Knowledge-Based Systems 163 (2019) 611–623

Contents lists available at ScienceDirect

Knowledge-Based Systems

journal homepage: www.elsevier.com/locate/knosys

A filter model for intrusion detection system in Vehicle Ad Hoc Networks: A hidden Markov methodology

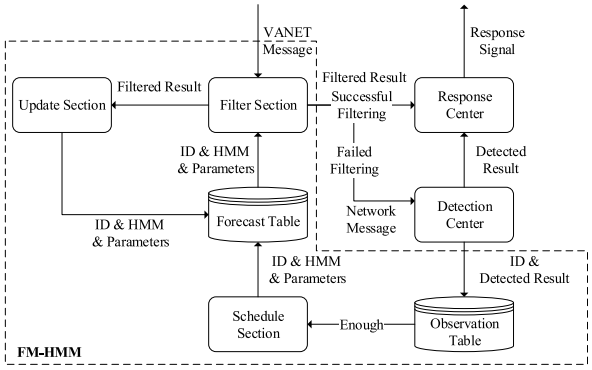
Junwei Liang [a,*], Maode Ma [a], Muhammad Sadiq [b], Kai-Hau Yeung [c]

[a] School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore
[b] School of Computer and Software Engineering, Shenzhen University, Shenzhen, China
[c] Department of Electronic Engineering, City University of Hong Kong, Hong Kong

HIGHLIGHTS

- Vehicle's state pattern is modeled as a hidden Markov model (HMM) for filtering.
- We propose a novel filter model based on HMM in IDS (we called it FM-HMM).
- FM-HMM can mitigate message congestion in VANETs with minor affection to the accuracy of IDS.

GRAPHICAL ABSTRACT



The Scheme of IDS with FM-HMM.

ARTICLE INFO

Article history:
Received 4 February 2018
Received in revised form 11 September 2018
Accepted 13 September 2018
Available online 22 September 2018

Keywords:
Vehicle Ad Hoc Networks (VANETs)
Intrusion detection system (IDS)
Hidden Markov model (HMM)
Filter model based on hidden Markov model (FM-HMM)

ABSTRACT

Although Vehicle Ad Hoc Network (VANETs) as a new technology is being used in wide range of applications to improve the driving experience as well as safety, it is vulnerable to various type of network attacks. Literature studies have revealed several reliable approaches based on intrusion detection system (IDS), to protect VANETs against attacks. However, by those solutions, the overheads of IDSs are serious which cause too long detection time, especially when the number of vehicles increases. In this paper, we propose a novel filter model based hidden Markov model (HMM) (FM-HMM) for IDS to reduce the overhead and time for detection without impairing detection rate. To the best of our knowledge, this is the first work in the literature to model the state pattern of each vehicle in VANETs as a HMM to quickly filter the messages from the vehicles instead of detecting these messages. The FM-HMM consists of three modules, i.e., schedule, filter and update. In the schedule module, Baum–Welch algorithm is used to produce a HMM and its parameters for each neighbor vehicle. In the filter module, multiple HMMs are used with their parameters to forecast the future states of neighbor vehicles with which the messages from them are filtered. In the update module, a timeliness method is used to update HMMs and their parameters. Experiments show that the IDS with FM-HMM has a better performance in terms of detection rate, detection time and overhead.

© 2018 Elsevier B.V. All rights reserved.

1. Introduction

Vehicle ad hoc networks (VANETs) can provide more information for the drivers to make the driving easier and thus improve highway safety. It will become more popular in the very near

* Corresponding author.
E-mail address: JUNWEI001@e.ntu.edu.sg (J. Liang).

https://doi.org/10.1016/j.knosys.2018.09.022
0950-7051/© 2018 Elsevier B.V. All rights reserved.

future and remarkably change our lives [1]. The tremendous safety, convenience and commercial potential of VANETs will drive its deployment. VANETs are multi-agent wireless networks that are designed mainly to solve traffic problems by allowing vehicles to communicate with each other. Nevertheless, VANETs can facilitate the applications and services of connected vehicles and intelligent transportation systems. The tremendous safety, convenience and commercial potential of VANETs can make the tasks easier for the drivers and thus improve safety.

VANETs provide a wireless communication among moving vehicles by using a dedicated short-range communication (DSRC) which is essentially IEEE 802.11a amended. The vehicles in VANETs use DSRC to communicate with each other, i.e., vehicle to vehicle (V2V), and with the infrastructure (Road Side Units- RSUs), i.e., vehicle to infrastructure (V2I) [2]. Although current traffic conditions are considered to be somewhat hazardous and confusing, these technological innovations for current vehicles will change the way we think about road travel by making it much safer and more productive. For example, VANETs can help drivers navigate safely in very dangerous driving conditions, such as fog, accidents, and black ice [3].

Since VANETs process the vital traffic information related to human safety, they should follow the security requirements such as privacy, confidentiality, integrity, and non-repudiation to provide secured communications against attackers, and malicious nodes. Various security attacks such as, false information attack [4], Sybil attack [5], denial of service (DOS) [6], Illution attack [7] and purposeful attack [8] not only affect the privacy of drivers and vehicles but also compromise traffic safety and eventually lead to loss of life [9]. Hence, a strategy to provide security in VANETs is necessary.

Intrusion detection system (IDS) is one of the most important approaches to protect vehicular networks against threats as it has the ability to detect both insider and external attacks with high accuracy [10,8]. However, there are some important issues that need to be addressed before using IDS to prevent attacks in VANETs. (i) It is difficult to use the same intrusion detection mechanisms that have been appropriately used in wired networks because of the wireless and mobile nature of VANETs. (ii) Encounters are short lived and the received information has to be processed quickly considering that VANETs are very fast moving and highly dynamic. In other words, the reliability of information needs to be ascertained both quickly and accurately [6]. (iii) Although IDS is a reliable approach to protect VANETs against attacks, it also brings the extra cost of detection time and overhead into VANETs. The costs cannot be ignored, especially when the number of vehicles increases, but few efforts have been made to address it.

The major contributions in this paper are as follows.

1. This is the first work in the literature to model the state pattern of each vehicle in VANETs as a hidden Markov process, which can be used to build up HMMs to quickly filter the messages from these vehicles.
2. FM-HMM is proposed for IDS to reduce overhead and the time for detection without impairing detection rate. It mainly consists of three modules including schedule, filter and update.
3. Using FM-HMM in IDS, the congestion of network message, i.e., broadcast storms, can be mitigated by the reduction of message scale and the improvement of processing efficiency. Moreover, FM-HMM just has minor affection to the performance of IDS, even when up to 40% of vehicles are malicious.

The remainder of this paper is organized as follows. Related work is discussed in Section 2. Preliminary and background of this paper are presented in Section 3. In Section 4, the detail of FM-HMM is provided. Section 5 demonstrates experiment results with the discussion in detail. Conclusion is presented in Section 6.

2. Related work

The security of VANETs is a highly important issue that has been the focus of research for many years [11–16]. One of the important challenges in this regard is the existence of malicious vehicles. Several approaches have been proposed in the literature to tackle this issue in VANETs. These approaches are usually classified three categories (i) trust or reputation based scheme, (ii) data-centric misbehavior detection scheme, and (iii) IDS.

2.1. Trust or reputation based scheme

By a trust or reputation based scheme, the assignment of a trust score to a vehicle is based on the previous or current interactions between the vehicle and others [17]. The trust score represents the reputation of a vehicle in the network. In [18,19], a decentralized infrastructure of trust based scheme has been adopted. A reputation management system has been built for each vehicle that enables to quickly adapt the change of local conditions and to establish trust relationships with other vehicles. In [20], a centralized infrastructure, named as an attack-resistant trust management scheme has been proposed. It can not only detect and cope with attacks, but also evaluate the trustworthiness of both data and vehicles in VANETs. The trust based schemes are useful but they cannot be used for detecting false emergency messages as trust needs to be built over a period of time. Moreover, if a false message comes from a legitimate vehicle, it is difficult to detect it.

2.2. Data-centric misbehavior detection scheme

A data-centric misbehavior detection scheme has been proposed in [21,22]. It has been used for shared data to improve the reliability of VANETs and used to detect and correct the errors of the data sent out by vehicles [21]. The messages that conform to a model are accepted, while otherwise are rejected. It also has been used to identify the false information of emergency message based on the message type and the subsequent behavior of vehicle [22]. However, due to messages are relayed and false information is identified based on both message and subsequent behavior of sending vehicles, the technique is difficult to be managed in VANETs and is not feasible for emergency messages which need to be acted quickly. Additionally, the technique produces enormous computation cost.

2.3. Intrusion detection system

Since building the trust or reputation based scheme is a very complex and time-consuming task and the data-centric misbehavior detection scheme has long been debated as it is difficult to maintain, update and use in VANETs [21], IDS is a more proper approach to protect VANETs against threats. It can detect attacks with a high accuracy and can protect the system from unknown attacks [23]. Several studies have been performed in the area of IDSs for VANETs. In [23,24], reputation scores and a framework of rule based detection have been used in IDSs. However, when the number of vehicles increases, its performance declines evidently, resulting in a longer detection time, high frequency of false alarms and heavier overhead. Although rule based IDS has high detection accuracy and efficiency, it just can detect known attacks and is invalid to unknown attacks [25,26]. In [27,28], the authors propose a watchdog for intrusion detection in VANETs. The former monitors all packets to decide if an attack is under progress. The latter monitors both the number of request to send/clear to send (RTS/CTS)

requests of the watchdogs and the detected vehicles at the MAC layer. The main shortcoming of this approach is that the misbehaving vehicles may be rewarded instead of being punished because they will no longer forward the packets from other vehicles but forward their own packets. In [21,29,30], statistic-based methods have been used for anomaly detection, which can detect attacks accurately and protect VANETs from unknown attacks. When using these statistic-based methods, the distraction of data has to be known in advance. However, it is usually difficult to get know. In addition, these statistic-based methods can only handle one feature at a time. In other words, when dealing with more than one feature, these methods have to be executed multiple times. In [31,32], support vector machine (SVM) based IDSs have been proposed. They use SVM-based classifier to monitor vehicles in order to classify vehicles either cooperative or malicious. These SVM-based detection mechanisms are placed in a single resource-constrained vehicle, which may result in overload because it needs to gather, propagate, store and analyze the training data sent out from a large number of vehicles around. In [27], authors propose an IDS which uses Bayesian game-theoretic methodology to switch its status (active or idle) to reduce overhead and detection time. Unfortunately, it is not able to detect the attacks from the vehicles that appear during the idle status of IDS.

