Defeating Eavesdropping Attacks with Inter-Cell Interference and Deep Reinforcement Learning

Nguyen Van Huynh[1], Diep N. Nguyen[2], Lorenzo Mucchi[3], Stefano Caputo[3], Massimo Piccardi[2], Dinh Thai Hoang[2], and Eryk Dutkiewicz[2]
[1] Department of Electrical Engineering and Electronics, University of Liverpool, UK
[2] School of Electrical and Data Engineering, University of Technology Sydney, Australia
[3] Department of Information Engineering, University of Florence, Italy
Emails: huynh.nguyen@liverpool.ac.uk, diep.nguye@uts.edu.au, {lorenzo.mucchi, stefano.caputo}@unifi.it, {massimo.piccardi, hoang.dinh, eryk.dutkiewicz}@uts.edu.au

***Abstract*—This paper introduces a novel joint user association and resource allocation framework to efficiently deal with eavesdropping attacks without requiring prior information about eavesdroppers. Specifically, the co-channel interference when reusing resource blocks is leveraged to disrupt the signal reception at eavesdroppers. To maximize the secure area, defined as the area where eavesdroppers cannot wiretap the channel due to co-channel interference, we first formulate the system by using the Markov decision process to capture the dynamics and uncertainty of mobile users and wireless communications. Then, a deep reinforcement learning (DRL)-based approach is proposed to obtain the joint optimal user association and resource allocation policy to utilize the co-channel interference and maximize the secure area. Extensive simulation results demonstrate that by intelligently associating users and allocating resource blocks to them, our proposed solution can help to effectively defeat eavesdropping attacks without requiring prior information of eavesdroppers which may not be readily available in practice. In addition, the proposed DRL-based algorithm can converge to the optimal policy quickly and achieve better performance compared to existing solutions.**

***Index Terms*—User association, resource block allocation, physical layer security, eavesdroppers, inter-cell interference, and deep reinforcement learning.**

I. Introduction

Ensuring privacy and security is a critical task in every wireless system. However, its broadcast nature makes wireless communication extremely vulnerable to eavesdropping attacks. With off-the-shelf circuits, an eavesdropper can easily "wiretap" the target wireless channel to acquire information transmitted between legitimate devices. As eavesdroppers operate passively and do not generate any active signals, accurately detecting and preventing eavesdropping attacks is very challenging. The most common countermeasure against eavesdropping is using cryptographic techniques [1], [2]. Although these techniques can mitigate the effects of eavesdroppers, they face various practical challenges. Specifically, encrypting and decrypting data introduce computing and resource burdens to wireless devices [3]. In addition, the diversity and mobility of wireless devices can pose challenges to the distribution and management of cryptographic keys [4]. Finally, with quantum computing, encryption techniques will soon become defenseless while post-quantum cryptography is still under its early development stage [5].

To overcome these issues, physical-layer security (PLS) has emerged as a prominent solution [7]–[9]. The fundamental principle of PLS is to obtain a positive secrecy capacity, which is determined by the difference between the channel capacity of the legitimate communication link and that of the eavesdropper's link [6]. With a positive secrecy capacity, the legitimate transmitter can adjust its transmission rate to allow the legitimate receiver to successfully decode the transmitted information while the intercepted data at the eavesdropper is limited and noisy [6]. For example, the authors in [8] develop a DRL-based friendly jamming approach that allows friendly jammers to optimize their jamming frequency, jamming power, and jamming duration to effectively deal with active eavesdroppers. Although achieving promising performance, existing approaches usually require prior eavesdropping channel state information. Unfortunately, it is challenging to obtain such information due to the passive nature of eavesdroppers. Moreover, approaches like using friendly jammers and performing beamforming can increase the system's complexity and deployment costs. Finally, existing solutions usually assume the availability of eavesdroppers' locations, which may not be the case in practice as eavesdroppers only passively listen to the channel.

Given the above, this paper proposes a novel joint user association and resource allocation framework to defeat eavesdropping attacks without requiring prior knowledge of eavesdroppers. To do that, we leverage the co-channel interference caused by resource block reuses to prevent potential eavesdroppers from "wiretapping" the legitimate channels. Instead of assuming the availability of eavesdroppers' locations, we investigate the case when eavesdroppers can be placed anywhere in the considered area. We then define a secure area that covers any locations where the secrecy capacity is positive and higher than a predefined threshold. Our aim is to maximize the secure area by optimally associating users and allocating resource blocks to them to efficiently exploit the inter-cell interference. However, it is very challenging for traditional optimization methods to obtain the optimal policy given the

dynamics and uncertainty of wireless communications as well as users' behaviors. Thus, we develop a novel DRL-based framework to obtain the optimal user association and resource allocation policy without requiring the system's parameters in advance. Simulations show that by leveraging the co-channel interference, our solution can efficiently deal with eavesdropping attacks without prior knowledge of eavesdroppers.

