A Systematic Learning Method for Optimal Jamming

SaiDhiraj Amuru[†], Cem Tekin[‡], Mihaela van der Schaar[‡], R. Michael Buehrer[†]
[†]Bradley Department of Electrical and Computer Engineering, Virginia Tech
[‡] Department of Electrical Engineering, UCLA
Email: {adhiraj, rbuehrer}@vt.edu, cmtkn@ucla.edu, mihaela@ee.ucla.edu

***Abstract*—Can an intelligent jammer learn and adapt to unknown environments in an electronic warfare-type scenario? In this paper, we answer this question in the positive, by developing a cognitive jammer that disrupts the communication between a victim transmitter-receiver pair. We formalize the problem using a novel multi-armed bandit framework where the jammer can choose various physical layer parameters such as signaling scheme, power level and the on-off/pulsing duration in an attempt to obtain power efficient jamming strategies. We first present novel online learning algorithms to maximize the jamming efficacy against static transmitter-receiver pairs i.e., the case when the victim does not change its communication technique despite the presence of interference. We prove that our learning algorithm converges to the optimal jamming strategy. Even more importantly, we prove that the rate of convergence to the optimal jamming strategy is sub-linear, i.e. the learning is fast, which is important in dynamically changing wireless environments. Also, we characterize the performance of the proposed bandit-based learning algorithm against adaptive transmitter-receiver pairs.**

I. Introduction

The vulnerabilities of a wireless system can be largely classified based on the capability of an adversary- a) an eavesdropping attack in which the eavesdropper (passive adversary) can listen to the wireless channel and decipher information, b) a jamming attack, in which the jammer (active adversary) can transmit energy in order to disrupt reliable communication and c) hybrid attack, in which the adversary can either passively eavesdrop or actively jam any ongoing transmission. In this paper, we study the effects of jamming attacks against static and adaptive victim transmitter-receiver pairs.

Most of the prior work studies jamming using an optimization or game-theoretic or information-theoretic framework [1]-[5]. The major disadvantage of these studies is that they assume the jammer has a lot of *a priori* information about the strategies used by the (malicious) transmitter-receiver pairs, channel gains, etc., which may not be available in practical scenarios. For instance, in our prior work [3], we showed that the optimal jamming signal follows a pulsed-jamming strategy, and derived the optimal pulse duration given that the jammer knows the transmission strategy of the victim transmitter-receiver pair. In contrast to prior work, in this paper we develop online learning algorithms for the jammer that learns the optimal jamming strategy by repeatedly interacting with the victim transmitter-receiver pair. Essentially, the jammer must learn to act in an unknown environment in order to maximize its total reward (e.g., jamming success rate).

Numerous approaches have been proposed to learn how to act in such unknown communication environments. A canonical example is reinforcement learning (RL) [6]-[11], in which a radio (agent) learns and adapts its transmission strategy using the transmission success feedback of the transmission actions it has used in the past. In general, this feedback is referred to as the *reward*, and over time the agent learns to take actions which yield higher rewards. For instance, the reward can be throughput, the negative of the energy cost, or a function of both these variables. In [6], [7], Q-Learning based algorithms are used to address jamming and anti-jamming strategies against adaptive opponents. It is well-known that such learning algorithms can guarantee optimality only asymptotically (as the number of packet transmissions goes to infinity). However, strategies with only asymptotic guarantees cannot be relied upon in mission-critical/ military applications, where failure to achieve the required performance level in a dynamic setting will have severe consequences. For example, in jamming applications, the jammer needs to learn and adapt its strategy against its opponent in a timely manner. Hence, the rate of learning matters.

In this paper we consider all of the above challenges, and develop novel multi-armed bandit (MAB)-based jamming algorithms that provide time-dependent (not asymptotic) performance bounds on the jamming performance against static and adaptive victim transmitter-receiver pairs. MAB-based algorithms [12]-[14] have been used in the context of wireless communications to address the selection of a wireless channel in either cognitive radio networks [8], [9] or in the presence of an adversary [10], or antenna selection in MIMO systems [11]. To the best of our knowledge, none of these works addressed jamming scenarios and an associated challenging problem of jointly adapting various physical layer parameters such as modulation/signaling scheme, signal power, etc., that can either come from a continuous space or a discrete space.

TABLE I: Comparison between related bandit works

	Finite armed bandits [12]	Continuum armed bandits [13]	Adversarial bandits [14]	Our work
Regret bounds (function of time)	Logarithmic	Sublinear	Sublinear	Sublinear
Action rewards	i.i.d	i.i.d	adversarial (worst-case)	i.i.d
Action set	finite	continuous	finite	mixed
Similarity between rewards	assumed	assumed	assumed	proven (Theorem 1)

The differences between our work and the prior bandit related works are summarized in Table I. We measure the jamming performance of a learning algorithm using the notion of *regret*, which is defined as the difference between the cumulative reward of the optimal jamming strategy when there is complete knowledge about the victim, and the cumulative reward achieved by the learning algorithm. Any algorithm with a sub-linear in time regret, will converge to the optimal strategy in terms of the average reward. Hence, the regret bounds provide a rate on how fast the jammer converges to the optimal strategy without having any *a priori* knowledge about the victim's strategy and the wireless channel.