To address the above challenges, we propose FM-HMM for IDS. Our proposed work is based on IDS that does not use trust or reputation and only relies on the analysis of the received data to detect intrusions in VANETs. It should be noted that the proposed FM-HMM is effective to almost all kinds of IDSs, which means FM-HMM is able to improve the performance of general IDSs. In addition, compared to the traditional IDSs, the IDS with FM-HMM has lower overhead and detection time, and there is almost no difference in their detection rates. The main reason is that FM-HMM uses HMM to filter vehicle messages and avoids several time-consumption procedures, such as feature extraction and detection.

3. Preliminary and background

3.1. VANETs model

A universal VANET model can be figured out according to previous literature [21,30,33,34], as shown in Fig. 1. In this model, each vehicle is equipped with several devices, such as a GPS, a velometer, an IDS and others. The GPS can obtain vehicle position, the velometer is used to measure vehicle speed and the IDS is used to detect attacks. There are three roles in vehicular communication, i.e., own vehicle, neighbor vehicle and target vehicle. The own vehicle represents the considered vehicle itself. The neighbor vehicles are the vehicles nearby with which the own vehicle can communicate directly. The target vehicle is a special one of the neighbor vehicles, whose message is being processed by the own vehicle. Moreover, each vehicle has several tables to store the messages from neighbor vehicles and the items within these tables are used for decision making to improve driving experience. In the following subsections, the details of the VANET model are provided.

3.1.1. VANET measurements

As shown in Fig. 1, the own vehicle can get the number of neighbor vehicles on the highway by checking their IDs in messages to obtain the density of vehicle ($Density_{own}$). Also, the own vehicle can easily get its speed by the velometer ($Speed_{own}$) embedded inside. By the GPS, the own vehicle can acquire its vehicle position, Pos_{own} ($Xpos_{own}$ and $Ypos_{own}$). Based on Green-shield's model [21] and free space model [35], the own vehicle can obtain average traffic flow ($AvgFlow_{own}$) and the distance between itself and the neighbor vehicle or between itself and the target vehicle ($D_{o\&n}$ or $D_{o\&t}$) respectively. Readers can refer to [21,34] and [35] for the detail of Green-shield's model and free space model.

According to the Green-shield's model, Eqs. (1) and (2) can be obtained, where *MaxSpeed* is the free flow speed when the density is zero and *MaxDensity* is the point at which the speed becomes zero and vehicles are stuck in a traffic jam. As a result, $AvgFlow_{own}$ can be derived as shown in Eq. (3), where $AvgFlow_{neg}{}'$ are the average traffic flow of neighbor vehicles at last moment and $n-1$ is the number of neighbor vehicles at the same moment.

$$Speed_{own} = MaxSpeed - \frac{Density_{own}}{MaxDensity} MaxSpeed \tag{1}$$

$$Flow_{own} = Speed_{own} \times Density_{own} \tag{2}$$

$$AvgFlow_{own} = \frac{1}{n}\left(\sum_{i=1}^{n-1} AvgFlow_{neg}{}' + Flow_{own}\right) \tag{3}$$

Base on the free space model [36], the own vehicle can calculate the distance between itself and one neighbor vehicle or between itself and its target vehicle by Eq. (4), in which RSS_j, WL_j and SP_j are the received signal strength, the wave length and the sending power of neighbor vehicle (or target vehicle) respectively.

$$D_{i\&j} = \sqrt{\frac{SP_j \times {WL_j}^2}{(4\pi)^2 RSS_j}},\ i = o\,(own)\,,\ j \in \{n\,(neg)\,,\ t(tag)\} \tag{4}$$

3.1.2. Message format

For communication with the neighbor vehicles, the own vehicle continuously broadcasts a beacon message (*BeaconMsg*) in the same time interval (*BeaconT*) or and an emergency message (*EmergencyMsg*) once it meets any accident. The formats of the two messages are shown in Eqs. (5) and (6), where ID_{own} is identity of the own vehicle and $Type_{own}$ is the type of emergency. It should be noted that, the own vehicle transmits ID_{own}, $Type_{own}$, $Density_{own}$, $Speed_{own}$, Pos_{own} and $AvgFlow_{own}$, to the neighbor vehicles and they will become ID_{neg}, $Type_{neg}$, $Density_{neg}$, $Speed_{neg}$, Pos_{neg} and $AvgFlow_{neg}$ at the end of the neighbor vehicles.

$$BeaconMsg(ID_{own}, Density_{own}, Speed_{own}, Pos_{own}, AvgFlow_{own}) \tag{5}$$

$$EmergencyMsg(ID_{own}, Type_{own}, Density_{own}, Speed_{own}, Pos_{own}, AvgFlow_{own}) \tag{6}$$

When the own vehicle needs to know the position of its target vehicle, it will broadcast request message (*RequestMsg*) as Eq. (7), where ID_{tag} is the identity of the target vehicle. Once its neighbor vehicles receive *RequestMsg*, they broadcast their positions and the distance between themselves and the target vehicle. During a certain waiting period, *WaitingT*, which is shorter than *BeaconT*, the own vehicle can receive response messages (*ResponseMsg*) which format is shown in Eq. (8), where $D_{n\&t}$ is the distance between the neighbor vehicle and the target vehicle.

$$RequestMsg(ID_{own}, ID_{tag}) \tag{7}$$

$$ResponseMsg(ID_{neg}, ID_{tag}, Pos_{neg}, D_{n\&t}) \tag{8}$$

3.1.3. Information tables

For storing the messages within a communication window, each vehicle has three tables, i.e., the current neighbor table, the last neighbor table and the position table, as shown in Fig. 1. The two neighbor tables are used to store *BeaconMsg* and *EmergencyMsg*, and the position table is used to store *ResponseMsg*. The lifespan of these items in the three tables is a constant. When time is end, the items in the last neighbor table and the position table are deleted. Then, the items in the current neighbor table are moved to the last neighbor table and are changed to $ID_{neg}{}'$, $Type_{neg}{}'$, $Density_{neg}{}'$, $Speed_{neg}{}'$, $Pos_{neg}{}'$ and $AvgFlow_{neg}{}'$.

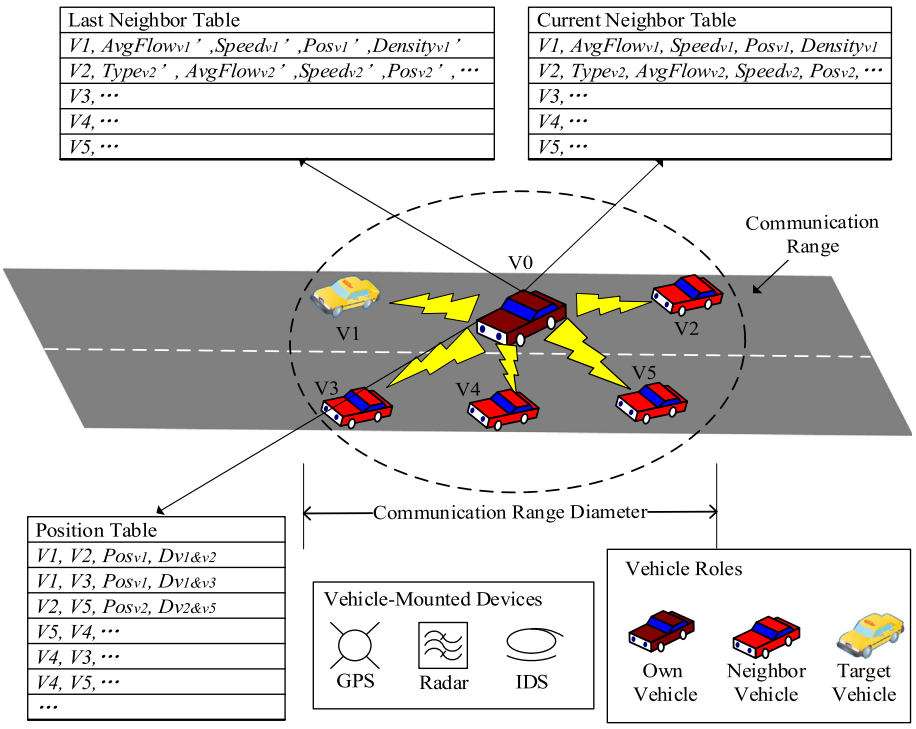


Fig. 1. VANETs model on the highway.

3.2. Attack model

Since IDS deals with the security of safety-oriented VANETs by preventing the occurrence of certain cyber-attacks, in the following, we will examine two important attacks, i.e., false information attack and Sybil attack. (i) False Information Attack [4]: Malicious vehicles can inject false messages into the network either on the purpose with malicious intent or due to faulty sensors, which can cause serious damage to the network. Under extreme conditions, the network can even be paralyzed. For example, malicious vehicles can falsify the low values of traffic flow to slow down the speed of their neighbor vehicles, and then they are able to surpass other vehicles easily. (ii) Sybil Attack [5]: Sybil attacks refer to the malicious vehicles that illegitimately use multiple identities. In VANETs, vehicles usually discover new neighbor vehicles by periodically broadcasting beacon messages. However, given the invisible nature of wireless communication, a malicious vehicle can easily claim multiple identities without being detected. For instance, a malicious vehicle can create false congestion condition by broadcast a lot of false *BeaconMsg* to intercept a certain vehicle for its selfish purposes.