II. System Model

We consider a wireless system with a large number of cells/base stations densely deployed in area A, with the presence of an eavesdropper E whose position is unknown. The eavesdropper aims to recover messages transmitted from base stations (BSs) to legitimate user equipment (UEs). In this paper, we assume that the universal frequency reuse scheme is used, and thus all BSs share the same frequency band [10]. As a result, UEs and the eavesdropper experience inter-cell interference, which will be exploited in this paper to prevent the eavesdropper from accessing legitimate information from UEs and BSs. In our system, C small cells are distributed randomly in the considered area A based on the uniform distribution. We denote $\boldsymbol{C}$ as the set of BSs in the system. We assume that at a given point in time, a number of UEs can access the network. Without loss of generality, UEs arriving at the system are indexed according to their order of arrival. At each time slot, a new UE arrives at the system with probability μ. The average resource occupation time of user u is λ time slots. When a UE leaves the system, its allocated resources will be released and the macro BS will update the status of the whole system. The positions of UEs are randomly generated and distributed based on the uniform distribution in the considered area A.

We denote $\boldsymbol{K} = \{1, \ldots, k, \ldots, K\}$ as the set of time-frequency resource blocks to serve UEs at each BS. Here, we assume that these K resource blocks are the same at all BSs [10]. UE u requests r_u number of resource blocks. If BS c serves UE u, it will allocate $r_u \leq K$ resource blocks to UE u, denoted by $\boldsymbol{R}_u \overset{\text{def}}{=} \{k_i : \forall k \in \boldsymbol{K}\}$ with $|\boldsymbol{R_u}| = r_u$. We denote $\boldsymbol{U}_c^t$ as the set of UEs currently served by BS c at time t. To guarantee that the total resource blocks allocated to UEs at any time do not exceed the total resource blocks available at BS c, we define the following constraint

$$\sum_{u \in \boldsymbol{U}_c^t} r_u \leq K, \forall u \in \boldsymbol{U}_c^t, \forall c \in \boldsymbol{C}. \tag{1}$$

We then define $\zeta_{u,c,k} = 1, \forall k \in \boldsymbol{R}_u$ to indicate that BS c serves UE u using resource block set $\boldsymbol{R}_u$. In this paper, we assume that each UE is served by only one BS, and the allocated resource blocks for each UE are fixed until it leaves the system. As such, we have the following constraint

$$\sum_{c \in \boldsymbol{C}} \sum_{k \in \boldsymbol{R}_u} \zeta_{u,c,k} = r_u, \forall u \in \boldsymbol{U}_c^t. \tag{2}$$

In addition, each resource block of a BS is used to serve only one UE belonging to this BS, expressed as $\boldsymbol{R}_u \cap \boldsymbol{R}_{u'} = \emptyset, \forall u, u' \in \boldsymbol{U}_c^t, \forall c \in \boldsymbol{C}$.

A. Channel Model

Suppose that UE u at location (x_u, y_u) connects to BS c at location (x_c, y_c), the channel between them can be modeled as $\boldsymbol{H}_{c,u} = h_{c,u}(\tau, \psi) \cdot d_{c,u}^{-b}$ [6], where $d_{c,u}$ is the Euclidian distance between UE u and BS c, and b represents the path loss exponent. $h_{c,u}(\tau, \psi)$ denotes the multipath fading effect considering the angle of dispersion and can be expressed as follows:

$$h_{c,u}(\tau, \psi) = \sum_{l=1}^{L} h_{c,u}^{(l)} \delta(\tau - \tau_l) \delta(\psi - \psi_l), \tag{3}$$

where τ_l and ψ_l are the delay and the angle of the arrival of l-th path, respectively. $h_{c,u}^{(l)}$ is the channel coefficient denoted by $h_{c,u}^{(l)} = a_{c,u}^{(l)} e^{-\beta_{c,u}^{(l)}}$ with $a_{c,u}^{(l)}$ is a stochastic variable follow Rayleigh distribution and $\beta_{c,u}^{(l)}$ is a stochastic random variable with uniform distribution in $(0, 2\pi)$.

