SATCOM Jamming Resiliency under Non-Uniform Probability of Attacks

Lan K. Nguyen*, Duy H. N. Nguyen‡, Nghi H. Tran†, Clayton Bosler♭, David Brunnenmeyer*
*LinQuest Corp, Los Angeles, CA
‡Dept. of Electrical Engineering, San Diego State University, San Diego, CA
† Dept. of Electrical Engineering, University of Akron, Akron, OH
♭LinQuest Corp, Dayton, OH

MILCOM 2021 - 2021 IEEE Military Communications Conference (MILCOM) | 978-1-6654-3956-5/21/$31.00 ©2021 IEEE | DOI: 10.1109/MILCOM52596.2021.9652944

Abstract —**This paper presents a new framework for SATCOM jamming resiliency in the presence of a smart adversary jammer that can prioritize specific channels to attack with a non-uniform probability of distribution. We first develop a model and a defense action strategy based on a Markov decision process (MDP). We propose a greedy algorithm for the MDP-based defense algorithm's policy to optimize the expected user's immediate and future discounted rewards. Next, we remove the assumption that the user has specific information about the attacker's pattern and model. We develop a Q-learning algorithm-a reinforcement learning (RL) approach-to optimize the user's policy. We show that the Q-learning method provides an attractive defense strategy solution without explicit knowledge of the jammer's strategy. Computer simulation results show that the MDP-based defense strategies are very efficient; they offer a significant data rate advantage over the simple random hopping approach. Also, the proposed Q-learning performance can achieve close to the MDP approach without explicit knowledge of the jammer's strategy or attacking model.**

Keywords — ***SATCOM Resiliency; Sweeping Jamming Attack; Markov Decision Process; Reinforcement Learning; Q-learning.***

I. INTRODUCTION

As noted in the U.S. 2018 National Defense Strategy [1] and Defense Science Board Report [2], the projected jamming attacks against satellite communication (SATCOM) have rapidly escalated in recent years. Jammers pose significant threats to both commercial and military SATCOM. By intentionally emitting jamming signals, adversaries can disrupt users' communication services, resulting in throughput degradation. Over the last decade, advanced anti-jamming approaches have been studied extensively in the literature. Game theory is considered an effective tool to address users' and jammers' interactions to determine the best defense strategies when facing the decision for both wireless communication and SATCOM [3-9]. For instance, the work in [3] appears to be one of the first anti-jamming studies that used game theory to exploit the flexible access to multiple channels to defeat jammers. Specifically, the paper presented an optimal defense strategy that employs a Markov decision process (MDP) to deal with smart sweeping attacks that occur uniformly across multiple channels. This approach was extended further in [8, 9] against the sweeping attacks under joint frequency hopping and rate adaptation.

The hopping strategies developed in [3, 8, 9] assume that an attacker can randomize the sweeping order; the attacking probability distribution is uniform across multiple channels. However, an effective and intelligent jammer can decide to attack with non-uniform probabilities. This strategy is more unpredictable and creating far more damaging effects. In this case, an optimal defense strategy requires a revised MDP-based hopping strategy that relies on the non-uniform probability assignments. To this end, we first derive a transition model and a defense action based on the new MDP. We then develop a greedy algorithm that maximizes the user's immediate reward and expected reward conditioned on the current action. Computer simulation results show that the proposed MDP-based strategy achieves significantly higher data rates than a traditional minimum hopping method.

Next, we remove the assumption that the user has specific knowledge about the jammer's attacking probability and strategy. In a dynamic environment, acquiring such information is unrealistic. Reinforcement learning (RL) [10], such as Q-learning and multi-agent RL techniques, allows the system to quickly and effectively fill in the missing information in response to an adversary's attack. They provide attractive alternatives in developing an efficient defense strategy without the need for an explicit jamming model [11-15]. This approach's novelty is that the user can learn an optimal countermeasure strategy without knowing the attacking strategy or the non-uniform probability of assignments. To this end, we develop a novel Q-learning defense strategy. First, we initiate a Q-table to approximate an action-value function. We then develop a Q-learning algorithm for approximating an optimal hopping strategy. Our numerical results demonstrate that this technique achieves close to the MDP-based defense strategy without explicit knowledge of the jammer's attacking model.

