Hindawi
Security and Communication Networks
Volume 2021, Article ID 9959310, 16 pages
https://doi.org/10.1155/2021/9959310

Research Article

RF Jamming Classification Using Relative Speed Estimation in Vehicular Wireless Networks

Dimitrios Kosmanos,[1] **Dimitrios Karagiannis**,[1] **Antonios Argyriou**,[1] **Spyros Lalis**,[1] **and Leandros Maglaras**[2]

[1]*Department of Electrical and Computer Engineering, University of Thessaly, Volos, Greece*
[2]*Department of Computing Technology, De Montfort University, Leicester, UK*

Correspondence should be addressed to Leandros Maglaras; leandrosmag@gmail.com

Received 10 March 2021; Revised 24 May 2021; Accepted 11 August 2021; Published 26 August 2021

Academic Editor: Emanuele Maiorana

Wireless communications are vulnerable against radio frequency (RF) interference which might be caused either intentionally or unintentionally. A particular subset of wireless networks, Vehicular Ad-hoc NETworks (VANET), which incorporate a series of safety-critical applications, may be a potential target of RF jamming with detrimental safety effects. To ensure secure communications between entities and in order to make the network robust against this type of attacks, an accurate detection scheme must be adopted. In this paper, we introduce a detection scheme that is based on supervised learning. The k-nearest neighbors (KNN) and random forest (RaFo) methods are used, including features, among which one is the metric of the variations of relative speed (VRS) between the jammer and the receiver. VRS is estimated from the combined value of the useful and the jamming signal at the receiver. The KNN-VRS and RaFo-VRS classification algorithms are able to detect various cases of denial-of-service (DoS) RF jamming attacks and differentiate those attacks from cases of interference with very high accuracy.

1. Introduction

A prevalent prediction is that fully autonomous vehicles, capable of self-navigating in unpredictable real-world environments with little human feedback, will flood the global market by 2025 [1]. Autonomous vehicle control imposes very strict security requirements for the wireless communication channels [2] which are used by a fleet of vehicles [3]. Specifically, the connected vehicles use the connected adapted cruise control (CACC) technology, in which the following vehicles learn the lead vehicle's dynamics via intervehicle communication and through them they determine their movement. However, an RF jamming attack can overload the wireless medium leading to large packet losses. So, the platoons of vehicles can become unsafe and collisions are possible. Moreover, with the intelligent vehicle grid technology, each vehicle becomes a sensor platform absorbing information from the environment or from other vehicles (also called Internet of Vehicles (IoV)). Vehicles also feed each other or infrastructure for assisting in safe navigation and traffic management.

Wireless communications, however, are vulnerable against a wide range of attacks. An attack that is particularly hard to detect in every wireless network is the RF jamming attack [4]. An RF jamming attack reduces the availability of the wireless medium making the successful detection of a jamming attack may be obstructed by several conditions that might occur in an urban environment, such as unintentional interference caused by other wireless nodes, poor link conditions, etc. In a VANET, RF jamming attack detection is even more challenging due to the constant and rapid changes in topology and the high mobility of the vehicles. Detection becomes even harder with the presence of a variety of jammers and unintentional interference sources in the same area. Jamming may affect the communication between vehicles (V2V communication) or the communication between vehicles and roadside units, namely, RSUs (V2R communication).

Over the last few years, there have been several experimental approaches for jamming detection [4–7], some of which suggest the use of machine learning techniques [6, 8]. However, only Puñal et al. [6] examine closely the adoption of machine learning techniques for jamming detection. None of the related works that focused on machine learning-based schemes has investigated the use of the speed of the involved vehicles as an extrafeature for classifying jamming attacks in VANETS. In this work, we show that this is a critical feature and, more specifically, the variations of relative speed (VRS) metric. VRS is used as a new feature for jamming classification in realistic scenarios with a minimum number of assumptions leading to increases in classification accuracy.

The proposed VRS metric, extracted at the application layer [9, 10], is combined with classic physical layer metrics leading to a cross-layer classification scheme. The intuition behind the use of the VRS is the following. In the general case, jamming reduces the receiver signal-to-interference-and-noise ratio (SINR), a problem that can be addressed with classic communication algorithms. However, SINR can be reduced due to unintentional interference, a problem very prevalent in dense populated areas where vehicles operate. Hence, for jamming detection, the actual reason behind the reduction in the SINR and the packet-delivery-ratio (PDR) has to be determined. The proposed VRS metric reveals the behavior of the jammer in relation to the receiver, specifically, the variations of its relative speed. An unintentional source of interference does not exhibit a specific pattern in its relative speed, allowing us thus to effectively differentiate the cases where a malicious intentional source of interference, namely, jammer, moves in ways that intend to disrupt communication. Our extensive results indicate that the proposed scheme can effectively differentiate the case of jamming attack from that of an interfering wireless source.

Accurate detection is also important because these two problems could be addressed differently; that is, in the case of interference, an interference cancellation (IC) scheme [11] is needed, while techniques such as spectral evasion (channel surfing and spatial retreats) scheme can be used for jamming attacks. With the use of more sophisticated techniques for alleviating the problem, the proposed scheme can be used as a first step of a process that aims at keeping alive the wireless communication between a transmitter and receiver, by detecting the exact cause of the wireless interference followed by appropriate actions related to the physical location of the nodes. Such actions could be the de-routing of the malicious vehicle from a specific area or the rerouting of legitimate vehicles towards different areas, free of any RF jamming or interference. Lack of such smart detection mechanisms could lead to incorrect de-routing decisions that may compromise the different objectives of applications that use intervehicle communications (IVC). For this reason, we tested both, a typical form of RF jamming, which is the continuous jamming, and a more smart reactive jamming.

The main contribution of this paper is the introduction of a proactive detection method against potential RF jamming attacks with fairly good detection results. This detection system is also able to differentiate interference from malicious RF jamming. Additionally, it is able to distinguish the unique characteristics of each attack especially when the proposed VRS metric is utilized among the other cross-layer features. The accuracy of the proposed detection method is about or over 90% under different supervised learning testing cases and under realistic values of the relative speed between the jammer and the receiver. This result is significantly improved as compared to other corresponding methods in the literature.

One key application area for our scheme is vehicle platoons in which an exterior or an interior attacker can cause significant instability in the CACC of the vehicle stream [12]. Our classifier could be used as a trustworthy indicator of a jamming attack; thus, the control model of the platoon could change from CACC to noncooperative adaptive cruise control (ACC), relying solely on radar techniques. This control mode switch can be considered as a mitigation technique to the impact of the attack [12]. For the evaluation of our approach, one interference-only scenario and two jamming attack scenarios have been designed and tested.

The rest of this paper is structured as follows. Section 2 provides an overview of related work in the domain of attack detection. Section 3 describes the topology and the channel model of our scenarios. Section 4.1 describes the methodology used for the estimation of the relative speed. Section 4.2 presents the proposed machine learning-based jamming detection system. Section 5 describes the simulation setup. Section 6 presents the experimental results and comparisons. Finally, Section 7 summarizes our findings and concludes our work.

2. Related Work

Several recent works have proposed machine learning-based techniques for attack detection in vehicular ad-hoc networks. Puñal et al. [6] used metrics that include the noise and channel busy ratio (CBR), packet delivery ratio (PDR), maximum inactive time (Max IT), and received signal strength (RSS), to detect attacks with machine learning techniques, and examined the cases of reactive and constant jammers.

Azogu et al. [5] proposed a new mechanism, called the hideaway strategy, according to which all nodes should remain silent while the network is under a jamming attack. Bißmeyer et al. [13] proposed a detection scheme that is based on the verification of vehicle movement data and on the assumption that a certain space will be occupied by only one vehicle at a certain time.

Malebary et al. [14] presented a two-phase jamming detection method that utilized metrics such as the RSS, the packet delivery/send ratio (PDSR), and the packet loss ratio (PLR), as well consistency, checks to distinguish a jamming from a no-jamming situation. In the first phase, which is the initialization phase, the values of the RSS, the packet delivery/send ratio (PDSR), and packet loss ratio (PLR) are calculated by the RSUs in a jammer-free network. Furthermore, a max value for the RSS is obtained for every PDSR value as well as two threshold values, equal to the

maximum PDSR and to the minimum PLR, respectively. In the second phase, when a PDSR value is lower than the defined threshold and a PLR value is higher than the respective threshold, a consistency check is conducted to determine whether the low PDSR value is consistent with the RSS value assigned in phase one, thus determining a jamming or no-jamming situation.