B. Received Power

Denote P_c as the transmit power of BS c, the power received by UE u can be calculated as $P_u = P_c |\boldsymbol{H}_{c,u}|^2 G_c(\theta_c, \phi_{c,u}) G_u(\theta_u, \phi_{u,c})$ [6], where $G_c(\theta_c, \phi_{c,u})$ is the antenna pattern gain of the BS, $\phi_{c,u}$ is the angle between the segment connecting BS c and UE u and the x-axis. θ_c represents the angle between the direction of the main lobe of BS c's antenna and the x-axis. Similarly, $G_u(\theta_u, \phi_{u,c})$ denotes the antenna pattern gain of UE u, $\phi_{u,c}$ is the angle between the segment connecting UE u and BS c and the x-axis, and θ_u is the angle between the direction of the main lobe of UE u's antenna and the x-axis. Denote $\bar{P}_{c,u} = P_c G_c(\theta_c, \phi_{c,u}) G_u(\theta_u, \phi_{u,c})$, we have $P_u = \bar{P}_{c,u} |\boldsymbol{H}_{c,u}|^2$.

C. Aggregate Interference

If BS c connects to UE u using resource blocks in $\boldsymbol{R}_u$, denoted by indicator $\zeta_{u,c,k} = 1$, and other BSs also use this resource block to serve their UEs, UE u will experience the inter-cell interference. The sum of the interference power (on resource block k) at UE u can be expressed as in (4), where $P_{c'}$ is the power emitted by BS c', $d_{c',u}$ is the Euclidian distance between BS c' and UE u, and $h_{c',u}$ is the the channel coefficient associated to the link [6]. Notably, the interference of BS c' does not affect UE u if the distance between them is too far as the distance path loss $d_{c',u}^{-2b}$ closes to zero.

D. Defeating Eavesdropper with Inter-cell Interference

In this section, we will present how inter-cell interference can help to prevent eavesdropper E from recovering transmitted messages. First, denoting M as a confidential message transmitted from BS c to UE u, the mutual information transmitted in the link can be calculated as follows [6], [10]:

$$\mathbb{I}_u = \mathbb{I}(M, Z_u) = \mathbb{H}(M) - \mathbb{H}(M|Z_u), \tag{5}$$

where Z_u is the received vector at UE u and $\mathbb{H}(.)$ denotes the entropy. Similarly, eavesdropper E aims to estimated message M based on its received vector Z_E. The information wiretapped by eavesdropper E can be expressed as $\mathbb{I}_E =$

$$\boldsymbol{F}_u^k = \sum_{c' \in \boldsymbol{C}} \sum_{u' \in \boldsymbol{U}_{c'}^t} \zeta_{u',c',k} P_{c'} G_{c'}(\theta_{c'}, \phi_{c',u}) G_u(\theta_u, \phi_{u,c'}) d_{c',u}^{-2b} |h_{c',u}|^2 = \sum_{c' \in \boldsymbol{C}} \sum_{u' \in \boldsymbol{U}_{c'}^t} \zeta_{u',c',k} \bar{P}_{c',u} |\boldsymbol{H}_{c',u}|^2, \tag{4}$$

$\mathbb{I}(M, Z_E) = \mathbb{H}(M) - \mathbb{H}(M|Z_E)$. The secrecy capacity then can be calculated as follows [6], [10]:

$$C_{\text{sec}} = \max_{p_M} \{\mathbb{I}_u - \mathbb{I}_E\} \geq \max_{p_M} \mathbb{I}_u - \max_{p_M} \mathbb{I}_E = C_u - C_E, \tag{6}$$

where C_u denotes the UE u's channel capacity and C_E denotes the eavesdropper E's channel capacity. p_M represents message M's marginal distribution. The capacity C_u of UE u's channel can be calculated as follows:

$$C_u = \frac{1}{2} \log \left(1 + \frac{P_u}{N_0 + \boldsymbol{F}_u} \right), \tag{7}$$

where N_0 is the Gaussian noise density at UE u. Similarly, the capacity of eavesdropper E's channel at generic point (x, y) can be expressed as follows:

$$C_E(x, y) = \frac{1}{2} \log \left(1 + \frac{P_E}{N_0 + \boldsymbol{F}_E} \right), \tag{8}$$

where P_E is the power received by eavesdropper E at generic point (x, y) which is defined as follows:

$$P_E = \bar{P}_{c,E} |\boldsymbol{H}_{c,E}|^2 = P_c G_c(\theta_c, \phi_{c,E}) G_E(\theta_E, \phi_{E,c}) |\boldsymbol{H}_{c,E}|^2, \tag{9}$$

where $G_c(\theta_c, \phi_{c,E})$ and $G_E(\theta_E, \phi_{E,c})$ are the antenna gains of BS c and eavesdropper E. $\phi_{c,E}$ denotes the angle between the segment connecting BS c and eavesdropper E and the x-axis. θ_E represents the angle between the direction of the main lobe of eavesdropper E's antenna and the x-axis. $\phi_{E,c}$ is the angle between the segment connecting eavesdropper E and BS c and the x-axis. $\boldsymbol{H}_{c,E}$ is the channel between BS c and eavesdropper E. $\boldsymbol{F}_E^k$ denotes the sum of interference power (on resource block k) at eavesdropper E (at generic point (x, y)) and can be expressed as in (10), where $P_{c'}$ is the power emitted by BS c', $d_{c',E}$ is the Euclidian distance between BS c' and eavesdropper E, and $h_{c',E}$ is the channel coefficient associated to the link.