II. SYSTEM MODEL

We consider a jamming scenario where a smart jammer attacks an uplink SATCOM user. We assume N orthogonal or alternate communication links or channels, each differing in either frequency band, beam, antenna, or transponder available to the user. In this paper, the term hopping means choosing an alternate path, which is different from the traditional hopping systems, where the user hops within the hopping bandwidth. When a user accesses a channel $n, 1 \le n \le N$, we define the signal-to-noise ratio (SNR) as $\gamma_{n,1}$ if there is no jammer on that channel. By contrast, the user SNR is $\gamma_{n,0}$ if there is a jammer on that channel. Clearly, $\gamma_{n,1} > \gamma_{n,0}$. When the jammer emits high-power jamming signals, the jammed channel is unusable. In each time slot, we assume the user can access one channel, and the jammer can only attack one channel at a time.

To simplify the analysis, we omit the transmission delay. We assume that the jammer knows after the fact that if its jamming attack at a given time slot is successful or not. The jammer employs a sweeping attack strategy with a non-uniform probability distribution until it successfully jams the user. The jammer will continue to jam that channel until the user hops to another channel. This strategy incapacitates a countermeasure strategy where the user can stay in a channel all the time. The jammer then restarts its sweeping jamming cycle.

We consider a game between the user and the jammer that repeats over multiple time slots. At time slot t, the user's payoff at the channel n is given by

$$\begin{aligned} U(t) = {} & R_n \cdot \mathbf{1}(Sucessful\ transmission\ on\ channel\ n) \\ & + L_n \cdot \mathbf{1}\ (Jammed\ on\ channel\ n) \\ & - C \cdot \mathbf{1}\ (Choosing\ the\ action\ to\ hop)\ , \end{aligned} \tag{1}$$

where C is the cost associated with the action to choose an alternate channel, subscript n denotes the channel index, R_n and L_n are the reward and loss for channel n, respectively, and $\mathbf{1}$ is the indicator function that returns a value of 1 when the statement holds true and 0 otherwise. The user aims to maximize the discounted sum of payoffs. It is given as

$$U_T = \sum_{t=1}^{\infty} \delta^t\, U(t), \tag{2}$$

where $\delta \in (0,1)$ is the discount factor.

III. ANALYSIS

A. *Anti-Jamming Game Agaisnt a Smart Jammer using Uniform Probability of Sweeping Attack*

This section examines an anti-jamming game against a smart jammer that uses a uniform probability sweeping attack strategy and how it reacts to the user's minimum hopping strategy [3, 11].

Markov Decision Process Model

We consider a Markov decision process (MDP) consisting of four components:

1. State space: a set of finite states,
2. Action space: a set of finite actions,
3. Transition probabilities: probability going from the current state to the next state by taking an action from the action space,
4. Reward: an immediate reward received after the state transition

State space: The MDP's state is defined as the number of consecutive time slots that the user has successfully occupied on that channel since the last time it occupied it. Let $\mathcal{X}$ denote the state space, then

$$\mathcal{X} = \{0, 1, 2, \cdots, L-1\}, \tag{3}$$

where 0 denotes the state that the user is jammed and $L-1$ is the highest number of consecutive time slots that the user can stay before hopping to another channel.

Action space: At the end of each time slot, the user decides whether to stay with action $a = s_n$ on the current channel-n or hop to a new channel-m with the action $a = h_m$. Thus, the action space is defined as

$$\mathcal{A} = \{s_1, s_2, \cdots, s_N, h_1, h_2, \cdots, h_N\}. \tag{4}$$

Transition probabilities: If the current state is jammed, the user may choose to hop to a new channel n or stay at the current channel. The jammer can also try to jam another channel or can continue to attack the current channel. Thus, the probability that the channel-n is jammed in the next time slot is given by

$$P(0|0, h_n) = 1/N, \tag{5}$$

and

$$P(1|0, h_n) = 1 - P(0|0, h_n) = 1 - 1/N. \tag{6}$$

If the channel is at state $X \neq 0$ (not jammed), the user decides to hop to a new channel-n. The user's next state can either be 0 (jammed) or 1 (not jammed). The next state is 1 if the jammer already swept channel n or has not swept channel n in the last X time slots and will sweep the next time slot. Thus,

$$P(1|X, h_n) = \frac{X}{N} + \left(1 - \frac{1}{N-X}\right)\frac{N-X}{N} = 1 - \frac{1}{N}, \tag{7}$$

and

$$P(0|X, h_n) = 1 - P(0|X, h_n) = 1/N. \tag{8}$$

The result shown in (8) indicates that if the user decides to hop to a new channel n, the jamming probability for that channel is still $1/N$.

