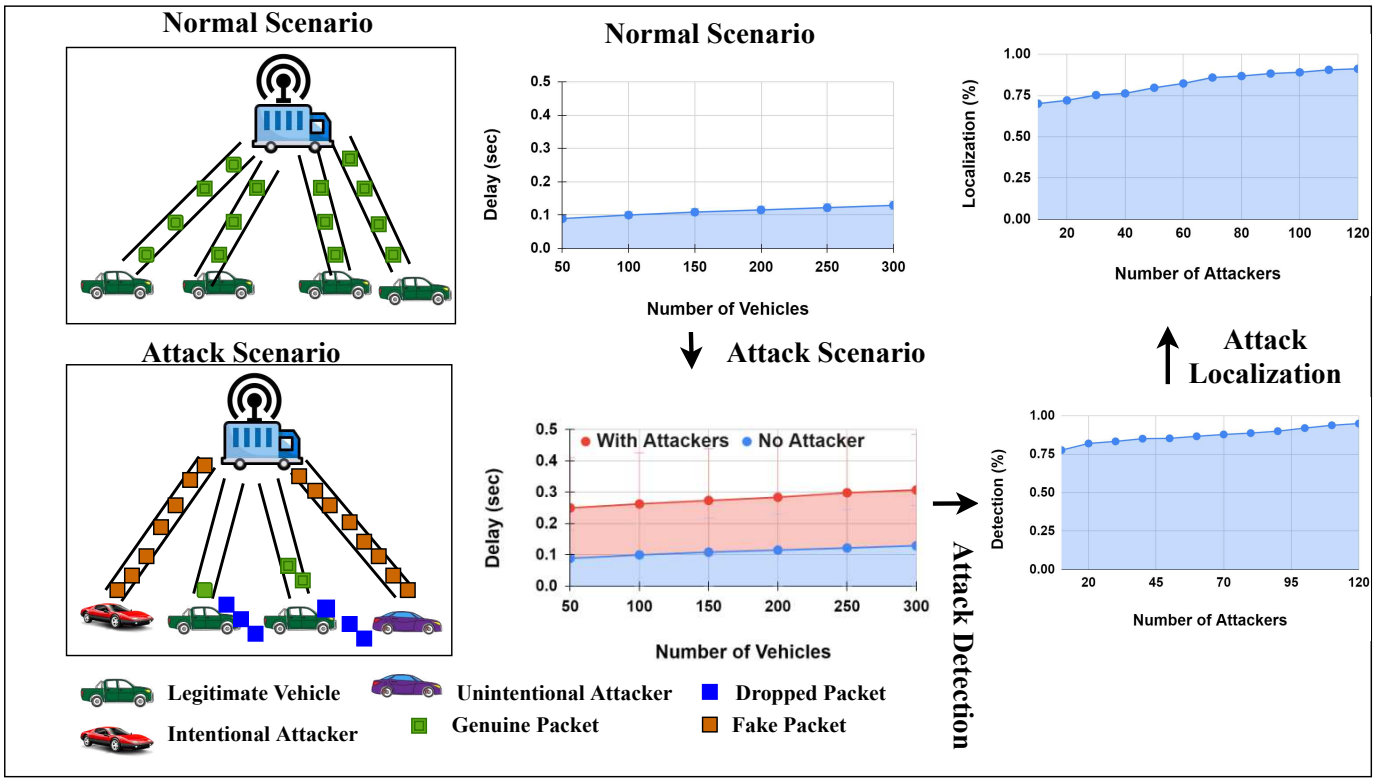


Fig. 1: Example of DoS Attack in LTE-based Vehicular Network

vehicles (attackers) can be used to launch DoS attacks. We assume that malicious vehicles will transmit packets in the uplink channel to their associated FAPs, with a higher transmission rate than legitimate vehicles' transmission rate.

LTE uses Resource Blocks (RBs) for data transmission. RB is a resource unit, defined in the time and frequency domain. RBs are allocated by a FAP to its associated vehicles which want to transmit data. The amount of RBs available for communication is fixed [11]. Therefore, a malicious vehicle can lead to higher contention for the fixed amount of available RBs by transmitting data at higher rates. This leads to higher average packet delay, low network throughput, and large packet drops. Such DoS attacks can be carried out in two possible ways:

1) Unintentional Attack (Case I): high amount of fake data transmission (unintentionally) by attacker using its own Vehicle Id [6].
2) Intentional Attack (Case II): high amount of fake data transmission (intentionally) by attacker using fake Vehicle Id [9].

Figure 1 shows the impact of DoS attack in LTE-Vnet.

B. Motivation

In this work, we focus on DoS attack. As few of the works in the literature have focussed on detection and localization of DoS attack in LTE-Vnet scenario, the following points motivated our work on detection and localization of attacker.

- The accuracy of PDR and DPC-based detection techniques vary with the number of attackers. As the number of attackers increases accuracy of these detection techniques also increases. In order to achieve a consistent accuracy, irrespective of the number of attackers, we use ML-based classification techniques to detect the attack.
- Triangulation method and DPC-based attack localization techniques are modelled according to the way of the attack. To localize the intentional attack we introduce the Triangulation method and for the unintentional attack, we use DPC-based detection. For the real-time scenario where intentional and unintentional both attacks can be possible simultaneously, we introduce an MR-based localization technique that can localize the attacker in case of both type of attacks.
- Since in our previous work [1], we have proposed detection and localization techniques for DoS attack, we also check the effectiveness of proposed techniques for DDoS.

C. Research Challenges

There are various research challenges for detection and localization of attackers due to the presence of highly mobile users as well as FAPs. We highlight three important challenges below:

- Mobile nodes: Due to high mobility of vehicles, it becomes imperative that detection and localization of any attacker is done before it moves out from the transmission range. DPC and PDR handle the detection and localization locally (at FAP) which leads to faster detection and localization.
- Large amount of data: In vehicular networks, each vehicle generates plenty of data (such as event

driven data, infotainment data, weather information, location information, etc.). Performing detection and localization with handling these data simultaneously is very challenging for FAP.
- Real-time constraints: In vehicular networks, majority of vehicular applications have strict delay constraints. Like any safety-critical system, violation of real-time constraints can lead to catastrophic (e.g., life threatening) consequences. Therefore, we have designed our algorithms very simple and lightweight to enable fast computation.

D. Major Contributions

Some works in the literature study DoS attacks in LTE-Vnet scenarios and provide possible solutions to detect the attacks [11]–[13]. However, these works have limited applicability (two layer architecture) with only MBSs. In this work, we consider a diverse scenario using MBSs as well as mobile FAPs in a vehicular network environment. We study the effects of DoS attacks by malicious vehicles in these diverse scenarios and propose efficient attack detection and localization techniques. Our primary objective is to detect attacks at the FAPs and then localize the attackers using only the information available at the FAPs.

- We propose a real-time DoS attack detection technique in LTE-Vnet with mobile FAPs based on Packet Delivery Ratio (PDR) and Data Packet Counter (DPC).
- We explore the effectiveness of machine learning algorithms in efficient detection of DoS attacks.
- We propose an efficient localization technique based on triangulation method, DPC and Measurement Report (MR)-based approach.
- We perform extensive performance analysis of average packet delay in the given scenario by modeling the system using $M/M/m$ queuing model. Simulation results demonstrate the effectiveness of the proposed framework.

E. Paper Organization

The rest of this paper is organized as follows. Section II surveys related efforts in DoS attack detection and localization. Section III describes our system architecture and provides an overview of our proposed framework. Section IV and Section V describe our proposed techniques for attack detection and localization, respectively. Section VI analyzes the performance of our proposed framework followed by experimental results in Section VII. Finally, Section VIII concludes the paper.

II. Related work

In this Section, we first discuss DoS attacks in LTE-Vnet. Next, we survey existing attack detection techniques and discuss their limitations. Finally, we describe prior efforts in machine learning based detection of DoS attacks.

A. DoS Attacks in LTE-Vnet

Automated vehicles and electric vehicles have gained ample interest of researchers in recent years [14]–[16]. LTE in recent years has emerged as a technology which can support a large number of vehicular applications. Its properties such as low delay, high data rate, and spatial reuse help to support such communications [2], [3]. However, features such as, high mobility of vehicles and constantly changing network topology make it vulnerable to several security attacks, including DoS attacks [4], [9]. DoS attacks in LTE can take place in many ways such as, attack in network components and in data and control channels [4], [9], [17]. Several works in the literature propose solutions for overcoming security challenges in LTE [12], [13], [17], [18]. In [9], the authors have developed a theoretical framework to explore the attack space in LTE. They have shown that the attack space can be in three dimensions– communication security services, planes of attack, and network components under attack. In [18], the authors have proposed a lightweight traffic based attack detection scheme in Voice-over-LTE (VoLTE) network. They have used Bayesian game to model their system but their work deals with static entities and the detection of attack is done at the MBS. Ambrosin et al. [13] propose a novel method to implement a distributed DoS attack on a target mobile operator's control network. They have exploited the lack of coordination between local and remote components of the LTE network during the roaming authentication process to realize a pulse DoS attack using temporal lensing. However, they have not proposed any attack detection or localization technique for their scenario. Authors in [17] talk about user-targeted DoS attack in LTE network. Their attack model is based on deploying a rogue base station which targets users to perform DoS attack. However they have not addressed mobility and have not proposed any detection and localization technique. Zhu et al. [12] study security flaws in platoon of vehicles in LTE-V2X networks but they have not proposed any attack detection or localization technique.