The authors in [15] proposed a data mining-based method for real-time detection of radio jamming DoS attacks in IEEE 802.11p V2V communications for platoon of vehicles. The state-of-the-art methods are compared with the proposed method which allows operating under the realistic assumption of random jitter accompanying every cooperative awareness message (CAM) transmission. However, only features from the network layer are utilized. Mokdad et al. [16, 17] proposed a scheme for detecting a jamming attack in vehicular ad-hoc networks that depends on the variations of the PDR. The approach is based on the premise in which only packets that originate from the sender are allowed through the cyclic redundancy check (CRC) and the PDR is equal to the ratio of these packets and the total number of packets received. Puñal et al., in [18], generated a set of jammers and implemented a variety of jamming scenarios, both indoor and outdoor, under different jamming behaviors (constant, reactive, and pilot jamming) in order to address the impact of an RF jammer in VANET communications.

Quyoom et al. [19] presented an RF jamming attack that consists of radio signals maliciously emitted to disrupt legitimate communications. This type of jamming is already known to be a big threat for any type of wireless network. With the rise in safety-critical vehicular wireless applications, this is likely to become a constraining issue for their deployment in the future.

RoselinMary et al. [20] proposed an approach that is based on the detection of malicious and irrelevant packets using the number of broadcast packets per second (frequency) and the velocity of the vehicle that the packets are sent from. This method calculates the frequency, e.g., the number of broadcast packets per second, and the velocity and then starts the detection algorithm. If the frequency and the velocity are both high and above a threshold, then the packets are malicious, whereas if they are between a low and a high threshold value, the packet is real.

A subcategory of related papers dealt with real-time medium access control- (MAC-) based jamming detection method to meet the requirements of safety applications in vehicular networks. These methods operate either under realistic assumption of random jitter accompanying every CAM transmission [21] or the decision of the detector (monitor) depends on the number of nearby vehicles and the number of successful transmissions and failed transmissions [22]. These detection methods can more accurately distinguish the causes of failed transmissions such as contention collisions, interference, and jamming attacks. In [23], the authors proposed a method for DoS attack detection in wireless sensor networks (WSNs). This method is based on the grouping of sensor nodes and the timestamp and the PDR calculated from one node to another one. However, all the above papers focused on simplistic jamming attacks such as "random jamming" or "ON-OFF jamming" without taking into account smarter jammers such as reactive jammer. Lastly, Mowla et al. [24] proposed a federated learning-based on-device jamming attack detection security architecture for flying ad-hoc network (FANET) using the RSSI and PDR features with a fairly good accuracy results in detecting the RF jamming attack. All the aforementioned jamming detection approaches used parameters only from the MAC or the physical layer for training and testing without exploiting upper layer features. Feng and Hua [25] proposed jamming detection schemes based on a variety of machine learning algorithms. They incorporate the information from the physical layer, the MAC layer, and the network layer (such as RSS, carrier sense time, noise, and PDR) for training and testing. Lastly, there are recent works in the literature that use either machine learning [26] or deep learning for a multistage jamming detection scheme in 5G networks: the cloud radio access network (C-RAN) [27]. However, these methods have not been tested on vehicular networks that have special features such as high-speed moving nodes. Only the authors, in [28], adopt a cross-layer approach incorporating also an application layer features for detecting and classifying different types of RF jamming attacks in VANETs. Specifically, the IDS that was proposed in [28] is able to differentiate a RF jamming attack from spoofing attacks in connected autonomous vehicles (CAVs).

Sharanya and Karthikeyan [29] proposed a support vector machine (SVM) algorithm with modified fading memory (MFM) for classifying legitimate and malicious nodes. The proposed classification scheme considers the following critical parameters to classify a node as malicious node, namely, power ratio, signal strength, packet delivery ratio, speed of node, number of packets generated, and transmission power. Their proposed system has two specific phases.

Lastly, Karagiannis and Argyriou [10] proposed an RF jamming attack detection scheme using unsupervised learning with clustering. The novelty of the above paper is that the relative speed metric is utilized between the jammer and the receiver, along with other parameters, in order to differentiate intentional from unintentional jamming as well as identify the unique characteristics of each jamming attack. However, this relative speed metric is assumed to be available without any form of estimation.

In all the previous works that were proposed, machine-learning based schemes, the estimated variations of the relative speed have not been considered as a classification feature. Our proposed system is the first one in the literature that uses the point-to-point RF communication in order to estimate the relative speed metric.

3. System Model

3.1. Topology. Our system topology is represented in Figure 1. In the left part (a), an interference scenario is presented, in which we assume that no jammer is present in the network. This scenario is important in order to be able to evaluate the efficiency of our method in differentiating jamming from interference. The vehicle travels, when, at

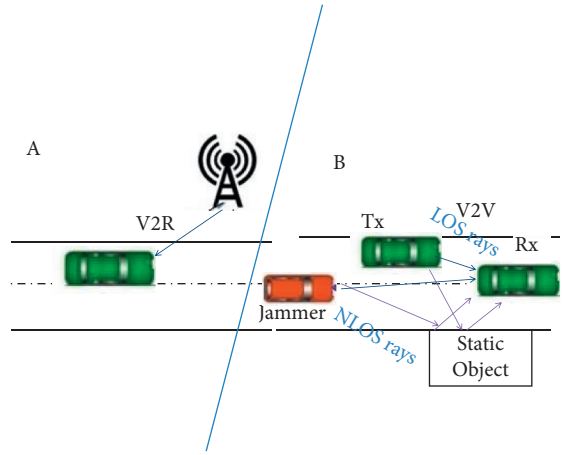

FIGURE 1: Topology. In the left part (a), the interference only situation is illustrated, while in the right part (b), we have the jamming situation with the presence of a moving jammer. Blue arrows represent the LOS V2V wireless communication. Purple arrows represent the NLOS components that are caused by reflections from a static object.

some point, it passes through an area with significant RF interference that is caused by a RSU. In the right part of this figure (b), a jamming situation is presented. The topology we adopt for this case involves a moving vehicle R_x, which serves as the target of the jammer, another vehicle T_x that is the transmitter of the useful signal, and the jamming vehicle J_x that tries to intervene in the communication between R_x and T_x. The travelling speed of R_x, namely, u_{R_x}, is equal to the travelling speed of T_x, namely, u_{T_x}. Moreover, we assume the presence of a static object in the area that causes multipath fading from reflections, as it is usually in urban environments. Upon spotting its target, the jammer begins following it and starts jamming either continuously or periodically (in order to stay undetected for as long as possible).

3.2. Rician Fading Model. In our work, we adopt the Rician fading model that is a channel model which includes path loss and also Rayleigh fading [30]. When a signal is transmitted, whether it is a useful signal or a jamming one, this model adds multipath fading in addition to thermal noise. It is assumed that a line-of-sight (LOS) ray and $N-1$ nonline-of-sight (NLOS) ray exist in the area. The combined baseband signal that the receiver receives from the jammer and the transmitter is

$$y(t) = \sum_{n=0}^{N-1} \left(\left(h_1(t,n) x_{\text{pilot}}[N-n] \sqrt{P_1} + h_2(t,n) s[N-n] \sqrt{P_2} \right) \right) + w(t), \tag{1}$$

where

$$h_1(t,n) = \text{Ray}_1(n) \left(+ \frac{1}{\text{dist}_1^2(t)} \right) e^{\left(j(2\pi/\lambda) \left(f_c + f_{d,\max} \cos\theta_1 \right) \tau_1(n) \right)} \delta\left(t - \tau_1(n) \right), \tag{2}$$

$$h_2(t,n) = \text{Ray}_2(n) \left(+ \frac{1}{\text{dist}_2^2(t)} \right) e^{\left(j(2\pi/\lambda) \left(f_c + f_{d,\max} \cos\theta_2 \right) \tau_2(n) \right)} \delta\left(t - \tau_2(n) \right), \tag{3}$$

where $h_1(t,n)$ and $h_2(t,n)$ are the Rician fading channel models between transmitter-receiver and jammer-receiver, respectively. This type of channel model includes path loss and also Rayleigh fading. $\text{Ray}_1(n)$ and $\text{Ray}_2(n)$ are complex Gaussian variables capturing the Rayleigh fading between transmitter-receiver and jammer-receiver, and $x_{\text{pilot}}[N-n]$ and $s[N-n]$ are the symbols that are transmitted from the transmitter and the jammer, respectively, for which the BPSK modulation is used. This modulation scheme is preferred because it achieves lower bit error rate providing a reliable communication between T_x and R_x. Moreover, this modulation scheme is the most robust in a high interference environment. In (2) and (3), f_c is the carrier frequency, $f_{d,\max}$ is the maximum Doppler

shift, P_1 and P_2 are the transmission power per symbol of the useful and of the jamming signal, respectively, and $w(t)$ is the channel noise at time instant t. The terms d_s and d_j correspond to the distance between the transmitter and the reflected object and between the jammer and the reflected object, respectively.