Here, a vehicle can assume N states, where at each state the vehicle could act normally or maliciously. Therefore, different attack models can be defined [27]: (i) Transitory misbehavior: The state of vehicle oscillates between legitimate and malicious in order to avoid being detected or other selfish intents. (ii) Permanent misbehavior: A vehicle persists to act maliciously and does not switch to legitimate state. Such misbehavior is considered as a lethal attack, and cause chaos in the transportation network by creating traffic jams, accidents.

To implement the above-mentioned attack model, there are two different scenarios. First, it is the normal scenario without any rogue vehicle in order to collect legitimated data to train IDS. After training, the IDS can detect attacks by recognizing the deviations from normal scenario. Second, it is the rogue scenario, in which a certain percentage of malicious vehicle (10%–40%) transmits attacks to their neighbor vehicles, while the rest still transmit legitimated messages. For the vehicles that broadcast attacks to others, there are different modes, i.e., transitory mode and permanent mode. Some of them persist to broadcast malicious messages to other, while others oscillate between legitimate and malicious. Furthermore, in order to transmit false information attack, a malicious vehicle will falsify the low values of traffic flow and send them to neighbor vehicles. To launch a Sybil attack, a malicious vehicle creates a lot of false *BeaconMsg* and *EmergencyMsg*, and then broadcasts these forged messages to neighbor vehicles.

3.3. The detection center of IDS

Here, an anomaly detection mechanism is used in the detection center of IDS, which relies on a growing hierarchical self-organizing map (GHSOM) that can achieve both the high accuracy of classification and the low overhead of computation [37]. The procedure of the detection center is provided as follow. Interested readers can refer to [38,37,39,40] for more details.

As shown in Fig. 2, in either of the training phase or the testing phase, once IDS receives a VANETs message, *VTMsg*(*BeaconMsg* and *EmergencyMsg*) from a target vehicle, the feature extraction module extracts a feature vector from *VTMsg* with help of the last neighbor table and the position table and then sends the feature vector and *VTMsg* to the classifier module. If the IDS needs assistance from neighbor vehicles in feature extraction process, it will broadcast *RequestMsg* and wait for *ResponseMsg* from the neighbor vehicles within *WaitingT*. Next, in the training phase, the feature vector is just used to train the classifier and *VTMsg* is added into the current neighbor table. In the testing phase, the classifier which has been trained checks if any deviation exists in *VTMsg* when it receives the feature vector. If the classifier judges that there is not any deviation in the feature vector, *VTMsg* is accepted and added to the current neighbor table. Otherwise, it is rejected and the classifier sends ID of this target vehicle to the response center to take corresponding actions.

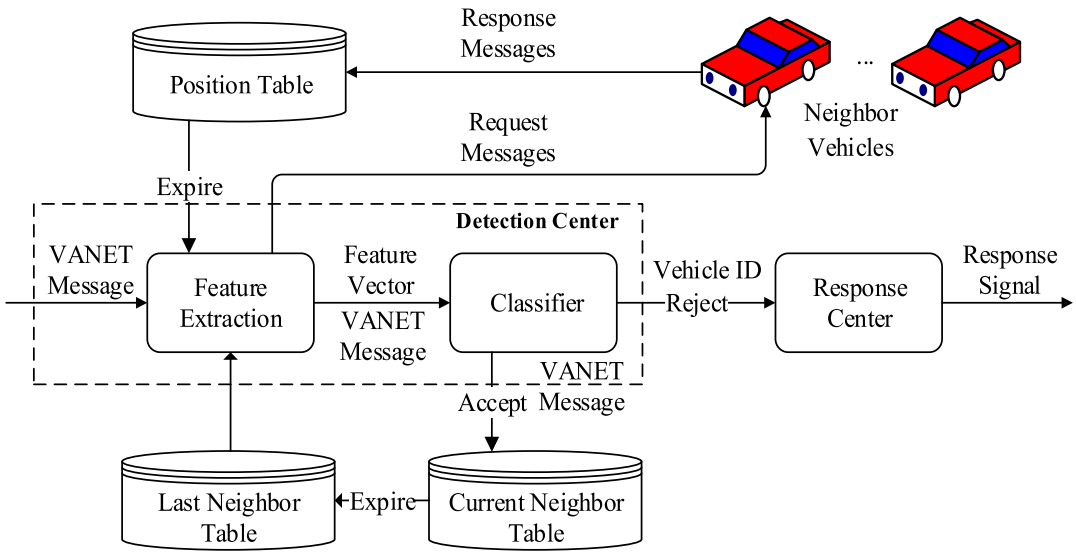


Fig. 2. The detection center in IDS.

4. The proposed FM-HMM for IDS

Although IDS is a reliable approach to protect VANETs against attacks, it will incur non-ignorable detection time and overhead once it is active to extract the features from vehicle messages and to classify these features [27]. These processes usually lead to long detection time and heavy overhead because they must exchange messages with other vehicles. The delay and collision of messages consume a lot of time in messages exchange process. Therefore, a tradeoff among accuracy, detection time and overhead should be considered. Thus, we have FM-HMM proposed, which can forecast the future state of neighbor vehicles to quickly filter the messages from these vehicles. In other words, FM-HMM is able to reduce the consumption of complex detection processes, such as feature extraction and classification.

In the following, for a better description of FM-HMM, the state pattern of a vehicle based on HMM is firstly described, which is the basics of the filter model. And then, FM-HMM is presented in detail. To the best of our knowledge, in this paper, we have produced the first work in the literature to use Markov model as the filter model to reduce the detection time and overhead of IDS in VANETs.

4.1. The vehicle's state pattern based on HMM

HMM is a statistical Markov model, by which the system being modeled is assumed to be a Markov process with unobserved states [38]. A HMM describes the joint probability of a collection of "hidden" and observed discrete random variables. It relies on the assumption that the t*th* hidden variable given the (t-1)*th* hidden variable is independent of previous hidden variables, and the currently observed variables depend only on the currently hidden state [41,42].

Here, we consider that the time axis is divided into equal time slots, which correspond to the time intervals between two continuous vehicle states. Let the state pattern of a vehicle be X_t, where t denotes the time instant. The security state of the vehicle, such as {'safe' and '*compromised*'}, can be divided into N discrete levels S: $\{s_1, \ldots, s_N\}$. The state transition from one state to another is shown in Fig. 3. The states s_t evolve following N-state Markov chains with state transition probability matrix $A(a_{ij})$ which is defined as Eq. (9), where S is the space of state.

$$A(a_{ij}) = \left[a_{ij}\right]_{i,j\in S}, \; where \; a_{ij} = \mathrm{P}\left(X_t = j | X_{t-1} = i\right) \tag{9}$$

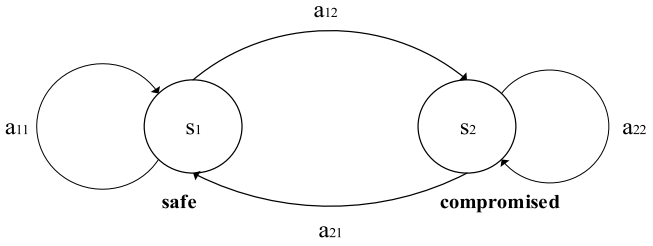


Fig. 3. The state transition of a vehicle.

In VANETs, the state of a vehicle could not be observed directly, thus it is HMM as shown in Fig. 4. The observation of the states can be represented by the detection results of IDS. The detection results, such as {'*normal*', '*abnormal*' and '*uncertain*'}, can be divided into M discrete levels O: $\{o_1, \ldots, o_M\}$. Let the observation of the state be Y_t at time slot t. If the security state X_t is equal to j at time slot t, the probability of obtaining observation, $b_j(y_t)$, is denoted as Eq. (10), where O is the state space of the observations of vehicle's state. The observation matrix $B(b_j(y_t))$ is defined as Eq. (11), which denotes Y_t acquired when X_t is picked at time t. Therefore, HMM is defined as $\lambda = (A, B, \theta)$, where θ is the probability vector of initial state. According to λ, predicted chain $Y_p = \{Y_{p_1} = y_{p_1}, Y_{p_2} = y_{p_2}, \ldots\}$ can be produced.

$$b_j(y_t) = \mathrm{P}\{Y_t = y_t | X_t = i\}, \text{ where } i \in S, y_t \in O \tag{10}$$

$$B(b_j(y_t)) = diag[b_1(y_t), \ldots, b_N(y_t)] \tag{11}$$

4.2. FM-HMM

As shown in Fig. 5, FM-HMM consist of three modules, i.e. schedule, update and filter, and two tables, which are observation table and forecast table. The observation table is used to store observed chain $Y_o = \{Y_{o_1} = y_{o_1}, Y_{o_2} = y_{o_2}, \ldots, Y_{o_T} = y_{o_T}\}$, where $y_{o_i}, y_{o_i} \in O$ is the detection result of IDS and T is the length of observed chain for each neighbor vehicle. The forecast table is used to store HMMs ($\lambda = (A, B, \theta)$) and theirs parameters, i.e., ID_{neg}, Y_p, the current step of a observed chain (π), and the error number of forecast (δ). The item of observation table and forecast table are shown as Eqs. (12) and (13) respectively. It has to be noted that the two tables are empty in the beginning, which means all items in the two tables are acquired by the online learning of IDS. The function