The rest of the paper is organized as follows. We introduce the system model in Section II. The jamming performance against a static transmitter-receiver pair is considered in Sec-

tion III, where we develop novel learning algorithms for the jammer and present high confidence bounds for the jammers' learning performance. Numerical results are presented in Section IV where we study the behavior of the learning algorithms in both single user and multi-user scenarios and finally conclude the paper in Section V.

II. System Model

We first consider a single jammer and a single victim transmitter-receiver pair in a discrete time setting ($t = 1, 2, \ldots$). We assume that the data conveyed between the transmitter-receiver pair is mapped onto an unknown digital amplitude-phase constellation. The low pass equivalent of this signal is represented as $x(t) = \sum_{m=-\infty}^{\infty} \sqrt{P_x} x_m g(t - mT)$, where P_x is the average received signal power, $g(t)$ is the real valued pulse shape and T is the symbol interval. The random variables x_m denote the modulated symbols assumed to be uniformly distributed among all possible constellation points. Without loss of generality, the average energy of $g(t)$ and modulated symbols $E(|x_m|^2)$ are normalized to unity.

It is assumed that $x(t)$ passes through an AWGN channel (received power is constant over the observation interval) while being attacked by a jamming signal represented as $j(t) = \sum_{m=-\infty}^{\infty} \sqrt{P_J} j_m g(t-mT)$, where P_J is the average jamming signal power as seen at the victim receiver and j_m denote the jamming signals with $E(|j_m|^2) \leq 1$. Assuming a coherent receiver and perfect synchronization, the received signal after matched filtering and sampling at the symbol intervals is given by $y_k = y(t = kT) = \sqrt{P_x} x_k + \sqrt{P_J} j_k + n_k,\ k = 1, 2, ..,$ where n_k is the zero-mean additive white Gaussian noise with variance denoted by σ^2. Let $\mathrm{SNR} = \frac{P_x}{\sigma^2}$ and $\mathrm{JNR} = \frac{P_J}{\sigma^2}$.

III. Jamming against a static transmitter-receiver pair

In this section, we consider the scenario where the victim uses a fixed modulation scheme with a fixed SNR. We propose an online learning algorithm for the jammer which learns the optimal power efficient jamming strategy over time, without knowing the victim's transmission strategy.

A. Set of actions for the jammer

At each time t the jammer chooses its signaling scheme, power level and on-off/pulsing duration. A joint selection of these is also referred to as an action. We assume that the set of signaling schemes has N_{mod} elements, while the set of power levels is $\mathrm{JNR} \in [\mathrm{JNR}_{\min}, \mathrm{JNR}_{\max}]$. The jamming signal $j(t)$ is defined by the signaling scheme and power level selected at time t. It is shown in [3] that the optimal jamming signal does not have a fixed power level, and it should alternate between two different power levels one of which is 0. In other words, the jammer sends the jamming signal j at power level JNR/ρ with probability ρ and at 0 (i.e., no jamming signal is sent) with probability $1 - \rho$. Notice that the pulsed-jamming strategies enable the jammer to create errors in the packet with a low average energy but a high instantaneous energy [3]. Hence, the optimal jamming signal is characterized by the signaling scheme, the average power level and the pulse duration $\rho \in (0, 1]$ which indicates the fraction of time that the jammer is turned on. The jammer should learn these optimal physical layer parameters first by transmitting the jamming signal and then by observing the reward obtained for its actions.

We formulate this learning problem as a *mixed multi-armed bandit* (mixed-MAB) problem. Different from prior work on MAB problems, in a mixed-MAB the action space consists of both finite (signaling set) and continuum (power level, pulse duration) sets of actions. Next, we propose an online learning algorithm called *Jamming Bandits* (JB) where the jammer learns by repeatedly interacting with the transmitter-receiver pair. As mentioned, the jammer receives feedback about its jamming actions which can be in terms of the symbol error rate (SER) or packet error rate (PER) inflicted by the jammer at the victim receiver, throughput allowed [15], among many others. In this paper, we consider the feedback to be in terms of the error rates SER and PER which is inherently a function of the jamming signal $j(t)$. Notice that the jammer can estimate the error rates by only observing the acknowledgment /no acknowledgement (ACK/NACK) packets that are exchanged between the transmitter-receiver pair[1] [16].

B. MAB formulation

The actions (also called the *arms*) of the mixed MAB are defined by the triplet [Signaling scheme, JNR, ρ]. For a given signaling scheme $\mathcal{J}$, the strategy set $\mathcal{S}$ (that constitutes JNR and ρ) is a compact subset of $(\mathbb{R}^+)^2$. For each time $t \in \{1, 2, 3, \ldots, n\}$, a cost function (feedback metric) $C_t : \{\mathcal{J}, \mathcal{S}\} \rightarrow \mathbb{R}$ is evaluated. Since we are interested in finding power efficient mechanisms to maximize the error rate at the victim receiver, we define $C_t = SER_t/\mathrm{JNR}_t$ or PER_t/JNR_t where JNR_t indicates the JNR used by the jammer at time t and SER_t, PER_t are the average symbol/packet error rate obtained by using a particular strategy $\{\mathcal{J}, \mathbf{s} \in \mathcal{S}\}$ at time t. This cost function is unknown to the jammer *a priori* and needs to be learned over time in order to optimize its jamming strategy.