If the user stays on channel-n for X time slots without being jammed and decides to stay one more time slot, we have

$$P(0|X, s_n) = \frac{1}{N-X}, \tag{9}$$

and

$$P(X+1|X, s_n) = 1 - P(0|X, s_n) = 1 - \frac{1}{N-X}. \tag{10}$$

Reward: Immediate reward at time slot t for channel-n after going from state X to state X' when an action $a \in \mathcal{A}$ is taken is given by [11]

$$U(X', X, a) = \begin{cases} R_n, & if\ X' \neq 0, a = s_n \\ L_n, & if\ X' = 0, a = s_n \\ R_n - C, & if\ X' \neq 0, a = h_n \\ L_n - C, & if\ X' = 0, a = h_n \end{cases}. \tag{11}$$

Note that if the user decides to stay, the next state X' can be either 0 (jammed) or $X+1$. Likewise, if the user chooses to hop, the next state X' can either be 0 or 1 (not jammed for the new channel for one-time slot).

MDP Based Defense Strategy

For an MDP, a policy π is defined as a mapping from a state $X \in \mathcal{X}$ to an action $a \in \mathcal{A}$. The action would indicate the current channel that the user should stay or a new channel that the user should hop to. Assuming the MDP starts from state X, we want to find an optimal policy π^* so that

$$\pi^*(X) = arg \max_{\pi} E(U_T | X_1 = X). \quad (12)$$

The above optimum policy maximizes the expected sum of the immediate and future discounted rewards conditioned on the current state and the action. A solution to equation (12) is the well-known Bellman equation for the expected utility maximization. It is given by

$$Q(X, a) = U(X, a) + \delta \sum_{X'} p(X'|X, a) V(X'), \quad (13)$$

where the expected immediate reward $U(X, a)$ is

$$U(X, a) = \sum_{X'} p(X'|X, a) U(X', X, a), \quad (14)$$

and $V(X')$ is the value of future reward at state X'. The optimal reward value evaluated at state X is given by [13]

$$V^*(X) = \max_{a \in \mathcal{A}} Q(X, a). \quad (15)$$

Using equation (15), [3, 11] proposed a value iteration algorithm for estimating the optimal policy π^* for equation (12). Note that under the uniform probability channel attack, the user always starts at a channel that yields the highest $U(0, a)$ until it decides to hop [8]. Readers refer to [16] for details of an optimal policy when the user utilizes its history to minimize the regret associated with its action over the weighted probability distribution of parameters.

B. Anti-Jamming Game Agaist a Smart Jammer using Non-Uniforml Probability of Sweeping Attack

This section investigates the jammer's attacking strategy and the user's MDP based defense strategy under non-uniform probability of sweeping attack. First, we assume that the user can build a belief vector for the channels that the jammer will attack; we then develop a greedy algorithm aimed to optimize the user's immediate reward. We also propose a value iteration algorithm to approximate the optimal strategy (12).

Non-Uniform Probability for Sweeping Attack Strategy

Suppose that the jammer will attack N channels at the start of sweeping sequence with different prior probabilities $\boldsymbol{q}^{(1)} = [q_1, q_2, \cdots, q_N]$, where $\sum_{m=1}^{N} q_m = 1$. Here, q_m is the jamming probability for channel-m in the first round, and $q_m > q_n$ implies that the jammer is more likely to attack channel-m than channel-n.