The terms $r_{1n}(t)$ and $r_{2n}(t)$ correspond to the distance between the transmitter and the receiver and between the jammer and the receiver. In (2) and (3), the travel distance of the LOS rays is equal to $\text{dist}_1(t) = r_{1n}(t)$ and $\text{dist}_2(t) = r_{2n}(t)$. On the contrary, the travel distance of the NLOS rays is $\text{dist}_1(t) = 2d_s - r_{1n}(t)$ and $\text{dist}_2(t) = 2d_j - r_{2n}(t)$, respectively. Moreover, θ_1 is the incidence angle of departure (AOD) between the vector of speed $\overrightarrow{u}_{T_x}$ and the signal vector of the transmitter, θ_2 is the incidence AOD between the vector of speed $\overrightarrow{u}_{J_x}$ and the signal vector of the jammer, ($\tau_1 = \text{dist}_1(t)/c, \tau_2 = \text{dist}_2(t)/c$) is the excess delay time for the transmitter and jammer signal ray (that may be caused due to ground reflection), and t is the current time instant. For the remainder of this paper, we will use the parameter γ_1 ($\gamma_1 = (\text{Ray}_1(n) + (1/r_{1n}^2(t)))$) as the transmitter-receiver complex amplitude associated with the LOS path and the parameter γ_2 ($\gamma_2 = (\text{Ray}_2(n) + (1/r_{2n}^2(t)))$) as the jammer-receiver complex amplitude. The above complex amplitude values are known at the receiver.

3.3. System Overview. In our system model, a fixed number of known pilot symbols are sent using the wireless IEEE 802.11p standard [14] over consecutive time instants from the transmitter to the receiver. At the same time, the jammer simultaneously transmits over consecutive time instants' random jamming symbols to the receiver. Using these pilots, the LOS channel and the $N - 1$ NLOS channels between the jammer and the receiver are estimated by the receiver.

The basic idea is to first estimate the relative speed between the jammer and the receiver, exploiting the RF Doppler shift. We use the variations of the estimated relative speed as a new feature in a supervised machine learning algorithm for RF jamming attack detection. Along with the relative speed from the application layer, we use cross-layer data that we obtain from the physical layer, such as the received signal strength indicator (RSSI), the SINR, and the PDR. Two classification algorithms are investigated, namely, the k-nearest neighbors (KNN) and the random forest (RaFo) algorithm, respectively.

3.4. Jamming Scenarios. We assume that the jammer continuously transmits so as to overload the wireless medium conducting a DoS attack [31]. We investigate three different attack scenarios, namely, interference scenario, smart attack scenario, and constant attack scenario, each representing a jamming attack case that could affect a VANET in real life.

In the interference scenario, we assume that no jammer is present in the network. This scenario is useful for evaluating the efficiency of our method in differentiating jamming from interference. The vehicle travels, when, at some point, it passes through an area with significant RF interference that affects the communication with other vehicles or the RSU. The smart attack scenario models an intelligent jammer behavior [32]. This smart jammer is designed to start transmitting in a reactive way upon sensing energy above a certain threshold. We set the latter to −75dBm as it is empirically determined to be an average threshold between jammer sensitivity and false transmission detection rate [18, 33]. Using this minimum threshold, each ongoing transmission can be detected by the reactive jammer. The standard protocol wireless access in vehicular environment (WAVE) IEEE 802.11p orthogonal frequency-division multiplexing (OFDM) frame format consists of the OFDM PHY layer convergence protocol (PLCP) preamble, PLCP header, MAC header, wave short message protocol (WSMP) header, PLCP service data unit (PSDU), tail bits, and pad bits. In the PLCP preamble field, the preamble consists of ten identical short training symbols and two identical long training symbols. The smart jammer is designed to affect the header of the 802.11p frame sent from T_x to R_x. When the next OFDM signal can be transmitted, there is an idle time of $T_{\text{prep}} = 10\,\mu\text{s}$ required to set up the next transmission. If the detected energy exceeds the threshold during a certain time span ($T_{\text{reaction}} = 12\mu\text{s}$), an ongoing 802.11p transmission is assumed by the jammer. The time interval of the detection is the sum between the idle time T_{prep} and a small value as the detection time $T_{\text{detection}} = 2\mu\text{s}$ to avoid reacting to sporadic noise power peaks. In the case where the detected energy exceeds the threshold during a certain time span, the jammer starts its transmission for a duration of ($T_{\text{duration}} = 64\mu\text{s}$) in order to jam a substantial part of the packet header to prevent being decoded by the receiver, as illustrated in Figure 2.

Specifically, a smart jammer starts following the victim vehicle, while transmitting a jamming signal. When the jammer reaches its target at a distance of about 10m, it retreats to a different position in order to stay undetected and transmits in a reactive way as described above. The most common approach in [33] is when the jammer keeps changing its transmission power, thus achieving the same disrupt or thwart in the communication (DoS attack) without the need of changing its distance from the target. With our smart attack, we aim at affecting the communication of the T_x-R_x pair, with the jammer detection being more difficult, pointing out the importance of the proposed VRS metric for the detection accuracy results. For that reason, the smart jammer alters also its position with the aim of staying undetected.

In the constant attack scenario, we study the case of a jammer that follows the receiver while transmitting constantly at a minimum power. When the jammer reaches its target, it begins transmitting constantly with its full power without any intention to stay undetected as in the smart attack scenario.

4. Proposed Detection System Based on Supervised Learning

4.1. Relative Speed Estimation. In this section, we present the basic idea regarding the estimation of the relative speed (Δu) between the jammer and the victim vehicle. Based on the

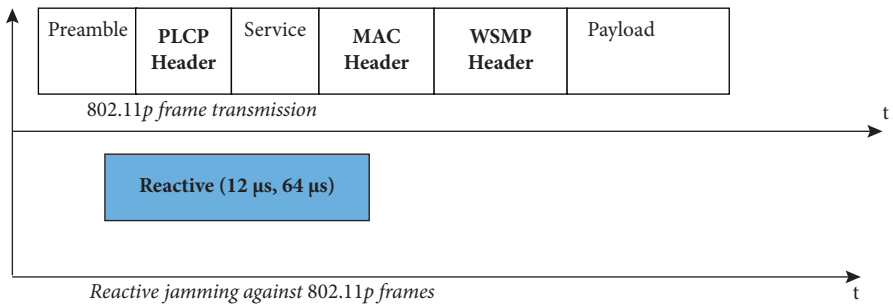

Figure 2: Reactive jammer profiles in the time domain compared with a default 802.11p transmission.

obtained values, the VRS metric is generated and then used for classification. The relative speed metric as defined in [10] is

$$\Delta u = \left| \vec{u}_{J_x} - \vec{u}_{R_x} \right|, \tag{4}$$

where $\vec{u}_{J_x}$ and $\vec{u}_{R_x}$ are the speed of the jammer and the speed of the receiver, respectively.

From (3), the N multipath combined channels $(h_1 + h_2)$ are estimated, using a minimum mean square error (MMSE) estimator [9]. By exploiting the Doppler phenomenon for modeling the LOS h_2^{LOS} channel between the jammer and the receiver, we estimate the above-defined relative speed metric, as described in [9]. Note that the jammer estimation method is based on the passive communication between J_x-R_x.

4.2. Proposed Algorithm. To make our detection method robust, apart from using physical and network metrics that were already used in related works, we use the VRS metric that is derived from the application layer and can be efficiently estimated from the RF signals (see Section 4.1). Our method uses this new metric, as an extrafeature in a cross-layer approach, along with other metrics from the physical layer for the classification process. All these metrics are presented in Table 1.

To generate the VRS metric for classification, we make three fundamental assumptions [34]:

(1) When the relative speed is equal to zero and remains unchanged, it indicates the existence of a constant jammer that follows the victim vehicle

(2) When the relative speed is not equal to zero and remains unchanged, it indicates the absence of a moving jammer as the relative speed is equal to the speed of the receiver and the speed of the jammer is equal to zero

(3) When the relative speed is not equal to zero for a period of time and then becomes zero while remaining unchanged, it indicates the existence of a jammer that begins following the target after reaching it

The common characteristic of these assumptions is that the speed of the participating nonmalicious vehicles remains unchanged and is always greater than zero.

Table 1: Metrics that are jointly processed by the classification algorithms.

ID	Model feature	Short description
1	VRS	Variations of relative speed (m/sec)
2	RSSI	Signal strength indicator (dBm)
3	SINR	Signal quantity indicator (dB)
4	PDR	Packet delivery ratio

However, in a real-life scenario, such as the one that we study, the speed—and as a consequence the relative speed—may not remain constant during the observation period. In other words, if we want to accurately model an urban environment, we have to consider the fact that the vehicles can alter their travelling speed. To handle these real-life situations, while still using the previously presented assumptions, we introduce the *Variations of Relative Speed* (Algorithm 1) (VRS algorithm).

The VRS algorithm detects changes in the relative speed of the training sample. To ensure that the relative speed in the current time instance along with the speed from previous as well as subsequent observations are used along with a series of control flow statements, the algorithm is divided into two main parts; the first considers the case in which the relative speed value is not equal to zero and the second the opposite case, each one with its own logical checks to determine the existence of a threat.