Given the above, we can calculate the secrecy capacity of the link between BS c and UE u given the location (x, y) of eavesdropper E as $C_{\text{sec}}^{(u)}(x, y) = C_u - C_E(x, y)$. Most existing works assume the exact or statistical information of the location as well as the channel state information (CSI) of the eavesdropper (with respect to the BS and the user u), so as that they can further progress different optimization (e.g., resource allocation, beamforming design, etc). However, such an assumption is unrealistic as eavesdroppers are often passive devices and their presence/existence is even unknown. For that, this paper takes a pragmatic approach to define a new secrecy metric, called secrecy area, that liberates us from the dependence on the information/location of potential eavesdroppers. Specifically, given $C_{\text{sec}}^{(u)}(x, y)$, the secrecy area of the user u at time t can be defined as follows [10]:

$$A_{\text{sec}}^{(u)}(t) = \frac{1}{A} \iint_A g(x, y) dx dy, \tag{11}$$

where

$$g(x, y) = \begin{cases} 1 & \text{if } C_{\text{sec}}^{(u)}(x, y) \geq \varphi, \\ 0 & \text{if } C_{\text{sec}}^{(u)}(x, y) < \varphi, \end{cases} \tag{12}$$

where φ is a threshold to make sure that the legitimate link of user u is secured. φ should be large enough as a small gap between C_u and $C_E(x, y)$ may not helpful in practice. The system's secure area at time t then can be formulated as:

$$A_{\text{sec}}(t) = \sum A_{\text{sec}}^{(u)}(t), \forall u \in \boldsymbol{U}_c^t, \forall c \in \boldsymbol{C}. \tag{13}$$

In this work, we aim to maximize the average secure area given the presence of the eavesdropper. The optimization problem can be formulated as follows:

$$\begin{aligned} \max_{\zeta_{u,c,k}} \quad & \mathbb{E}(A_{\text{sec}}(t)), \\ \text{s.t.} \quad & \zeta_{u,c,k} \in \{0, 1\}, \forall u, \forall c, \forall k, \\ & \sum_{c \in \boldsymbol{C}} \sum_{k \in \boldsymbol{R}_u} \zeta_{u,c,k} = r_u, \forall u \in \boldsymbol{U}_c^t, \\ & \boldsymbol{R}_u \cap \boldsymbol{R}_{u'} = \emptyset, \forall u, u' \in \boldsymbol{U}_c^t, \forall c \in \boldsymbol{C}, \\ & \sum_{u \in \boldsymbol{U}_c^t} r_u \leq K, \forall u \in \boldsymbol{U}_c^t, \forall c \in \boldsymbol{C}. \end{aligned} \tag{14}$$

It can be observed that the secure area maximization problem in (14) is an NP-hard problem, and thus usually intractable to be solved by traditional optimization methods. More importantly, in our considered problem, the parameters in (14) are deterministic caused by the dynamics of mobile users and wireless communications. With these considerations, existing static optimization approaches may not be feasible to effectively obtain the joint optimal user association and resource allocation policy. Hence, in the following, we adopt the Markov decision process (MDP) [11] to capture the dynamics of our system and formulate a new optimization problem. Then, Q-learning and deep Q-learning based approaches are developed to obtain the optimal user association and resource allocation strategy to maximize the average secure area.

III. Problem Formulation

A. State Space

To effectively capture all properties of the system, we define the system state as a combination of the locations and resources available of all BSs in the system, the current user's location, the number of resource blocks and the serving time requested by the current user, and the current secure area of the system. We first denote $\boldsymbol{\Omega} \stackrel{\text{def}}{=} \{(x_c, y_c, \boldsymbol{R}_u) : \forall u, c\}$ as the set of the states of all BSs in the system, consisting of their locations and available resources. Then, the system state space $\mathcal{S}$ can be defined as follows:

$$\mathcal{S} \triangleq \{(x_u, y_u), \boldsymbol{\Omega}, r_u, t_u, A_{\text{sec}}\}, \tag{15}$$