If the jammer attacks channel-m in the first round and the user has not occupied that channel, the jammer needs to revise the attacking probability for channel $n \neq m$ in the second round. We consider two rules for adjusting the attacking probability vector as follows:

Rule 1: Split q_m proportionally to q_n such that

$$q_n^{(2)} = \frac{q_n}{1-q_m} = q_n + \frac{q_n q_m}{1-q_m}. \quad (16)$$

Rule 2: Split q_m equally to all other channels such that

$$q_n^{(2)} = q_n + \frac{q_m}{N-1}. \quad (17)$$

Using the above rules, we can generalize to find the jamming probability for all channels in the $(K+1)^{th}$ round. Specifically, given $\mathcal{N}_K$ as the set of attacked channels in the first K rounds, the next channel $n \notin \mathcal{N}_K$ has the following jamming probability, depending on what rule the jammer shall follow:

Rule 1: Split $\sum_{m \in \mathcal{N}_K} q_m$ proportionally to q_n such that

$$q_n^{(K+1)} = \frac{q_n}{1-\sum_{m \in \mathcal{N}_K} q_m} = q_n + \frac{q_n \sum_{m \in \mathcal{N}_K} q_m}{1-\sum_{m \in \mathcal{N}_K} q_m}. \quad (18)$$

Rule 2: Split $\sum_{m \in \mathcal{N}_K} q_m$ equally to all other channels such that

$$q_n^{(K+1)} = q_n + \frac{\sum_{m \in \mathcal{N}_K} q_m}{N-K}. \quad (19)$$

Note that the updates in (18) and (19) ensure that $\sum_{m \notin \mathcal{N}_K} q_m^{(K+1)} = 1$ from the jammer's perspective.

During the sweeping attack, the user assumes to occupy channel-n and the user does not know which channel the jammer has visited in the previous K rounds. The user only knows that the jammer has not attacked channel $n \notin \mathcal{N}_K$ in these K rounds. For this reason, the user can only build a belief vector on the channels that the jammer will attack next.

MDP-based Defense Strategy

Let $\boldsymbol{\gamma}^{(1)} = [\gamma_1, \gamma_2, \cdots, \gamma_N]$ be the belief vector at the onset of the sweeping sequence. We assume that $\boldsymbol{\gamma}^{(1)} = \boldsymbol{q}$ is known a priori by the user. Suppose that in round 1, the jammer does not attack channel-n. Indeed, the attacking probability for channel-n increases in the next round. Since the user does not know which channel-m the jammer has attacked in the first round, it takes an approximate $q_m \approx (1 - q_n)/(N-1)$. The belief that the jammer will attack channel-n in the second round based on the jammer's attacking strategy is

Rule 1: The user would split $(1 - q_n)/(N-1)$ proportional to q_n such that

$$\gamma_n^{(2)} = \frac{q_n}{1 - \frac{1-q_n}{N-1}} = q_n + \frac{q_n}{1 - \frac{1-q_n}{N-1}} \cdot \frac{1-q_n}{N-1}, \quad (20)$$

where the second term on the right-hand side (RHS) of (20) is the added belief that the jammer will attack channel-n next. On the other hand, the belief of attacking different channels, say

$m \neq n$, should decrease. We then split $\frac{q_n}{1-\frac{1-q_n}{N-1}} \cdot \frac{1-q_n}{N-1}$ proportionally to q_m as follows:

$$\gamma_m^{(2)} = q_m - \frac{q_m}{1-q_n} \cdot \frac{q_n}{1-\frac{1-q_n}{N-1}} \cdot \frac{1-q_n}{N-1} = q_m - \frac{q_m q_n}{N+q_n-2}. \quad (21)$$

Rule 2: The user would split $(1-q_n)/(N-1)$ equally and increase the attacking belief for channel-n as

$$\gamma_n^{(2)} = q_n + \frac{1-q_n}{(N-1)^2}. \quad (22)$$

On the other hand, user will split the added belief $\frac{1-q_n}{(N-1)^2}$ equally to q_m, for $m \neq n$ as

$$\gamma_m^{(2)} = q_m - \frac{1-q_n}{(N-1)^3}. \quad (23)$$

It is easy to verify that both updating rules maintain the condition $\sum_{n=1}^{N} \gamma_n^{(2)} = 1$.