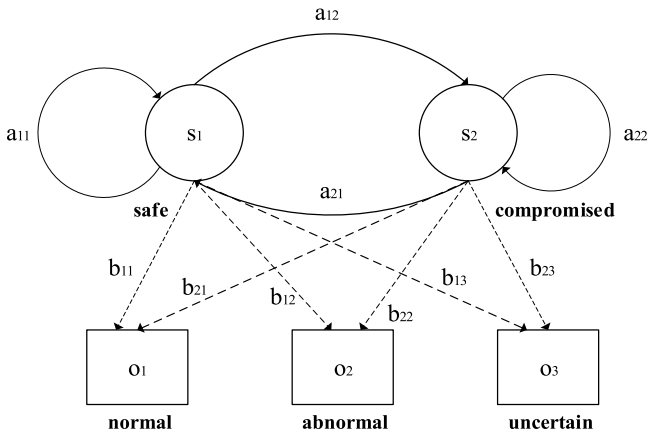


Fig. 4. The hidden Markov model of vehicle's states.

of the schedule module is to produce λ and its parameters for each neighbor vehicle using the data from the observation table. The filter module filters the messages from each neighbor vehicle by its λ and parameters if they exist. According to the result from the filter module, the update module can update λ and its parameters for each neighbor vehicle in the forecast table.

$$ObItem(ID_{neg}, Y_o = \{Y_{o_1} = y_{o_1}, \ldots, Y_{o_T} = y_{o_T}\}) \quad (12)$$

$$ForeItem(ID_{neg}, \lambda = (A, B, \theta), Y_p = \{Y_{p_1} = y_{p_1}, Y_{p_2} = y_{p_2}, \ldots\}) \quad (13)$$

The procedure of FM-HMM is presented in Fig. 5 and Algorithm 1. Once a VANETs message, *VTMsg* (*BeaconMsg* or *EmergencyMsg*), is received, it is handled by FM-HMM. First, the filter module checks the forecast table according ID_{neg} of the received *VTMsg* to find out λ and its parameters. Then, the filter module handles *VTMsg*. If λ and X_p exist, the filter signal (*IFF*) is equal to '*Filtered*', which indicates successful filtering. The filter result (*FRe*) is sent to both of the response center and update module. In the response center, corresponding response signals are produced to protect VANETs. When the *FRe* arrives at the update module, the update function will be active, which can update λ and its parameters, and then updates them in the forecast table. Otherwise, *IFF* is equal to '*Unfiltered*', which indicates failed filtering. The detection center will receive *VTMsg* from the filter module, and then detection result (*DRe*) is produced. *DRe* is sent to the response center to carry out corresponding actions, and (ID_{neg}, *DRe*) is sent to observation table to produce λ and its parameters. Once Y_o of any neighbor vehicle is enough, they will be sent to the schedule module to produce this vehicle's λ and its parameters.

We have an example to describe the operation of FM-HMM more intuitively. V0 is a vehicle running on the highway. V1 as a newcomer just enters into the communication range of V0, which implies that V0 and V1 can transmit their messages to each other. At first, when V0 got messages from V1, V0 cannot filter the messages due to lack of the history of V1, thus V0 have to use detection center to detect the messages by several time-consumption procedures, including feature extraction and detection. Once V0 obtain enough messages from V1, it can build a HMM for V1, and then use the HMM to quickly filter the subsequent messages from V1. When the error of filter is beyond threshold, V0 abandons the HMM. The process will be repeated if V1 still in the communication range of V0.

In the following, the main parts of FM-HMM, i.e., filter, update and schedule modules, are described in detail.

4.2.1. The procedure of filter module

Filter module is one of important modules in FM-HMM, which can quickly identify the states of neighbor vehicles by HMM. It can reduce the detection time and overhead of IDS without time-consuming detection processes, such as feature extracting and classification.

As shown in Algorithm 2, the filter module firstly checks whether λ is empty. If it is empty, it means there is no enough Y_o to produce HMM. Thus, *IFF* and *FRe* will be assigned '*Unfiltered*' and '*empty*' respectively to inform the other modules of IDS that filtering is unsuccessful. Otherwise, it needs to be identified whether π already reaches the maximum step or not. If $\pi \leq Length\left(C_p\right)$, *IFF* will be assigned '*Filtered*' and then *FRe* will be assigned Y_{p_π} (Y_{p_π} is the πth element in Y_p). If $\pi > Length\left(Y_p\right)$, *IFF* and *FRe* will be assigned '*Unfiltered*' and '*empty*' respectively.

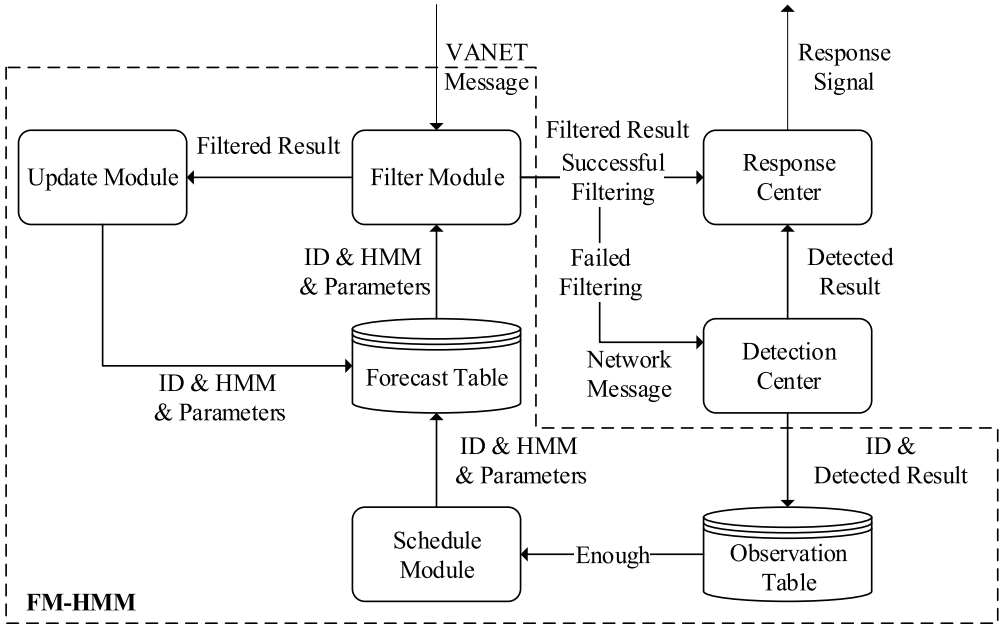


Fig. 5. The scheme of the IDS with FM-HMM.

Algorithm1
FM-HMM procedure

Input: $VTMsg$
Output: $RepSigal$
1: **for** each $VTMsg$ **do**
2: Acquire λ, Y_p, π, δ by ID_{neg} of $VTMsg$ from forecast table
3: $(IFF, FRe) = \mathtt{Filter}(\lambda, Y_p, \pi)$
4: **if** (IFF = '$Filtered$') **then**
5: $(Y_p, \pi, \delta, \lambda) = \mathtt{Update}(VTMsg, FRe, Y_p, \pi, \delta, \lambda)$
6: Update (ID_{neg}, λ, C_p, π, δ) of forecast table
7: Send FRe to response center to produce $RepSigal$
8: **else**
9: Calculate DRe by detection center
10: Send DRe to response center to produce $RepSigal$
11: Send (ID_{neg}, DRe) to observation table
12: **if** any Y_o of ID_{neg} is enough **then**
13: $(Y_p, \pi, \delta, \lambda) = \mathtt{Schedule}(Y_o)$
14: Add (ID_{neg}, λ, C_p, π, δ) to forecast table as a new item
15: **end if**
16: **End if**
17: **End for**

Algorithm2
Filter module (λ, Y_p, π)

Input: λ, Y_p, π
Output: IFF, FRe
1: **if** $\lambda \neq \emptyset$ **then**
2: **if** $\pi \leq Length(Y_p)$ **then**
3: $IFF = 'Filtered'$
4: $FRe = Y_{p_\pi}$ % Y_{p_π} is the π-th element in Y_p
5: **else**
6: $IFF = 'Unfiltered'$, $FRe = 'empty'$
7: **end if**
8: **else**
9: $IFF = 'Unfiltered'$, $FRe = 'empty'$
10: **End if**

4.2.2. The procedure of update module

To guarantee the detection rate of filter, the update module is proposed in FM-HMM. A timeliness method is used to reduce the production of error forecast, which consists of two scenarios. First, when the number of error forecast is below a fix error threshold, which implies that the error forecast is mainly caused by computational errors, such as false positive and false negative. At that moment, we need to use HMM to amend its observation chain. Second, when the number of error forecast is over the error threshold, which implies that the state pattern of a neighbor vehicle has changed and its HMM is out of date. In such case, the HMM and its parameters should be abandoned.

The procedure of the update module is shown in Algorithm 3. At the beginning, the values between *FRe* and *DRe* are compared. If *FRe* is equal to *DRe*, which means the result of filter is correct, and the step of Y_p increases as $\pi = \pi + 1$. Then, if $\pi > Length(Y_p)$, Y_p are reproduced by λ. However, if $FRe \neq DRe$, there are two scenarios in following processes. When δ is larger than the threshold of error number (*MaxError*), it means λ is out of date. Thus, λ and its related parameters (π, δ, C_p) are abandoned. If $\delta < MaxError$, Y_p have to be replaced by *DRe* and the elements behind Y_{p_π} have to be reproduced by λ. The procedure of chain reproduction is shown in Fig. 6. Here, we have to note that both the detection center and the update module use the same detection mechanism. However, the update module does not incur any additional detection time and overhead because it does not affect the modules of message handling in FM-HMM.