$$\boldsymbol{F}_E^k = \sum_{c' \in \boldsymbol{C}} \sum_{u' \in \boldsymbol{U}_{c'}^t} \zeta_{u',c',k} P_{c'} G_{c'}(\theta_{c'}, \phi_{c',E}) G_E(\theta_E, \phi_{E,c'}) d_{c',E}^{-2b} |h_{c',E}|^2 = \sum_{c' \in \boldsymbol{C}} \sum_{u' \in \boldsymbol{U}_{c'}^t} \zeta_{u',c',k} \bar{P}_{c',E} |\boldsymbol{H}_{c',E}|^2 \tag{10}$$

where r_u, t_u, and (x_u, y_u) are the number of resource blocks and serving time required by the current user as well as its location, respectively. A_{sec} is the system's current secure area.

B. Action Space

When UE u arrives at the system, we need to jointly find BS c to serve this UE and select resource blocks in $\boldsymbol{R}_u$ for its communications. As such, the action space at state $s \in \mathcal{S}$ can be expressed as follows:

$$\mathcal{A}_s \triangleq \{a\} = \{(c, \boldsymbol{R}_u)\}, \forall c, \forall \boldsymbol{R}_u, \tag{16}$$

where a is the action taken at state s. It is worth noting that when a UE leaves the system, the secrecy of the whole system will be updated using (13) as the interference changes.

C. Immediate Reward

In this work, we aim to maximize the secure area of the system. Thus, the immediate reward after performing action a_t at state s_t is defined as the total secure area at the current time slot as follows:

$$r(s_t, a_t) = \sum A_{\text{sec}}^{(u)}(t), \forall u \in \boldsymbol{U}_t. \tag{17}$$

D. Optimization Formulation

The long-term secure area maximization problem can be formulated as follows:

$$\max_{\pi} \qquad \mathcal{R}(\pi) = \lim_{T \to \infty} \frac{1}{T} \sum_{t=1}^{T} \mathbb{E}\left(r_t(s_t, \pi(s_t))\right), \tag{18}$$

where π is the joint user association and resource allocation policy, $r_t(s_t, \pi(s_t))$ is the immediate reward after performing an action following policy π at state s_t. $\mathcal{R}(\pi)$ represents the average long-term secure area given policy π. In the next section, we present the Q-learning and deep Q-learning algorithms to solve this optimization problem.

IV. Defeating Eavesdroppers with Reinforcement Learning Algorithms

A. Q-learning

The key idea of the Q-learning algorithm is to learn the value of an action and a particular state, namely Q-value. To do that, it employs a Q-table to store the Q-values of all state-action pairs. By interacting with the environment, Q-learning can gradually update the Q-values and obtain the optimal policy in which the optimal action for a given state is the one that has the highest Q-value among all possible actions for that state. In particular, at state s_t, the algorithm determines action a_t by using the ϵ-greedy method [11]. Then, it observes the system's next state s_{t+1} and immediate reward r_t (defined in (17)). From this observation, the Q-value of state-action pair (s_t, a_t) will be updated by using the following equation:

$$\begin{aligned} \mathcal{Q}_{t+1}(s_t, a_t) = \mathcal{Q}_t(s_t, a_t) + \tau \Big[& r_t(s_t, a_t) \\ & + \gamma \max_{a_{t+1}} \mathcal{Q}_t(s_{t+1}, a_{t+1}) - \mathcal{Q}_t(s_t, a_t) \Big], \end{aligned} \tag{19}$$

where γ is the discount factor and τ is the learning rate that accounts for the impact of new experiences on the current Q-value. By frequently updating the Q-values in the Q-table through (19), Q-learning can converge to the optimal policy. Unfortunately, its performance in practice is limited, especially for high-dimensional state and action spaces as discussed in [11]. Hence, in the following, we propose a novel joint user association and resource allocation framework based on deep Q-learning to effectively learn the considered complex system and maximize the average secure area.

B. Deep Q-learning

To solve the aforementioned problems of Q-learning, deep Q-learning replaces the Q-table with two deep neural networks, namely Q-network and target Q-network, to approximate the Q-value of each state-action pair. The main procedure of our proposed deep Q-learning based approach is presented in Algorithm 1. In particular, the algorithm first initiate the Q-network $\mathcal{Q}$ and target Q-network $\hat{\mathcal{Q}}$ with random weights $\hat{\boldsymbol{\theta}} = \boldsymbol{\theta}$. Under the ϵ-greedy method, the algorithm selects action a_t at state s_t, performs this action, and observes immediate reward r_t and next state s_{t+1} from the environment. The experience (s_t, a_t, r_t, s_{t+1}) is then store in memory pool $\boldsymbol{D}$. During the training step, a number of random experiences will be taken from the memory pool and fed into the Q-network for training. By doing this, past experiences can be learned by the Q-network multiple times, resulting in a more efficient learning process [11]. In addition, by using the deep neural network, the system state can be directly fed into the input layer of the network, without requiring to be discrete as in Q-learning.