We can generalize to a scenario where a user is on channel-n and has not been attacked for the first K consecutive rounds based on the attacker's strategy. The belief on the attacking probability in the next round is as follows:

Rule 1: The belief on channel-n is increased as

$$\gamma_n^{(K+1)} = \frac{q_n}{1-K\cdot\frac{1-q_n}{N-1}} = q_n + \frac{q_n K(1-q_n)}{N-1-K(1-q_n)}, \quad (24)$$

and the belief on channel $m \neq n$ is reduced as follows:

$$\gamma_m^{(K+1)} = q_m - \frac{q_m}{1-q_n} \cdot \frac{q_n K(1-q_n)}{N-1-K(1-q_n)}. \quad (25)$$

Rule 2: The belief on channel-n is increased as

$$\gamma_n^{(K+1)} = q_n + \frac{K(1-q_n)}{(N-1)(N-K)}, \quad (26)$$

and the belief on channel $m \neq n$ is reduced as follows:

$$\gamma_m^{(K+1)} = q_m - \frac{K(1-q_n)}{(N-1)^2(N-K)}. \quad (27)$$

Note that should $K = N-1$, updating rules (24) and (26) result $\gamma_n^{(N)} = 1$ as expected, i.e., channel-n is the last attacked channel.

Based on the belief vector $\boldsymbol{\gamma}^{(K+1)}$, we can define the transition model $T\big((x',n')\big|(x,n),a\big)$, where (x,n) denotes the number of consecutive time slots that the jammer has not attacked channel-n and a is the user's action. Here we define the action space $\mathcal{A} = \{a_1, a_2, \cdots, a_N\}$, where a_n means that the user will access channel-n. The transition state (x',n') and the transition model $T(\cdot \mid \cdot, a)$ depend on the action $a \in \mathcal{A}$ and the channel being jammed in the next time slot is given as follows: If the user decides to stay on channel-n, i.e., $a = a_n$, the next state is $n' = n$ and x' is either 0 (jammed) or $x+1$. Furthermore,

$$T\big((0,n)\big|(x,n),s\big) = \gamma_n^{(x+1)}, \quad (28)$$

and

$$T\big((x+1,n)\big|(x,n),s\big) = 1 - \gamma_n^{(x+1)}. \quad (29)$$

On the other hand, if the user decides to hop to channel-m, $m \neq n$, i.e., $a = a_m$, the next state is $n' = m$ and again, x' is either 0 (jammed) or $x+1$. Furthermore,

$$T\big((0,m)\big|(x,n),h_m\big) = \gamma_m^{(x+1)}, \quad (30)$$

and

$$T\big((1,m)\big|(x,n),h_m\big) = 1 - \gamma_m^{(x+1)}. \quad (31)$$

Also, if the user hops to a new channel, the belief vector is reset to $\boldsymbol{\gamma}^{(1)} = \boldsymbol{q}$. If the user is jammed on a particular channel-n at time slot t, the jammer will continue to jam channel-n in time slot $t+1$. Thus, the belief $\boldsymbol{\gamma}^{(0)}$ is temporarily set to an all-zero-vector, except $\gamma_n^{(0)} = 1$, i.e., $\boldsymbol{\gamma}^{(0)} = \big[0,\cdots,0,\gamma_n^{(0)} = 1,0,\cdots,0\big]$. In the next time slot $t+2$, since the jammer will restart a new sweeping sequence, the belief vector is thus reset to $\boldsymbol{\gamma}^{(1)} = \boldsymbol{q}$.

Greedy Algorithm

The goal of the algorithm is to optimize the user's expected immediate reward. Suppose that the user has stayed at a given channel-n for x time slots without being jammed. Herein, if $x = 0$, it means that the jammer just jammed the user. The user then evaluates the expected payoff $\mu_m^{(x+1)}$ for accessing channel-m in the next time slot, based on the belief vector $\boldsymbol{\gamma}^{(x+1)} = \big[\gamma_1^{(x+1)},\cdots,\gamma_N^{(x+1)}\big]$ as follows:

$$\mu_m^{(x+1)} = \begin{cases} \gamma_m^{(x+1)}(L_m - C) + \big(1-\gamma_m^{(x+1)}\big)(R_m - C), & if\ m \neq n \\ \gamma_n^{(x+1)} L_n + \big(1-\gamma_n^{(x+1)}\big)R_n, & if\ m = n \end{cases}. \quad (32)$$

This evaluation means that if the user decides to stay at channel-n, it does not have to pay for the hopping cost. If the user is attacked, the belief vector will be reset to $\boldsymbol{\gamma}^{(1)}$. Based on this expected payoff evaluation, an ϵ-greedy algorithm can be used to decide the next action for the user. This includes staying at the current channel-n or hopping to a new channel-m. The user generates a probability vector assignment $\boldsymbol{p} = [p_1, p_2, \cdots, p_N]$ to select the next channel, where

$$p_n = \begin{cases} 1-\epsilon+\frac{\epsilon}{N}, & if\ n = arg\max_m \mu_m^{(x+1)} \\ \frac{\epsilon}{N}, & otherwise \end{cases}. \quad (33)$$