4.2.3. The procedure of schedule module

In the schedule module, Baum–Welch algorithm [43] is used to find the unknown parameters of a hidden Markov model including the state probability associated with hidden states and observation probability. The algorithm uses the well-known expectation maximization (EM) algorithm [44] to find maximum likelihood estimate of the parameters of a HMM based on a given observation chain that consist of a series of observed states.

Algorithm3

Update module($VTMsg$, FRe, Y_p, π, δ, λ)

Input: $VTMsg$, FRe, Y_p, π, δ, $\lambda = (A, B, \theta)$

Output: Y_p, π, δ, λ

1: Calculate DRe of $VTMsg$ by the detection mechanism of update module
2: **if** $FRe = DRe$ **then**
3: $\pi = \pi + 1$
4: **if** ($\pi > Length(Y_p)$) **then**
5: Reproduce Y_p by $\lambda = (A, B, \theta)$
6: $\pi = 0$, $\delta = 0$
7: **End if**
8: **else**
9: **if** $\delta > MaxError$ **then** % $MaxError$ is the threshold of error number
10: Abandon λ and its parameters (π, δ, Y_p)
11: **else**
12: $Y_{p_\pi} = DRe$ % Y_{p_π} is the π-th element in Y_p
13: Reproduce the elements that behind Y_{p_π} in Y_p
14: $\delta = \delta + 1$, $\pi = \pi + 1$
15: **end if**
16: **End if**

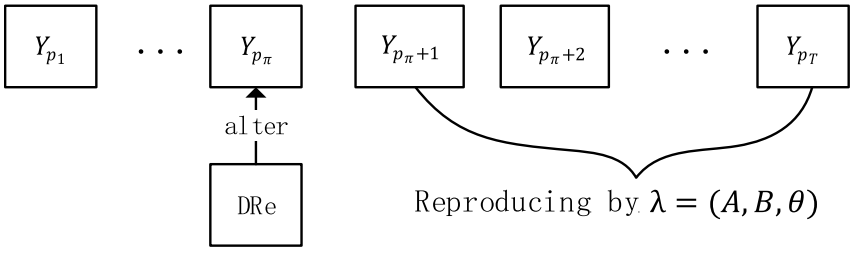


Fig. 6. The procedure of chain reproduction.

As shown in Algorithm 4, at steps 1, $\lambda = (A(a_{ij}), B(b_j(y_t)), \theta)$ is set with random initial conditions. Steps 2–10 is a loop, which consists of a forward procedure, a backward procedure and an update procedure. The loop is repeated iteratively until a desired level of convergence. First, the forward procedure is processed. Let $\alpha_i(t) = P(Y_{o_1} = y_{o_1}, \ldots, Y_{o_t} = y_{o_t}, X_t = i|, \lambda)$ that is the probability both at $y_{o_1}, y_{o_2}, \ldots, y_{o_t}$ and in state i at time t. It can be found in Eq. (14), where N is the number of vehicle state.

$$\alpha_i(t) = \begin{cases} \theta_i b_i(y_1), & t = 1 \\ b_i(y_{t+1}) \sum_{i=1}^{N} \alpha_j(t)\, a_{ji}, & t > 1 \end{cases} \tag{14}$$

Second, the backward procedure is processed. Let $\beta_i(t) = P(Y_{o_{t+1}} = y_{o_{t+1}}, \ldots, Y_{o_T} = y_{o_T}|X_t = i, \lambda)$ that is the probability of finding $y_{o_{t+1}}, y_{o_{t+2}}, \ldots, y_{o_T}$ given starting state i at time t. $\beta_i(t)$ can be calculated as Eq. (15).

$$\beta_i(t) = \begin{cases} \sum_{j=1}^{N} \beta_j(t+1)\, a_{ij} b_j(y_{t+1}), & t < 1 \\ \beta_i(T) = 1, & t = T \end{cases} \tag{15}$$

Third, the update procedure is processed. According to Bayes' theorem, the temporary variables $\gamma_i(t)$ and $\xi_{ij}(t)$ can be calculated as Eq. (16) and Eq. (17) respectively. $\gamma_i(t)$ is the probability of being in state i at time t given Y_o and λ. $\xi_{ij}(t)$ is the probability of being in state i and j at time t and t+1 respectively given Y_o and λ. Now, a_{ij} and $b_j(y_t)$ can be obtained using $\gamma_i(t)$ and $\xi_{ij}(t)$ as Eq. (18) and Eq. (19). In Eq. (19), $\varphi(y_t, v_k)$ is the expected times of that the y_t have been equal to v_k.

$$\gamma_i(t) = \frac{\alpha_i(t)\,\beta_i(t)}{\sum_{j=1}^{N} \alpha_j(t)\,\beta_j(t)} \tag{16}$$

$$\xi_{ij}(t) = \frac{\alpha_i(t)\,\beta_j(t+1)\,a_{ij} b_j(y_{t+1})}{\sum_{i=1}^{N}\sum_{j=1}^{N} \alpha_i(t)\,\beta_j(t+1)\,a_{ij} b_j(y_{t+1})} \tag{17}$$

$$a_{ij} = \frac{\sum_{t=1}^{T-1} \xi_{ij}(t)}{\sum_{t=1}^{T-1} \gamma_i(t)} \tag{18}$$

$$b_j(y_t) = \frac{\sum_{t=1}^{T-1} \gamma_i(t) * \varphi(y_t, v_k)}{\sum_{t=1}^{T-1} \gamma_i(i)},$$

$$\text{where } \varphi(y_t, v_k) = \begin{cases} 1 \text{ if } (y_t = v_k) \\ 0\,(otherwise) \end{cases} \tag{19}$$

At steps 12–13, for forecasting the states of a neighbor vehicle in the future, a new observation chain Y_p is produced by λ, and then zero is assigned to both π and δ.

5. Performance evaluation

5.1. Simulation setup

Our simulation is based on network simulator 2 (NS2) [45] and simulation of urban mobility (SUMO) [46]. The NS2, which has been

Algorithm4

Schedule module (Y_o)

Input: Y_o

Output: λ, Y_p, π, δ

Initiate $\lambda = (A(a_{ij}), B(b_j(y_t)), \theta)$ randomly
while $A(a_{ij})$, $B(b_j(y_t))$ doesn't converge **then**
  Obtain $\alpha_i(t)$ according to Y_o by Eq. (14)
  Obtain $\beta_i(t)$ according to Y_o by Eq. (15)
  Obtain $\gamma_i(t)$ according to λ, $\alpha_i(t)$ and $\beta_i(t)$ by Eq. (16)
  Obtain $\xi_{ij}(t)$ according to λ, $\alpha_i(t)$ and $\beta_i(t)$ by Eq. (17)
  for each a_{ij}, $b_j(y_t)$ in A and B **do**
    Update a_{ij} using $\gamma_i(t)$, $\xi_{ij}(t)$ by Eq. (18)
    Update $b_j(y_t)$ using $\gamma_i(t)$, $\xi_{ij}(t)$ by Eq. (19)
  End for
End while
Produce observation chain Y_p according to λ
$\pi = 0, \delta = 0$

Table 1
Simulation parameters.

Parameter	Value
Scenario	2 Lane Highway
Highway Length	5 km
Max vehicle speed	100 km/h
Wireless protocol	802.11p
Transmission range	500 m in each direction
Simulation time	165 s (35 s to 200 s)
Vehicle arrival interval	1 s
Transmission interval (*BeaconT*)	Every 0.3 s
The dimension of input vector of HMM	1 (Euclidean Space)
Input classes of HMM	Safe, Compromised
Output classes of HMM	Normal, Abnormal, Uncertain
Maximum error threshold (*MaxError*)	3
The length of observation chain (T)	57

highly validated by the networking research community, presents many well-developed low-layer protocols with easy programming interfaces. The SUMO is a software tool used to generate vehicular traffic by specifying the speed, types, behavior and number of vehicles. It can also set up road types and conditions. In our simulation, the SUMO is used to generate mobility trace files, and the NS2 is used to load these trace files and run the IDS with FM-HMM.

5.2. Experimental environment

The scenario is performed with the parameters shown in Table 1. In the simulations, vehicles can run on a 2-lane highway with a maximum speed of 100 km/h and each lane is 5 km length. Each vehicle can communicate with other vehicles within 500 m transmission range according to the IEEE 802.11p standard. To avoid generation of too much data in one simulation, we set the simulation time as 165 s, the vehicle inter-arrival interval as 1 s and transmission interval, i.e., *BeaconT* as 0.3 s Regarding to the parameters of HMM, the dimension of input vector is set as 1 and its type is Euclidean space. Input and output classes of HMM are {"safe", "compromised"} and {"normal", "abnormal", "uncertain"} respectively. After several runs of the simulation experiments, it is suitable to set the maximum error threshold (*MaxError*) as 3. The observation chain length (T) will be described in Section 5.6 in detail.