Apart from the estimated relative speed, in order to handle cases of speed alterations, the speed of the receiver has to be examined as well. If Δu is not equal to zero, then either there is no jammer present (and only interference may potentially affect the wireless communication) or there is a jammer that has not yet reached the receiver. To identify in which case we are, we have to examine whether or not there has been a variation in the relative speed compared to a previous time instance.

Observing a variation in the relative speed, however, it is not, by itself, a clear indicator of the presence or absence of a jammer. For that reason, the speed of the receiver u_{Rx} is, also, used. The equality between the relative speed (Δu) and the speed u_{Rx}, while Δu changes, indicates the absence of a jammer, since the speed of the jammer u_{Jx} is equal to zero and the speed of the receiver u_{Rx} is in fact the relative speed. On the contrary, a difference between Δu and u_{Rx} indicates the presence of a jammer that follows the receiver.

```
(1) M← number of observations
(2) vrs←matrix(nrow = M, ncol = 1)
(3) Δu[M]← array of estimated relative speed values
(4) u_Rx[M]←array of real travelling speed values
(5) k←1
(6) if Δu[k] = Δu[k + 1] then
(7)    vrs←NA
(8)    trigger←0
(9) else if Δu[k] ≠ Δu[k + 1] then
(10)   vrs←A
(11)   trigger←1
(12) end if
(13) k + +
(14) while (k < M) do
(15)   if Δu[k] ≠ 0 then
(16)     if Δu[k] ≠ Δu[k − 1] then
(17)       if Δu[k] = u_Rx[k] then
(18)         vrs←NA
(19)         trigger←0
(20)       else if Δu[k] ≠ u_Rx[k] then
(21)         vrs←A
(22)         trigger←1
(23)       end if
(24)     else if Δu[k] = Δu[k − 1] then
(25)       if Δu[k] ≠ u_Rx[k] then
(26)     vrs←A
(27)     trigger←1
(28)   else if Δu[k] = u_Rx[k] then
(29)     if (Δu[k − 1] = u_Rx[k − 1] &&
(30)     Δu[k + 1] = u_Rx[k + 1]) then
(31)       vrs←NA
(32)       trigger←0
(33)     else
(34)       vrs←A
(35)       trigger←1 end
(36)     end if
(37)   end if
(38)   else if
(39)     Δu[k] = 0 then
(40)       if u[k] ≠ 0 then
(41)         vrs←A
(42)         trigger = 1
(43)       else if u_Rx[k] = 0 then
(44)         if Δu[k − 1] = u_Rx[k − 1] then
(45)         if trigger = 0 then
(46)           vrs←NA
(47)           trigger←0
(48)         else
(49)           vrs←A
(50)           trigger←1
(51)         end if
(52)         else if Δu[k − 1] ≠ u_Rx[k − 1] then
(53)           vrs←A
(54)           trigger←1
(55)         end if
(56)       end if
(57)     end if
(58)   end while
(60) return vrs
```

Algorithm 1: The VRS algorithm

On the contrary, if no alteration of the relative speed is observed while the relative speed value is not equal to the speed value, a possible presence of a jammer is registered. This could occur in a situation where the target vehicle would reduce its speed due to an obstacle. Following our assumption, the jammer would, also, decrease its travelling speed, thus keeping the relative speed unchanged but also different from the travelling speed of the receiver. Contrary to the previous, if no alteration in the relative speed value is observed (for the previous and the next measurement), while having $\Delta u = u_{Rx}$, we conclude that a jammer is not following the receiver.

Having examined the case where the observed relative speed value is not equal to zero, we proceed to the opposite case. With $\Delta u = 0$, a simplistic form of the proposed algorithm (*VRS algorithm*) is presented, indicating the existence of a jammer that has reached its target and follows it closely with the same speed. A real-life environment, however, is more complicated. If the travelling speed u_{Rx} of the receiver is not equal to zero, while $\Delta u = 0$, a jammer has reached the receiver and follows it while disrupting the communications. On the contrary, if the travelling speed is zero (while $\Delta u = 0$), there might be a jammer present that has stopped moving (following the behavior of the target). In that case, we have to examine the previous observation for equality between relative speed and travelling speed as well as the trigger value to determine the situation.

The variables Δu and u_{Rx} represent an array of estimated relative speed values and real travelling speed values of the receiver, respectively, M is the number of the available observations upon which the algorithm operates, vrs is an array used to store the classification result (A for attack or NA for not attack) of the current observation, and *trigger* is a binary variable which indicates the presence of a jammer (value is equal to 1) or its absence (value is equal to 0). The NA and A values are two extreme and distinct values able to differentiate the attack from the no attack cases and guide the classification process.

4.3. Supervised Learning Algorithms. The supervised learning methods that are used in this work are KNN [35] and random forests [36]. Their choice does not affect the efficiency of our algorithm as our proposed feature is not constrained by the type of the supervised learning algorithm that is used. The VRS (Algorithm 1) generates the new metric which is used as an extrafeature for classification.

Both supervised learning techniques are very popular, with the KNN being robust against noisy training data like the ones obtained from a real-life urban environment and random forests being one of the most accurate algorithms, due to the fact that it reduces the chance of overfitting (by averaging several trees, there is a significantly lower chance of overfitting). As it is previously stated, our detection scheme is currently based on offline training that leverages the use of a dataset of collected measurements in order to train the classifier.

5. Simulation Setup

Figures 3(a)–3(c) illustrate the behavior of the jammer by plotting how SINR varies in time for each of the three scenarios, namely, interference scenario, smart attack scenario, and constant attack scenario.

5.1. Supervised-Learning Testing Cases. Apart from the scenarios that we use to evaluate the performance of the overall system, we also created a series of test cases that are presented in Table 2. They allow for a deeper exploration of the proposed method depending on the set of observations that are utilized for both training and testing.

These cases only affect how the training and testing is performed, without any further implications in the scenarios. They are created in such a way so as to provide insight about the importance of using the VRS metric for classification under different circumstances [37]. Specifically, it is evaluated for the cases that use or omit the VRS metric as an extrametric for the classification process. For the sake of completeness, the trained prediction model is also tested using data that were collected under a receiver speed of 25 m/s, that is, under a speed different from the 15 m/s that we trained the prediction model. We also conducted additional experiments using data measurements from the 25 m/s receiver speed range used for training. Finally, the data are normalized prior to their use for training and testing. By normalization, we refer to the process of changing the data so as to belong in the 0-1 range. It should be noted that, in all the other cases than those declared, the data are not normalized prior to their use for training and testing.

5.2. VANET Simulation Assumptions. Regarding the details of our simulation setup, the speed of the vehicles involved in the legitimate communication (u_{T_x,R_x}), the initial distance between the jammer and the pair of R_x-T_x ($\text{dist}_{\text{initial}}$), the distance that separates the receiver from the transmitter throughout the course of the simulation (dist_{T_x,R_x}) as well as the power of all the transmitted signals (P_{T_x,J_x}), and the reference distance (dist_{ref}), with which the path loss component is estimated, are presented in Table 3.

The power of all transmitted signals is measured in milliwatts (mW) and is converted in the dBm scale prior to using them in the algorithm. Each signal that is transmitted from both the jammer and the transmitter consists of streams that are 500 bits long. In all scenarios, 1000 packets are transmitted from the transmitter to the receiver. Using a time sample of 0.1 sec, we simulate the system for 100 seconds and obtain 1000 measurements.

We used Veins that combines the Simulation of Urban Mobility (SUMO) and the OMNET++/VEINS [38]. SUMO is adopted as our traffic simulator and OMNET++ is used to simulate the wireless communication. Furthermore, the GEMV (a geometry-based efficient propagation model for V2V) [39] tool was integrated into the VEINS network simulator for a more realistic simulation of the PHY layer [32]. For describing the modeled area, GEMV takes the map of a real area as an input and uses the outlines of vehicles,