Since the action set is a continuum of arms, it is assumed that the arms that are close to each other (in terms of the Euclidean distance), yield similar expected costs. Such assumptions on the cost function will at least help in learning strategies that are close to the optimal strategy (in terms of the achievable cost function) if not the optimal strategy, especially when we consider learning continuous parameters [13]. Formally, the expected or average cost function $\bar{C}(\mathcal{J}, \mathbf{s}) : \{\mathcal{J}, \mathcal{S}\} \rightarrow \mathbb{R}$ is assumed to be uniformly locally Hölder continuous with constant $L \in [0, \infty)$ and exponent $\alpha \in (0, 1]$. More specifically, the Hölder condition (which is described with respect to the continuous arm parameters) is given by,

$$|\bar{C}(\mathcal{J}, \mathbf{s}) - \bar{C}(\mathcal{J}, \mathbf{s}')| \leq L||\mathbf{s} - \mathbf{s}'||^{\alpha}, \tag{1}$$

for all $\mathbf{s}, \mathbf{s}' \in \mathcal{S}$ with $||\mathbf{s} - \mathbf{s}'|| \leq \delta > 0$ [17] ($||\mathbf{s}||$ denotes the Euclidean norm of the vector $\mathbf{s}$). The best strategy $\mathbf{s}^*$ satisfies $\arg\min_{\mathbf{s} \in \mathcal{S}} \bar{C}(\mathcal{J}, \mathbf{s})$ for a signaling scheme $\mathcal{J}$. We assume that the jammer knows (1) i.e., L and α. The next theorem shows that this similarity assumption holds true when the cost function is SER.

Theorem 1. *SER is uniformly locally Hölder continuous.*

Proof: See the longer version of this paper [18] for the proof and examples that validate this Theorem.

[1] The number of NACKs gives an estimate of the PER. SER can be estimated as $1 - (1 - PER)^{1/N_{sym}}$ where N_{sym} is the number of symbols in one packet.

The result in this theorem is crucial for deriving the regret and high confidence bounds of the proposed learning algorithm. Unlike existing works in MAB, which assume Hölder (or Lipschitz) continuity to derive the regret bounds, the above theorem proves that this condition holds in our setting, i.e., it is not an assumption but rather an intrinsic (proven) feature of our problem.

Corollary 1. *PER and PER/JNR are Hölder continuous.*

C. Proposed Algorithm

The proposed Jamming Bandits (JB) algorithm is shown in Algorithm 1. At each time t, JB forms an estimate $\hat{C}_t$ on the cost function $\bar{C}$, which is an average of the costs observed over the first $t-1$ time slots. However, since some dimensions of the joint action set are continuous, JB discretizes them and then approximately learns the cost function among these discretized versions. For example, ρ is discretized as $\{1/M, 2/M, \ldots, 1\}$ and JNR is discretized as $\mathrm{JNR}_{\min} + (\mathrm{JNR}_{\max} - \mathrm{JNR}_{\min}) * \{1/M, 2/M, \ldots, 1\}$, where M is the *discretization* parameter. Later, we will compute the optimal values of M.

JB divides the entire time horizon n into several rounds with different durations. Within every round (the duration T of each round is also adaptive as shown in Alg. 1), JB uses a different discretization parameter M to create the discretized joint action set, and learns the best jamming strategy over this set, as shown in Fig. 1. The resolution M increases in the number of rounds. Its value given in line 2 of Algorithm 1 is chosen such that the regret is optimized.

Algorithm 1 Jamming Bandits (JB)

T$\leftarrow 1$
1: **while** $T \leq n$ **do**
2: $M \leftarrow \lceil (\sqrt{\frac{T}{\log T}} L 2^{\alpha/2})^{\frac{1}{1+\alpha}} \rceil$
3: Initialize UCB1 algorithm [12] with strategy set $\{\text{AWGN,BPSK,QPSK}\} \times \{1/M, 2/M, \ldots, 1\} \times \mathrm{JNR}_{\min} + (\mathrm{JNR}_{\max} - \mathrm{JNR}_{\min}) * \{1/M, 2/M, \ldots, 1\}$, where $\times$ indicates the Cartesian product.
4: **for** $t = T, T+1, \ldots, \min(2T-1, n)$ **do**
5: Get strategy $\{\mathcal{J}_t, \mathbf{s}_t\}$ from UCB1 [12]
6: Play $\{\mathcal{J}_t, \mathbf{s}_t\}$ and receive the feedback $C_t(\mathcal{J}_t, \mathbf{s}_t)$
7: For each arm in the strategy set, update its index using $C_t(\mathcal{J}_t, \mathbf{s}_t)$.
8: **end for**
9: $T \leftarrow 2T$
10: **end while**

Another advantage of JB is that the jammer does not need to know the time horizon n. Time horizon n is only given as an input to JB to indicate the stopping time. All our results in this paper hold true for any time horizon n. This is achieved by increasing the time duration of the inner loop in JB to $2T$ at the end of every round. The inner loop can use any of the standard finite armed MAB algorithms such as UCB1 [12].

D. Upper bound on the regret

The n-step regret R_n is the expected difference in the total cost between the strategies chosen by the proposed algorithm i.e., $\{\mathcal{J}_1, \mathbf{s}_1\}, \{\mathcal{J}_1, \mathbf{s}_2\}, \ldots, \{\mathcal{J}_n, \mathbf{s}_n\}$ and the best strategy $\{\mathcal{J}^*, \mathbf{s}^*\}$. More specifically, we have $R_n = \mathbf{E}\Big[\sum_{t=1}^{n} C_t(\mathcal{J}^*, \mathbf{s}^*) - C_t(\mathcal{J}_t, \mathbf{s}_t)\Big]$, where the expectation is taken over the random feedback signals.