Value Iteration for Approximating the Optimal Policy

Assuming the MDP starts from state (x,n), we want to find an optimal policy π^* that maximizes the expected immediate reward and the expected future discounted reward conditioned on the current action $a \in \mathcal{A}$. It is given as

$$\pi^* = arg \max_{\pi} E\big(U_T\big|(x,n)\big). \quad (34)$$

A solution for (34) is the Bellman equation; it is given as

$$Q\big((x,n),a\big) = U\big((x,n),a\big) + \delta \sum_{(x',n')} T\big((x',n')\big|(x,n),a\big)V(x',n'), \quad (35)$$

where the immediate reward is $U\big((x,n),a\big)$; it is given as

$$U\big((x,n),a\big) = \sum_{(x',n')} T\big((x',n')\big|(x,n),a\big) \cdot U\big((x',n'),(x,n),a\big). \quad (36)$$

Again, $\delta \in (0,1)$ is the future discounted reward and $V(\cdot)$ is the value of the future reward. The optimal value of the future reward at a given state (x,n) is given by [13]

$$V^*(x,a) = \max_{a\in\mathcal{A}} Q\big((x,n),a\big). \quad (37)$$

The value iteration algorithm for (37) that solves (34) is summarized in Algorithm 1.

Algorithm 1 Value Iteration for Estimating π^*

Input: Parameter ϵ as the stopping criterion and parameter δ as the discount factor
Initialization: $V(x,n), \forall x, \forall n,\ \Delta > \epsilon$
while $\Delta > \epsilon$ **do**
  $\Delta \leftarrow 0$
  for each x, n **do**
  $v \leftarrow V(x,n)$
  $Q\big((x,n),a\big) \leftarrow U\big((x,n),a\big) + \delta \sum_{(x',n')} T\big((x',n')\big|(x,n),a\big)V(x',n')$
  $V(x,n) \leftarrow \max_{a\in\mathcal{A}} Q\big((x,n),a\big)$
  $\Delta \leftarrow max(\Delta, |v - V(x,n)|)$
  end for
end while
Output: A deterministic policy π such that
$$\pi(x,n) = arg \max_{a\in\mathcal{A}} Q\big((x,n),a\big)$$

C. Q-Learning Solution

While the approach discussed in the previous section is effective, the derivation of the belief vector for the MDP-based defense strategy requires the knowledge of the non-uniform attacking probability $\boldsymbol{q}$ apriori. In a dynamic jamming scenario, this information may not be readily available for the user. To overcome this drawback, we propose a Q-learning algorithm to learn the optimal defense strategy without explicit knowledge of the jammer's strategy or attacking probability $\boldsymbol{q}$. The user initiates a Q-learning table to approximate an action-value function [10]. Depending on the current state, $s_t = (x_t, n_t)$, the user takes action a_t to stay at channel n_t or hop to a new channel n', resulting a new state s_{t+1}. The user then observes its payoff U_{t+1} from assessing the channel and updates the action-value function $Q(s_t, a_t)$ as follows:

$$Q(s_t,a_t) \leftarrow Q(s_t,a_t) + \alpha\Big[U_{t+1} + \delta \max_{a}\big(Q(s_{t+1},a) - Q(s_t,a_t)\big)\Big], \quad (38)$$

where α is the learning rate. The learned action-value function $Q(\cdot)$ shown above approximates the optimal action-value function [10]. An approximate optimal policy π^* is then drawn from the $Q(\cdot)$ function (38). Algorithm 2 summarizes the Q-learning algorithm for approximating the optimal policy π^*.

Compared to the value iteration in Algorithm 1, the Q-learning algorithm does not require the transition model $T(\cdot \mid \cdot, a)$ or the belief vector's derivation $\boldsymbol{\gamma}$. However, the Q-learning algorithm requires more computational power and takes a longer time to find the optimal policy than Algorithm 1.

