Computers, Materials & Continua
DOI:10.32604/cmc.2021.016260
Article

An Efficient Impersonation Attack Detection Method in Fog Computing

Jialin Wan[1], **Muhammad Waqas**[1,2], **Shanshan Tu**[1,*], **Syed Mudassir Hussain**[3], **Ahsan Shah**[2], **Sadaqat Ur Rehman**[4] and **Muhammad Hanif**[2]

[1]Engineering Research Center of Intelligent Perception and Autonomous Control, Faculty of Information Technology, Beijing University of Technology, Beijing, 100124, China
[2]Faculty of Computer Science and Engineering, Ghulam Ishaq Khan Institute of Engineering Sciences and Technology, Topi, 23460, Pakistan
[3]Department of Electronics Engineering, FICT, Balochistan University of Information Technology, Engineering and Management Sciences, Quetta, 87300, Pakistan
[4]Department of Computer Science, Namal Institute, Mianwali, 42200, Pakistan
[*]Corresponding Author: Shanshan Tu. Email: sstu@bjut.edu.cn

Received: 28 December 2020; Accepted: 28 January 2021

Abstract: Fog computing paradigm extends computing, communication, storage, and network resources to the network's edge. As the fog layer is located between cloud and end-users, it can provide more convenience and timely services to end-users. However, in fog computing (FC), attackers can behave as real fog nodes or end-users to provide malicious services in the network. The attacker acts as an impersonator to impersonate other legitimate users. Therefore, in this work, we present a detection technique to secure the FC environment. First, we model a physical layer key generation based on wireless channel characteristics. To generate the secret keys between the legitimate users and avoid impersonators, we then consider a Double Sarsa technique to identify the impersonators at the receiver end. We compare our proposed Double Sarsa technique with the other two methods to validate our work, i.e., Sarsa and Q-learning. The simulation results demonstrate that the method based on Double Sarsa outperforms Sarsa and Q-learning approaches in terms of false alarm rate (FAR), miss detection rate (MDR), and average error rate (AER).

Keywords: Fog computing; double Sarsa; attack detection; physical layer key security

1 Introduction

Due to the growth of the Internet of Things (IoT) and mobile devices, many applications are sensitive to bandwidth and perceived latency [1,2]. Besides, the processing of these applications requires higher computation and communication costs [3]. In IoT, interconnected things generate a large amount of data. The data is usually processed in the cloud, prone to severe network congestion and load [4,5]. This is highly problematic, especially for the overall performance of time-sensitive applications and services. In particular, the recently developed 5G wireless networks have a higher demand for bandwidth, computing resource, security, and delay [6–9]. To address the problems mentioned above, fog computing was introduced to facilitate the computing, networking, and storage operations between end-users and cloud data centers. The added fog layer is located between the cloud and end-users. The primary function of fog computing is to process IoT data locally, thereby bringing the convenience of data storage, calculation, and transmission. It has the characteristics of low latency, location awareness, and large-scale IoT applications support [4]. However, due to its additional and unique features, it is faced with communication and data security issues.

In FC, most of the devices are connected wirelessly [10]. As a result, due to the wireless transmission medium's broadcast characteristics, there is a security risk in communication between fog nodes and end-users [11]. For example, the attackers may pretend to be legitimate fog nodes or end-users, thereby sending signals to other nodes so that the receivers may receive wrong signals. To prevent the attack from disguised attackers and ensure the wireless network's security in FC, both parties need to use a common key for encrypted transmission. Therefore, we propose the fusion of physical layer security (PLS) technologies based on the key generation and reinforcement learning algorithm to detect impersonation attacks.

On the one hand, PLS protects the information by exploiting the communications medium's intrinsic characteristics or principles and is a promising wireless security technique [12]. Thus, we employ physical layer key generation technology based on PLS. Unlike traditional cryptography methods with computational complexity, it can provide real-time, non-distribution key generation means for both communicating parties with channel coding, pre-coding, signal processing, etc. On the other hand, reinforcement learning is the training of machine learning algorithms to help make a sequence of decisions. In this scheme, a user learns to achieve a goal in an uncertain and potentially complex environment via trial and error [13]. The learning is based on a game-like situation where the users obtain rewards through continuous interaction with the environment and attain the optimal strategies [14]. Thus, we propose a detection method with a Double Sarsa algorithm to defend against impersonation attacks. The proposed detection method based on Double Sarsa [14] utilizes the channel state information (CSI). It can obtain the optimal test threshold value to detect impersonation attacks in the dynamic environment. The main contributions of this work are summarized below.

- We investigate to tackle the impersonation attacks between fog nodes and legitimate end-users in fog computing (FC). For this purpose, we apply a reinforcement learning algorithm to detect the impersonation attacks.
- We formulate a zero-sum game between the illegitimate node (fog node or end user) and receiver. We obtain the optimal threshold value to detect impersonation attacks by establishing a hypothesis test based on channel state information (CSI) at the receiving end.
- We validate the performance of the proposed method by comparing with Sarsa and Q-learning in false alarm rate (FAR), miss detection rate (MDR), and average error rate (AER) based on a test threshold value. We find out that the FAR, MDR, and AER are decreased significantly as the experiments progress based on the appropriate threshold value with the proposed method. This shows that our proposed technique has a better detection capability.

The rest of the paper is organized as follows. In Section 2, we discuss the related work. Sections 3 and 4 describe our proposed system and proposed method, respectively. Section 5 presents the performance evaluation of our proposed method. Finally, the paper is concluded in Section 6.

2 Related Works

Key generation technology exploits the randomness and reciprocity of wireless channels. In recent years, this approach is widely studied by many researchers. Zhang et al. [15] presented a survey of the key generation technologies related to the wireless channels. An information theory secret key generation (SKG) method is proposed in [16] for time-division duplexing (TDD) based orthogonal frequency-division multiplexing (OFDM) systems over multipath fading channels. In [17], the authors proposed a novel key generation scheme for wireless sensor networks, achieving a high key reconciliation rate. The authors in [18] developed an algorithm to protect the keys secret from eavesdropper and non-trusted selected relays, which improves the secret key generation rate (SKGR). FC cannot be considered fully secure, and it is faced with security vulnerabilities. To deal with this problem, different approaches have been proposed in the literature. Huang et al. [19] proposed an access tree structure and attribute-based signature technology in FC to implement ciphertext updating and computing outsourcing. To ensure the end-to-end security of IoT devices at the fog layer, Abbas et al. [20] proposed a novel fog security service based on two cryptographic schemes, i.e., identity-based encryption and identity-based signature. Tian et al. [21] proposed a three-layer storage framework to prevent an attack from the inside of the cloud server and protect data privacy. However, the traditional security methods are not suitable in a dynamic environment. These conventional methods have no self-learning ability and cannot make timely adjustments according to the changing environment, which causes difficulty in optimization.