We denote $y_j = r_j + \gamma \max_{a'_j} \hat{\mathcal{Q}}(s'_j, a'_j; \hat{\boldsymbol{\theta}})$ as the target value for transition j in the random experience batch, where s'_j is the next state after state s_j and a'_j is a possible action at state s'_j, $\hat{\boldsymbol{\theta}}$ is the weights of the target Q-network. It is worth noting that we denote s'_j as the next state of state s_j instead of s_{j+1} because the transitions in the random sample set are randomly taken from the memory pool, and thus they may occur at different period during the training process. Then, the loss function of the deep Q-learning algorithm can be defined as follows:

$$L_j(\theta_j) = \mathbb{E}_{(s_j, a_j, r_j, s'_j) \sim U(\mathbf{D})} \Big[y_j - \mathcal{Q}(s_j, a_j; \boldsymbol{\theta}) \Big)^2 \Big], \tag{20}$$

where $\boldsymbol{\theta}$ represents the primary Q-network's weights, respectively. By minimizing the loss function, the weights of the

Q-network can be optimized and the optimal policy can be obtained. To do that, we first differentiate (20) with respect to the Q-network's weights to derive the gradient in (21).

$$\nabla_{\theta_j} L(\theta_j) = \mathbb{E}_{(s_j,a_j,r_j,s'_j)}\Big[\Big(y_j - \mathcal{Q}(s_j,a_j;\boldsymbol{\theta})\nabla_{\boldsymbol{\theta}}\mathcal{Q}(s_j,a_j;\boldsymbol{\theta})\Big)\Big]. \tag{21}$$

Algorithm 1 Deep Q-learning based User Association and Resource Allocation Algorithm

Initialize replay memory $\mathbf{D}$ to capacity $\mathcal{D}$.
Initialize the Q-network $\mathcal{Q}$ with random weights $\boldsymbol{\theta}$.
Initialize the target Q-network $\hat{\mathcal{Q}}$ with weight $\hat{\boldsymbol{\theta}} = \boldsymbol{\theta}$.
for *iteration=1 to I* **do**
    Perform action a_t based on ϵ-greedy method and observe reward r_t and next state s_{t+1}.
    Store transition (s_t, a_t, r_t, s_{t+1}) in $\mathbf{D}$.
    Sample random mini-batch of transitions (s_j, a_j, r_j, s'_j) from $\mathbf{D}$.
    $y_j = r_j + \gamma \max_{a'_j} \hat{\mathcal{Q}}(s'_j, a'_j; \hat{\boldsymbol{\theta}})$.
    Perform a gradient descent step on $(y_j - \mathcal{Q}(s_j, a_j; \boldsymbol{\theta}))^2$ with respect to $\boldsymbol{\theta}$.
    Every F steps reset $\hat{\mathcal{Q}} = \mathcal{Q}$.
end for

To calculate the gradient in (21), stochastic gradient descent can be adopted to minimize the loss function rather than calculating the above gradient's full expectations [12]. In this work, we use the Adam optimizer to minimize the loss function due to its superiority compared to other other stochastic optimization methods, thanks to the adaptive estimation of the first-order and second-order moments. Finally, the target Q-network's weights are updated with the Q-network's weights after every F iterations. By performing this process, the deep Q-learning based approach can obtain the optimal joint user association and resource allocation policy after a finite number of training iterations.

V. Performance Analysis and Simulation Results

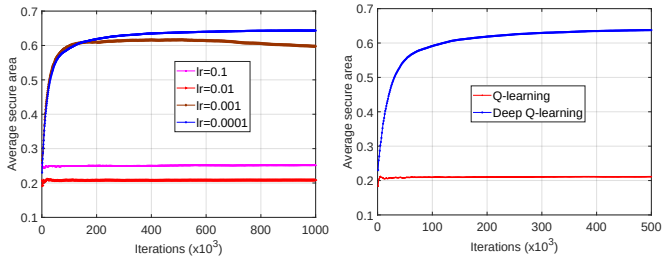
A. Parameter Setting

In our simulations, we consider an area of 1,000 meters $\times$ 1,000 meters with 5 BSs, and each BS has 10 resource blocks. The power radiated from corresponding serving BSs is set at 38 dBm for all users. The path loss model used in our simulations is the COST 231 Hata model, which is commonly used for big city scenarios [13]. Unless otherwise stated, μ, λ, and r_u are set at 0.8, 5, and 1, respectively. Since we consider the cases in which the locations of eavesdroppers are unknown, we discretize the considered area into a grid by a distance of 50 meters and the eavesdropper's location can be at any point in this grid for simulation purposes. The secrecy capacity threshold φ is set at 5, and the frequency is set at 1.8 GHz.