5.3. Evaluation metric

We have tested the IDS with FM-HMM by computing detection rate (DR), detection time (DT), and overhead (OV). It should be noted that DR, DT and OV are equally important in VANETs because these factors determine whether a vehicle can quickly recognize legitimate messages and react according to these messages. The metrics used are described below:

(1) DR: This is the detection rate of vehicles, i.e., the sum of the percentages of the malicious vehicles that are classified as malicious and the legitimate vehicles that are classified as legitimate. This is also referred to as accuracy and is calculated as Eq. (20).

$$DR = \frac{No.\ of\ vehicles\ been\ classified\ correctly}{No.\ of\ total\ vehicles} \tag{20}$$

(2) DT: Detection time is the average working time of the IDS with FM-HMM at each vehicle, which is given as Eq. (21). It is an important metric to measure the efficiency of IDS.

$$DT = \frac{\sum processing\ time\ of\ each\ vehicle}{No.\ of\ total\ vehicles} \tag{21}$$

(3) OV: Overhead (Kb/s) measures the amount of the messages that are handled by the IDS with FM-HMM in a unit time. It is another important metric to measure the efficiency of IDS. It is given as Eq. (22).

$$Overhead = \frac{No.\ of\ messages\ \times\ Bytes\ of\ a\ message}{running\ time\ \times\ 1000} \tag{22}$$

5.4. Effectiveness of FM-HMM

To demonstrate the effectiveness of FM-HMM, initially, we show whether the IDS with FM-HMM is able to detect attacks in VANETs or not. Then, we examine the DR, DT and OV in the IDSs with and without filter model to verify that FM-HMM can improve the performance of IDS.

5.4.1. Attack detection in VANETs

Here, we monitor the changes of the average flow of a randomly selected vehicle ($AvgFlow_{own}$) and its received flow from neighbor vehicles ($AvgFlow_{neg}$) under different scenarios, i.e. normal scenario (without malicious vehicles) and rogue scenario (involving malicious vehicles). First, data are gathered for the detection of

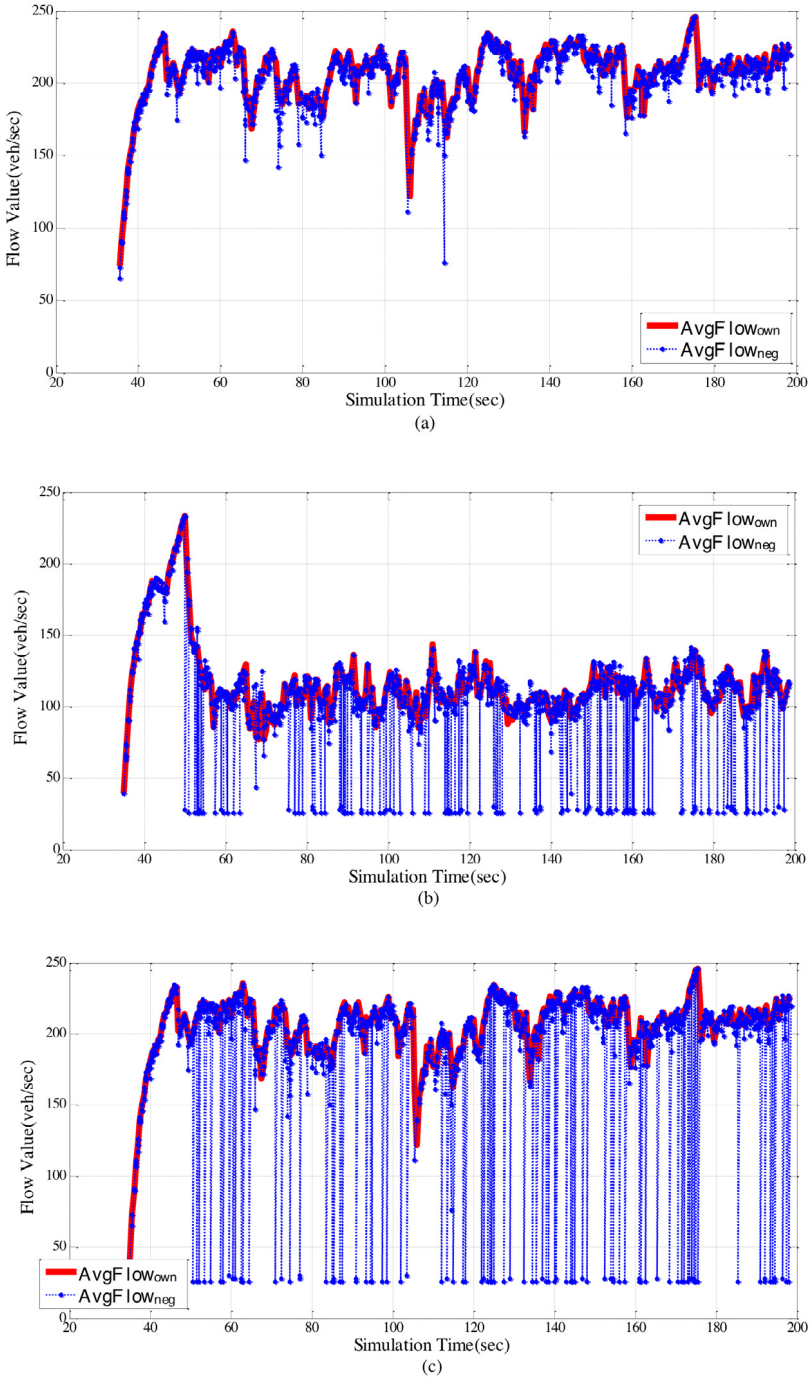


Fig. 7. Scenario: 30% of malicious vehicles and false messages triggered at $t = 50$ s (a) The vehicle under normal scenario, (b) The vehicle without IDS under rogue scenario, (c) The vehicle with IDS under rogue scenario.

deviation under the normal scenario. Furthermore, the rogue scenario is used in both cases of scenarios with and without the IDS.

In this simulation, $AvgFlow_{own}$ and $AvgFlow_{neg}$ are shown in Fig. 7. First, Fig. 7(a) shows the recorded data under the normal scenario. In this figure, both $AvgFlow_{own}$ and $AvgFlow_{neg}$ are quite close to each other because the own vehicle and its neighbor vehicles are close to each other in the same situations [21]. Then, 30% of malicious vehicles are inserted, which transmit the false low values of traffic flow (the blue dots that close to the bottom of Fig. 7(b) and (c)) after $t = 50$ s, and the results of $AvgFlow_{own}$ and $AvgFlow_{neg}$ have shown in Fig. 7(b) and (c). In the absence of the IDS with FM-HMM (Fig. 7(b)), $AvgFlow_{own}$ reduce as all messages are accepted and $AvgFlow_{neg}$ (the blue dots that close to the red curve) also reduce because they are also affected by the malicious vehicles. In other words, the own vehicle and its legitimate neighbor vehicles are affected by the malicious vehicles. On the contrary, as shown in Fig. 7(c), the IDS with FM-HMM can reject the false messages from the malicious vehicles so that the flow of the own vehicle and its legitimate neighbor vehicles are similar to that in Fig. 7(a). It means the IDS with FM-HMM is effective in the detection of attacks.

5.4.2. *Effectiveness of FM-HMM to IDS*

When the percentage of malicious vehicles increase from 10% to 40%, several comparisons have been carried out between the IDSs with FM-HMM and without filter. The IDS without filter is the same as the proposed IDS except it does not has FM-HMM, which means

Table 2
The comparison between the IDSs with and without filter model.

Malicious vehicle percentage	The IDS without filter model			The IDS with FM-HMM		
	DR	DT	OV	DR	DT	OV
10%	99.52	14.57	3.99	99.24	2.92	0.80
20%	99.33	14.72	4.04	98.77	3.67	0.97
30%	99.09	14.79	4.06	98.14	4.58	1.26
40%	98.76	14.64	4.02	97.38	5.71	1.58

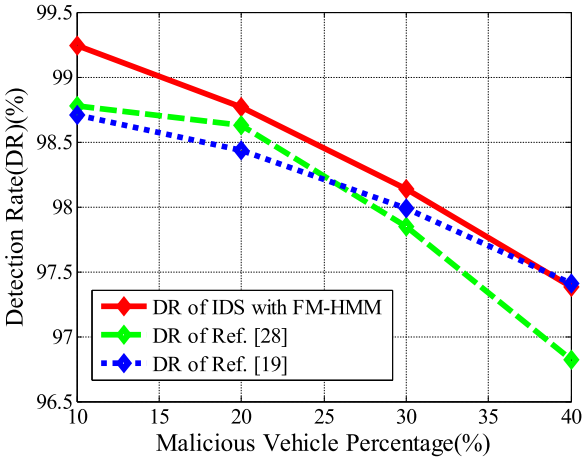


Fig. 8. The comparison of DRs of three IDSs.

the two IDSs have the same detection mechanism that is based on GHSOM. To compare with the IDS without FM-HMM, we can verify the efficiency of the proposed filter model. The results are shown in Table 2 which indicate that the performance of the IDS with FM-HMM is superior to that of another.

As shown in Table 2, the DR of the IDS with FM-HMM is just a little lower than that of the IDS without filter model. However, for the IDS with FM-HMM, its OV and DT get evident improvement. Even if malicious vehicles occupy 40% of the total vehicles, they are evidently better with nearly three times than that of the IDS without filter model. Therefore, FM-HMM is effective to reduce the OV and DT of IDS with only minor affection to the DR of IDS.

5.5. The effectiveness of the IDS with FM-HMM

5.5.1. The performance evaluation of the IDS with FM-HMM

In this section, the IDS with FM-HMM is compared with the IDSs in two previous literatures, i.e. [27,21]. The differences between the proposed method with Ref. [27,21] are the detection mechanism and FM-HMM. As described in Ref. [27], a novel framework is provided in IDS, which uses Bayesian game-theoretic methodology to switch the status of IDS between active and idle to reduce the detection time and overhead of IDS. In Ref. [21], a hypothesis testing is used to detect whether there is deviation in the messages of VANETs. It should be noted that we do not cite the results from Ref. [27,21], the IDSs in this paper and the two references are experimented in our simulation environment. We can demonstrate the IDS with FM-HMM has better performance than other recently proposed IDSs by comparing with Ref. [27,21].