Besides, reinforcement learning algorithms have been used by researchers to improve wireless network security. For instance, Xiao et al. [22] proposed a spoofing detection scheme based on Q-learning to detect spoofing attacks in wireless networks, and the efficiency of the proposed strategy is improved. In [23], the authors applied a Q-learning algorithm with reinforcement learning to detect the impersonation attacks, enhancing detection accuracy. However, these methods mentioned above only used one Q-table, which may make large changes in the threshold and slow convergence. Our proposed can solve the problem with two Q-tables, and the optimum test threshold value is obtained for each other's estimation errors. It has better performance in detecting impersonation attacks in FC security.

3 Proposed System

This section illustrates the impersonation attack model and the security model based on the key generation. First, we discuss the system architecture to consider legitimate and illegitimate users. Then, a key generation model is presented to discuss the process of secret keys.

3.1 System Architecture

As depicted in Fig. 1, we consider a three-layer system architecture model of the FC network, including the cloud layer, fog layer, and end-user layer. The fog layer is composed of fog nodes, which can be computers, smartphones, etc. The end-user layer includes end-user devices and IoT devices. In this regard, let consider there are $E = \{1, 2, \ldots, e\}, \forall e \in E$ transmitters. $F = \{1, 2, \ldots, f\}, \forall \in F$ are receivers in the network. For legitimate and non-legitimate users, we consider a set of $H = \{1, 2, \ldots, h\}, \forall h \in H$ and $I = \{1, 2, \ldots, i\}, \forall \in I$, respectively.

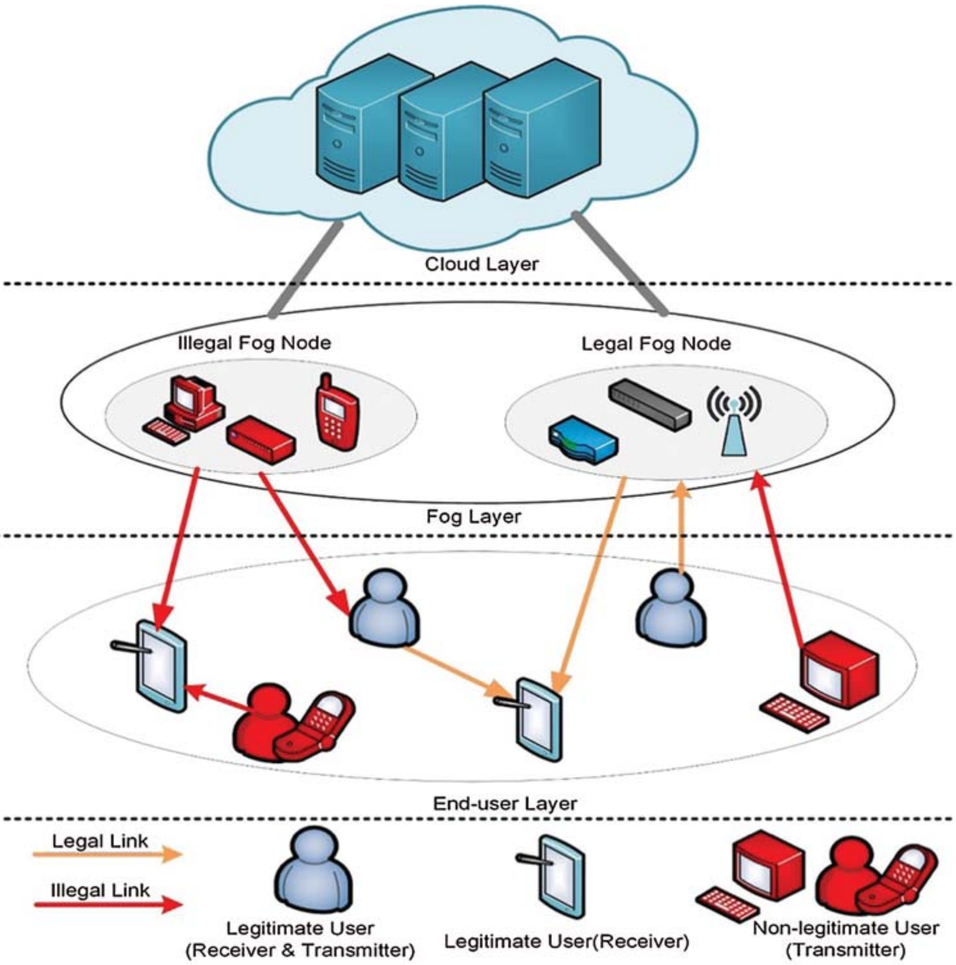


Figure 1: Impersonation attack network model

The fog nodes are used to provide computing, storage, and communication. They provide temporary storage and real-time analysis for terminal devices' data and send their data summary to the cloud. Illegal users pretend to be legitimate users to enjoy the fog node's services or pretend to be legal fog nodes to offer false services to end-users. These non-legitimate users (fog nodes or end-user nodes) claim to be legitimate and launch attacks on other network nodes. Consequently, it causes the wireless communication link between fog nodes and end-users' insecure.

3.2 Key Generation Model

As mentioned in the previous section, the purpose is to generate a random and secure key for secure communication between fog nodes and end-users [24]. Therefore, we exploit the physical layer channel parameter, i.e., CSI, to generate the secret keys. The proposed key generation model is depicted in Fig. 2. In this model, we assume that the fog nodes and the end-users communicate in a time division duplex (TDD) mode. First, both parties send training signals to each other in a time slot. Based on the received signals, they will estimate the channel and measure the channel parameter. As the attacker may launch an attack at any time, the signal transmitted by the channel may be non-legitimate. The detection step is introduced to identify the legitimacy of the received signals. If the signal is non-legitimate, the two sides continue to transmit training signals to extract CSI.