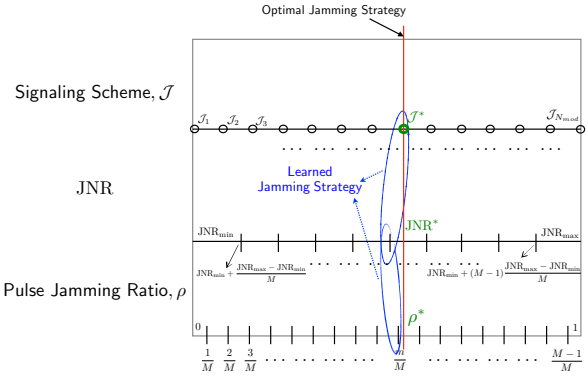

Fig. 1: An illustration of learning in one round of JB.

Theorem 2. *The regret of JB is $\mathcal{O}(N_{mod} n^{\frac{\alpha+2}{2(\alpha+1)}} (\log n)^{\frac{\alpha}{2(\alpha+1)}})$.*

Proof: The proof of the Theorem is based on the Hölder continuity properties of the cost function established in Theorem 1. See [18] for more details.

Remark 1. *The upper bound on regret increases as N_{mod} increases. This is because the jammer now has to spend more time in identifying the optimal jamming signaling scheme. This does not mean that the jammer is doing worse, since as N_{mod} increases, the jamming performance of the* benchmark *against which the regret is calculated also gets better. Hence, the jammer will converge to a better strategy, though it learns slowly. Further, the regret decreases as α increases because higher values of α indicate that it is easier to separate strategies that are close (in Euclidean distance) to each other.*

Corollary 2. *The average cumulative regret of JB converges to 0. Its convergence rate is given as $\mathcal{O}(n^{\frac{-\alpha}{2(\alpha+1)}} (\log n)^{\frac{\alpha}{2(\alpha+1)}})$.*

The average cumulative regret converges to 0 as n increases. These results establish the learning performance i.e., the rate of learning (how fast the regret converges to 0) of JB and indicate the speed at which the jammer learns the optimal jamming strategy using Algorithm 1.

E. High Confidence Bounds

The confidence bounds provide an *a priori* probabilistic guarantee on the desired level of jamming performance (e.g., SER or PER) that can be achieved at a given time.

The sub-optimality gap of the ith arm, denoted by $\{\mathcal{J}^i, \mathbf{s}^i\}$ (recall that $N_{mod}M^2$ arms can be chosen in one round of JB), is defined as $\bar{C}(\mathcal{J}^*, \mathbf{s}^*) - \bar{C}(\mathcal{J}^i, \mathbf{s}^i)$. We say that an arm is sub-optimal if it belongs to the set $U_>$, which is defined in the Appendix. Let $u_i(t)$ denote the total number of times the ith arm has been chosen until time t and $U(T)$ indicate the set of time instants $t \in [1, T]$ for which $u_i(t) \leq \frac{8\log(T)}{\Delta_i^2}$ for some sub-optimal arm $i \in \mathcal{U}_>$ with a sub-optimality gap Δ_i [18].

Theorem 3. *Let $\delta = 2 \times 2^{\frac{3\alpha+2}{2(1+\alpha)}} L^{\frac{1}{1+\alpha}} \left(\frac{\log T}{T}\right)^{\frac{\alpha}{2(1+\alpha)}}$. Then for any $t \in [1, T] \backslash U(T)$, with probability at least $1 - 2(N_{mod} + M^2)t^{-4}$, the expected cost of the chosen jamming strategy $(\mathcal{J}_t, \mathbf{s}_t)$ is at most $\bar{C}(\mathcal{J}^*, \mathbf{s}^*) + \delta$. In other words, $P\left(\bar{C}(\mathcal{J}^*, \mathbf{s}^*) - \bar{C}(\mathcal{J}_t, \mathbf{s}_t) > \delta\right) \leq 2(N_{mod} + M^2)t^{-4}$.*

We also have $E[|U(T)|] \leq \sum_{t=1}^{T} P(a\ sub\text{-}optimal\ arm\ i \in \mathcal{U}_{>}\ is\ chosen\ at\ t) \leq 8\sum_{i\in\mathcal{U}_{>}}\left(\frac{\log T}{\Delta_i^2}\right) + \left(1+\frac{\pi^2}{3}\right)|U_{>}|$, *which means that our confidence bounds hold in all except logarithmically many time slots in expectation. As the number of rounds increases we have* $T \to \infty$, *which implies that*

$$\lim_{T\to\infty}\lim_{t\to T} P\left(\bar{C}(\mathcal{J}^*, \mathbf{s}^*) - \bar{C}(\mathcal{J}_t, \mathbf{s}_t) > \delta\right) = 0.$$

Hence, the one-step regret converges to zero in probability.

Proof: See [18] for the proof and more details on how the jammer can estimate $U(T)$.