In our simulations, we deploy a simple deep neural network with four fully connected hidden layers, each consisting of 128 neurons. γ is set at 0.99. We set ϵ at 1 at the beginning of the training and gradually reduce it to 0.01 with a decay factor of 0.9999. In our simulation, the target Q-network is updated after every 5,000 iterations. The memory pool can store up to 10,000 experiences, and in each training step the algorithm randomly takes 32 experiences from the memory pool for training. To evaluate the performance of our proposed solution, we consider three baselines, including Q-learning, Random, and Greedy policies. As mentioned, the Q-learning algorithm requires the state space to be discrete to employ the Q-table. As such, it cannot use the same state space as our proposed deep Q-learning based approach, which has tens of millions of states if we discretize the state space. For that, we use a new state space for the Q-learning algorithm which consists of the number of available resource blocks at each BS and the current secure area. The secure area is discretized into 10 levels. The learning rate and discount factor of Q-learning are set at 0.1 and 0.9, respectively. The Random method randomly chooses an action to perform. Finally, the Greedy method selects an available BS that is closest to the current user, and the allocated resource block is randomly selected among available resources.

B. Simulation Results

1) Convergence Analysis: We first evaluate the convergence of the deep Q-learning algorithm with different learning rates, as shown in Fig. 1(a). As can be observed, with the learning rate of 0.0001, the deep Q-learning algorithm converges after 5×10^5 training iterations while the algorithm cannot learn the joint user association and resource allocation policy with the learning rates of 0.1 and 0.01. This is because, with large learning rates, the weights in the Q-network will be changed by large values, making the algorithm overshoot the minimum loss. In contrast, low learning rates can help to stabilize the learning process and avoid the overshooting problem. Given the above, we will use 0.0001 as the learning rate for the rest of the simulations.



(a) (b)

Fig. 1: Convergence of deep Q-learning (a) with different learning rates and (b) vs. Q-learning.

Next, Fig. 1(b) shows the learning processes of the proposed deep Q-learning and Q-learning based approaches. As can be observed, the Q-learning algorithm cannot learn the considered environment after 5×10^5 training steps. This is because the Q-learning algorithm discretizes the state space, resulting in

information loss. In addition, the Q-table is less efficient in learning the environment compared to the Q-network. This is demonstrated by the fact that the deep Q-learning algorithm can gradually learn the environment and converge to a much higher secure area within 5×10^5 training iterations.

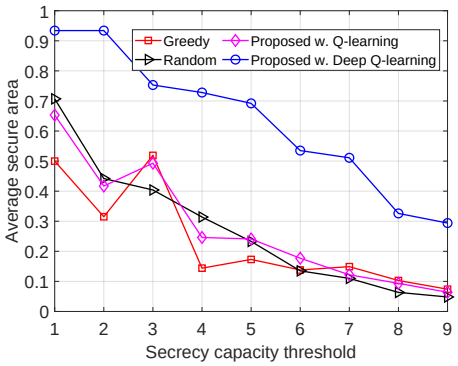


Fig. 2: Average secure area vs. secrecy capacity threshold φ.

2) Performance Evaluation: In this section, we perform simulations to evaluate the average secure areas obtained by our proposed deep Q-learning based approach and other baselines. In particular, we first vary the threshold φ defined in (12) and observed the average secure areas obtained by all the methods, as shown in Fig. 2. It is worth noting that a UE can only be considered in the secured area, i.e., where eavesdroppers cannot "wiretap" the channel when the secrecy capacity of the UE is higher than the threshold φ. As such, as can be observed in Fig. 2, the average secure area decreases when φ is increased. This is because, with higher values of φ, it is more difficult for a UE to be immune from eavesdropping attacks. However, in all cases, our proposed deep Q-learning algorithm can obtain the best performance as it can learn UEs' properties and the dynamics of the system to obtain the joint user association and resource allocation policy.

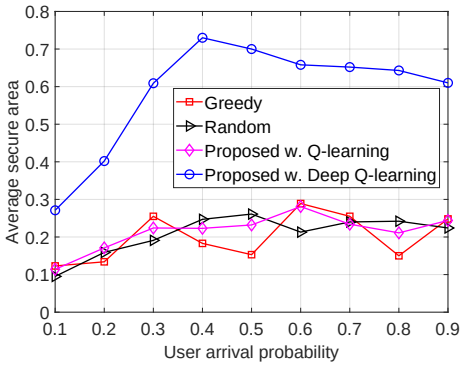


Fig. 3: Average secure area vs. UE arrival probability μ.