The results of comparison are shown in Fig. 8. As noted in the figures, the DR of the IDS with FM-HMM is better than that in Ref. [27,21] from 10%–40% of malicious vehicles. It is because GHSOM is used to find the possible deviation of VANET message, which has been demonstrated to have higher classification accuracy in [38,37]. In addition, in FM-HMM, we provide HMM to accurately predict the states of vehicles as well as an update module to maintain the accuracy of forecast. Therefore, both accuracy and stability of the IDS with FM-HMM are better than that in Ref. [27,21].

Similarly, as shown in Fig. 9(a) and (b), the OV and DT of the IDS with FM-HMM are better than that in Ref. [27,21] when the malicious vehicles increase from 10% to 40%. The reasons are

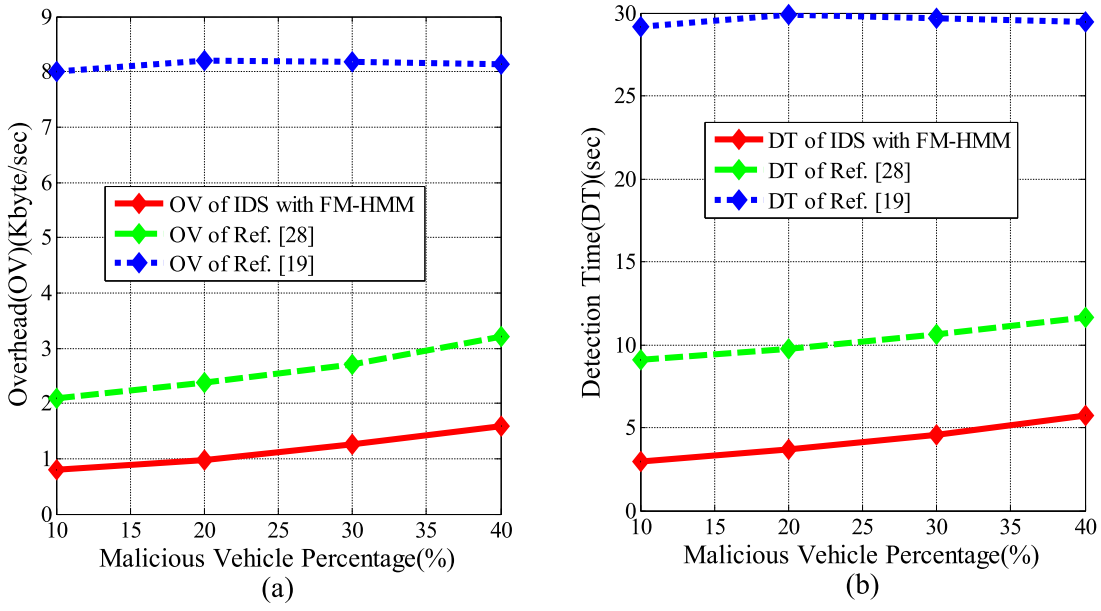


Fig. 9. The comparison among Three IDSs (a) the OVs among three IDSs, and (b) the DTs among Three IDSs.

Table 3
The comparison among the IDS with FM-HMM, Refs. [47,48].

Malicious vehicle percentage	The IDS with FM-HMM			Ref. [47]			Ref. [48]		
	DR	DT	OV	DR	DT	OV	DR	DT	OV
10%	99.24	2.92	0.80	98.32	54.74	8.02	97.60	14.68	3.98
20%	98.77	3.67	0.97	97.80	54.71	8.03	97.02	14.65	4.01
30%	98.14	4.58	1.26	97.27	54.69	8.07	96.51	14.73	4.01
40%	97.38	5.71	1.58	96.58	52.67	8.05	95.93	14.72	4.02

Table 4
The optimal length of observation chain in different malicious vehicle percentage.

Different measurements	Malicious vehicle percentage (%)			
	10%	20%	30%	40%
DR/OV	$T = 58$	$T = 57$	$T = 57$	$T = 56$
DR/DT	$T = 57$	$T = 57$	$T = 56$	$T = 57$

that Ref. [21] do not have any mechanism to reduce OV and DT, and the solution in Ref. [27] has to constantly switch statues between idle and active, which lead to a lot of computation resource consumption. For the IDS with FM-HMM, it not only uses FM-HMM to reduce OV and DT, but also avoids the additional computation for the switch of status. As a result, the efficiency of the IDS with FM-HMM are much lower than that in the two references.

5.5.2. The comparisons with other existing IDSs

To further demonstrate the performance of proposed method, the IDS with FM-HMM is compared with other existing researches, which are the IDSs of Refs. [47,48]. As described in Ref. [47], authors proposed a novel SVM-based IDS by combining dolphin swarm algorithm with SVM. It is claimed that the dolphin swarm algorithm is able to improve the performance of SVM-based IDS. In Ref. [48], a game theory based intrusion detection framework and a novel clustering algorithm have been developed for VANETs. By using their specification rules and a lightweight neural network based classifier for detecting malicious vehicle, the communication overhead of IDS is reduced.

The IDS with FM-HMM is compared with the IDSs of Ref. [47,48], and the results of DR, DT and OV are shown in Table 3 in which the percentage of rogue vehicles increases from 10% to 40%. As shown in the table, the DR of the IDS with FM-HMM is better than that of Ref. [47,48]. It is because that GHSOM used in our IDS can accurately judge if there is deviation in VANET messages compared with other neural networks [38,37]. For the DT and OV in Table 3, the two metrics of proposed method are much lower than that in Ref. [47,48]. This is because that FM-HMM can reduce several time-consumption procedures (feature extraction and detection) by predicting the states of vehicles to filter vehicle messages. Evidently, the performance of the IDS with FM-HMM is the best, which not only achieves higher classification accuracy but also requires lower computational complexity.

5.6. The discussion of the important parameter

As described in Section 4, the DR, DT and OV are affected by T. When T is low, the DR of FM-HMM decreases as it does not have enough data to build appropriate HMM. On the contrary, when T is high, the DT and OV of FM-HMM are too high. Thus, it is important to find the optimal values of T.

Here, an example is taken under the situation of 30% of malicious vehicles. Fig. 10 shows the results when $T = 30, 40, 50,$ and 60. As shown in Fig. 10(a) and (b), it is clear that beyond the point $T = 57$, the DR increases more slowly, while the OV and DT

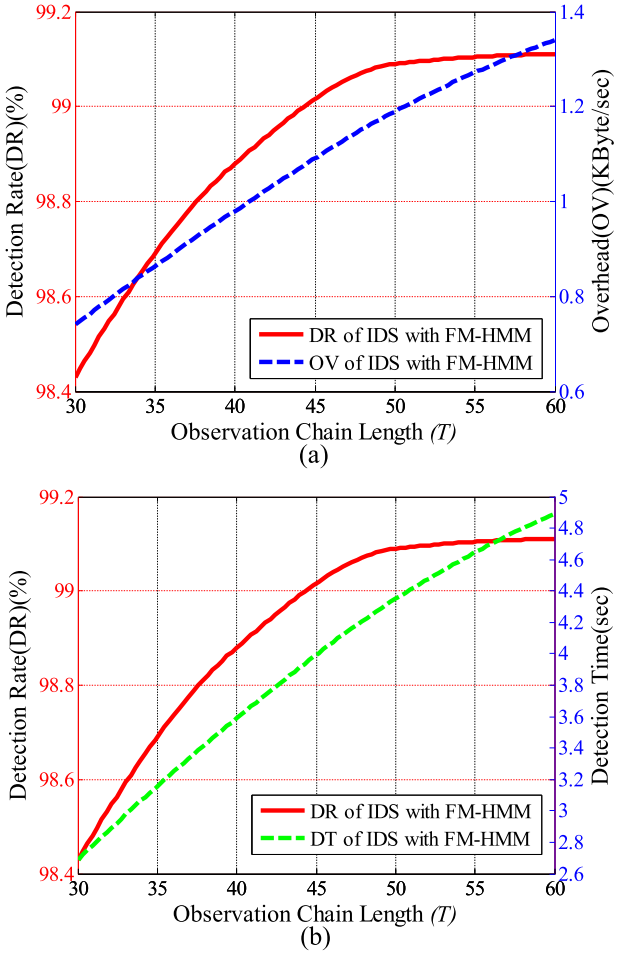


Fig. 10. The selection of FM-HMM parameter (a) The relation between DR and OV, (b) The relation between DR and DT.

increase more rapidly. Therefore, we identify the point, $T = 57$, as an optimal point. In addition, the optimal T in different percentages of malicious vehicles are shown in Table 4.

6. Conclusion

In this paper, we have proposed FM-HMM to reduce the overhead and detection time of IDS in VANETs. The proposed FM-HMM is effective to almost all kinds of IDSs, which means FM-HMM is able to improve the performance of general IDSs. To the best of our knowledge, this is the first work in the literature to model the state pattern of each vehicle in VANETs as a HMM to quickly filter the messages from the vehicles instead of detecting these messages. In FM-HMM, schedule, filter and update modules are proposed to reduce overhead and the time for detection without impairing detection rate. Simulations show the network message congestion (e.g., broadcast storms) can be mitigated by the reduction

of message scale and the improvement of processing efficiency. Moreover, the performance of the proposed IDS is still remarkable even when up to 40% of vehicles are rogue vehicles.

References

[1] Pathan, Al-Sakib Khan, Security of Self-organizing Networks: MANET, WSN, WMN, VANET, CRC press, 2016.

[2] Nasrin Taherkhani, Samuel Pierre, Centralized and localized data congestion control strategy for vehicular ad hoc networks using a machine learning clustering algorithm, IEEE Trans. Intell. Transp. Syst. 17 (11) (2016).

[3] Jovan Radak, Bertrand Ducourthial, Detecting road events using distributed data fusion: experimental evaluation for the icy roads case, IEEE Trans. Intell. Transp. Syst. 17 (1) (2016).