To achieve a desired confidence level (e.g., about the SER inflicted at the victim receiver) δ at each time step, the probability of choosing a jamming action that incurs regret more than δ must be very small. In order to achieve this objective, the jammer can set M as $\max\{(\frac{2^{\frac{\alpha+4}{2}}L}{\delta})^{1/\alpha}, \lceil(\sqrt{\frac{T}{\log T}}L2^{\alpha/2})^{\frac{1}{1+\alpha}}\rceil\}$. By doing this, the jammer will not only guarantee a small regret at every time step, but also chooses an arm that is within δ of the optimal arm at every time step with high probability. Hence, the one time step confidence about the jamming performance can be translated into overall jamming confidence. It was however observed that the proposed algorithm performs significantly better than predicted by this bound (Section IV).

Theorem 4. *Let* $\delta = 2\times 2^{\frac{5\alpha+4}{2(1+\alpha)}} L^{\frac{1}{1+\alpha}} \left(\frac{\log T}{T}\right)^{\frac{\alpha}{2(1+\alpha)}}$. *Then, for any* $t \in [1,T]\backslash U(T)$, *the jammer knows that with probability at least* $1-2(N_{mod}+M^2)t^{-4}-t^{-16}$, *the true expected cost of the optimal strategy is at most* $\hat{C}(\mathcal{J}_t, \mathbf{s}_t)+\delta$, *where* $\hat{C}(\mathcal{J}_t, \mathbf{s}_t)$ *is the sample mean estimate of the expected reward of strategy* $(\mathcal{J}_t, \mathbf{s}_t)$ *selected by the jammer at time* t.

Proof: See Appendix. This Theorem presents a high confidence bound on the estimated cost function of any strategy used by the jammer. Such high confidence bounds will enable the jammer to make decisions on the jamming duration and jamming budget, which is explained below with an example.

Remark 2. *Fig. 2 summarizes the importance and usability of Theorems 3 and 4 in real-time environments. The high confidence bounds for the regret help the jammer decide the number of symbols (or packets) to be jammed to disrupt the communication between the transmitter-receiver pair. For example, such confidence is necessary in scenarios where the victim uses erasure or rateless codes and/or HARQ-based transmission schemes. For instance, when* $M = 15$, *we have at large time* t, $\delta > 0.01$, *i.e.,* $P(SER^* - S\hat{E}R_t > 0.01) = 0$, *where* SER^* *is the optimal average* SER *achievable and* $S\hat{E}R_t$ *is the estimated* SER *achieved by the strategy used at time* t. *If the jammer estimates* SER *as* 0.065 *then the best estimate of the* SER^* *indicates that it is less than or equal to* 0.075. *Using such knowledge, the jammer can identify the minimum number of packets it has to jam so as to disrupt the communication and prevent the exchange of a certain number of packets, which in applications such as video transmission can completely break down the system.*

IV. Numerical Results

We first discuss the learning behavior of the jammer against a transmitter-receiver pair that employs a static strategy and later consider the performance against adaptive strategies. To validate the learning performance, we compare the results against the optimal jamming signals that are obtained when the jammer has complete knowledge about the victim [3]. It is assumed that the victim and the jammer send 1 packet with $N_{sym} = 10000$ symbols at any time t. Each time instant is typically of the order of μs (micro seconds) as is usually the case in modern day wireless standards such as LTE. A packet is said to be in error if at least 10% of the symbols are received in error at the victim receiver so as to capture the effect of error correction coding schemes. The minimum and the maximum $\mathrm{SNR}, \mathrm{JNR}$ levels are taken to be $0\ dB$ and $20\ dB$ respectively. The set of signaling schemes for the transmitter-receiver pair is $\{BPSK, QPSK\}$ and for the jammer is $\{AWGN, BPSK, QPSK\}$ [3] i.e., $N_{mod} = 3$.

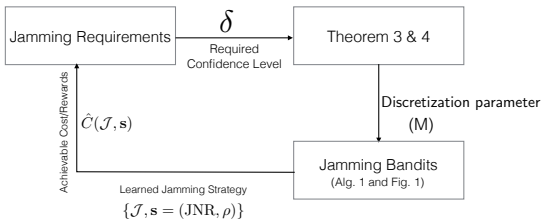

Fig. 2: Using Theorems 3 and 4 in a real time jamming environment.

A. Jamming Performance Against a Static Victim

To enable comparison with [3], we first consider a scenario where the JNR is fixed and the jammer chooses the signaling scheme $\mathcal{J}$ and ρ. Note that unlike [3], the jammer here does not know the signaling parameters of the victim signal, and hence it cannot solve the optimization problems in [3] to find the optimal jamming strategy. In contrast, it learns over time the optimal strategy by simply learning the expected reward of each strategy it tries.

For a fair comparison with [3], we initially assume that the jammer can estimate the SER as seen at the victim receiver. We will shortly discuss the more practical setting in which the jammer can only estimate PER. Fig. 3 shows the average SER attained by JB as a function of time. This figure also shows the performance of ϵ-greedy learning algorithm with exponentially decreasing exploration probability $\epsilon^{\frac{t}{10}}$ (to allow high exploration, the initial exploration probability ϵ is set to 0.9) and resolution factors $M = 5, 10, 20$ (arbitrarily chosen since the optimal value is not known *a priori*). The performance of ϵ-greedy algorithm highly depends on M, and it can be suboptimal if M is chosen incorrectly. However, in our learning setting it is not possible to know the optimal M *a priori*. Also, the performance of AWGN jamming (which is the most widely used jamming signal when the jammer is not intelligent) is significantly lower than the performance of JB.