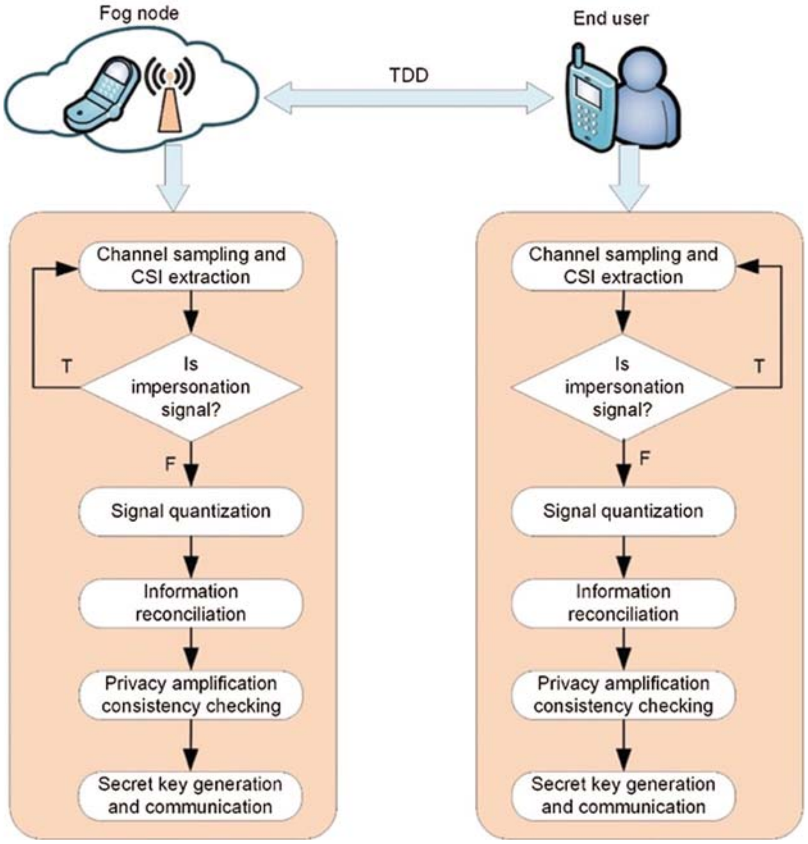

Fog node
End user
TDD
Channel sampling and CSI extraction
Is impersonation signal?
T
F
Signal quantization
Information reconciliation
Privacy amplification consistency checking
Secret key generation and communication
Channel sampling and CSI extraction
Is impersonation signal?
T
F
Signal quantization
Information reconciliation
Privacy amplification consistency checking
Secret key generation and communication

Figure 2: Security model based on key generation

Alternatively, the quantization step is performed to get the initial key bit sequence. The information reconciliation is then employed to correct mismatch bits and privacy amplification consistency checking to ensure the agreed key sequence security. After these steps, both parties' common and secure secret key sequence is generated for secure communications. In this way, the communication between the fog node and the end-user node is coarse-grained detected, which reduces the communication and calculation overhead.

4 Proposed Method

According to the models presented in the previous section, we establish the hypothesis test at the receiver. We then exploit the zero-sum game and the Double Sarsa algorithm [14] to present the proposed method.

4.1 Hypothesis Test

For the hypothesis test, the fog node and the end-user send training signals to each other in the channel sampling stage. The receiving end estimates its related CSI and extracts the signal's channel vector after receiving each signal. The channel vector of the signal accepted by the receiving end is called the channel record. The channel vector and the channel record are used to determine the sender's legitimacy. The hypothesis test is built at the receiving end to detect the attack. The channel gain of each received training signal is sampled and estimated at the receiving end. Let $V_{\alpha_i}^{\beta_i}$ denote the channel vector that the β_i signal transmitted by the α_i sending end. The received channel vector is called the channel record and is represented by $R_{\alpha_i}^{\beta_i}$. The channel gain of the channel vector is denoted by $G(V_{\alpha_i}^{\beta_i})$. Suppose null hypothesis H_0 indicates that the legitimate node sends the training signal. Alternative hypothesis H_1 indicates that the training signal is sent by an illegitimate node, given by the following equations.

$$H_0 : G(V_{\alpha_i}^{\beta_i}) \geq \Upsilon_{M,N} \tag{1}$$

$$H_1 : G(V_{\alpha_i}^{\beta_i}) \neq \Upsilon_{M,N} \tag{2}$$

Where

- M, N represent the fog node and end-user, respectively;
- $\Upsilon_{M,N}$ is the estimated channel gain of the legitimate end user.

Due to wireless channels' spatial decorrelation, the CSIs of each legitimate node, attacker, and malicious nodes are different [25] but can be determined uniquely. As a result, if $V_{\alpha_i}^{\beta_i} = R_{\alpha_i}^{\beta_i}$, the training signal is sent by legitimate sender; otherwise, the training signal is non-legitimate. To judge whether the signal's sender is legal, we set the test statistics of the hypothesis test.

$$\mathrm{L}\left(V_{\alpha_i}^{\beta_i}, R_{\alpha_i}^{\beta_i}\right) = \frac{\left\| V_{\alpha_i}^{\beta_i} - R_{\alpha_i}^{\beta_i} \right\|^2}{\left\| R_{\alpha_i}^{\beta_i} \right\|^2} \tag{3}$$

The symbol $\|\cdot\|$ denotes the Euclidean norm, and $\mathrm{L}\left(V_{\alpha_i}^{\beta_i}, R_{\alpha_i}^{\beta_i}\right)$ is the normalized Euclidean distance between $V_{\alpha_i}^{\beta_i}$ and $R_{\alpha_i}^{\beta_i}$. We compare $\mathrm{L}\left(V_{\alpha_i}^{\beta_i}, R_{\alpha_i}^{\beta_i}\right)$ with the test threshold λ. If it is less than λ, we consider it is sent by a legitimate user and the receiver accepts H_0. Otherwise, if it is greater than λ, we consider it is sent by an attacker and the receiver accepts H_1. The hypothesis test is given by

$$\mathrm{L}\left(V_{\alpha_i}^{\beta_i}, R_{\alpha_i}^{\beta_i}\right) < \lambda \Longrightarrow H_0 \tag{4}$$

$$\mathrm{L}\left(V_{\alpha_i}^{\beta_i}, R_{\alpha_i}^{\beta_i}\right) > \lambda \Longrightarrow H_1 \tag{5}$$