B. Detection and Localization of DoS Attacks

A vast majority of the existing security research efforts in LTE networks have one of the following fundamental limitations– (i) they do not deal with highly mobile scenarios like vehicles, (ii) they cannot deal with heterogeneous network entities like FAPs, or (iii) they do not propose any attack detection or localization technique for their threat models. Several literature proposed the DoS attack detection in [19]–[22]. In [19], the authors have used the idea of market trading and set up trading rules to restrict the malicious vehicle's spread of false messages, and to encourage vehicles to contribute to the traffic event monitoring and verification. In this work, they have considered static RSU whereas,in our work we have considered mobile FAP and detection and localization of attack has been done in FAP. In [20], the authors have proposed multivariate stream analysis to detect and mitigate the DDoS attack. Multivariate stream analysis algorithm maintains multiple stages like classification, pre-processing and analysis of data packets that come from

vehicles. They have considered VANET network with static RSU, whereas, in our scenario we have considered a heterogeneous vehicular network with mobile FAPs, and have done detection and localization in FAP itself. In [21], the authors have proposed a centralized reputation management scheme to detect malicious nodes in the vehicular network. Their main objective is to appropriately handle the reputation system to maximize the accuracy of message validation. However, our proposed algorithm is based on detecting and localising DoS attack. In [22], the authors have used evolutionary game theory to model the evolution process of malicious users' attacking strategies and discussed the methodology of the evaluation simulations. As they have considered non-heterogeneous vehicular network with static RSU to flag malicious users whereas, we have considered LTE-Vnet with mobile FAP to detect and localize DoS attack.

Some works have been done in the literature to detect the attacker in various scenarios such as UAV, SDN netorks etc. [23]–[27]. In [23], the authors have proposed an intrusion detection system to predict the future misbehaviour of a malicious vehicle using the game theory technique. However, they have only focused on attack detection part. In [24], the authors have used SDN architecture and blockchain technology for routing calculation and configuration management of the Unmanned Aerial Vehicle (UAV) network. As the suggested work is based on UAV scenario whereas, our work is based on LTE-Vnet scenario. In [25], the authors have used a machine learning-based multiple-kernel clustering intrusion detection scheme for 5G and IoT networks. This work is based on static nodes and non-heterogeneous scenarios whereas, our work is based on mobile nodes and heterogeneous scenario. In [26], the authors have done a survey on analysis of DoS attack detection and mitigation in the vehicular networks. In [27], the authors have proposed a Weight-Based Ensemble Machine Learning Algorithm (WBELA) to identify abnormal messages of vehicular Controller Area Network (CAN) bus network. They have also derived a model based on many-objective optimization for intrusion detection of CAN bus network. However, this work is based on CAN bus network which is different from heterogeneous vehicular networks.

Li et al. [11] have addressed DoS attack in cellular-V2X network where attacker maliciously reserves communication resources such that legitimate vehicles get little or no resources. They have proposed an attack detection technique which is carried out at the Mobile Edge Computing (MEC) server based on the information received from the MBS. They did not consider heterogeneity in the network. Also, performing detection at MEC server and MBS may lead to high delay which makes it unsuitable for real-time attack detection. The authors have not addressed the issue of localization of attacker. While there are promising approaches for DoS attack detection and localization in network-on-chip architectures [28], [29], they are not suitable for vehicular networks.

C. DoS Attack Detection using Machine Learning

Detection and localization of attackers in LTE networks has been explored extensively in the literature [30]–[33]. In [33], the authors have used ML-based techniques to detect DoS attack in static nodes. In [31] the authors have proposed some ML-based techniques to detect the misbehaviour of attacker in vehicular network but they have not considered the heterogeneous scenario. In [30], the authors have proposed tree based ML algorithms to detect attacks. However, they have not considered a heterogeneous scenario.

We have used some supervised classification algorithms in LTE-Vnet scenario to detect DoS attack in mobile FAPs. ML-based techniques are considered to give better accuracy but they are very complex and take high computational cost. We use the feature extraction technique to reduce computational complexity. The presence of mobile FAPs helps in spatial reuse [3] but introduces challenges in detecting the attack as well as in localizing the attacker(s). *To the best of our knowledge, our work is the first attempt in detection and localization of DoS attacks in heterogeneous vehicular networks.*

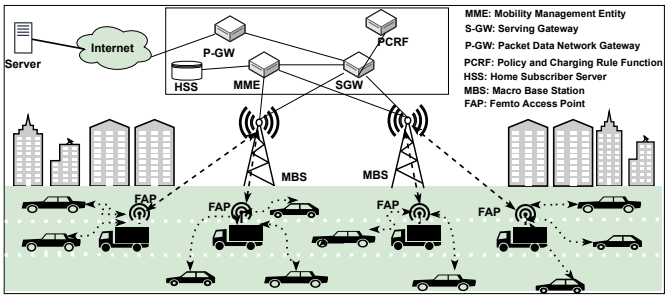


Fig. 2: System model of an LTE-based vehicular network

III. System Model and Problem Formulation

Here, we first describe the system model. Next, we provide an overview of our proposed framework.

A. System Model

Our system model consists of a city scenario with multiple roads and intersections. The system architecture considered is given in Figure 2. Vehicles are equipped with devices which help them to communicate via cellular network. FAPs are placed on larger vehicles, such as, buses and trucks. A vehicle associates/joins with a FAP if it receives strong signal strength from the FAP. In the absence of a FAP in the neighborhood, a vehicle associates with MBS. Vehicles generate fixed size packets and the inter-arrival duration of packet generation follows an exponential distribution. Packets generated by vehicles remain in their buffer until they get a chance to transmit to a FAP or MBS. Vehicles can be either active or inactive. Active vehicles, when associated with FAPs, send packets periodically, while inactive vehicles do not send packets even after associating with a FAP.

B. Overview

Figure 3 shows a broad overview of our proposed framework. We consider both types of attacks outlined in

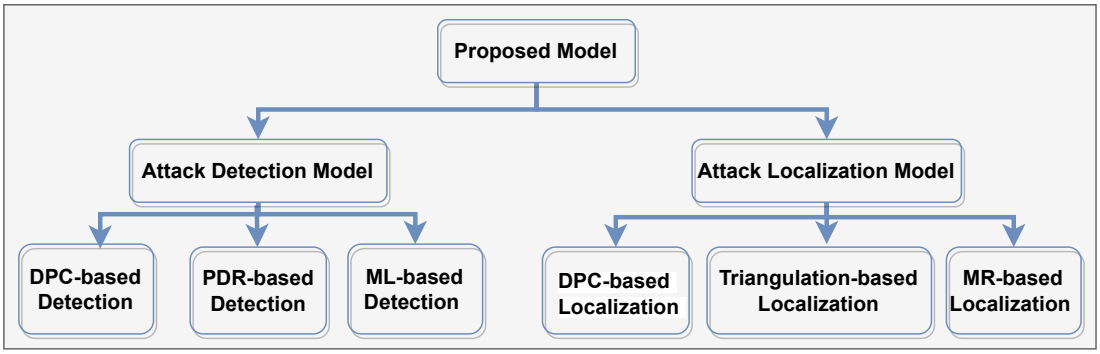


Fig. 3: Overview of our proposed framework

Section I-A. We perform DoS attack detection using the following three approaches:

- DPC based Detection: DPC is a counter which is initialized to the maximum number of packets that can be transmitted by a vehicle. This approach is described in Section IV-A
- PDR based Detection: PDR is defined as the ratio between the number of packets received to the number of generated packets. This approach is described in Section IV-B.
- Detection using ML algorithms: We use some ML-based classification algorithms to detect attack. We consider K Nearest Neighbour (KNN), Support Vector Machine (SVM), Decision Tree (DT), Random Forest (RF), Extra Tree (ET), eXtreme Gradient Boosting (XGBoost), Stacking, Feature Selection XGBoost (FS XGBoost), Feature Selection (FS Stacking). This is described in Section IV-C.

To enable efficient localization of malicious vehicles, we explore the following two approaches:

- Triangulation-based localization: Triangulation method uses the distance of a known vehicle (legitimate vehicle) from FAP, an unknown vehicle (attacker vehicle), and the measured angle between the vehicles [34] to calculate the location of the attacker as described in Section V-B.
- Measurement Report based localization: Section V-C describes our proposed MR-based localization of DoS attacks.