Fig. 4 shows the learning performance in terms of the average PER (by observing the ACKs/NACKs) inflicted by the jammer at the victim receiver. While the jammer learns to use BPSK as the optimal signaling scheme, the optimal ρ value learned in this case is 0.23 which is different from the value of ρ learned in Fig. 3. This is because PER is used as the cost function in learning the jamming strategies. It is clear that both the AWGN jamming and ϵ-greedy learning algorithm (that uses a suboptimal value of M) achieve a $PER = 0$ based on the SER results in Fig. 3. Even in this case, JB outperforms

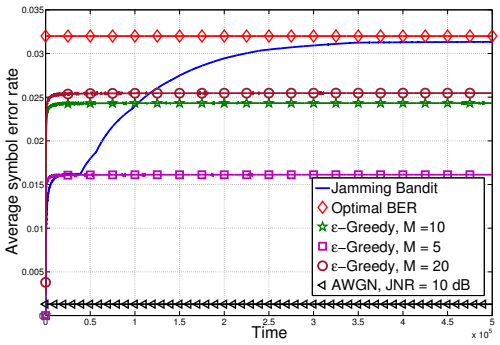

Fig. 3: Average SER achieved by the jammer when JNR $= 10dB$, SNR $= 20dB$ and the victim uses BPSK. The jammer learns to use BPSK with $\rho = 0.078$ using JB. The learning performance of the ϵ-greedy learning algorithm with various resolution factors is also shown.

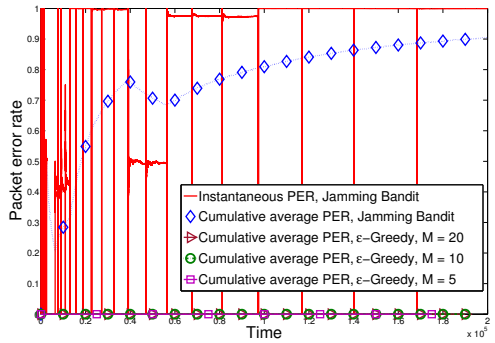

Fig. 4: Average PER inflicted by the jammer at the victim receiver, SNR $= 20$ dB, victim uses BPSK and JNR $= 10$ dB. The jammer learns to use BPSK signaling scheme with $\rho = 0.23$.

traditional jamming techniques that use AWGN or the ϵ-greedy learning algorithm.

Fig. 5 shows the confidence levels as predicted by the one-step regret bound in Theorem 3 and that is achieved by JB. The cost function is taken as $\max(0, (PER - 0.8)/\text{JNR})$ (it is Hölder continuous and is bounded in $[0, 1]$) to ensure that the jammer only chooses strategies which achieve at least 80% PER (achieving a target PER is a common requirement). The optimal reward is estimated by performing an extensive grid search $(M = 100)$ over the entire strategy set. The steps in $\log\delta$ seen in Fig. 5 are due to change in M as shown in

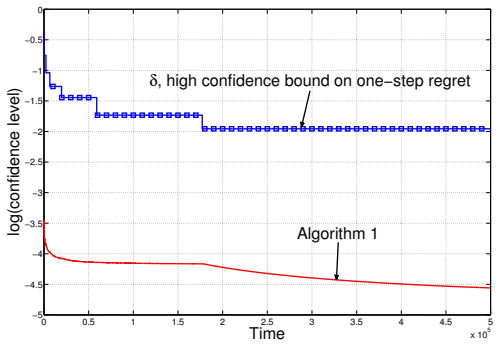

Fig. 5: Confidence level (optimal reward-achieved reward) predicted by Theorem 3 and that achieved by Jamming Bandits.

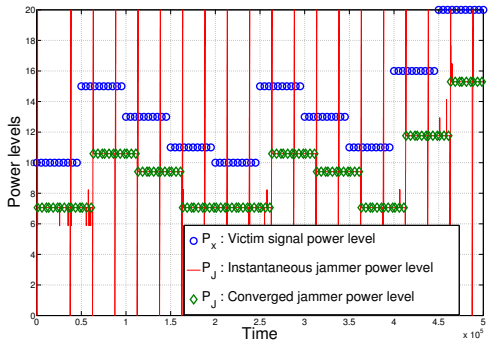

Fig. 6: Learning against a victim with stochastic strategies. The figure shows the power levels adaptation by the jammer using a drifting algorithm and that of the victim.

Algorithm 1. As mentioned before, the algorithm performs much better than predicted by Theorem 3.

B. Jamming Performance Against an Adaptive Victim

When the victim changes its strategy rapidly, JB cannot track the changes perfectly because it learns over all past information, and prior information may not convey knowledge about the current strategy used by the victim which can be completely different from the prior strategy. In such cases, it is important to learn only from recent past history, which can be achieved by using JB on a recent window of past history (for instance, a sliding window-based algorithm to track changes in the environment) [19]. Specifically, we consider the sliding-window method proposed in [19] to run multiple instances of JB with a window length 25000. For this modified version of JB Fig. 6 shows the jammers' power level adaption when the victim is varying its power levels across time. The dips seen at regular intervals in Fig. 6 are due to the proposed sliding window-based algorithm where the user resets the algorithm at regular intervals to adapt to the changing wireless environment. The PER achieved by this algorithm is similar to the results shown in Figs. 4, 5 in comparison to other jamming techniques. These results successfully illustrate the adaptive capabilities of the proposed learning algorithms and also their universal applicability across various jamming scenarios.