The value of the test threshold has a certain relationship with the detection accuracy. The false alarm rate (FAR) is high for a small threshold value, while a large threshold value results in a high miss detection rate (MDR). The FAR represents the probability that a legal signal sent by a legitimate node is detected as an illegitimate signal. The MDR represents the probability that an illegitimate signal is detected as a legal signal. Based on the hypothesis test, the FAR and MDR are calculated as below [23].

$$P_1(\lambda) = 1 - F_{X_{2M}^2}\left(\frac{2\lambda\omega}{2\omega^2 + n\omega\xi^2}\right) \tag{6}$$

$$P_2(\lambda) = F_{X^2_{2M}}\left(\frac{2\lambda\omega}{2\omega^2 + (1+m)\omega\xi^2}\right) \tag{7}$$

The AER is the probability of error in detecting impersonation attacks, which is given by

$$P_A(\lambda) = 1 - F_{X^2_{2M}}\left(\frac{2\lambda\omega}{2\omega^2 + n\omega\xi^2}\right) + F_{X^2_{2M}}\left(\frac{2\lambda\omega}{2\omega^2 + (1+m)\omega\xi^2}\right) \tag{8}$$

Where

- ω is the signal to interference plus noise ratio of the legal training signals (SINR);
- n is the relative change rate of channel gain;
- ξ^2 is the average power gain of the signal received by the receiver from the transmitter;
- m is the ratio of the channel gain of the impersonator to that of the transmitter, and the symbol $F_{X^2_{2M}}$ stands for cumulative distribution function with a degree of freedom 2M.

4.2 Methodology of Impersonation Detection

Based on the impersonation attack model presented in Section 3.1, the interactions between an illegitimate node and a receiver can be considered a zero-sum game [26]. The probability that an illegitimate node sends an illegal signal is p_k. The set of illegitimate signals sent by illegitimate nodes is $\Phi = [p_k]$. It is assumed that only one illegitimate node performs an impersonation attack in a time slot, so the receiver's probability of illegitimate signal is $\sum_{k=1}^{Y} p_k$. We also considered the Bayes risk of impersonation detection is given as [27].

$$B(\lambda,\Phi) = (l_1(1 - P_1(\lambda)) - r_2 P_1(\lambda))\left(1 - \sum_{k=1}^{Y} p_k\right) + (l_2(1 - P_2(\lambda)) - r_1 P_2(\lambda))\left(\sum_{k=1}^{Y} p_k\right) \tag{9}$$

Where

- l_1 is the gain of receiving legal signals, l_2 is the gain of refusing illegal signals;
- r_1 is the cost of receiving illegal signals, r_2 is the cost of refusing legal signals;
- $P_1(\lambda)$ represents FAR, and $P_2(\lambda)$ represents MDR.

Therefore, according to the zero-sum game, the utility at the receiving end is as follows.

$$\begin{aligned}\Psi_Z(\lambda,\Phi) &= -\Psi_Y(\lambda,\Phi) = B(\lambda,\Phi)\\ &= (l_2 - l_1)\sum_{k=1}^{Y} p_k - (l_2 + r_1) P_2(\lambda)\sum_{k=1}^{Y} p_k - (l_1 + r_2) P_1(\lambda)\left(1 - \sum_{k=1}^{Y} p_k\right) + l_1 \end{aligned} \tag{10}$$

In (10), the illegitimate nodes and the receivers play games in the static environment. However, the practical fog computing environment is dynamic. Specifically, the mobility of fog nodes and terminals can cause changes in the wireless channel, and the change rate has a particular impact on the key generation rate (KGR). In the dynamic scenario, it is difficult for the receiver to recognize the training signal's legitimacy. Double Sarsa algorithm in reinforcement learning can be used to find out the optimal strategy with insufficient information in a dynamic environment [14,28].

In reinforcement learning, there are several important constants. First, the discount rate is used to control the proportion of current and future rewards. For the large value, the agent pays much attention to future rewards. However, for a small value, the agent focuses on the current rewards. Next is learning efficiency. For high efficiency, a large proportion of results are obtained with new attempts and vice versa. In Double Sarsa algorithm [14], two Q-tables are used to compensate for each other's estimation errors. A small change in the Q-value can determine a steady policy and increase the reward. The receiver evaluate the T training signals sent by transmitter in each time slot via the hypothesis test. Meanwhile, the receiver employs the ε-greedy strategy to choose the test threshold until it reaches the optimum, which is determined whether the sender of the signal is a legitimate node or illegitimate node. In each state, the receiving end randomly chooses the λ with a probability ε and chooses the largest λ value in Q-table with a probability $1-\varepsilon$, that is, the λ satisfying $\Pi(S_t)$. According to (10), we can use it as the reward function in the Double Sarsa algorithm, so the current utility of the receiver in each state is

$$\Omega_t = \sum_{t'=(t-1)T+1}^{\mathrm{t}T} \Psi_Z(\lambda, \Phi) \tag{11}$$

In the Double Sarsa algorithm, aiming at reducing the impact of estimation errors and converging to the optimal Q value, the update rule of two Q-tables is given by

$$Q^A(\mathrm{S}_t, \lambda) = (1-\mu)\, Q^A(\mathrm{S}_t, \lambda) + \mu \left(\Omega_t + \gamma Q^B\left(\mathrm{S}_{t+1}, \lambda'_A\right)\right) \tag{12}$$

$$Q^B(\mathrm{S}_t, \lambda) = (1-\mu)\, Q^B(\mathrm{S}_t, \lambda) + \mu \left(\Omega_t + \gamma Q^A\left(\mathrm{S}_{t+1}, \lambda'_B\right)\right) \tag{13}$$

Where

- μ is the learning efficiency, $\mu \in [0, 1]$;
- γ is the discount rate, $\gamma \in [0, 1]$;
- S_t is the state of the receivers at time t, i.e., the FAR and MDR value at time $\mathrm{t}-1$;
- λ'_A and λ'_B are the test thresholds for maximizing Q value of Q^A and Q^B in state S_{t+1}, respectively. Their calculation formulas are as following.

$$\lambda'_A = \arg max\, \mathrm{Q}^A\left(\mathrm{S}_{t+1}, \lambda'\right) \tag{14}$$

$$\lambda'_B = \arg max\, \mathrm{Q}^B\left(\mathrm{S}_{t+1}, \lambda'\right) \tag{15}$$

$$\Pi(S_t) = \max\left[\frac{Q^A(\mathrm{S}_t, \lambda) + Q^B(\mathrm{S}_t, \lambda)}{2}\right] \tag{16}$$

$\Pi(S_t)$ denotes the maximum mean value of $Q^A + Q^B$ in current state. The optimal test threshold λ^* is given by

$$\lambda^* = \arg \Pi(S_t) \tag{17}$$

Aiming to get an optimal action and maximize the utility, the proposed method to resist impersonation attack is described in Algorithm 1. We initialize some parameters, then judge the legitimacy of training signals according to the test threshold. Finally, we find the optimum test threshold value based on the Double Sarsa algorithm to detect impersonation attacks.