IV. Detection of DoS Attacks

This section describes our proposed attack detection using DPC, PDR as well as ML algorithms.

A. DPC-based Detection

In this case, we consider detecting unintentional attacks at the FAPs by using DPC counters. All uplink packets from vehicles are transmitted via their associated FAPs. DPC counters for each vehicle, associated to a FAP, are initialized to the maximum number of packets that can be delivered by the corresponding vehicles in a given time interval. Finding the initial value of DPC counter is crucial. We show its calculation below.

Let us assume that there are m RBs and x vehicles are associated to a FAP. These x vehicles include both legitimate and malicious vehicles. Total time is T which is divided into time slots of duration τ. Let α be the packet generation rate. The probability of accessing any one of the m RBs by a vehicle is given by

$$\delta = e^{-\alpha\left(\left[\frac{x-1}{m}+1\right]\right)\tau} \tag{1}$$

Now, let us assume that j out of x vehicles generate packets in a given time interval. These j vehicles can be selected in the following way:

$$\binom{x}{j} = \frac{x(x-1)}{j} \tag{2}$$

Let g_t be the probability of generating packets by a vehicle in a given time slot, t. The probability of generating packets by j vehicles is

$$\Gamma_j = g_t^j (1-g_t)^{x-j} \tag{3}$$

The probability that exactly j of x vehicles transmit packets in a given time slot is given by

$$\beta_j = j \times \Gamma_j = j \times g_t^j \left(1-g_t\right)^{x-j} \tag{4}$$

By using Equations 1 and 4, we can calculate the average number of packets transmitted per vehicle:

$$E\left[X\right] = \delta \times \beta_j \tag{5}$$

The maximum number of packets that can arrive at a FAP from a vehicle depends on the distance between the FAP and the vehicle, the Signal-to-Interference-plus-Noise Ratio (SINR) value and Reference Signal Received Power (RSRP) value. SINR is calculated as:

$$SINR = \frac{S}{N+I} \tag{6}$$

where S denotes the signal strength, N denotes the noise and I denotes the interference in the channel. Based on SINR, the maximum channel capacity can be calculated by using Shannon's Channel Capacity theorem [3]:

$$W = B \times log_2\left(1+SINR\right) \tag{7}$$

where B denotes the bandwidth of the channel. Now, by using Equation 5 and 7, the average number of packets received by a FAP, per vehicle, is calculated as:

$$E\left[r\right] = \frac{W}{E\left[X\right] \times x} \tag{8}$$

Therefore, with packet size l, the DPC value can be calculated as:

$$DPC = max\left(\frac{W}{l}, E\left[r\right]\right) \quad (9)$$

Algorithm 1 outlines the sequence of steps to detect the attacker(s). In this algorithm, at each time step (10 ms), FAPs find their associated vehicles (line 7). At every d ms, FAP checks for association with new vehicles. If present, it calculates its DPC value and initializes its packet counter $PCount$, among other values (lines 9-16). Otherwise, the uplink buffer is checked and $PCount$ is updated (lines 17-19). Based on the condition in line 20, the attack is detected and the corresponding VId added into the A_VID list (20-22).

Algorithm 1: DPC-based Detection of DoS Attacks

Input: fap: set of FAPs
Output: A_VID: List of attackers VId
1 n: total number of vehicle;
2 $Ubuf$: Temporary uplink buffer with packet transmitted by each vehicle at FAP;
3 $PCount$:Packet count;
4 $temp = 0$;
5 **for** $((i = 1; i <= T; i++)$ **do**
6 **for** $(f = 1; f <= fap; f++)$ **do**
7 $\bar{V}$ = $vehicles\_associated\_to\_fap(f)$;
8 **for** $(k = 1; k <= n; k++)$ **do**
9 **if** $(i\%d == 0)$ **then**
10 **for** $(v = 1; v <= \bar{V}; v++)$ **do**
11 **if** $(k.VId == v.VId)$ **then**
12 **if** $(v.f_{i-1} \neq v.f_i)$ **then**
13 $v.PCount = 0$;;
14 $v.Attacker$=False;
15 $v.Ubuf$= null ;
16 $v.DPC =$ $Calculate\_DPC(v)$;
17 $temp = v.Ubuf.size()/l$;
18 **for** $(j = 1; j <= temp; j++)$ **do**
19 $v.PCount++$;
20 **if** $(v.PCount > v.DPC)$ **then**
21 $v.Attacker = True$;
22 $A\_VID \leftarrow v.VId$;

B. PDR-based Detection

In this section, we consider PDR to detect intentional attacks. We assume that the attacker uses the ID of inactive vehicles, associated to a FAP, to launch attack and hides its own identification [9].

- Each vehicle (both legitimate and attacker) periodically calculates its own PDR with the help of acknowledgments received in each time slot.
- Vehicles then send their PDR values to FAP through control plane (the attack is happening over data plane).
- FAP keeps track of PDR values of each vehicle associated to it and uses these values to calculate the Average PDR (APDR) of each vehicle. The objective behind calculating APDR is to avoid flagging legitimate vehicles as potential attackers.
- If APDR$< \vartheta$, a threshold [35], the vehicle is considered to be a legitimate vehicle, otherwise it is flagged as a possible attacker. The APDR value of inactive vehicles will be zero as they do not send packets. Thus, the FAP will now have two different APDR values for the same vehicle ID – one belonging to the legitimate but inactive vehicle and the other belonging to the attacker.
- If FAP gets two APDR values for the same vehicle ID, it compares both the values with the threshold and starts localizing the vehicle with the higher APDR using the method given in Section V-B.

Algorithm 2: PDR-based Detection of DoS Attacks

Input: Ubuf: uplink buffer of each vehicle
Output: AList: List of Attackers
1 temp=0;
2 **for** $(i = 1; i <= T; i++)$ **do**
3 **for** $(f = 1; f <= fap; f++)$ **do**
4 $\bar{V}$ = $vehicles\_associated\_to\_fap(f)$;
5 $v.Ubuf \longleftarrow v.PDR\_Calculation()$;
6 **for** $(v = 1; v <= \bar{V}; v++)$ **do**
7 **for** $(s = 1; s <= q; s++)$ **do**
8 $PList \leftarrow v.Ubuf.get(pdr)$;
9 $v.AvgPdr \leftarrow PList/q$;
10 **if** $(v.AvgPdr > \vartheta)$ **then**
11 $AttackerTable.put(VId, v.AvgPdr)$;
12 $EntryTable.put(VId, temp++)$;
13 **if** $(EntryTable.getvalue() > 1)$ **then**
14 **if** $(AttackerTable.getvalue() > 0)$ **then**
15 $AList.add(EntryTable.getkey())$
16 **Function** `PDR_Calculation()`:
17 **for** $(v = 1; v <= \bar{V}; v++)$ **do**
18 **if** $(v.Dbuf \neq null)$ **then**
19 $v.RList \leftarrow v.Dbuf$;
20 $v.pdr \leftarrow v.RList.size()/v.G$;
21 $v.Ubuf \leftarrow v.pdr$;

Algorithm 2 outlines the sequence of steps for detecting the attack and produces a list of possible attackers. The PDR values are added to a temporary list (*Plist*) and average PDR is calculated (lines 6-9). If the average PDR is greater than a threshold value ϑ then the vehicle ID is added to a possible attackers list *AttackerTable* (line 11) and IDs of all associated vehicles (active and inactive) is added in another list, *EntryTable* (line 12). If there is any vehicle which appears more than once in the *EntryTable*, this means that it may

have taken ID of an inactive vehicle. Then, its presence in the *AttackerTable* is checked. Thus, if a vehicle has taken inactive vehicle's ID and has transmitted packets more than the threshold ϑ, then it is flagged as an attacker (lines 13-15). Function PDR_Calculation outlines the sequence of steps to calculate the PDR of each vehicle. Here, each vehicle associated to a FAP f, checks its downlink buffer (*Dbuf*) for acknowledgments received and adds the contents in a temporary list *RList* (lines $18-19$). PDR is calculated using the method given in line 20 and the corresponding value is added in the vehicle's uplink buffer *Ubuf* (line 21).

C. DoS Attack Detection using Machine Learning

The approaches mentioned in Sections IV-A and IV-B give better accuracy when number of attackers is more. We explore ML-based detection to cover scenarios where there are very few attackers. ML-based detection algorithms do not depend on number of attackers, they can perform better with any number of attackers. Figure 4 depicts the ML-based attack detection procedure. Here, we have enumerated both the scenarios- normal as well as attack scenario and analysed the network traffic to gather the data and extract the features. We train the ML model with extracted features and then classify it to detect the attack.