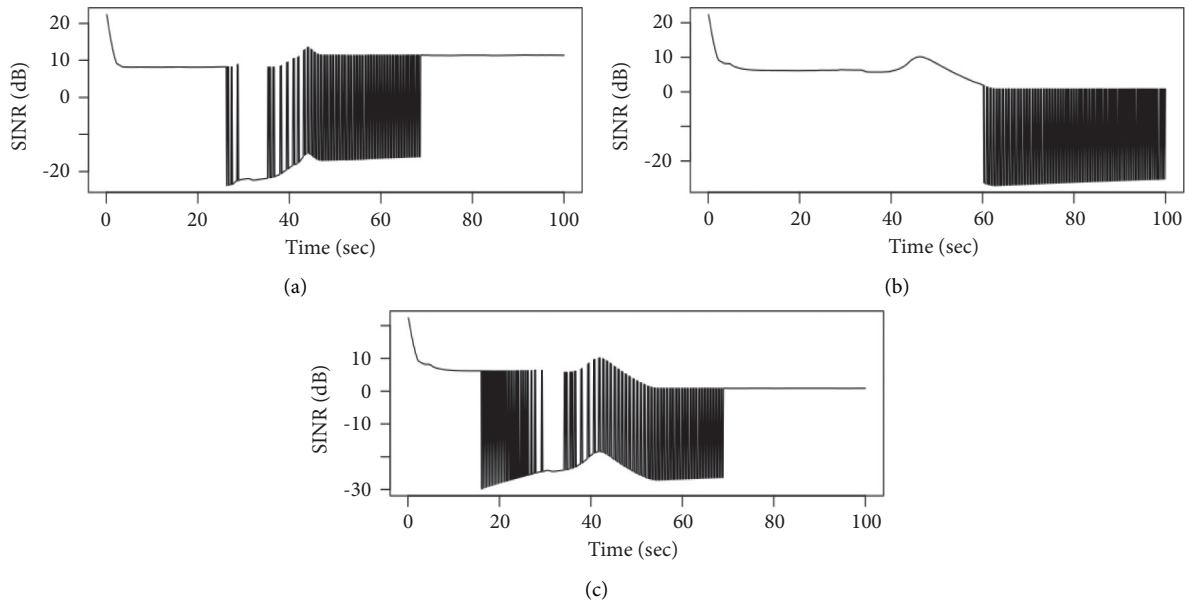

Figure 3: SINR vs. time for an interference scenario and each of the two attack scenarios. (a) SINR vs. time for the Rician fading model in the interference scenario, (b) SINR vs. time for the Rician fading model in the smart Attack Scenario, and (c) SINR vs. time for the Rician fading model in the constant attack scenario.

Table 2: The classification process under different testing cases.

Cases	VRS metric utilization (m/s)	Normalization (m/s)	R_x speed for training (m/s)	R_x speed for testing (m/s)
Same_KNN-VRS and Same_RaFo-VRS	✓		15	15
Same_KNN and Same_RaFo			15	15
Different_KNN-VRS and Different_RaFo-VRS	✓		15	25
Different_KNN and Different_RaFo			15	25
Same_KNN-VRS_25 m/s and Same_RaFo-VRS_25 m/s	✓		25	15 or 25
Same_KNN_25 m/s and Same_RaFo_25 m/s			25	15 or 25
Norm_KNN-VRS and Norm_RaFo-VRS	✓	✓	15	15
Norm_KNN and Norm_RaFo		✓	15	15

buildings, and foliage. Based on the outlines of the objects, it forms R-trees. R-tree is a tree data structure in which objects in the field are bound by rectangles and are hierarchically structured based on their location in space. Hence, GEMV employs a simple geometry-based small-scale signal variation model and calculates the additional stochastic signal variation and the number of diffracted and reflected rays based on the information about the surrounding objects. Last, to set up and test our classification algorithms for the RF jamming attacks detection on the previously obtained data, we chose to use the programming language R [40]. Part of the Erlangen city (see the evaluation setup in [41]) is used for conducting the simulations.

6. Evaluation

6.1. Detection System Evaluation Setup. To underline the significance of our proposed system, we implement and analyze the performance of our model under the different cases presented previously. In particular, for each supervised learning testing case presented in Table 2, we execute a simulation which lasts for 300 seconds and is equally split in the three jamming scenarios discussed in Section 3.4 so that the first 100 sec represent the smart attack scenario, the next 100 sec represent the interference scenario, and the last 100 sec represent the constant attack scenario. All the above scenarios are independent from each other and are run at consecutive time instants.

To avoid testing with "previously seen data," thus leading to biased classification results, we have to ensure that the training and testing sets are completely separated. So, prior to presenting the classification results, we have to define the size of the training and testing sets as well as the total number of observations used, so as to make them more interpretable. The overall simulation utilizes a set of 3000 observations equally split into the three attack scenarios examined. To avoid overfitting (overfitting occurs when the classifier tends to memorize the training set and thus

generalize poorly when facing previously unseen data), only 30% of the total number of the observations are used for training, while the remaining 70% are used for testing.

Based on the ratio above, the number of the observations in the training set is 941 (that is, 293 observations from the interference scenario, 319 from the smart attack scenario, and 329 from the constant attack scenario), whereas the number of the observations in the testing set is 2059 (that is, 703 observations from the interference scenario, 685 from the smart attack scenario, and 671 from the constant attack scenario), randomly chosen but almost equally split among the three scenarios in both cases.

To present the classification results, the *confusion matrix* is used, where the rows represent classification output and the columns represent the ground truth. To evaluate the performance of our detection system in the various scenarios previously described, we use the *accuracy of the prediction model*. Accuracy is a measure that is obtained from the confusion matrix and *is equal to the ratio of all the correctly predicted labels over all the predictions*. The correctly predicted labels are the labels of the main diagonal of the confusion matrix. As an example of the above-defined confusion matrix for the accuracy calculation of our prediction model for the Same_KNN case compared to the Same_KNN-VRS case, we present the subsequent confusion matrices for the KNN algorithm (see Table 4).

6.2. Same_KNN-VRS vs. Same_KNN and Same_RaFo-VRS vs. Same_RaFo Case Classification Results. Starting from the first case, the accuracy of the prediction model achieved while using the VRS metric as an extrafeature in the classification process is 82.27% for the KNN and 80.04% for the random forest algorithm.

On the contrary, when omitting the VRS metric, we observe not only a drop in the classification accuracy but also a high confusion between interference and jamming cases. The accuracy of the prediction model is now equal to 79.16% and 76.54% for the KNN and the random forest algorithms, respectively, so the impact of the VRS metric is evident. Apart from the fact that it increases the success rate of the classification (compared to the cases where the VRS metric is omitted) it ensures, almost perfectly, the differentiation between the cases of intentional and unintentional jamming (see Table 5).

6.3. Different_KNN-VRS vs. Different_KNN and Different_RaFo-VRS vs. Different_RaFo Case Classification Results. As stated previously, these cases examine the situation in which training and testing are based on observations that were collected under different speeds. The accuracy achieved while using the VRS metric as an extrafeature in the classification process is equal to 66.97% for KNN and 69.84% for random forest, respectively.

On the contrary, when the VRS metric is not used, the accuracy of the prediction model is reduced to 56% for the KNN and to 55.37% for the random forest algorithm. Figures 4 and 5 provide insight to the results for the random forest, respectively.

Table 3: Simulation parameters.

Evaluation parameters in Veins simulator	Values
u_{T_x,R_x}	15 m/sec or 25 m/sec
dist_{T_x,R_x}	35 m
$\text{dist}_{\text{initial}}$	200 m
P_{T_x,J_x}	100 mW
Minimum sensitivity (P_{th})	−85 dBm
Transmission range	130–300 meters
f_c	5.9 GHz
Doppler shift for Δu = 120km/h	±655.5 Hz
dist_{ref}	100 m

Table 4: Confusion matrix for the Same_KNN case.

Scenario	Interference	Smart attack	Constant attack
Interference	682	38	33
Smart attack	17	470	160
Constant attack	4	177	478

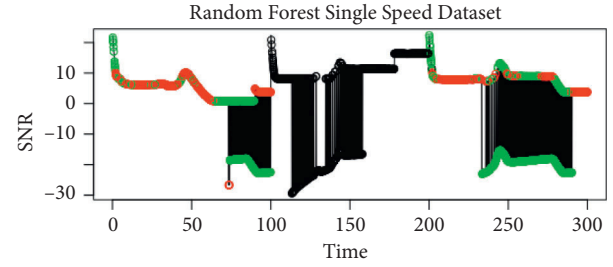

Figure 4: SINR vs. time for the Different_RaFo-VRS case, with the smart attack scenario represented by the red, the interference scenario represented by the black, and the constant attack scenario represented by the green color.

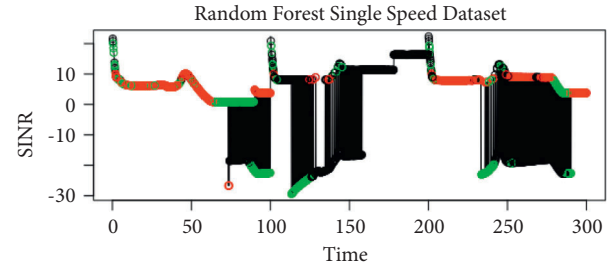

Figure 5: SINR vs. time for the Different_RaFo case, with the smart attack scenario represented by the red, the interference scenario represented by the black, and the constant attack scenario represented by the green color.

The color of the figures indicates the class in which each observation is predicted to belong to. The smart attack scenario is represented by the red and lasts for the first 100 seconds, the interference scenario is represented by the black lasts for the time interval 100–200 seconds, and the constant attack scenario is represented by the green color and lasts for the time time interval between 200 and 300 seconds(as described in Section 5.2). In Figure 4, we explain in more detail the detection process for each scenario:

(1) The appearance of black or green colors in the smart attack scenario (0–100 seconds) indicates the misclassification of this scenario with the interference and the constant attack scenarios, respectively. On the contrary, the points with red color indicate a correct detection of this attack scenario.