Algorithm 1: Detection method based on Double Sarsa algorithm

```
Step 1: Initialization
$\mu$, $\gamma$, $\varepsilon$, $Q^A(S_t,\lambda) = Q^B(S_t,\lambda) = 0$, $\Pi(S_t) = 0$;
Step 2: Processing:
For $t = 1,2,3,\ldots,N$ do
    Select $\lambda$ in current state $S_t$ according to $\varepsilon$-greedy strategy;
    for $T = 1,2,\ldots,20$ do
      Extract CSI of the training signals,$\Upsilon_{M,N}$, $\Upsilon_{M|N,I}$, $V_{\alpha_i}^{\beta_i}$ and $R_{\alpha_i}^{\beta_i}$;
      Calculate $L\left(V_{\alpha_i}^{\beta_i}, R_{\alpha_i}^{\beta_i}\right)$ via (3);
      If $L\left(V_{\alpha_i}^{\beta_i}, R_{\alpha_i}^{\beta_i}\right) < \lambda$, accept the training signals;
      Else
      Reject the training signals;
    end for
Step 3: Updating:
Enter next state $S_{t+1}$;
Update $Q^A(S_t,\lambda)$ via (12) and (14) with probability 0.5;
Otherwise, Update $Q^B(S_t,\lambda)$ via (13) and (15);
Update $\Pi(S_t)$ via (16);
End For
```

5 Performance Evaluation

In this section, to assess the proposed method's performance, we set four indicators, such as the utility at the receiver, FAR, MDR, and AER. And three different methods are compared for preventing impersonation attacks in 500 experiment trails.

5.1 *Performance Metrics and Initial Parameters*

In this study, we set the experimental environment as a space of 50×50 m^2, in which several fog nodes, attackers, and legitimate end-users are scattered. All channel gains follow the normal distribution (0, 1), as used in [23]. We use a computer with MATLAB software for simulation under the Windows operating system. Assuming the receiver receives $T = 20$ training signals in the same time slot, the center frequency is 2.4 GHz. The names, symbols, and initial values of the correlative parameters are shown in Tab. 1.

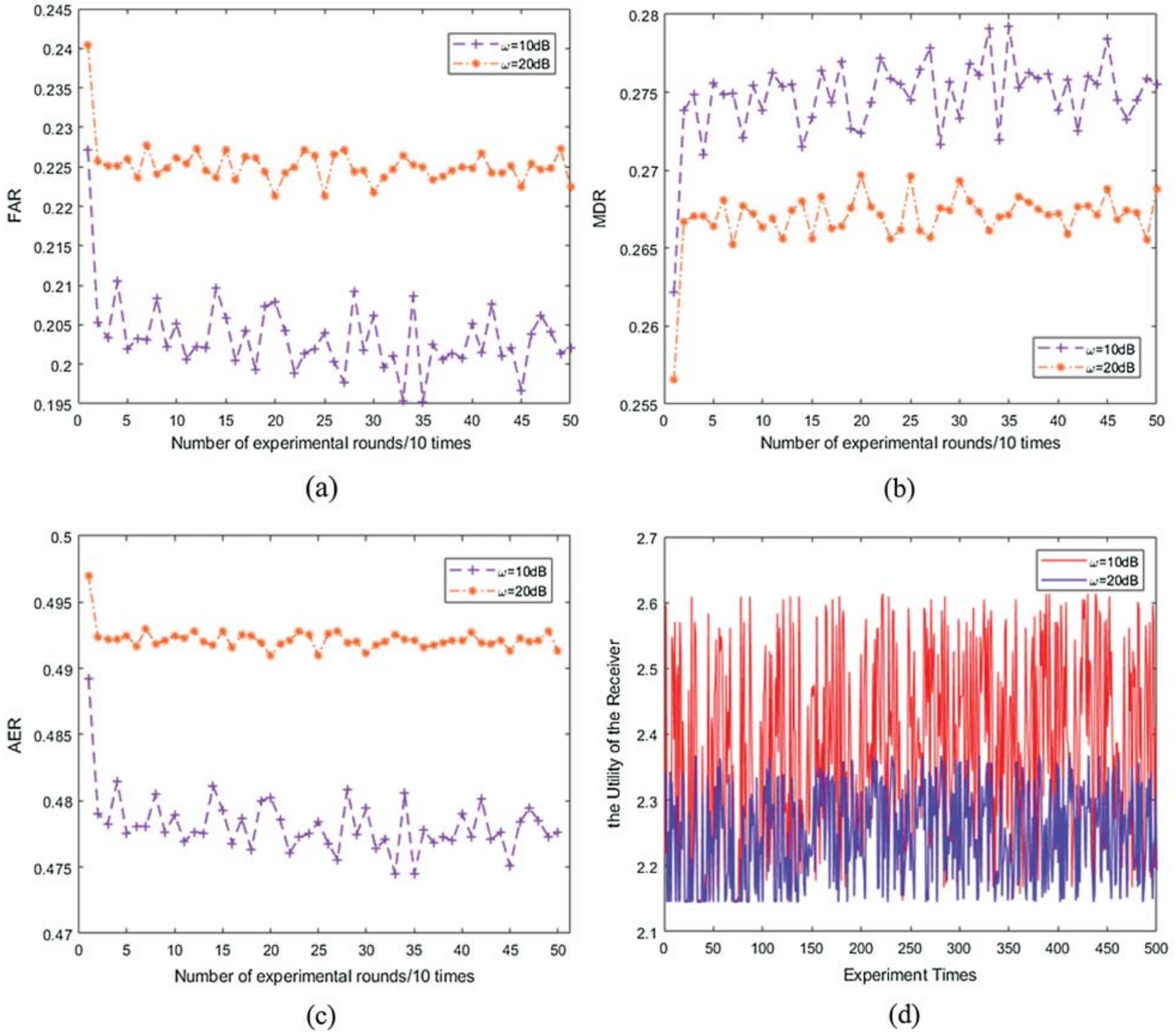
Table 1: List of symbols and names for parameters