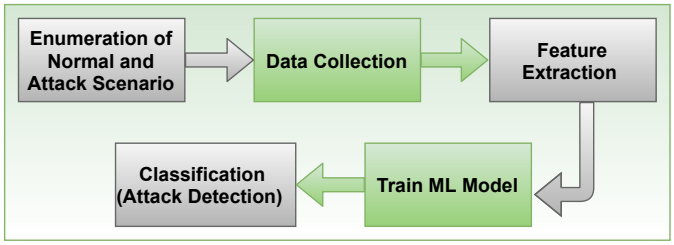


Fig. 4: ML-based attack detection

1) *Data Collection*: The first step in ML-based attack detection model is data collection. In this step, a huge amount of data is collected under the normal scenario as well as the attack scenario. The data can be collected from FAP. FAP contains the information (packet size, data rate, throughput, delay, inter-arrival time,and PDR) about its associated vehicles. We have used real-time Simulation of Urban MObility (SUMO) [36] data to feed in the ML classification algorithm. The data generation is outlined in Section VII-A. As in [32], we too have considered 300077 data instances for the experiments.
2) *Feature Extraction*: We have used some common network features and named them as data rate (f0), throughput (f1), delay (f2), inter-arrival time (f3), and PDR (f4). The computational complexity may increase due to the large dimension of the data set, therefore, we try to extract the features. We use ensemble feature selection technique XGBoost tree-based algorithm to improve the confidence of selected features as it calculates the importance of each feature of each tree and aggregates the value of each tree to make the result more reliable [37]. Figure 5 shows the importance of each feature with respect to the F score. In this figure, we can clearly observe that PDR (f4) has the highest F score and data rate (f0) has the lowest. We ignore the packet size because its F score is less than 0.

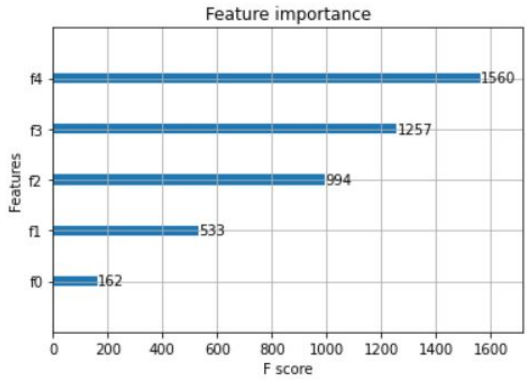


Fig. 5: Feature Importance

3) *Train ML Model*: ML-based classification algorithms are widely used to detect the attack. As our desired output is in the form of binary (like attacker or legitimate), a supervised ML algorithm has been considered to perform attack detection. We have used KNN, SVM, DT, RF, ET, XGBoost, FS XGBoost, and FS Stacking. The reason behind considering these algorithms are as follows:
a) These algorithms have the capability to handle the huge dimension of data and they can take care of non-linear data.
b) It has been observed that DT, RF, ET and XGBoost have lower computational time than other algorithms [38]. The ML models KNN and SVM are the common classification based algorithms. These algorithms are popular because of their significant performance and simple implementation [39] [40]. DT is a tree-based classification algorithm that has decision nodes and leaf nodes, and they represent the features and results, respectively. DT requires low computational cost and it can deal with redundant attributes in less time [41]. ET, RF, and XGBoost are ensemble learning classification algorithms that improve the performance of classification algorithm [37] [42]. To train the above algorithms, the ratio of training and testing set is considered as 0.7 and 0.3 and based on that result is evaluated [33].
4) *Classification (Attack detection)*: We use all the algorithms (mentioned above) to detect the attack. After verifying the classification result, it can be observed in Table II that DT outperforms than other algorithms using our data set.

V. Localization of DoS Attacks

This section describes our proposed attack localization techniques using DPC, triangulation method, and MR.

A. DPC-based Localization

In this section, we propose a technique to localize attacker(s) detected in DPC-based detection technique where the attack

is unintentional. In the DPC-based attack detection technique (Algorithm 1) we got the list of attackers' vehicle ID (A_VID). Algorithm 3 outlines the sequence of steps to localize the attacker(s). In this algorithm, at each time step (10 ms), FAPs find their associated vehicles (line 4). FAP checks if any VId matches with the attacker's VId. Then it adds corresponding location to A_Loc.

Algorithm 3: DPC-based Localization of DoS Attacks

Input: A_VID: List of attackers
Output: A_Loc: List of Location
1 n: total number of vehicle;
2 **for** $((i = 1; i <= T; i++)$ **do**
3 **for** $(f = 1; f <= fap; f++)$ **do**
4 $\bar{V}$ = $vehicles\_associated\_to\_fap(f)$;
5 **for** $(v = 1; v <= \bar{V}; v++)$ **do**
6 **if** $(v.VId == VId.A\_VID)$ **then**
7 $A\_Loc \leftarrow v.Location$;

B. Triangulation-based Localization

In this section, we propose our technique for localization of intentional attackers detected using PDR-based detection technique. It should be noted that localization in LTE-Vnet framework faces the following major challenges:

- There is no direct communication between vehicles.
- Single-hop communication takes place between vehicles and base station (FAP or MBS).
- High mobility of vehicles.

The following information is available about each vehicle– the Signal to Interference & Noise Ratio (SINR), Received Signal Strength (RSS), PDR, and limited number of transmitted packets. Considering the challenges and the information available about each vehicle, we propose using triangulation method for localizing the attackers. Triangulation method uses the distance of a known vehicle (legitimate vehicle) from FAP, an unknown vehicle (attacker vehicle), and the measured angle between the vehicles [34] to calculate the location of the attacker, as shown in Figure 6. As mentioned in Section III-A, we have considered a city scenario. Therefore, it is safe to assume that the number of vehicles is high, resulting in same relative speed between them. Although we have considered a city scenario but our work is also applicable in highway scenarios because even with high speed of vehicles, the relative speed between vehicles will remain the same. Due to low relative speed between associated vehicles and FAPs, we can use triangulation method for localization.

Let us assume that A is a possible attacker obtained from Algorithm 2 at a FAP F, as shown in Figure 6. F calculates the distance between itself to A and legitimate vehicle V using RSS. This distance can be calculated in the same way as given in [43]. Let γ be the reference distance between transmitter A and receiver F. The received signal power ψ can be calculated as:

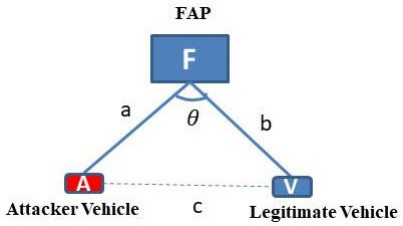


Fig. 6: Triangulation method for localization

$$\psi\,(in\,dBm) = u - R(\gamma) \tag{10}$$

where, $R(\gamma)$ denotes reference power for γ and u as transmit power of A. Now, using Equation (10) the RSS value is calculated as:

$$RSS(in\,dBm) = \psi - 10h\log a \tag{11}$$

where, h represents path loss exponent and depends on specific propagation environment. It measures the rate at which the RSS decreases with distance. a is the distance from A to F. Maximum RSS value RSS_{max} can be obtained by computing the maximum of RSS values. Using RSS_{max} value, distance a is calculated as:

$$a(in\,meters) = 10^{\left(\frac{\psi - RSS_{max}}{10}\right)\times h} \tag{12}$$

The distance between V and F denoted as b is also calculated in a similar way. After calculation of distance, the angle θ between a and b is calculated by FAP using following formula:

$$\theta = \tan^{-1}\left(\pm\frac{\Delta_1 - \Delta_2}{1 + \Delta_1\Delta_2}\right) \tag{13}$$

where Δ_1 and Δ_2 are the slope of lines F-A and F-V. To localize the attacker, FAP needs to find the distance between A and V. As the value of a, b and θ are known, the distance between A and V, which is denoted as c, can be calculated as follows:

$$c = \sqrt[2]{a^2 + b^2 - 2abCos\theta} \tag{14}$$

After calculating the distance, FAP calculates the common meet point using the distance c and a. Then, it calculates the coordinates for a common point, and based on the coordinates it localizes the attacker.

C. MR-based Localization

In our previous technique, attack detection and localization were done at the FAP. As in [30], the authors have suggested that attack detection can be done in the network gateway so we can also perform the detection and localization in FAP. The beauty of this localization technique is the localization of both intentional and unintentional attackers.

We have used ML-based classification algorithms to detect the attack and to localize the attacker. We propose a four-step localization approach as shown in Figure 4. We know that we have only the following attributes to perform any of the tasks in our scenario – PDR, delay, throughput, time intervals, measurement report (Vehicle ID, Cell ID, RSRP, RSRQ, connection timestamp). Figure 7 depicts the localization procedure which takes place on FAP. The very 1st step is the PDR check. In this step, FAP calculates the

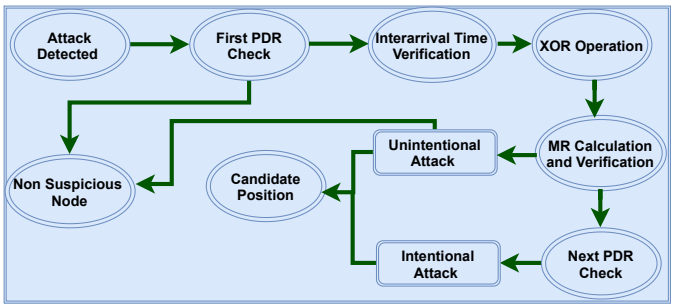


Fig. 7: MR-based localization procedure

PDR value of each vehicle in each run and compares it with the predefined threshold value. If it is greater than the threshold, FAP performs the 2nd step which is inter-arrival time verification, otherwise, the particular vehicle is flagged as legitimate.