Algorithm 2 Q-learning for Estimating π^*

Input: Parameter α as the learning rate, parameter δ as the discount factor, and small ϵ
Initialization: $Q\big((x,n),a\big), \forall x, \forall n$
for each episode **do**
  Initialize (x,n)
  for loop for each step of the episode: **do**
    choose α using policy derived from Q, e.g., $\epsilon-$greedy.
    Take action a, observe payoff U, and new state (x',n')
    $Q\big((x,n),a\big) \leftarrow (1-\alpha)Q\big((x,n),a\big) + \alpha\Big[U + \delta \max_{a} Q\big((x',n'),a\big)\Big]$
    $(x,n) \leftarrow (x',n')$
  end for
end for
Output: A deterministic policy π such that
$$\pi(x,n) = arg \max_{a\in\mathcal{A}} Q\big((x,n),a\big)$$

IV. SIMULATION RESULTS

We performed Monte Carlo simulations over 4000-time slots to demonstrate the effectiveness of the proposed defense strategies. In the simulations, the discount rate is set at $\delta = 0.95$, the cost associated with hopping is set at $C = 1$, the initial set of probabilities q is proportional to SNR, and the data rate $log(1+SNR)$ is the payoff. Under sweeping attack, the SNR of the jammed channel is reduced by 20 dB. Figure 1 shows the user's data rates achieved by the MDP-based iteration value Algorithm 1 and the greedy algorithm. In this simulation, the user has access to 10 orthogonal communication links with SNR in the range $[10, 19]$ (dB). For comparison, we also include the data rate achieved by a minimum random hopping scheme,

where the user would only hop to a random channel once the jammer hits the channel. The proposed MDP-based defense strategy offers a significant data rate advantage over the minimum random hopping method. Figure 2 displays the user's data rate results for the iterative Q-learning Algorithm 2 for the defense strategy. In this simulation, the user's SNR in the range $[10,\ 25]$ (dB), and there are 16 orthogonal channels available to the user. Over time, the Q-learning Algorithm 2 achieves similar results as that of the MDP-based Algorithm 1 without the need to have prior knowledge of the attacker's strategy.

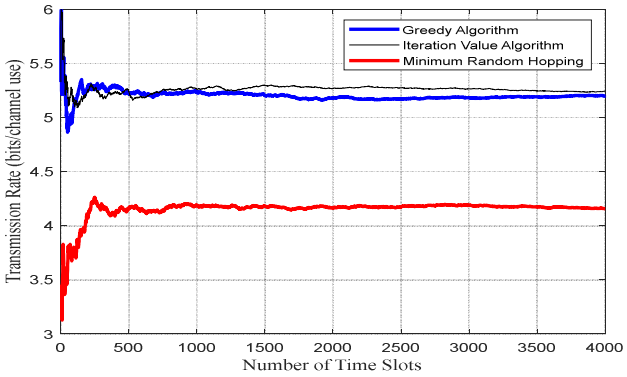

Fig. 1. Transmission rates achieved by different defense strategies, 10 orthogonal channels, SNR = [10, 19] (dB).

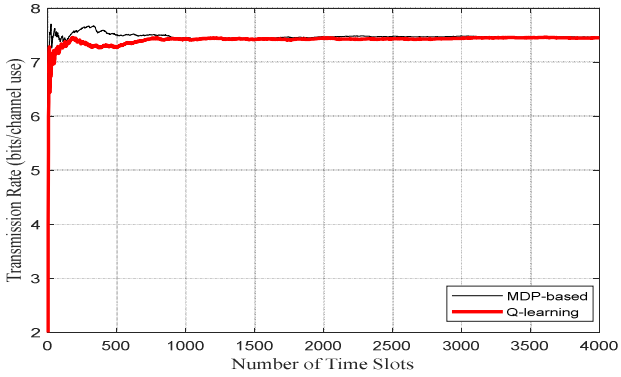

Fig. 2. Transmission rates achieved by different defense strategies, 16 orthogonal channels, SNR = [10, 25] (dB).

V. Conclusion

This paper presented a novel approach for SATCOM jamming resiliency under the non-uniform probability of sweeping attacks. We first developed an MDP-based defense strategy. We next developed a greedy algorithm to optimize the expected user's immediate rewards and future discounted rewards. Computer simulation results showed that the proposed MDP-based defense algorithm exceeded the traditional random hopping approach's performance. Also, we proposed a Q-learning algorithm for the user. We showed that at steady state the Q-learning algorithm achieved similar performance as the MDP-based defense algorithm without knowing the jammer's attacking model or strategy. The results suggest that the proposed framework can effectively provide a countermeasure against any smart jammers that employ a non-uniform probability of sweeping attacks.