Name	Symbol	Value
The cost of receiving illegal signals	r_1	4
The cost of refusing legal signals	r_2	2
The gain of receiving legal signals	l_1	6
The gain of refusing illegal signals	l_2	9
The learning efficiency	μ	0.9
The discount rate	γ	0.6
The strategy selection rate	ε	0.5
The relative change rate of channel gain	n	3 dB
The average power gain of the signal received by the receiver from the transmitter	ξ^2	5
The ratio of the channel gain of the impersonator to that of the legitimate transmitter	m	0.2
The signal to interference plus noise ratio of the legal training signals (SINR)	ω	10/20 dB
The center frequency	f_0	2.4 GHz

5.2 *Simulation Results*

To improve the detection accuracy at the receiving end, we analyze the performance of the proposed method under the four indicators by changing the value of ω. Figs. 3a–3c show that the change of detection error rate with the number of experimental rounds under different ω, and the Q table is updated 10 times in each round. As shown in Fig. 3a, under the condition that the ratio of the channel gain of the impersonator to that of the legitimate transmitter m is constant, the FAR decreases with ω. This is because SINR has a direct relationship with the channel estimation error. The low SINR indicates that the channel estimation error is low. Furthermore, the channel estimation error has a particular influence on the degree of training signal quantization, affecting FAR and MDR. Furthermore, for different ω, the FAR decreases first, and then stabilize a certain value. This is because Double Sarsa algorithm optimizes the threshold in the process of searching for the optimal threshold. The threshold changes greatly at the beginning, the FAR changes rapidly accordingly. As the threshold reaches the optimum, the FAR changes smaller. If $\omega = 10$, it fluctuates within a small range of 0.20. After 50 rounds of experiments, the false alarm rate is 0.2021, which is 9.21% lower than $\omega = 20$. Similarly, in Fig. 3b, the MDR changes quickly at the start, and then keeps stable in a range. Its value is 0.2755 with $\omega = 10$, which is 2.49% higher than $\omega = 20$. Fig. 3c shows that the AER decreases with ω. For example, after 50 rounds of trails, if $n = 3$dB, the average error rate for $\omega = 10$dB decreases to 0.4776 from 0.4914 for $\omega = 20$dB.

In addition, as the detection error rate decreases correspondingly, the average utility of the receiver decreases with ω, as shown in Fig. 3d. For example, after 500 experiment times, the average utility of the receiver is 2.4112 with $\omega = 10$, which is 9.9% higher than $\omega = 20$. FAR, MDR, AER and the average utility of the receiver with different ω is shown in Tab. 2. Therefore, the performance is better with $\omega = 10$, based on the above four indicators. To further illustrate the performance gain of the proposed method, we compare the proposed method with Sarsa algorithm and Q-learning algorithm [29].



(a)

(b)

(c)

(d)

Figure 3: Average impersonation detection performance in the experiments. (a) False alarm rate (FAR). (b) Miss detection rate (MDR). (c) Average error rate (AER). (d) Utility of the receiver

Table 2: FAR, MDR, AER and the average utility with ω

ω	FAR	MDR	AER	The average utility
10	0.2021	0.2755	0.4776	2.4112
20	0.2226	0.2688	0.4914	2.1930

Fig. 4 depicts the comparison of FAR of three algorithms for 500 iterations. It can be seen that the FAR of these three algorithms reduces at the start. The proposed method gradually stabilizes after 100 iterations. As the iterations continue, the three methods are roughly kept within a constant range. The proposed method is stable at 20.0%–20.2%. Because the receiving end knows nothing about the channel and gradually gets understanding as the experiment goes on. The optimal λ will be chosen according to previous experience when detecting attacks. The FAR does not change when reaching a certain number of experiments. But because of the wireless channel's fading characteristics, it can cause a slight change in the value and eventually steady. In most experiments, FAR is lower and stable with the proposed method than Sarsa-based and Q-learning-based. This is because it uses two Q tables to update each other, which helps slow down the threshold's change rate and converge faster. It is easier to get the optimal threshold for detecting impersonation attacks.

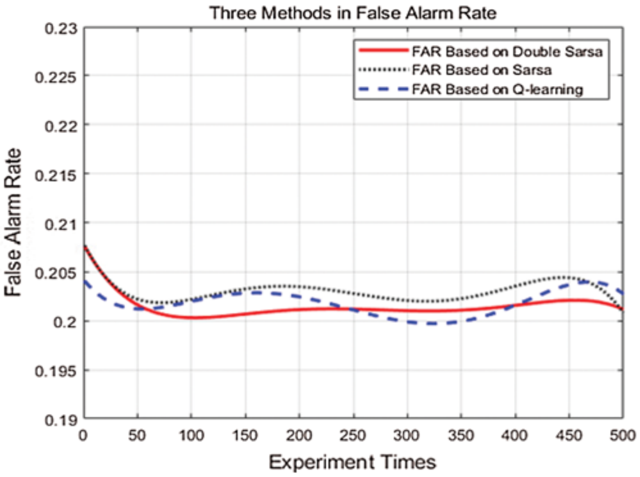


Figure 4: Comparison of double Sarsa, Sarsa, Q-learning for false alarm rate (FAR)

Fig. 5 describes the MDR of the three algorithms. Because the receiving end has no idea of the system details. The initial Q-table value is all zero, and the threshold at the beginning of the experiment is small. During continuous training, the threshold gradually increases. According to Eq. (7), the MDR value of three methods increases first. As the threshold reaches the optimum gradually, the change range of MDR value is getting smaller. Besides, the channel fading also affects the value, so it fluctuates slightly. The MDR of the proposed method is steady at about 27.6%, with the experiment is iterated 500 times. Compared with the Sarsa-based method, the MDR of the Double Sarsa-based method is decreased by 5.5% and by an average of 5.8% compared with the Q-learning-based method. Furthermore, Double Sarsa uses two Q-value tables to compensate for errors each other and prevent large changes in the threshold value, so it is more stable than the other two methods. The proposed method increases the probability of illegitimate signals being ignored so that the communication quality between fog nodes and end-users is improved.