Algorithm 4: MR-based Localization of Attacker

Input: PDR_val, IA_time : $InterArrivalTime, V\_List\_MR\_values$
Output: $ALoc$: Attacker location

1 $Unint\_Attacker = False$;
2 $Int\_Attacker = False$;
3 $Un\_Suspicious = False$;
4 **for** $(i = 1; i <= T; i++)$ **do**
5 **for** $(f = 1; f <= fap; f++)$ **do**
6 $\bar{V}$ = $vehicles\_associated\_to\_fap(f)$;
7 **for** $(v = 1; v <= \bar{V}; v++)$ **do**
8 **if** $(PDR\_val > \vartheta)$ **then**
9 **if** $(IA\_time > normal\_time)$ **then**
10 **for** $(i = 0; i < c; i++$ **do**
11 $v.XOR\_val =$ $v.XOR\_operation)$;
12 **if** $(v.XOR\_val == 0)$ **then**
13 Calculate $v.MR\_value$;
14 Verify $v.MR\_value$ with $V\_List\_MR\_values$;
15 **if** $(v.MR\_values ==$ $V\_List\_MR\_values)$ **then**
16 **if** $(PDR\_val > \vartheta)$ **then**
17 $v.Int\_Attacker =$ $True$;
18 $v.ALoc$ = $v.location$;
19 $v.Unint\_Attacker = True$;
20 $v.ALoc = v.location$;
21 $v.Un\_Suspicious = True$;

In the inter-arrival time verification, the inter-arrival time verification takes place. Since in our attack model attacker sends huge amount of packets, the inter-arrival time of packets is much less than legitimate vehicles. For example, the legitimate vehicle sends the packet in the time interval of 5ms, but the attacker sends the packet in the interval of 1ms. FAP checks the frequency of the time interval for each run, if it is more than the legitimate one then it performs the 3rd step which is the XOR operation. In the XOR operation step, if the above two conditions hold, FAP performs the XOR operation continuously for each run for a particular vehicle ID. If the output of the XOR operation comes as 0 then control shifts to the next step (MR verification).

In the MR verification step, 1st, we calculate the two values (RSRP, RSRQ). After all three steps as depicted in Figure 7, the FAP calculates the particular vehicle position by using the MR value. MR value contains Vehicle ID, Cell ID, RSRP, RSRQ, and connection time stamp. We can calculate RSRP using RSS:

$$RSRP(dBm) = RSS(dBm) - 10log(12 \times N)$$

where N is the number of resource blocks [11]. Using RSS and RSRP, RSRQ can be calculated [44]:

$$RSRQ = \frac{RSRP}{RSS/N}$$

After calculation of MR value, we check the suspicious vehicles' MR values with all other vehicles' MR values that are associated with the FAP. If MR values match, we check the PDR value of suspicious vehicles. If PDR is greater than the threshold value, we flag it as an intentional attack, otherwise it is flagged as a non-suspicious vehicle. If the MR value is distinct from others, we flag it as unintentional attacker. For finding the candidate position, we use RSRP value and localize the unintentional as well as intentional attacker [33] [43].

Algorithm 4 outlines the sequence of steps to localize the attacker(s). In this algorithm, at each time step (10 ms), FAPs find their associated vehicles (line 7). Then FAP compares the PDR_val (line 8) and the IA_time (line 9). After that XOR operation of corresponding VID is performed (lines 10-11). Based on XOR value MR_value is calculated and MR_value is verified with $V\_List\_MR\_values$ (lines 12-14). If the condition (line 15) holds good then PDR_val is checked and the vehicle is flagged as Int_Attack otherwise $Unint\_Attack$ and localize them by using RSRP values (lines 16-20).

VI. Performance Analysis

In this section, we analyze the average delay of a packet being transmitted from a vehicle to its associated base station (FAP or MBS). As mentioned in Section I, at a given time, a fixed number of RBs are available for communication and FAPs are responsible for allocating them to vehicles. Now, if we consider our attack scenarios (both Case I and Case II), we can conclude that given a fixed amount of available resources, when number of data packets increase, contention for availing the limited amount of resources will also increases, leading to high delay. For our analysis, we trace each packet from its generation until its delivery to the associated FAP. As mentioned earlier, in the uplink, vehicles generate packets which are stored in their buffers until RB is allocated for transmission. The maximum number of vehicles that a FAP can serve is equal to the number of available RBs. On

assignment of a RB, a vehicle can transfer its packets to the associated FAP. Let us assume that there are x vehicles associated to a FAP, where x includes both legitimate and malicious vehicles. Also, there are only m RBs available such that $m < G.x$, i.e., number of available RBs is less than the number of packets generated by the associated vehicles. Hence, for transmission from vehicles to FAP, vehicles have to contend for RBs. Let $E[W_v]$ be the expected waiting time incurred by packets generated at vehicles to reach FAPs. The buffers in vehicles associated to a FAP can now be modeled as an $M/M/m$ queue where m RBs act as servers. Let the packet arrival rate be λ and the service time be exponentially distributed with mean $1/\mu$. Thus, the expected waiting time for packets in vehicles is given by,

$$E[W_v] = E[P]/\lambda \tag{15}$$

where, $E[P]$ is the expected number of packets in the queue including the ones in service. $E[P]$ is given by,

$$E[P] = \frac{\rho\eta}{1-\rho} \tag{16}$$

Here, ρ is the server utilization factor and η is defined as the probability of queuing i.e., the probability that there more packets waiting in the queue to be served. The server utilization factor for $M/M/m$ queue is given by

$$\rho = \frac{\lambda}{m\mu} \tag{17}$$

η is given by

$$\eta = \frac{(m\rho)^m}{m!(1-\rho)}J \tag{18}$$

J represents the probability that all servers are idle and there are no packets in the system to serve.

$$J = \left[1 + \frac{(m\rho)^m}{m!(1-\rho)} + \sum_{i=1}^{m-1}\frac{(m\rho)^i}{i!}\right]^{-1} \tag{19}$$

We analyse our detection and localization algorithm in the context of time, and found that the time complexity is $O(fV)$ where V is denoted as number of vehicles and f is denoted as number of FAPs.

VII. Experiments

In this section, we first describe our experimental setup. Next, we present the experimental results.

A. Experimental Setup

The simulation scenario consists of a city scenario with road of length 10 km with multiple lanes and intersections. The traffic is bidirectional. We are comparing our approach with [11], where the authors have considered only MBS and detection takes place at MEC server. To simulate our scenario, we have used a discrete event simulator based on Java. We have used Simulation of Urban MObility (SUMO) to generate vehicular movements. The simulation results have been averaged over 100 runs. Parameters considered for simulation are given in Table I. The choice of parameters is consistent with prior studies [3], [9]. For ML-based classification algorithms we have considered 300077 data instances generated by our java based simulator.

TABLE I: Parameters used in simulation [3], [9]

Parameter	Value
Number of Vehicles	300
Number of FAPs	50
Packet Size	160 bytes
Packet generation rate (legitimate vehicle)	1 Packet/5ms
Packet generation rate (attacker)	1 Packet/1ms
MBS Transmission Range	10 km
FAP Transmission Range	50 m
MBS Transmission Power	43 dBm
FAP Transmission Power	23 dBm
Vehicle Speed	30-80 Km/hr
Path Loss Coefficient	FAP:3.5 MBS:2.5

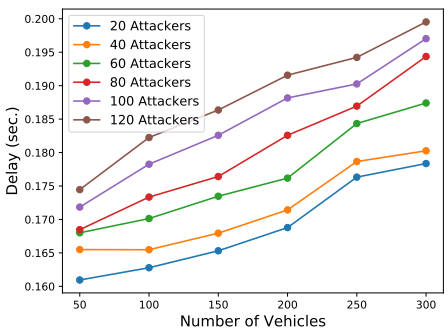
B. Delay and PDR Results

Figure 8a represents the variation in average delay with increase in number of vehicles in the scenario. Vehicles include both legitimate and malicious vehicles. As expected, with increase in number of vehicles, the average delay increases. This is because, more number of vehicles will contend for the fixed amount of available resources. With the inclusion of malicious vehicles, more packets will be generated, which will eat up more resources resulting in higher contention, leading to higher delay. As we can see, higher delay can be observed when the number of malicious vehicles is more. Similarly, in Figure 8b, it can be observed that when the number of attackers increases, the average PDR percentage decreases. This is because, with more attackers, contention for available RBs increases, leading to lower PDR.