We then vary the UE arrival probability and show the average secure area of all the approaches in Fig. 3. As can be seen, the average secure area is increased when μ is increased from 0.1 to 0.4. This is because with more UEs, the algorithm can better leverage the interference from UEs to prevent eavesdroppers from "wiretapping" the channel. However, when μ is higher than 0.4, the average secure area is decreased from around 0.73 to 0.6. The reason is when there are many UEs served in the considered area, the probability that the eavesdropper can "wiretap" their channels is higher. Nevertheless, our proposed solution still achieves the best performance compared to other methods.

VI. Conclusion

In this paper, we have proposed a novel anti-eavesdropping solution by leveraging co-channel interference to disrupt the reception of potential eavesdroppers. Given the dynamics of users and the uncertainty of wireless environments, it is challenging to associate users to base stations and at the same time to allocate resources to serve them in a way that the co-channel interference can be utilized. Thus, we proposed a deep Q-learning based approach to learn all the environment's properties and then obtain the optimal policy to maximize the secure area. Extensive simulations then demonstrate the effectiveness of our solution in terms of the average secure area and the learning efficiency, compared to other baselines.

References

[1] H. Jeon, J. Choi, S. W. McLaughlin, and J. Ha, "Channel aware encryption and decision fusion for wireless sensor networks," *IEEE Trans. Inf. Forensics Security*, vol. 8, pp. 619–625, Apr. 2013.

[2] N. V. Huynh, N. Q. Hieu, N. H. Chu, D. N. Nguyen, D. T. Hoang, and E. Dutkiewicz, "Defeating Eavesdroppers with Ambient Backscatter Communications," *IEEE WCNC*, Glasgow, United Kingdom, Mar. 2023.

[3] N. H. Chu, N. V. Huynh, D. N. Nguyen, D. T. Hoang, S. Gong, T. Shu, E. Dutkiewicz, and K. T. Phan, "Countering Eavesdroppers with Meta-learning-based Cooperative Ambient Backscatter Communications," *IEEE Trans. Wireless Commun.*, Early Access, 2024.

[4] X. Lu, E. Hossain, T. Shafique, S. Feng, H. Jiang, and D. Niyato, "Intelligent reflecting surface enabled covert communications in wireless networks," *IEEE Network*, vol. 34, no. 5, pp. 148-155, Sept/Oct. 2020.

[5] D. J. Bernstein and T. Lange, "Post-quantum cryptography," *Nature*, vol. 549, no. 7671, pp. 188-194, Sept. 2017.

[6] L. Mucchi, L. Ronga, X. Zhou, K. Huang, Y. Chen, and R. Wang, "A new metric for measuring the security of an environment: The secrecy pressure," *IEEE Trans. Wireless Commun.*, vol. 16, no. 5, pp. 3416-3430, May 2017.

[7] P. Siyari, M. Krunz, and D. N. Nguyen, "Friendly jamming in a MIMO wiretap interference network: A nonconvex game approach," *IEEE J. Sel. Areas Commun.*, vol. 35, no. 3, pp. 601-614, Mar. 2017.

[8] K. Li, Y. Ren, Z. Lin, and L. Xiao, "Reinforcement Learning Based Friendly Jamming for Digital Twins Against Active Eavesdropping," *IEEE MSN*, 14-16 December 2023, Nanjing, China.

[9] Y. Zhou, P. L. Yeoh, C. Pan, K. Wang, Z. Ma, B. Vucetic, and Y. Li, "Caching and UAV friendly jamming for secure communications with active eavesdropping attacks," *IEEE Trans. Veh. Technol.*, vol. 71, no. 10, pp. 11251-11256, Oct. 2022.

[10] D. Marabissi, L. Mucchi, and S. Casini, "Physical-layer security metric for user association in ultra-dense networks," *IEEE ICNC*, Big Island, HI, USA, 17-20 Feb. 2020.

[11] D. T. Hoang *et al.*, *Deep Reinforcement Learning for Wireless Communications and Networking: Theory, Applications and Implementation*. Wiley-IEEE Press, 2023.

[12] V. Mnih, K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, M. G. Bellemare, A. Graves, et al., "Human-level control through deep reinforcement learning," *Nature*, vol. 518, no. 7540, pp. 529-533, Feb. 2015.

[13] R. V. Akhpashev and A. V. Andreev, "COST 231 Hata adaptation model for urban conditions in LTE networks," *IEEE EDM*, Erlagol, Russia, 2016.