C. Multiple Users

In this subsection, we consider a case when the jammer uses an omnidirectional antenna and intends to jam two transmitter-receiver pairs (users) in a network. Similar to the previous subsection, we assume that the users are adaptive. The jammer considers the mean PER seen at both these users as feedback to gauge the performance of its jamming actions (it is assumed that the jammer can differentiate between the two users' ACK/NACK packets). Fig. 7 shows the performance of the JB algorithm against the two users that are randomly changing their power levels to overcome interference (this captures a much more difficult scenario as compared to standard adaptive mechanisms in which the user increases its power level until it reaches a maximum so as to overcome interference). Although each user has a different adaption cycle (specifically, user 1 changes its power levels based on the performance history over the past 50000 time instants and user 2 adapts its power levels over a window of size 30000 time

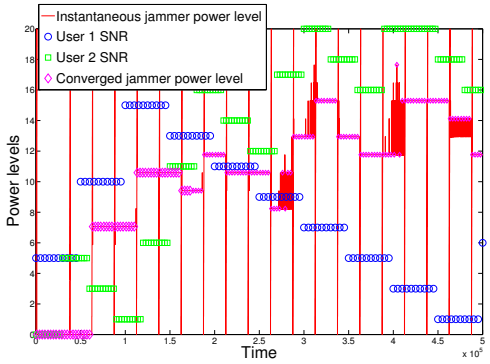

Fig. 7: PER achieved by the jammer against 2 stochastic users in the network. Both the users use BPSK signaling. The jammer learns to use BPSK to achieve power efficient jamming strategies and also tracks the changes in the users' strategies.

instants), the jammer is capable of tracking these changes in a satisfactory manner. Further, by using a weighted PER metric rather than a mean PER metric, the jammer can prioritize jamming one victim against the others.

V. Conclusion

In this paper, we studied whether or not a cognitive jammer can learn the optimal physical layer jamming strategy in an electronic warfare-type scenario without having any *a priori* knowledge about the system dynamics. Novel learning algorithms based on the multi-armed bandit framework were proposed to optimally jam malicious transmitter-receiver pairs. The learning algorithms were capable of learning the optimal jamming strategies that were known from previous works and were also capable of tracking the different strategies used by adaptive transmitter-receiver pairs. Moreover, they come with strong theoretical guarantees on the performance including confidence bounds which are used to estimate the probability of successful jamming at a particular time instant.

References

[1] M. Azizoglu, "Convexity properties in binary detection problems," *IEEE Trans. Inf. Theory*, vol. 42, pp. 1316-1321, Jul. 1996.

[2] S. Bayram *et al.* "Optimum power allocation for average power constrained jammers in the presence of non-Gaussian noise," *IEEE Commun. Lett.,* vol. 16, no. 8, pp. 1153-1156, Aug. 2012.

[3] S. Amuru and R. M. Buehrer, "Optimal jamming strategies in digital communications - impact of modulation," in *Proc. GLOBECOM*, Austin, TX, Dec. 2014.

[4] K. Dabcevic *et al.* "A fictitious play-based game-theoretical approach to alleviating jamming attacks for cognitive radios", in *Proc. ICASSP*, Florence, Italy, May 2014.

[5] S. Shamai (Shitz) and S. Verdú, "Worst-case power constrained noise for binary-input channels," *IEEE Trans. Inf. Theory,* vol. IT-38, no. 5, pp. 1494-1511, Sep. 1992.

[6] B. Wang, Y. Wu and K. J. R. Liu, "An anti-jamming stochastic game in cognitive radio networks," *IEEE J. Sel. Areas Commun.,* vol. 29, no. 4, pp. 877-889, Apr. 2011.

[7] Y. L. Gwon *et al.,* "Competing Mobile Network Game: Embracing antijamming and jamming strategies with reinforcement learning," in *Proc. CNS,* Washington, D.C., Oct. 2013, pp. 28-36.

[8] C. Tekin and M. Liu, "Online learning in opportunistic spectrum access: a restless bandit approach," in *Proc. INFOCOM*, Shanghai, China, Apr. 2011, pp. 2462-2470.

[9] Y. Gai, B. Krishnamachari and R. Jain, "Learning multiuser channel allocations in cognitive radio networks: a combinatorial multi-armed bandit formulation," in *Proc. DYSPAN*, Singapore, April 2010.

[10] Q. Wang, P. Xu, K. Ren, and X.-Y. Li, "Towards optimal adaptive ufh-based anti-jamming wireless communication," *IEEE J. Sel. Areas Commun.*, vol. 30, no. 1, pp. 16-30, Jan. 2012.

[11] N. Gulati and K. R. Dandekar, "Learning state selection for reconfigurable antennas: a multi-armed bandit approach", *IEEE Trans. Antenna Propag.,* vol. 62, no. 3, pp. 1027-1038, Mar. 2014.

[12] P. Auer *et al.,* "Finite-time analysis of the multiarmed bandit problem," *Machine Learning*, vol. 47, no. 2-3, pp. 235–256, 2002.