(2) The points with green or red colors in the interference scenario (100–200 seconds) indicate the misclassification of this scenario with the constant attack scenario and the smart attack scenario, respectively. The black color indicates a correctly detected of the interference scenario.

(3) Lastly, for the constant attack scenario during the time interval between (200–300 seconds), the presence of points with black or red colors indicates the misclassification with the interference and the smart attack scenarios, respectively. On the contrary, the appearance of the green color indicates a proper detection of the sonstant attack scenario.

Based on the classification results presented above, we can reach an important conclusion. When testing the prediction model with observations from a different speed—compared to the one used in training—we observe an overall reduction in accuracy. Nevertheless, the use of the VRS metric significantly increases prediction accuracy (in both supervised algorithms examined in this paper), while also achieving a clear separation between interference and jamming.

6.4. Norm_KNN-VRS vs. Norm_KNN and Norm_RaFo-VRS vs. Norm_RaFo Case Classification Results. In these two cases, we try to determine whether normalizing the data prior to using them in training and testing affects the classification results, with and without the use of the VRS metric. The accuracy achieved while using the VRS metric is equal to 81.25% for the KNN algorithm and 80.09% for the random forest, with its omission leading to an accuracy equal to 78.1% and 76.4%, respectively. Once more, the use of the VRS metric in the classification process leads to a upturn in the accuracy of the prediction model. In addition to that, if we compare the previous classification results of the Same_KNN and Same_RaFo cases with the respective ones that derive when no normalization is applied to the data prior to their use, we observe that there is no significant increase in accuracy results. Thus, we conclude that a normalization of the measurements is not necessary. It should be noted that in all the previous and the next presented classification results, the data are not normalized prior to their use for training and testing.

6.5. Same_KNN-VRS_25 m/s vs. Same_KNN_25 m/s and Same_RaFo-VRS_25 m/s vs. Same_RaFo_25 m/s Case Classification Results. As already stated, our RF jamming attack detection system is based on offline training, using a dataset of measurements collected under a speed of 15 m/s so as to train the classifier prior to its use for testing. For the sake of completeness, we examine the Same_KNN-VRS and Same_RaFo-VRS and Same_KNN and Same_RaFo cases presented previously using the data measurements from a higher speed at about 25 m/s speed range for training.

For the Same_KNN-VRS_25 m/s and Same_RaFo-VRS_25 m/s cases, the accuracy of the prediction model achieved is equal to 94.46% for the KNN and 94.61% for the random forest algorithm. For the Same_KNN_25 m/s and Same_RaFo_25 m/s cases, on the contrary, the calculated accuracy is equal to 88.68% for the KNN and 89.22% for the random forest algorithm, respectively.

From the classification results presented above, an important observation can be made. There is an increase in classification accuracy when the training is done using data from a higher speed. The higher classification accuracy comes from the fact that the increase in speed adversely influences the effects of jamming. More concretely, in the constant attack scenario, the jammer overtakes the sender-receiver pair faster, in the interference scenario, the sender-receiver pair remains in the jamming area for a shorter period of time, and in the smart attack scenario, the jammer reaches its target at a higher speed, thus the gradual effect of the jamming observed at lower speeds is greatly reduced. All the above lead to a significant increase in the quality of the measurements obtained, hence leading to higher classification accuracy as well as to better distinction between the different types of jammers affecting the communication, as seen in Figure 6, for the KNN algorithm.

We also investigate more thoroughly the effect of the relative speed metric in the detection probability of a RF jamming attack in a multiclass environment with three classes (class of reactive jamming attack, class of continuous jamming attack, and class of interference). In Figure 7, we present the detection probabilities of the proposed model using the KNN algorithm for a range of relative speed Δu $[0, 25]\text{m}/s$. We observe that, in the medium range of Δu values, we achieve a perfect RF jamming detection result. This result is attributed to the specific characteristics of each type RF jamming attack. Specifically, the continuous jammer transmits continuously deteriorating the wireless communication between the transmitter-receiver. On the contrary, the reactive jammer starts its activity only when it retreats to a safe position (close to the receiver). So, for a small range of Δu values, both types of RF jamming attackers (reactive and continuous) have started their attack leading to several misclassification errors between the two corresponding classes. Finally, at higher Δu values over $20\text{m}/s$, we have some misclassification errors between the classes of reactive jamming and interference because the relative speed value is approximately equal to the speed of the receiver ($25\text{m}/s$) when there is no attacker in the area but only a static RSU that interferes the wireless communication between the transmitter-receiver.

In Table 6, we summarize the classification accuracy, exploiting the usage of the proposed VRS metric as an extra feature, achieved while training with measurements from a speed of 15 m/s and a speed of 25 m/s, respectively.

6.6. Result Summary and Comparison with State of the Art. Figure 8 summarizes classification accuracy percentages that are presented above. These are achieved by both the KNN

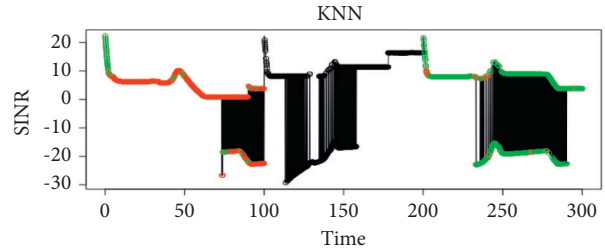

Figure 6: Plot using the 25 m/s speed for testing and training for the Same KNN 25 m/s case; with the smart attack scenario represented by the red, the interference scenario represented by the black, and the constant attack scenario represented by the green color.

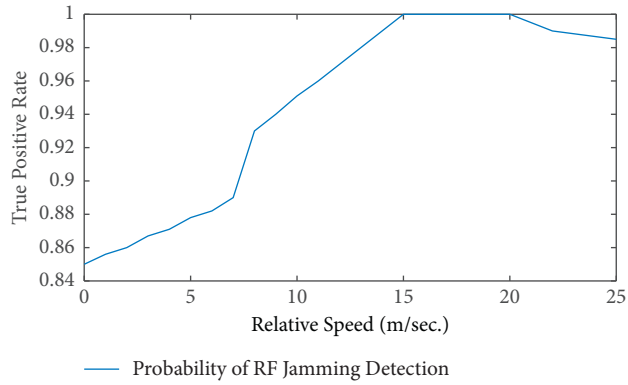

Figure 7: Effect of Δu in RF jamming detection results using the KNN algorithm.

Table 5: Confusion matrix for the Same_KNN-VRS case.

Scenario	Interference	Smart attack	Constant attack
Interference	703	0	0
Smart attack	0	494	174
Constant attack	0	191	497

Table 6: Classification accuracy percentages while training lower and higher speed measurements, respectively

	Train with 15 m/s (%)	Train with 25 m/s (%)
Test with 15 m/s (KNN)	82.27	74.31
Test with 15 m/s (RaFo)	80.04	74.41
Test with 25 m/s (KNN)	66.97	94.46
Test with 25 m/s (RaFo)	69.84	94.61

and the random forest algorithms when based only on the features previously used in the literature for jamming attack detection [26], compared to the proposed approaches, KNN-VRS and RaFo-VRS, that use the VRS metric. The VRS metric increases the accuracy of the classifier and ensures almost perfect differentiation between cases of intentional and unintentional jamming. When using the VRS metric while testing with data from the same speed, there is an increase up to about 4% in the classification accuracy. When testing with data of a different speed, the increase in accuracy is even greater up to about 14%.

We also compare the accuracy of the proposed scheme versus recent state-of-the-art work. We compare RF jamming detection methods with the same complexity and without using extrahardware (e.g., multiple antennas at the receiver [42]). For collecting the used jamming detection metrics, we assume only a completely passive scheme that is based on RF communication between the transmitter-receiver under the presence of a jammer in the area.

As we explained earlier, the authors in [24] proposed a federated learning-based on-device jamming attack detection security architecture for FANET. In order to compare our proposed RF jamming detection method with this method (Federated-Nischat-2019), we preprocess the simulated datasets to derive two unbalanced subdatasets. The first subdataset contains a higher percentage of nonjamming instances (80%) and a lower percentage of jamming instances (20%). We show in Figure 9 that the method (Federated-Nischat-2019) achieves an accuracy of 89.73% under the ns-3 simulated FANET dataset. This performance is better only for the cases where VRS metric is not used.