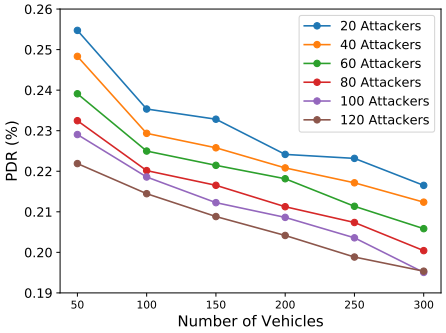
C. Detection and Localization Time

Time and accuracy play a crucial role in vehicular networks because inaccurate information and delay may be critical to life. Hence, we have considered ML model for attack detection. Based on our data set, Decision Tree performs better in less time and with higher accuracy compared to existing ML-based detection techniques [30], as shown in Table II, IV and Table III.Subsequently, we have compared our proposed algorithms with some existing approaches with respect to time.

Figure 9 depicts the comparison of our detection technique with the MEC-based technique with respect to detection and localization time. In [11], the attack detection is done at MBS and localization of attacker is done at MEC server [11]. This takes more time and makes detection more complex. Figure 9a represents the attack detection time of DPC-based detection, PDR-based detection and MEC-based detection technique. As expected, with the increase in number of attackers, the attack detection time decreases. We can see that the detection time taken in DPC-based detection is much less than that of PDR-based and MEC-based detection technique. This is because, in DPC-based technique, the detection is done at FAPs by calculating DPC values, whereas, in PDR-based detection technique, PDR calculated at each vehicle is transmitted to FAP which uses it for detection of attack. In the MEC-based detection technique, in [11] authors have used MBS to perform detection. As the MBS is far from the vehicles and traffic load

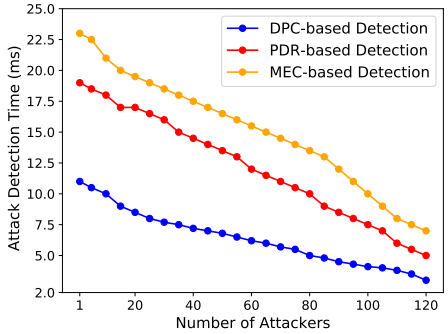


(a) Impact on average packet delay

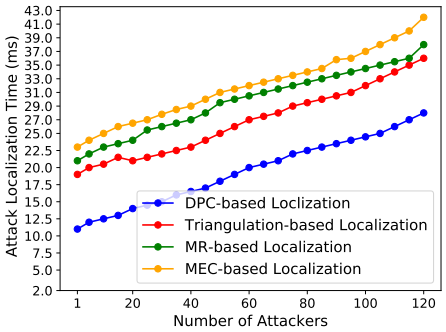


(b) Impact on PDR percentage

Fig. 8: Impact of DoS attacks with multiple attackers

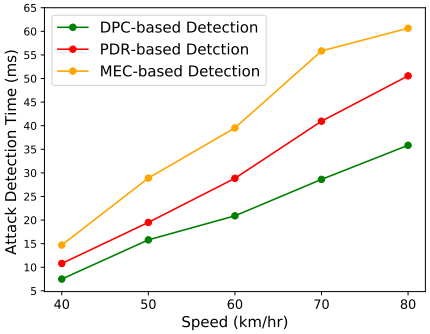


(a) Detection time with attackers

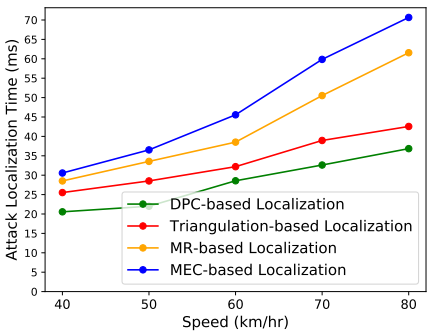


(b) Localization time with attackers

Fig. 9: Comparison of proposed technique with MEC-based approach [11]

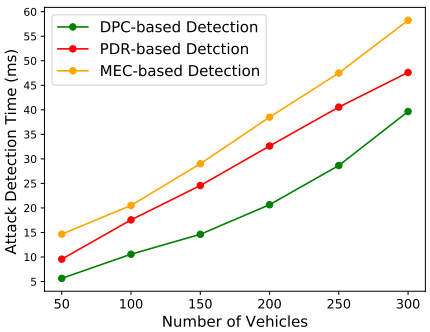


(a) Attack detection time

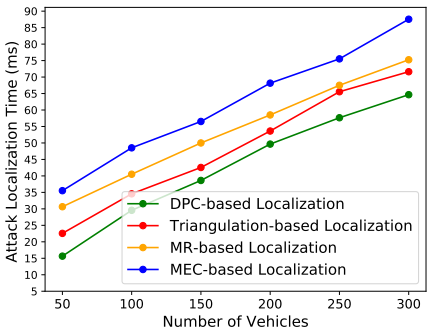


(b) Attack localization time

Fig. 10: Impact of speed in localization and detection time



(a) Performance of detection algorithms



(b) Performance of localization algorithms

Fig. 11: Time required for detection and localization with respect to number of vehicles

at MBS is higher than FAP, it takes more time to detect than DPC and PDR-based detection techniques.

Figure 9b represents the time for localizing the attacker. It can be seen that DPC-based localization technique takes less time for localizing than both PDR-based localization technique and MEC-based localizing technique. This is because, in DPC-based localization, FAP localizes the attacker using its vehicle Id, whereas, in PDR-based localization, FAP uses triangulation method to localize the attacker, and in the MEC-based localization technique neural network based method is used for localization.

Figure 10a and 10b represent time required for attack detection and localization with respect to speed. It can be observed that DPC and PDR perform better than MEC-based detection. This is because, MEC performs complex computations to detect and localize the attacker which makes the detection and localization procedure time consuming and the attacker may move away by then. Figure 11a and 11b represent the time required for detection and localization with respect to number of vehicles. It can be clearly seen that, as the number of vehicles increases, detection and localization time also increases. Because of the simple mathematical procedure in DPC and PDR-based detection and localization procedures, they outperform MEC and MR-based techniques.

D. Detection and Localization Accuracy

Figure 12 shows the performance comparison for detection and localization algorithms with MEC algorithm [11]. Figure 12a represents the variation in the percentage of attackers detected with increases in the number of attackers. As the vehicle Id is known in the solution of PDR-based detection, the attacker is easily detected. Hence, it performs better than DPC-based detection and the MEC-based detection technique. In Figure 12b, it can be observed, the MR-based localization technique performs better than others. The reason behind this is the multi step filtering technique of MR-based localization. Also, we can observe that ML-based detection (Decision Tree) outperforms all the approaches with better accuracy. We pre-labelled our data set and divided the target value (result) as categorical data (attack and legitimate) hence DT performs better with our data set.

TABLE II: Comparison with [30]

ML algorithm	Detection Accuracy		F1 score		Detection Time	
	[30]	Our Work	[30]	Our Work	[30]	Our Work
KNN	96.6%	99.88%	0.966	0.998	14(s)	12.08(s)
SVM	98.01%	99.12%	0.978	0.991	489.49(s)	380.26(s)
DT	99.72%	100%	0.998	1	13.7(s)	10.2(s)
RF	98.37%	98.61%	0.983	0.986	48.6(s)	48.08(s)
ET	93.43%	97.9%	0.934	0.979	62.6(s)	58.78(s)
XGBoost	99.78%	97.2%	0.997	0.972	96.2(s)	96.50(s)
Stacking	99.86%	99.9%	0.998	0.999	190.5(s)	109.71(s)
FS XGBoost	99.7%	99%	0.996	0.99	45(s)	38.07(s)
FS Stacking	99.82%	99%	0.997	0.99	74.8(s)	47.36(s)

In Table II, we have compared our technique with the work in [30] and got better accuracy and F1 score with less time. Also in Table III, we varied the number of attackers (5 to 120) and got decent accuracy and F1 score in less time. Hence, ML-based detection algorithms give better accuracy with any number of attackers. It can be observed that the DT algorithm performs better in our scenario with less time. We have compared our work with [31] and observed that our approach performs better as shown in Table IV. As our data-set is pre-labelled and categorised into two categories (attack and legitimate) and also we use specific features, therefore, DT took less time with better accuracy.