As shown in Fig. 6, we compare the average error rate of detection attacks under three algorithms. All three methods have a downward trend in the early period of the experiment and then stabilizes. This is because the receiver has a better understanding of the channel after receiving a large number of training signals and can use accumulated experience to choose the optimal threshold for detection. At the same time, the AER value eventually stabilizes at a constant within range for the channel's characteristics. AER based on the Double Sarsa algorithm is lower than the other two algorithms. When the experiment is carried out 500 times, the AER of the proposed method is stable at about 47.72%, which is less than the Q-learning-based method by 3.0% and by 3.1% Sarsa-based method. In addition, our proposed method converges more slowly because two Q tables are used to update alternately. Therefore, considering the overall performance, the proposed method has lower FAR, MDR and AER, and it can detect attack behavior more accurately in balanced exploration and utilization, which can improve the security of FC.

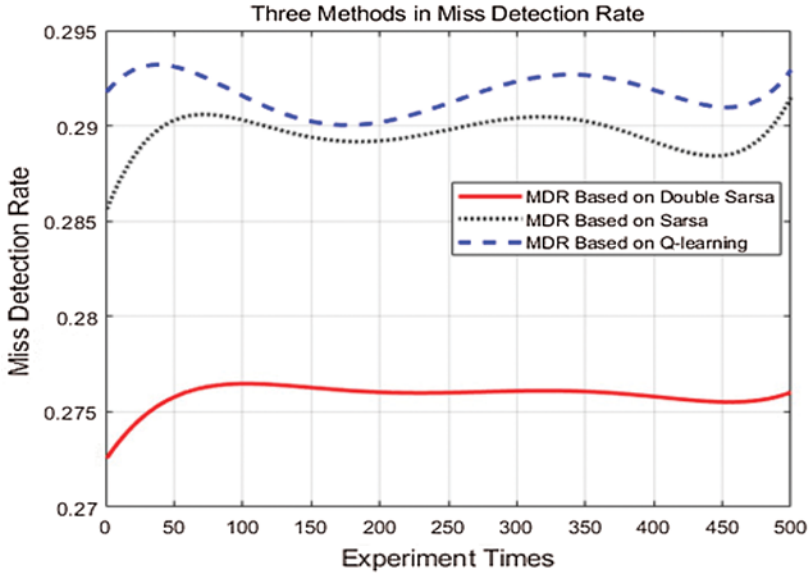


Figure 5: Comparison of double Sarsa, Sarsa, Q-learning for miss detection rate (MDR)

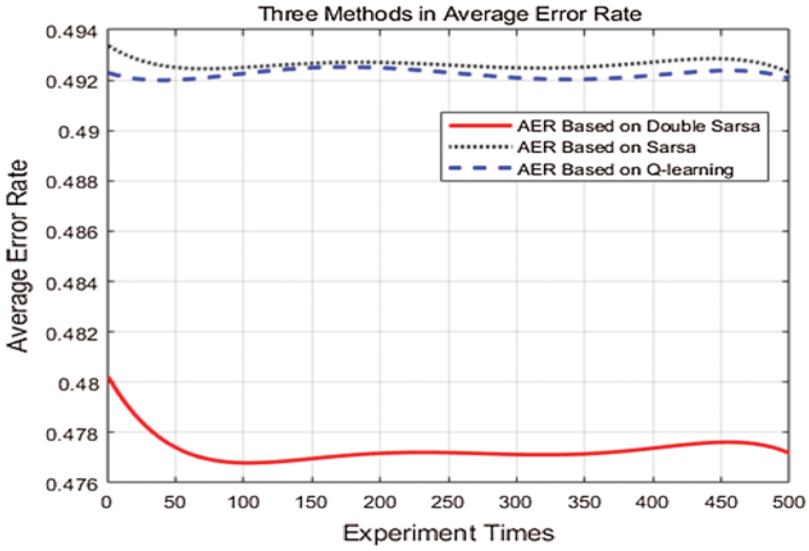


Figure 6: Comparison of double Sarsa, Sarsa, Q-learning for average error rate (AER)

6 Conclusion

In this paper, we proposed an impersonation attack detection method based on the Double Sarsa algorithm in FC. First, the impersonation attack model and the key generation model based on PLS are built. Then, the hypothesis test is established at the receiver. Next, the zero-sum game between illegitimate nodes and receivers is designed to calculate utility. Double Sarsa algorithm with reinforcement learning is used to choose the optimal test threshold to detect malicious users' impersonation attacks. The experimental results illustrate that the proposed method can reduce the FAR, MDR, and AER. Comparing with the Sarsa algorithm and Q-learning algorithm, the performance gain of the proposed method is proved. The proposed method in this study can protect the FC environment more effectively. In the future, reinforcement learning and deep learning will be combined to study and solve the fog computing environment's security problem.

Funding Statement: This work was supported by Natural Science Foundation of China (61801008), The China National Key R&D Program (No. 2018YFB0803600), Scientific Research Common Program of Beijing Municipal Commission of Education (No. KM201910005025) and Chinese Postdoctoral Science Foundation (No. 2020M670074).

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

[1] K. H. Abdulkareem, M. A. Mohammed, S. S. Gunasekaran, M. N. Al-Mhiqani, A. A. Mutlag *et al.,* "A review of fog computing and machine learning: Concepts, applications, challenges, and open issues," *IEEE Access*, vol. 7, pp. 153123–153140, 2019.

[2] L. Xu, C. Xu, Z. Liu, Y. Wang and J. Wang, "Enabling comparable search over encrypted data for IoT with privacy-preserving," *Computers, Materials & Continua*, vol. 60, no. 2, pp. 675–690, 2019.

[3] P. Bellavista, J. Berrocal, A. Corradi, S. K. Das, L. Foschini *et al.,* "A survey on fog computing for the internet of things," *Pervasive and Mobile Computing*, vol. 52, pp. 71–99, 2019.