[13] R. Kleinberg, "Nearly tight bounds for the continuum-armed bandit problem," in *Proc. NIPS*, 2004.

[14] P. Auer, N. Cesa-Bianchi, Y. Freund, and R. E. Schapire, "The non-stochastic multiarmed bandit problem", *SIAM J. Comput.,* vol. 32, no. 1, pp. 48-77, 2002.

[15] S. Amuru and R. M. Buehrer, "Optimal jamming using delayed learning," in *Proc. Military Commun. Conf.,* Baltimore, MD, Oct. 2014.

[16] E. Bayraktaroglu *et al.,* "On the performance of IEEE 802.11 under jamming," in *Proc. INFOCOM,* Phoenix, AZ, Apr. 2008, pp. 1265-1273.

[17] J-Y. Audibert, *et al.* "Exploration-exploitation trade-off using variance estimates in multi-armed bandits," *Theor. Comput. Sci.,* vol. 410, no. 19, pp. 1876-1902, Apr. 2009.

[18] S. Amuru, *et al.,* "Jamming bandits," in arXiv preprint arXiv:1411.3652.

[19] C. Tekin , L. Canzian, and M. van der Schaar, "Context adaptive big data stream mining", in *Proc. Allerton,* Monticello, USA, Oct. 2014.

Appendix

$U_> \subseteq \mathcal{J}_> \cup \mathcal{S}_>$, where $\mathcal{J}_>$ is the set of signaling schemes $\{\mathcal{J}^k\}_{k=1}^{N_{mod}}$ with sub-optimality gap $\Delta_k^{\mathcal{J}} = \bar{C}(\mathcal{J}^*, \mathbf{s}^*) - \bar{C}(\mathcal{J}^k, \mathbf{s}^*) > \delta/4$ and $\mathcal{S}_>$ is the set of $\{\text{JNR}, \rho\}$ with sub-optimality gap $\Delta_k^{\mathcal{S}} = \bar{C}(\mathcal{J}, \mathbf{s}') - \bar{C}(\mathcal{J}, \mathbf{s}^k) > \delta/4 \ \forall k \in [1, M^2]$ and any signaling scheme $\mathcal{J}$. Here $\mathbf{s}'$ is the closest discretized strategy to $\mathbf{s}^*$ among M^2 strategies. See [18] for more details.

Proof of Theorem 4:

A high confidence bound on the mean estimate of the reward/cost function for any strategy that is used at time t by the jammer is presented. To do so, we evaluate $P(\bar{C}(\mathcal{J}^*, \mathbf{s}^*) - \hat{C}(\mathcal{J}_t, \mathbf{s}_t) > \delta)$ as follows,

$$P(\bar{C}(\mathcal{J}^*, \mathbf{s}^*) - \hat{C}(\mathcal{J}_t, \mathbf{s}_t) > \delta) \leq P(\bar{C}(\mathcal{J}^*, \mathbf{s}^*) - \bar{C}(\mathcal{J}_t, \mathbf{s}_t) > \frac{\delta}{2}) + P(\bar{C}(\mathcal{J}_t, \mathbf{s}_t) - \hat{C}(\mathcal{J}_t, \mathbf{s}_t) > \frac{\delta}{2}), \quad (2)$$

where $\bar{C}(\mathcal{J}_t, \mathbf{s}_t)$ is the actual mean reward/cost of the strategy $(\mathcal{J}_t, \mathbf{s}_t)$. The first term can be bounded using Theorem 3 where it can be shown to be less than $2(N_{mod} + M^2)t^{-4}$ for all $\delta > 2^{\frac{5\alpha+4}{2(1+\alpha)}} L^{\frac{1}{1+\alpha}} \left(\frac{\log T}{T}\right)^{\frac{\alpha}{2(1+\alpha)}}$. For the second term, notice that we are comparing the actual and estimated mean rewards of the strategy $(\mathcal{J}_t, \mathbf{s}_t)$ which can be bounded using the Chernoff-Hoeffding bound and the properties of the UCB1 algorithm [12] as follows,

$$P(\bar{C}(\mathcal{J}_t, \mathbf{s}_t) - \hat{C}(\mathcal{J}_t, \mathbf{s}_t) > \frac{\delta}{2}) \leq \exp(-\frac{u_t \delta^2}{2}), \quad (3)$$

where u_t is the total number of times the strategy $(\mathcal{J}_t, \mathbf{s}_t)$ has been used until time t. Since we use the UCB1 algorithm within JB, when each arm is chosen atleast $\frac{8\log t}{\Delta_t^2}$ number of times until time t (where $\Delta_t = \bar{C}(\mathcal{J}^*, \mathbf{s}^*) - \bar{C}(\mathcal{J}_t, \mathbf{s}_t)$ is the regret incurred by the strategy $(\mathcal{J}_t, \mathbf{s}_t)$) the probability of choosing a suboptimal arm $< 2t^{-4}$ (see [12]). By using the bound on the first term in (2) which is established in Theorem 3, we have that $\Delta_t \leq \delta/2$ with high probability. Thus, we have for the second term that $P(\bar{C}(\mathcal{J}_t, \mathbf{s}_t) - \hat{C}(\mathcal{J}_t, \mathbf{s}_t) > \frac{\delta}{2}) \leq \exp(-16\log t) = t^{-16}$ which converges to 0 as t increases.