We also compare the KNN and the RaFo algorithm for reactive and constant jamming attack detection using the cross-layer combination of metrics proposed by Feng et al. named (feng2018-KNN) and (feng2018-RaFo). We observe in Figure 10 that when the receiver moves at low speeds of 15m/s, we have the same accuracy. When the receiver

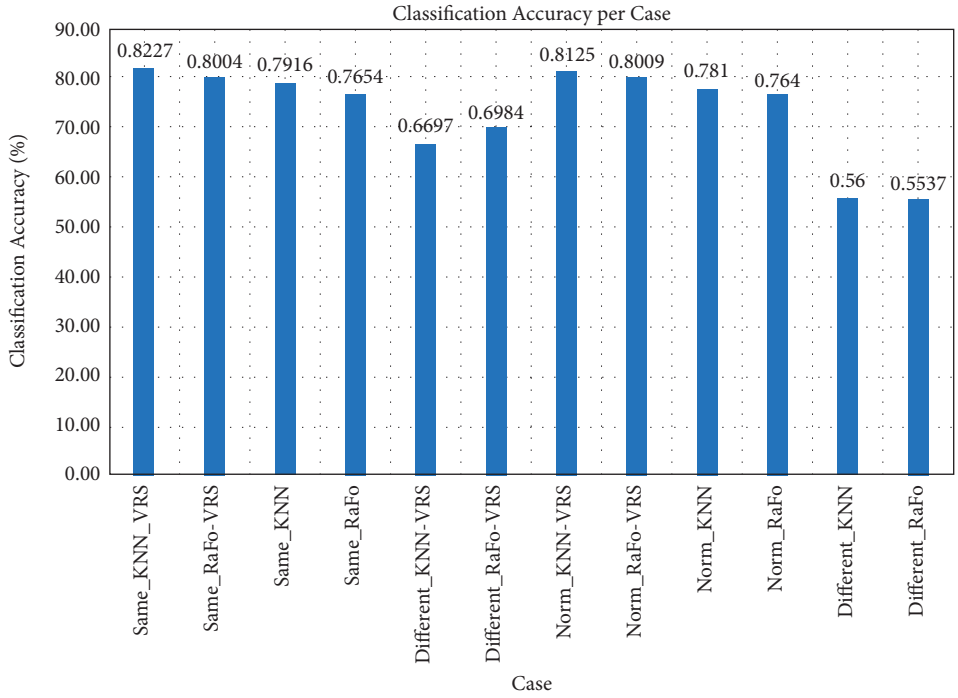

Figure 8: Comparison between the standard KNN and random forest classification algorithms and the proposed KNN-VRS and RaFo-VRS algorithms based on the accuracy percentage achieved in every case.

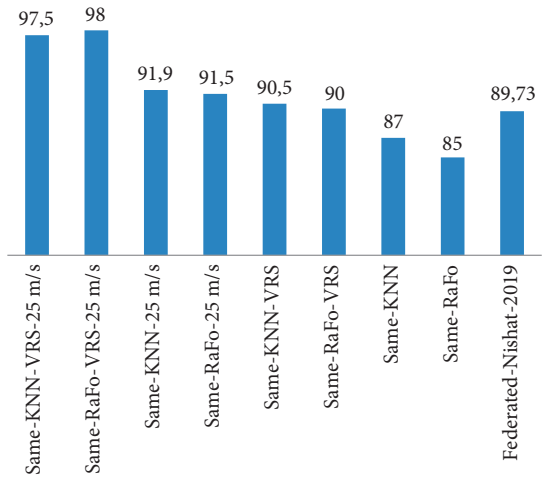

Figure 9: Comparison between the proposed KNN-VRS and RaFo-VRS algorithms, in terms of accuracy, with other state-of-the-art methods.

increases its speed and it moves with a speed of 25m/s, our proposed jamming detection scheme using the VRS metric achieves a much better accuracy (an increase of about 13%) than the competing methods.

Finally, we compare with the work of Lyamin et al. [21], where the authors use historical observation of events in the V2V channel for the jamming detection. The method is evaluated for two jamming models: random and ON-OFF jamming. To represent random jamming in our model, the reactive jammer transmits its jamming signal randomly and independently with a probability $p = 0.7$ when it is triggered. When comparing the jamming detection results of the method [21] with the proposed jamming detection method for a random jammer with $p = 0.7$, we have a probability of attack detection (true positive rate) at about 0.95 for the random reactive jammer, while the method in [21] achieves a probability of attack detection at about 0.85 for the same type of jamming. Additionally, a priori knowledge about a platoon is employed for this method to achieve better detection results. Only when the number of receivers increases to 20 in the form of a platoon of vehicles (also increasing the

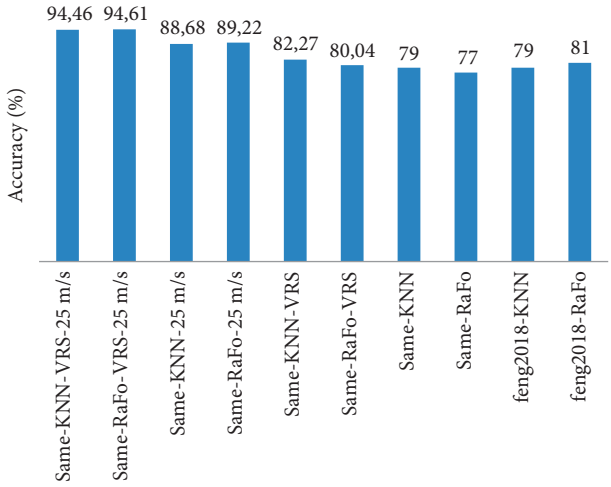

Figure 10: Comparison between the proposed KNN-VRS and RaFo-VRS algorithms, in terms of accuracy, with other state-of-the-art methods.

received observations for the training phase), the method in [21] manages to reach the probability of attack detection that we achieve using our proposed jamming detection method with a single receiver.

From this set of comparative results, the effect of the proposed VRS metric in RF jamming classification is clear. Especially, when the receiver increases its speed to a speed of 25m/s, the accuracy of the proposed method increases by over 90%. This performance is much higher than the other corresponding methods in the literature.

7. Conclusions

In this paper, we presented a method for detecting a specific type of DoS attack, namely, RF jamming, based on a cross-layer set of features and supervised machine learning. We introduced a novel metric from the application layer, namely, the variations of the relative speed between the jammer and the target. The relative speed is passively estimated from the combined value of the desired and the jamming signal at the target vehicle combined with metrics from the network and physical layer. To evaluate the significance of the proposed metric and its estimation algorithm, we implemented three different scenarios: two with a jammer and one with interference only.

With our work, we introduced a proactive approach against potential RF jamming attacks which is able to differentiate interference from malicious RF jamming. Additionally, it is able to distinguish the unique characteristics of each attack, especially when the offline training is conducted with a higher speed than 15m/s. Through our evaluation results, we were able to highlight the vital role of the relative speed and its variations, in addition to other metrics obtained from the physical layer and in jamming detection and unintentional jamming cases differentiation, as well as in the overall increase in the prediction accuracy.

As part of our future work, we plan to investigate the efficiency of our idea in complex vehicular networks with a large number of communicating nodes and several attackers. The target of this classification process will be the characterization of the behavior of a node as malicious or as regular node, mainly using the proposed VRS metric. The classification results can be collected and managed from a Trusted Central Authority (TCA) in an area with V2X communication. Having this information, the TCA could reroute vehicles towards more jamming friendly areas.

Data Availability

The data presented in this study are available from the corresponding author upon request. The data are not publicly available due to privacy reasons.

Disclosure

A preliminary version of the article appears on arxiv at https://arxiv.org/abs/1812.11886.

Conflicts of Interest

All authors declare no conflicts of interest.

References

[1] J. Jo and M. Gerla, "Internet of vehicles and autonomous connected car-privacy and security issues," in *Proceedings of the 2017 26th IEEE International Conference on Computer Communication and Networks (ICCCN)*, Vancouver, BC, USA, July 2017.

[2] A. Herm, "Assume self-driving cars are a hacker's dream? think again," https://www.theguardian.com/technology/2017/aug/30/self-driving-cars-hackers-security, 2017, [Online; accessed 30-August-2017].

[3] D. Cottingham, "What is vehicle platooning? driving tests," 2017, https://www.drivingtests.co.nz/resources/what-is-vehicle-platooning/.

[4] A. Hamieh, J. Ben-Othman, and L. Mokdad, "Detection of radio interference attacks in VANET," in *Proceedings of the Global Telecommunications Conference*, pp. 1–5, GLOBECOM 2009. IEEE, Honolulu, HI, USA, December 2009.

[5] I. K. Azogu, M. T. Ferreira, J. A. Larcom, and H. Liu, "A new anti-jamming strategy for VANET metrics-directed security defense," in *Proceedings of the 2013 IEEE Globecom Workshops (GC Wkshps)*, pp. 1344–1349, Atanta, GA, USA, December 2013.

[6] O. Puñal, I. Aktaş, C. Schnelke, G. Abidin, K. Wehrle, and J. Gross, "Machine learning-based jamming detection for IEEE 802.11: design and experimental evaluation," in *Proceedings of the 2014 IEEE 15th International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM)*, pp. 1–10, Sydney, Australia, June 2014.

[7] W. Xu, W. Trappe, Y. Zhang, and T. Wood, "The feasibility of launching and detecting jamming attacks in wireless networks," in *Proceedings of the 6th ACM international symposium on Mobile ad hoc networking and computing*, pp. 46–57, New York, NY, USA, May 2005.