TABLE III: Performance with varying attackers

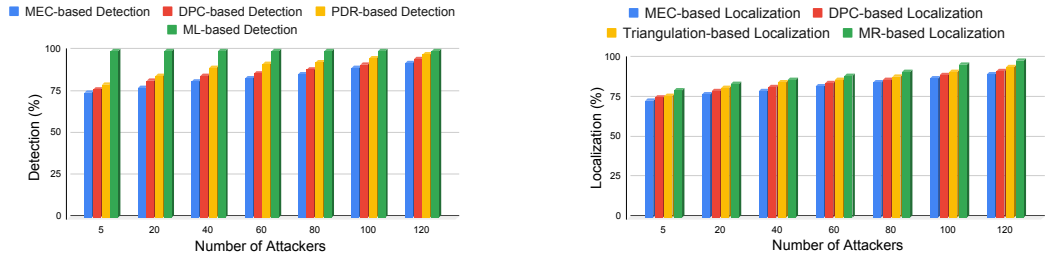
ML algorithm	Detection Accuracy	F1 score	Detection Time
KNN	96.8	0.968	12.08(s)
SVM	98.76	0.987	380.26(s)
DT	99.71	0.997	10.2(s)
RF	98.52	0.985	48.08(s)
ET	92.66	0.926	58.78(s)
XGBoost	99.15	0.991	96.50(s)
Stacking	99.97	0.999	109.71(s)
FS XGBoost	99.95	0.999	38.07(s)
FS Stacking	99.98	0.999	47.36(s)

Figure 14a depicts the comparison of MR-based localization technique and context-based localization technique [32] with respect to accuracy and error. In the figure, it can be seen that our localization technique outperforms the context-based technique because in our localization technique we have filtered the vehicle information in four ways which makes our model less erroneous. We also compare our localization techniques (MR-based and Triangulation-based) with the MEC-based technique [11] with respect to their localization percentage.

In Figure 15a and 15b we can observe that DPC, PDR and triangulation algorithms perform better than others with variations in vehicle speed because these algorithms take less time than others to detect and localize the attacker. Also, we can observe that variation of speed does not impact much in ML-based algorithm hence ML-based detection approach outperforms others.

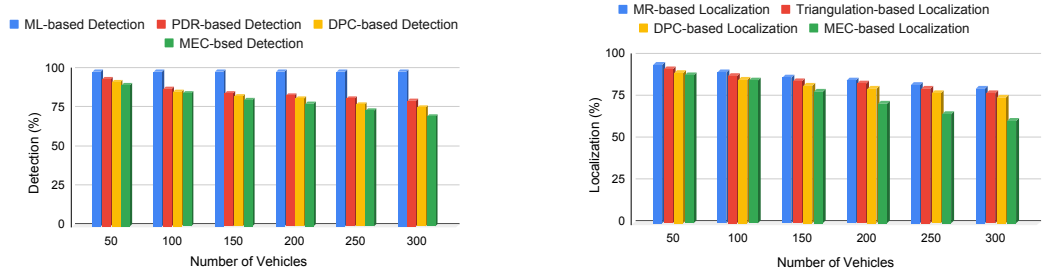
We have also checked the effectiveness of our proposed algorithms in the context of a distributed denial-of-service attack (DDos) which is shown in Figure 16. In Figure 16a we can observe that ML-based detection outperforms all the approaches with better accuracy because we have filtered the pre-labelled data-set before classification. As DPC and PDR-based detection algorithms execute locally and take less time, they outperform the MEC-based detection approach. In Figure 16b, we can observe that MR-based localization performs better than others because of four-step (PDR check, inter-arrival time verification, XOR operation, MR verification) filtering approach. Also, we can observe that PDR and DPC-based localization approach performs decently because they localize the attacker locally (at FAP itself).

We have also considered the effectiveness of our proposed algorithm with the variation of vehicles as shown in Figure 13. It can be clearly observed that the detection and localization accuracy decreases while number of vehicles increases. In Figure 13a we can observe that ML-based detection techniques perform better than all other detection techniques because the variation in number of vehicles does not put much impact on ML-based detection techniques. In Figure 13b it can be seen that the MR-based localization technique outperforms others because of four-step filtering techniques, making it more accurate.



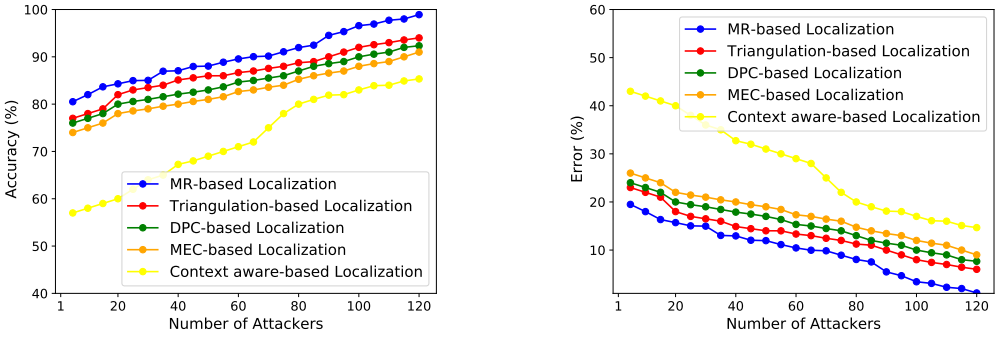
(a) Percentage Comparison of detection algorithms (b) Percentage Comparison of localization algorithms

Fig. 12: Percentage Comparison of MEC-based detection and localization technique [11] with proposed detection and localization techniques



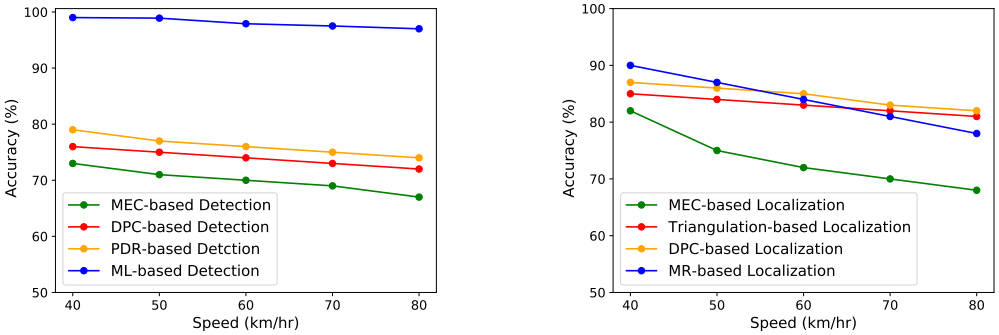
(a) Performance of detection algorithms (b) Performance of localization algorithms

Fig. 13: Accuracy of detection and localization with respect to number of vehicles



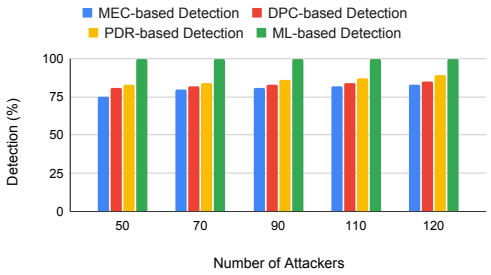
(a) Localization Accuracy (b) Localization Error

Fig. 14: Accuracy and Error comparison of our localization techniques with context aware based localization [32] and MEC-based localization [11]

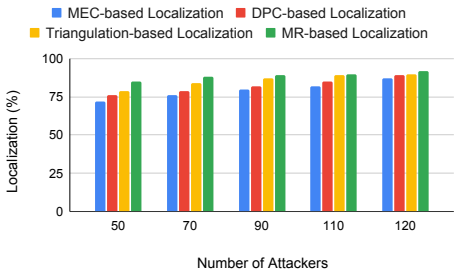


(a) Detection accuracy (b) Localization accuracy

Fig. 15: Impact of speed in localization and detection algorithms



(a) Performance of detection algorithms



(b) Performance of localization algorithms

Fig. 16: Accuracy of detection and localization in DDoS scenario

TABLE IV: Comparison with [31]

Model	**Precision**		**F1 score**	
	[31]	Our Work	[31]	Our Work
KNN	0.861	0.98	0.841	0.969
Logistic regression	0.788	0.84	0.687	0.81
Decision tree classifier	0.978	0.992	0.941	0.993
Bagging	0.99	0.999	0.98	0.992
Random Forest	0.99	0.999	0.98	0.992

VIII. Conclusions & Future Work

In this work, we have developed an efficient framework for real-time detection and localization of Denial-of-Service (DoS) attacks in LTE-based vehicular networks. We have explored DoS attack detection using average packet delivery ratio as well as data packet counter values. We have utilized machine learning based algorithms to detect the attacks with high accuracy. We have used triangulation method to localize the attacker which uses fake vehicle identification to perform the attack. Also, we have introduced DPC-based localization to localize the attacker which introduces an attack unintentionally. We have developed an MR-based localization technique that can localize both intentional as well as unintentional attacks irrespective of the number of attackers in the scenario. We have also performed a detailed analysis of average packet delay using the M/M/m queuing model. Our experimental results demonstrate that our approach can significantly outperform state-of-the-art methods. In future, we plan to explore various mitigation techniques for both type of attacks.

Acknowledgments

This research work was supported in part by the Department of Science and Technology (DST), Government of India vide project grant ECR/2018/000917. This work was also partially supported by the U.S. National Science Foundation (NSF) grant SaTC-1936040.