[4] J. Ni, K. Zhang, X. Lin and X. S. Shen, "Securing fog computing for internet of things applications: Challenges and solutions," *IEEE Communications Surveys & Tutorials*, vol. 20, no. 1, pp. 601–628, 2017.

[5] M. De Donno, K. Tange and N. Dragoni, "Foundations and evolution of modern computing paradigms: Cloud, IoT, edge, and fog," *IEEE Access*, vol. 7, pp. 150936–150948, 2019.

[6] M. Waqas, Y. Niu, Y. Li, M. Ahmed, D. Jin *et al.,* "A comprehensive survey on mobility-aware D2D communications: Principles, practice and challenges," *IEEE Communications Surveys & Tutorials*, vol. 22, no. 3, pp. 1863–1886, 2020.

[7] N. Hassan, K. L. A. Yau and C. Wu, "Edge computing in 5G: A review," *IEEE Access*, vol. 7, pp. 127276–127289, 2019.

[8] D. Kim and S. Kim, "Network-aided intelligent traffic steering in 5G mobile networks," *Computers, Materials & Continua*, vol. 65, no. 1, pp. 243–261, 2020.

[9] K. Shaque, B. A. Khawaja, F. Sabir, S. Qazi and M. Mustaqim, "Internet of things (IoT) for next-generation smart systems: A review of current challenges, future trends and prospects for emerging 5G-IoT scenarios," *IEEE Access*, vol. 8, pp. 23022–23040, 2020.

[10] R. K. Naha, S. Garg, D. Georgakopoulos, P. P. Jayaraman, L. Gao *et al.,* "Fog computing: Survey of trends, architectures, requirements, and research directions," *IEEE Access*, vol. 6, pp. 47980–48009, 2018.

[11] Y. Wu, A. Khisti, C. Xiao, G. Caire, K. K. Wong *et al.,* "A survey of physical layer security techniques for 5G wireless networks and challenges ahead," *IEEE Journal on Selected Areas in Communications*, vol. 36, no. 4, pp. 679–695, 2018.

[12] N. Wang, P. Wang, A. Alipour-Fanid, L. Jiao and K. Zeng, "Physical layer security of 5G wireless networks for IoT: Challenges and opportunities," *IEEE Internet of Things Journal*, vol. PP, no. 99, pp. 1, 2019.

[13] H. Zhang, K. Zheng, X. Wang, S. Luo and B. Wu, "Strategy selection for moving target defense in incomplete information game," *Computers, Materials & Continua*, vol. 62, no. 2, pp. 763–786, 2020.

[14] M. Ganger, E. Duryea and W. Hu, "Double Sarsa and double expected Sarsa with shallow and deep learning," *Journal of Data Analysis and Information Processing*, vol. 4, no. 4, pp. 159–176, 2019.

[15] J. Zhang, T. Q. Duong, A. Marshall and R. Woods, "Key generation from wireless channels: A review," *IEEE Access*, vol. 4, pp. 614–626, 2016.

[16] Y. Peng, P. Wang, W. Xiang and Y. Li, "Secret key generation based on estimated channel state information for TDD-OFDM systems over fading channels," *IEEE Transactions on Wireless Communications*, vol. 16, no. 8, pp. 5176–5186, 2017.

[17] K. Moara-Nkwe, Q. Shi, G. M. Lee and M. H. Eiza, "A novel physical layer secure key generation and refreshment scheme for wireless sensor networks," *IEEE Access*, vol. 6, pp. 11374–11387, 2018.

[18] M. Waqas, M. Ahmed, Y. Li, D. Jin and S. Chen, "Social-aware secret key generation for secure device-to-device communication via trusted and non-trusted relays," *IEEE Transactions on Wireless Communications*, vol. 17, no. 6, pp. 3918–3930, 2018.

[19] Q. Huang, Y. Yang and L. Wang, "Secure data access control with ciphertext update and computation outsourcing in fog computing for internet of things," *IEEE Access*, vol. 5, pp. 12941–12950, 2017.

[20] N. Abbas, M. Asim, N. Tariq, T. Baker and S. Abbas, "A mechanism for securing IoT-enabled applications at the fog layer," *Journal of Sensor and Actuator Networks*, vol. 8, no. 1, pp. 16, 2019.

[21] T. Wang, J. Zhou, X. Chen, G. Wang, A. Liu *et al.,* "A three-layer privacy preserving cloud storage scheme based on computational intelligence in fog computing," *IEEE Transactions on Emerging Topics in Computational Intelligence*, vol. 2, no. 1, pp. 3–12, 2018.

[22] L. Xiao, Y. Li, G. Han, G. Liu and W. Zhuang, "PHY-layer spoofing detection with reinforcement learning in wireless networks," *IEEE Transactions on Vehicular Technology*, vol. 65, no. 12, pp. 10037–10047, 2016.

[23] S. Tu, M. Waqas, S. Rehman, M. Aamir, O. Rehman *et al.,* "Security in fog computing: A novel technique to tackle an impersonation attack," *IEEE Access*, vol. 6, pp. 74993–75001, 2018.

[24] J. Zhang, S. Rajendran, Z. Sun, R. Woods and L. Hanzo, "Physical layer security for the internet of things: Authentication and key generation," *IEEE Wireless Communications*, vol. 26, no. 5, pp. 92–98, 2020.

[25] N. Aldaghri and H. Mahdavifar, "Physical layer secret key generation in static environments," *IEEE Transactions on Information Forensics and Security*, vol. 15, pp. 2692–2705, 2020.

[26] J. R. Riehl and M. Cao, "A centrality-based security game for multihop networks," *IEEE Transactions on Control of Network Systems*, vol. 5, no. 4, pp. 1507–1516, 2018.

[27] Z. Ma and A. Leijon, "Bayesian estimation of beta mixture models with variational inference," *IEEE Transactions on Pattern Analysis & Machine Intelligence*, vol. 33, no. 11, pp. 2160–2173, 2011.

[28] H. Zhang, K. Zheng, X. Wang, S. Luo and B. Wu, "Strategy selection for moving target defense in incomplete information game," *Computers, Materials & Continua*, vol. 62, no. 2, pp. 763–786, 2020.

[29] M. Waqas, S. Tu, S. Rehman, Z. Halim, S. Anwar *et al.,* "Authentication of vehicles and road side units in intelligent transportation system," *Computers, Materials & Continua*, vol. 64, no. 1, pp. 359–371, 2020.