ACKNOWLEDMENT

The authors would like to thank the reviewers' comments.

References

[1] J. Mattis, "Summary of the 2018 national defense strategy of the United States," *Washington: Department of Defense*, 2018.

[2] DoD, "Military satellite communication and tactical networking," *Department of Defense: Defense Science Board Report*, Mar. 2017.

[3] Y. Yu, B. Wang, K. J. R. Liu, and T. C. Clancy, "Anti-jamming games in multi-chanel cognitive radio networks," *IEEE Journal on Selected Areas in Communications*, vol. 30, no. 1, pp.4-15, Jan 2012.

[4] D. Yang, G. Xue, J. Zhang, A. Richa, and X. Fang, "Coping with a smart jammer in wireless networks: A Stackelberg game approach," *IEEE Transactions on Wireless Communications*, vol. 12, no. 8, pp. 4038–4047, Aug. 2013.

[5] A. Garnaev, Y. Liu, and W. Trappe, "Anti-jamming strategy versus a low-power jamming attack when intelligence of adversary's attack type is unknown," *IEEE Transactions on Signal and Information Processing, over Networks*, vol. 2, no. 1, pp. 49–56, Mar. 2016.

[6] L. Jia, F. Yao, Y. Xu, Y. Sun, S. Feng, and A. Anpalagan, "Stackelberg game approach for anti-jamming defense wireless networks," *IEEE Wireless Communications*, vol. 25, no. 6, pp. 120–128, Dec. 2018.

[7] C. Han, A. Liu, H. Wang, L. Huo, and X. Liang, "Dynamic anti-jamming coalition for satellite-enabled Army IoT : a distrubuted game approach," *IEEE Internet of Things Journal*, vol. 7, pp. 10932-10944, Nov. 2020.

[8] M. K. Hanawal, M. J. Abdel-Rahman, and M. Krunz, "Joint adaptation of frequency hopping and transmission rate for anti-jamming wireless systems," *IEEE Transactions on Mobile Computing*., vol. 15, no. 9, pp. 2247-2259, Sep. 2016.

[9] M. K. Hanawal, M. J. Abdel-Rahman, and M. Krunz, "Game theoretic anti-jamming dynamic frequency hopping and rate adaptation in wireless systems," *12th Internaltional Symposium on Modeling and Optimzation in Mobile, AdHoc, and Wireless Networks (WiOpt), May 2014*.

[10] R. S. Sutton, A. G. Barto, *Reinforement Learning: An Introduction*, 2nd Ed., MIT Press, 2018.

[11] L. Xiao, Y. Li, J. Liu, and Y. Zhao, "Power control with reinforcement learning in cooperative cognitive radio networks against jamming," *The Journal of Supercomputing, vol. 71, 04 2015.*

[12] Y. Gwon, S. Dastangoo, C. Fossa, and H. T. Kung, "Competing mobile network game: embracing antijamming and jamming strategies with reinforcement learning," *EEE Conference on Communications and Network Security (CNS)*, pp. 28-36, 2013.

[13] N. Adem and B. Hamdaoui, "Jamming resiliency and mobility management in cognitive communication networks," *IEEE* International Conference on Communications *(ICC),* pp. 1-6, 2017.

[14] M. A. Aref, S. K. Jayaweera, and S. Machuzak, "Multi-agent reinforcement learning based cognitive anti-jamming," *IEEE Wireless Communications and Networking Conference (WCNC)*, pp. 1-6, 2017.

[15] L. Xiao, D. Jiang, D. Xu, H. Zhu, Y. Zhang, and H. V. Poor, "Two-dimensional antijamming mobile communication based on reinforecement learning*," IEEE Transaction on Vehicular Technology*, vol. 67, no. 10, pp. 9499-9512, Oct. 2018.

[16] Y. Sagduyu, Y. Shi, A. MacKenzie, H. Zhu, T. Hou, "Regret Minimization for primary/secondary access to satellite resources with cognitive interference*," IEEE Transaction on Wireless Communications*, vol. 17, no. 5, pp. 3512-3523, May 2018.