References

[1] M. R. Dey *et al.*, "Real-Time Detection and Localization of Denial-of-Service Attacks in Heterogeneous Vehicular Networks," in *Design Automation & Test in Europe (DATE)*, 2021, pp. 1434–1439.

[2] G. Araniti *et al.*, "LTE for vehicular networking: a survey," *IEEE Communications Magazine*, vol. 51, no. 5, pp. 148–157, 2013.

[3] M. Patra *et al.*, "Improving delay and energy efficiency of vehicular networks using mobile femto access points," *IEEE Transactions on Vehicular Technology*, vol. 66, no. 2, pp. 1496–1505, 2017.

[4] R. Piqueras Jover, *"Security attacks against the availability of LTE mobility networks: Overview and research directions"*. WPMC, 2013.

[5] M. Lichtman *et al.*, *"Detection and mitigation of uplink control channel jamming in LTE"*. IEEE Military Communications Conference, 2014.

[6] S. Mukherjee *et al.*, *"Practical DoS Attacks on Embedded Networks in Commercial Vehicles"*. Springer International Publishing, 2016.

[7] M. Tayyab *et al.*, *"A survey on handover management: From LTE to NR"*. IEEE Access, 2019, vol. 7.

[8] J. Chen *et al.*, "Femtocells–architecture & network aspects," *Qualcomm*, vol. 28, pp. 1–7, 2010.

[9] S. Bhattarai *et al.*, *"On simulation studies of cyber attacks against LTE networks"*. ICCCN, 2014.

[10] F. S. d. Lima Filho *et al.*, "Smart detection: an online approach for dos/ddos attack detection using machine learning," *Security and Communication Networks*, vol. 2019, 2019.

[11] Y. Li *et al.*, *"An MEC-based DoS attack detection mechanism for C-V2X Networks"*. GLOBECOM, 2018.

[12] M. Zhu *et al.*, *"Security Analysis of LTE-V2X and A Platooning Case Study"*. IEEE INFOCOM, 2020.

[13] M. Ambrosin *et al.*, *"A roaming-based denial of service attack on LTE networks: poster"*. Association for Computing Machinery, 2017.

[14] G. Velikic, *"Intelligent Cars: Are We There Yet?"*. IEEE Consumer Electronics Magazine, 2020, vol. 9.4.

[15] D. Baek *et al.*, "Build Your Own EV: A Rapid Energy-Aware Synthesis of Electric Vehicles," *IEEE Design Test*, pp. 40–47, 2018.

[16] H. Thapliyal *et al.*, *"Emerging Paradigms in Vehicular Cybersecurity"*. IEEE Consumer Electronics Magazine, 2019, vol. 8, no. 6.

[17] R. Ghannam *et al.*, *"User-targeted Denial-of-Service Attacks in LTE Mobile Networks"*. WiMob. IEEE., 2018.

[18] N. Ruan *et al.*, *"A Traffic Based Lightweight Attack Detection Scheme for VoLTE.* IEEE GLOBECOM, 2016.

[19] Z. Tian, *et al.*, "Vcash: a novel reputation framework for identifying denial of traffic service in internet of connected vehicles," *IEEE Internet of Things Journal*, vol. 7, no. 5, pp. 3901–3909, 2019.

[20] R. a. o. Kolandaisamy, "A multivariant stream analysis approach to detect and mitigate ddos attacks in vehicular ad hoc networks," *Wireless communications and mobile computing*, vol. 2018, 2018.

[21] S. Su, *et al.*, "A reputation management scheme for efficient malicious vehicle identification over 5g networks," *IEEE Wireless Communications*, vol. 27, no. 3, pp. 46–52, 2020.

[22] Z. Tian, *et al.*, "Evaluating reputation management schemes of internet of vehicles based on evolutionary game theory," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 6, pp. 5971–5980, 2019.

[23] H. Sedjelmaci, *et al.*, "Predict and prevent from misbehaving intruders in heterogeneous vehicular networks," *Vehicular Communications*, vol. 10, pp. 74–83, 2017.

[24] N. Hu, *et al.*, "Building agile and resilient uav networks based on sdn and blockchain," *IEEE Network*, vol. 35, no. 1, pp. 57–63, 2021.

[25] N. Hu *et al.*, "A multiple-kernel clustering based intrusion detection scheme for 5g and iot networks," *International Journal of Machine Learning and Cybernetics*, vol. 12, no. 11, pp. 3129–3144, 2021.

[26] Z. Zhang *et al.*, "A many-objective optimization based intelligent intrusion detection algorithm for enhancing security of vehicular networks in 6g," *IEEE Transactions on Vehicular Technology*, vol. 70, no. 6, pp. 5234–5243, 2021.

[27] A. Verma, R. Saha, G. Kumar, and T.-h. Kim, "The security perspectives of vehicular networks: A taxonomical analysis of attacks and solutions," *Applied Sciences*, vol. 11, no. 10, p. 4682, 2021.

[28] S. Charles, *et al.*, "Real-time detection and localization of DoS attacks in NoC based SoCs," pp. 1160–1165, 2019.

[29] S. Charles, *et al.*, "Real-time detection and localization of distributed DoS attacks in NoC based SoCs," *IEEE TCAD*, 2020.

[30] L. Yang, *et al.*, "Tree-based intelligent intrusion detection system in internet of vehicles," in *2019 IEEE Global Communications Conference (GLOBECOM)*, 2019, pp. 1–6.

[31] S. Gyawali *et al.*, "Machine learning and reputation based misbehavior detection in vehicular communication networks," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 8, pp. 8871–8885, 2020.

[32] Y. Zhang *et al.*, "Context-aware telco outdoor localization," *IEEE Transactions on Mobile Computing*, pp. 1–1, 2020.

[33] L. Ni, Y. Wang *et al.*, "Accurate localization using lte signaling data," in *2017 IEEE International Conference on Computer and Information Technology (CIT)*, 2017, pp. 268–273.

[34] C. Savarese *et al.*, *"Location in distributed ad-hoc wireless sensor networks"*. ICASSP, 2001.

[35] M. Lichtman *et al.*, *"LTE/LTE-A jamming, spoofing, and sniffing: threat assessment and mitigation"*. IEEE Communications, 2016.

[36] P. A. Lopez *et al.*, "Microscopic traffic simulation using sumo," in *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*. IEEE, 2018, pp. 2575–2582.

[37] T. Chen *et al.*, "Xgboost: extreme gradient boosting," *R package version 0.4-2*, vol. 1, no. 4, pp. 1–4, 2015.

[38] M. J. Kearns, *The computational complexity of machine learning*. MIT press, 1990.

[39] S. o. Zhang, "Efficient knn classification with different numbers of nearest neighbors," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 29, no. 5, pp. 1774–1785, 2018.

[40] M. Pal *et al.*, "Feature selection for classification of hyperspectral data by svm," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 48, no. 5, pp. 2297–2307, 2010.

[41] R. C. Barros *et al.*, "A survey of evolutionary algorithms for decision-tree induction," *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 42, no. 3, pp. 291–312, 2012.

[42] P. Geurts *et al.*, "Extremely randomized trees," *Machine learning*, vol. 63, no. 1, pp. 3–42, 2006.

[43] ""ukessays. distance measurement using rssi method in wsn."." November 2018. [Accessed 25 September 2021]. [Online]. Available: https://www.ukessays.com/essays/computer-science/distance-measurement-using-rssi-method-8607.php?vref=1.

[44] F. Afroz *et al.*, ""SINR, RSRP, RSSI and rsrq measurements in long term evolution networks"," *International Journal of Wireless & Mobile Networks*, 2015.

Meenu Rani Dey received the B.Tech. degree in Information Technology from the Chhattisgarh Swami Vivekananda Technical University, Chhattisgarh, India, and the M.Tech. degree in Computer Science from the International Institute of Information Technology Bhubaneswar, Odisa, India. She is currently working towards her Ph.D. degree in Indian Institute of Technology Guwahati, Assam, India. Her research interests include Intrusion Detection System in Cellular based Vehicular Networks.

Moumita Patra received the Ph.D. degree from the Department of Computer Science and Engineering, Indian Institute of Technology Madras, Chennai, India. She is currently an Assistant Professor with the Department of Computer Science and Engineering, Indian Institute of Technology Guwahati, Assam, India. Her research interests include the areas of vehicular ad hoc networks, Internet of Vehicles, Internet of Things, and 5G communication networks.

Prabhat Mishra is a Professor in the Department of Computer and Information Science and Engineering at the University of Florida. He received his Ph.D. in Computer Science from the University of California at Irvine in 2004. His research interests include embedded systems, hardware security, energy-aware computing, system-on-chip validation, machine learning, and quantum computing. He currently serves as an Associate Editor of IEEE Transactions on VLSI Systems and ACM Transactions on Embedded Computing Systems. He is an IEEE Fellow, an AAAS Fellow, and an ACM Distinguished Scientist.