[4] False Attack. [Online]. Available: https://en.wiktionary.org/wiki/falseattack, Accessed on: 05.05.16.

[5] J. Newsome, E. Shi, D. Song, A. Perrig, The Sybil attack in sensor networks: analysis & defenses, in: Proc. of the Third International Symposium on Information Processing in Sensor Networks, IPSN 2004, 2004, pp. 259–268.

[6] B. Parno, A. Perrig, Challenges in securing vehicular networks, in: Proc. of the Fourth Workshop on Hot Topics in Networks, HotNets-IV, 2005.

[7] Jinyuan Sun, Yuguang Fang, A defense technique against misbehavior in VANETs based on threshold authentication, in: Military Communications Conference MILCOM 2008, IEEE, 2008, pp. 1–7.

[8] S. Ruj, M.A. Cavenaghi, Z. Huang, A. Nayak, I. Stojmenovic, On data-centric misbehavior detection in VANETs, in: Proc. IEEE Veh. Technol. Conf. (VTC Fall), San Francisco, CA, USA, 2011, pp. 1–5.

[9] Al-Kahtani Mohammed Saeed, Survey on security attacks in Vehicular Ad hoc Networks (VANETs), in: Signal Processing and Communication Systems (ICSPCS), 2012 6th International Conference on, IEEE, 2012, pp. 1–9.

[10] A. Daeinabi, A.G.P. Rahbar, A. Khademzadeh, VWCA: An efficient clustering algorithm in vehicular ad hoc networks, J. Netw. Comput. Appl. 34 (1) (2011) 207–222.

[11] M. Raya, J-P. Hubaux, Securing vehicular adhoc networks, journal of computer security, Spec. Issue Secur. Ad Hoc Sens. Netw. 15 (1) (2007) 39–68.

[12] P. Papadimitratos, A. Kung, J.P. Hubaux, F. Kargl, Privacy and identity management for vehicular communication systems: a position paper, in: Workshop on Standards for Privacy in User-Centric Identity Management, Zurich, Switzerland, 2006.

[13] R. Lu, X. Lin, H. Zhu, P.H. Ho, X. Shen, ECPP: Efficient conditional privacy preservation protocol for secure vehicular communications, in: Proc. IEEE 27th Conf. Comp. Commun. 2008, pp. 1229–1237.

[14] D. Huang, S. Misra, G. Xue, M. Verma, PACP: An efficient pseudonymous authentication based conditional privacy protocol for VANETS, IEEE Trans. Intell. Transp. Syst. 12 (3) (2011) 736–746.

[15] A.-N. Shen, S. Guo, D. Zeng, G. Mohsen, A lightweight privacy-preserving protocol using chameleon hashing for secure vehicular communications, in: Proc. IEEE Wireless Commun. Netw. Conf. (WCNC), 2012, pp. 2543–2548.

[16] R. Lu, X. Li, T.H. Luan, X. Liang, X. Shen, Pseudonym changing at social spots: an effective strategy for location privacy in VANETs, IEEE Trans. Veh. Technol. 61 (1) (2012) 86–96.

[17] P. Wex, J. Breuer, A. Held, T. Leinmuller, L. Delgrossi, Trust issues for vehicular ad hoc networks, in: Veh. Technol. Conf. (VTC Spring 2008), 2008, pp. 2800–2804, 11–14.

[18] U. Minhas, J. Zhang, T. Tran, R. Cohen, Towards expanded trust management for agents in vehicular ad hoc networks, Int. J. Comput. Intell. Theory Pract. 5 (1) (2010) 3–15.

[19] A. Patwardhan, A. Joshi, T. Finin, Y. Yesha, A data intensive reputation management scheme for vehicular ad hoc networks, in: Proc. 3rd Annual Int. Conf. Mobile Ubiquitous Systems, 2006, pp. 1–8.

[20] L. Qin, A. Malip, K.M. Martin, S. Ng, J. Zhang, A reputation-based announcement scheme for VANETs, IEEE Trans. Veh. Technol. 61 (2012) 4095–4108.

[21] Kamran Zaidi, Milos Milojevic, Host based intrusion detection for VANETs: a statistical approach to rogue node detection, IEEE Trans. Veh. Technol. (2016) http://dx.doi.org/10.1109/TVT.2480244.

[22] S. Ruj, M.A. Cavenaghi, Z. Huang, A. Nayak, I. Stojmenovic, On data-centric misbehavior detection in VANETs, in: Veh. Technol. Conf. (VTC Fall), 2011 IEEE, IEEE, 2011, pp. 1–5.

[23] H. Sedjelmaci, S.M. Senouci, M. Feham, An efficient intrusion detection framework in cluster-based wireless sensor networks, Secur. Commun. Netw. (2013) 1211–1224.

[24] Hichem Sedjelmaci, Sidi Mohammed Senouci Ss, An accurate and efficient collaborative intrusion detection framework to secure vehicular networks, Comput. Electr. Eng. (2015) 33–47.

[25] Hichem Sedjelmaci, Sidi Mohammed Senouci, An efficient and lightweight intrusion detection mechanism for service-oriented vehicular networks, IEEE Internet Things J. 1 (6) (2014).

[26] Lynda Mokdad, Jalel Ben-Othman, DJAVAN: Detecting jamming attacks in vehicle ad hoc networks, Perform. Eval. (2015) 47–59.

[27] Hichem Sedjelmaci, Sidi Mohammed Senouci, Intrusion detection and ejection framework against lethal attacks in UAV-Aided networks: a Bayesian game-theoretic methodology, IEEE Trans. Intell. Transp. Syst. (2017).

[28] Baiad Raghad, et al., Cooperative cross layer detection for blackhole attack in VANET-OLSR, in: Wireless Communications and Mobile Computing Conference (IWCMC), 2014 International, IEEE, 2014.

[29] Bo Yu, Cheng-Zhong Xu, Detecting Sybil attacks in VANETs, J. Parallel Distrib. Comput. (2013) 746–756.

[30] Shihao Yan, Student Member, IEEE, Robert Malaney, Optimal information-theoretic wireless location verification, IEEE Trans. Veh. Technol. 63 (7) (2014).

[31] Wahab Omar Abdel, et al., CEAP: SVM-based intelligent detection model for clustered vehicular ad hoc networks, Expert Syst. Appl. 50 (2016) 40–54.

[32] Alheeti, Khattab M. Ali, Anna Gruebler, Klaus D. McDonald-Maier, On the detection of grey hole and rushing attacks in self-driving vehicular networks, in: Computer Science and Electronic Engineering Conference (CEEC), 2015 7th., IEEE, 2015.

[33] Saif Al-Sultan, Moath M. Al-Doori, A comprehensive survey on vehicular ad hoc network, J. Netw. Comput. Appl. (2014) 380–392.

[34] Greenshields model. [Online]. Available: http://www.webpages.uidaho.edu/niattabman-ual/chapters/trafficflowtheory/theoryandconcepts/GreenshieldsModel.html, Accessed on: 05.05.16.

[35] Free space Model, http://www.isi.edu/nsnam/ns/d-oc/node217.html.

[36] F. Kargl, P. Papadimitratos, L. Buttyan, M. Muter, E. Schoch, B. Wieder-sheim, Ta-Vinh Thong, G. Calandriello, A. Held, A. Kung, J-P. Hubaux, Secure vehicular communication systems: implementation, performance, and research challenges, IEEE Commun. Mag. 46 (11) (2008) 110–118.

[37] Yingying Zhu, Junwei Liang, Jianyong Chen, Zhong Ming, An improved NSGA-III algorithm for feature selection used in intrusion detection, Knowl.-Based Syst. 116 (2017) 74–85.

[38] T. Sim, S. Zhang, R. Janakriaman, S. Kumar, Continuous verification using multimodal biometrics, IEEE Trans. Pattern Anal. Mach. Intell. 29 (4) (2007) 687–700.

[39] T. Kohonen, Self-Organizing Maps, third ed., Springer, 2001.

[40] A. Rauber, D. Merkl, M. Dittenbach, The growing hierarchical self-organizing map: exploratory analysis of high-dimensional data, IEEE Trans. Neural Netw. (2002) 1331–1341.

[41] Shengrong Bu, F. Richard Yu, Structural results for combined continuous user authentication and intrusion detection in high security mobile ad-hoc networks, IEEE Trans. Wireless Commun. 10 (9) (2011).

[42] Jie Liu, F. Richard Yu, Optimal combined intrusion detection and biometric-based continuous authentication in high security mobile ad hoc networks, IEEE Trans. Wireless Commun. 8 (2) (2009).

[43] Jeff A. Bilmes, A Gentle Tutorial of the EM Algorithm and its Application to Parameter Estimation for Gaussian Mixture and Hidden Markov Models, International Computer Science Institute, CA, 1998, pp. 7–13.

[44] EM Algorithm. [Online]. Available: http://en.wikipedia.org/wiki/Expectation/maximiza-tion_algorithm, Accessed on: 05.05.17.

[45] NS2: Network Simulator. [Online]. Available: http://www.isi.edu/nsnam/ns/, Accessed on: 05.05.17.

[46] M. Behrisch, L. Bieker, J. Erdmann, D. Krajzewicz, Sumo simulation of urban mobility: an overview, in: SIMUL 2011, 3rd Int.Conf. on Advances in System Simulation, 2011, pp. 63–68.

[47] Sharma Sparsh, Ajay Kaul, Hybrid fuzzy multi-criteria decision making based multi cluster head dolphin swarm optimized IDS for VANET, Veh. Commun. 12 (2018) 23–38.

[48] Subba Basant, Santosh Biswas, Sushanta Karmakar, A game theory based multi layered intrusion detection framework for VANET, Future Gener. Comput. Syst. 82 (2018) 12–28.