[8] J. Grover, N. K. Prajapati, V. Laxmi, and M. S. Gaur, "Machine learning approach for multiple misbehavior detection in VANET," *Advances in Computing and Communications* in *Proceedings of the International Conference on Advances in*

Computing and Communications, vol. 192, pp. 644–653, ACC, Kochi, India, July 2011.

[9] D. Kosmanos, A. Argyriou, and L. Maglaras, "Estimating the relative speed of RF jammers in VANETs," *Security and Communication Networks*, vol. 2019, Article ID 2064348, 2019.

[10] D. Karagiannis and A. Argyriou, "Jamming attack detection in a pair of RF communicating vehicles using unsupervised machine learning," *Vehicular Communications*, vol. 13, pp. 56–63, 2018.

[11] M. Azizian, S. Cherkaoui, and A. S. Hafid, "Link activation with parallel interference cancellation in multi-hop VANET," in *Proceedings of the 2016 IEEE 84th Vehicular Technology Conference (VTC-Fall)*, Montreal, QC, Canada, September 2016.

[12] B. DeBruhl, S. Weerakkody, B. Sinopoli, and P. Tague, "Is your commute driving you crazy?: a study of misbehavior in vehicular platoons," in *Proceedings of the WiSec '15 Proceedings of the 8th ACM Conference on Security and Privacy in Wireless and Mobile Networks*, vol. 22, New York, NY, USA, June 2015.

[13] N. Bißmeyer, C. Stresing, and K. M. Bayarou, "Intrusion detection in vanets through verification of vehicle movement data," in *Proceedings of the Vehicular Networking Conference (VNC)*, pp. 166–173, IEEE, New York, NY, USA, December 2010.

[14] S. Malebary, W. Xu, and C.-T. Huang, "Jamming mobility in 802.11 p networks: modeling, evaluation, and detection," in *Proceedings of the 2016 IEEE 35th International on Performance Computing and Communications Conference (IPCCC)*, pp. 1–7, Las Vegas, NV, USA, December 2016.

[15] L. Nikita, K. Denis, D. Quentin, and V. Alexey, "Real-time jamming DoS detection in safety-critical V2V C-ITS using data mining," *IEEE Communications Letters*, vol. 23, pp. 442–445, 2019.

[16] L. Mokdad, J. Ben-Othman, and A. T. Nguyen, "DJAVAN: detecting jamming attacks in vehicle ad hoc networks," *Performance Evaluation*, vol. 87, pp. 47–59, 2015.

[17] A. T. Nguyen, L. Mokdad, and J. Ben Othman, "Solution of detecting jamming attacks in vehicle ad hoc networks," in *Proceedings of the 16th ACM international conference on Modeling, analysis & simulation of wireless and mobile systems*, pp. 405–410, New York, NY, USA, November 2013.

[18] O. Puñal, C. Pereira, A. Aguiar, and J. Gross, "Experimental characterization and modeling of RF jamming attacks on VANETs," *IEEE Transactions On Vehicular Technology*, vol. 64, no. 2, pp. 524–540, 2015.

[19] A. Quyoom, R. Ali, D. N. Gouttam, and H. Sharma, "A novel mechanism of detection of denial of service attack (DoS) in VANET using malicious and irrelevant packet detection algorithm (MIPDA)," in *Proceedings of the 2015 IEEE International Conference on Computing, Communication & Automation (ICCCA)*, pp. 414–419, Greater Noida, UP, May 2015.

[20] S. RoselinMary, M. Maheshwari, and M. Thamaraiselvan, "Early detection of DoS attacks in VANET using attacked packet detection algorithm (apda)," in *Proceedings of the 2013 IEEE International Conference on Information Communication and Embedded Systems (ICICES)*, pp. 237–240, Chennai, Tamil Nadu, India, February 2013.

[21] N. Lyamin, D. Kleyko, Q. Delooz, and A. Vinel, "Real-time jamming dos detection in safety-critical v2v c-its using data mining," *IEEE Communications Letters*, vol. 23, no. 3, pp. 442–445, 2019.

[22] A. Benslimane and H. Nguyen-Minh, "Jamming attack model and detection method for beacons under multichannel operation in vehicular networks," *IEEE Transactions on Vehicular Technology*, vol. 66, no. 7, pp. 6475–6488, 2017.

[23] S. Hymlin Rose and T. Jayasree, "Detection of jamming attack using timestamp for wsn," *Ad Hoc Networks*, vol. 91, Article ID 101874, 2019.

[24] N. I. Mowla, N. H. Tran, I. Doh, and K. Chae, "Federated learning-based cognitive detection of jamming attack in flying ad-hoc network," *IEEE Access*, vol. 8, pp. 4338–4350, 2020.

[25] Z. Feng and C. Hua, "Machine learning-based rf jamming detection in wireless networks," in *Proceedings of the 2018 Third International Conference on Security of Smart Cities, Industrial Control System and Communications (SSIC)*, pp. 1–6, Shanghai, China, October 2018.

[26] G. Kasturi, A. Jain, and J. Singh, "Machine learning-based rf jamming classification techniques in wireless ad hoc networks," in *Proceedings of the WIDECOM 2020:nternational Conference on Wireless, Intelligent and Distributed Environment for Communication*, Toronto, Canada, May 2020.

[27] M. Hachimi, G. Kaddoum, G. Gagnon, and P. Illy, "Multi-stage jamming attacks detection using deep learning combined with kernelized support vector machine in 5g cloud radio access networks," in *Proceedings of the 2020 International Symposium on Networks, Computers and Communications (ISNCC)*, pp. 1–5, Montreal, QC, USA, October 2020.

[28] D. Kosmanos, A. Pappas, F. J. A. Navarro et al., "Intrusion detection system for platooning connected autonomous vehicles," in *Proceedings of the SEEDA-CECNSM conference*, Piraeus, Greece, September 2019.

[29] S. Sharanya and S. Karthikeyan, "Classifying malicious nodes in VANETS using support vector machines with modified fading memory," *ARPN Journal of Engineering and Applied Sciences*, vol. 12, no. 1, pp. 171–176, 2017.

[30] D. Tse and P. Viswanath, *Fundamentals of Wireless Communication*Cambridge University Press, Cambridge, UK, USA, 2005.

[31] S. Munazza, K. Muazzam, K. Umair Shafiq, and S. Nazar, "Detection and prevention of distributed denial of service attacks in VANETs," in *Proceedings of the 2016 International Conference on Computational Science and Computational Intelligence (CSCI)*, Las Vegas, NV, USA, December 2016.

[32] D. Kosmanos, N. Prodromou, A. Argyriou, L. A. Maglaras, and H. Janicke, "MIMO techniques for jamming threat suppression in vehicular networks," *Mobile Information Systems*, vol. 2016, pp. 1–9, Article ID 8141204, 2016.

[33] M. Strasser, B. Danev, and C. Srdjan, "Detection of reactive jamming in sensor networks," *ACM Transactions On Sensor Networks*, vol. 7, 2010.

[34] D. Kosmanos, D. Karagiannis, A. Argyriou, S. Lalis, and L. Maglaras, "RF jamming classification using relative speed estimation in vehicular wireless networks," 2018, https://arxiv.org/abs/1812.11886.

[35] O. Sutton, "Introduction to k nearest neighbour classification and condensed nearest neighbour data reduction," University lectures, University of Leicester, Leicester, England, 2012.

[36] A. Liaw and M. Wiener, "Classification and regression by random forest," *R News*, vol. 2, no. 3, pp. 18–22, 2002.

[37] D. Karagiannis, *Wireless Jamming Detection in Vehicular Networks Using Machine Learning and Cross-Layer Data*, University of Thessaly, Department of Electrical and Computer Engineering, Thessaly, Greece, 2018, http://hdl.handle.net/11615/48295.

[38] C. Sommer, R. German, and F. Dressler, "Bidirectionally coupled network and road traffic simulation for improved IVC analysis," *IEEE Transactions On Mobile Computing*, vol. 10, no. 1, pp. 3–15, 2015.
[39] M. Boban, J. Barros, and O. K. Tonguz, "Geometry-based vehicle-to-vehicle channel modeling for large-scale simulation," *IEEE Transactions on Vehicular Technology*, vol. 63, p. 4146, 2016.
[40] "What is r?," 2017, https://www.r-project.org/about.html.
[41] D. Kosmanos, L. Maglaras, M. Mavrovouniotis et al., "Route optimization of electric vehicles based on dynamic wireless charging," *IEEE Access*, vol. 6, pp. 42 551–42 565, 2018.
[42] S. Xu, W. Xu, C. Pan, and M. Elkashlan, "Detection of jamming attack in non-coherent massive simo systems," *IEEE Transactions on Information Forensics and Security*, vol. 14, no. 9, pp. 2387–2399, 2019.