Jamming Bandits—A Novel Learning Method for Optimal Jamming

SaiDhiraj Amuru, *Member, IEEE*, Cem Tekin, *Member, IEEE*, Mihaela van der Schaar, *Fellow, IEEE*, and R. Michael Buehrer, *Senior Member, IEEE*

***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 adaptively and optimally disrupts the communication between a victim transmitter–receiver pair. We formalize the problem using a multiarmed bandit framework where the jammer can choose various physical layer parameters such as the signaling scheme, power level and the on-off/pulsing duration in an attempt to obtain power efficient jamming strategies. We first present online learning algorithms to maximize the jamming efficacy against static transmitter–receiver pairs and prove that these algorithms converge to the optimal (in terms of the error rate inflicted at the victim and the energy used) jamming strategy. Even more importantly, we prove that the rate of convergence to the optimal jamming strategy is sublinear, i.e., the learning is fast in comparison to existing reinforcement learning algorithms, which is particularly important in dynamically changing wireless environments. Also, we characterize the performance of the proposed bandit-based learning algorithm against multiple static and adaptive transmitter–receiver pairs.**

***Index Terms*—Jamming, optimal, learning, multiarmed bandits, regret, convergence.**

I. Introduction

THE INHERENT openness of the wireless medium makes it susceptible to adversarial attacks. 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 try to infer information (which if leaked may severely compromise data integrity) [2], [3], b) a jamming attack, in which the jammer (active adversary) can transmit energy or information in order to disrupt reliable data transmission or reception [5]–[7] and c) a hybrid attack in which the adversary can either passively eavesdrop or actively jam any ongoing transmission [8], [9]. In this paper, we study the ability of an agent to learn efficient jamming attacks against static and adaptive victim transmitter-receiver pairs.

Jamming has traditionally been studied by using either optimization or game-theoretic or information theoretic principles, see [10]–[17] and references therein. 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 (victim) transmitter-receiver pairs, channel gains, etc., which may not be available in practical scenarios. For instance, in our prior work [12], we showed that it is not always optimal (in terms of the error rate) to match the jammer's signal to the victim's signaling scheme and that the optimal jamming signal follows a pulsed-jamming strategy. However, these optimal jamming strategies were obtained by assuming that the jammer has *a priori* knowledge regarding the transmission strategy of the victim transmitter-receiver pair. In contrast to prior work (both ours and others), in this paper we develop online learning algorithms that learn 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 unknown communication environments. A canonical example is reinforcement learning (RL) [18]–[27], 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. Specifically, it learns the optimal strategy by repeatedly interacting with the environment (for example, the wireless channel). During these interactions, the agent receives feedback indicating whether the actions performed were good or bad. The performance of the action taken is measured as a reward or cost, whose meaning and value depends on the specific application under consideration. For instance, the reward can be throughput, the negative of the energy cost, or a function of both these variables. In [20]–[22], Q-Learning based algorithms were proposed to address jamming and anti-jamming strategies against adaptive opponents in multi-channel scenarios. It is well-known that such learning algorithms can guarantee optimality only asymptotically, for example as the number of packet transmissions goes to infinity. However, strategies with only asymptotic guarantees cannot be relied upon in mission-critical applications, where failure to achieve the required performance level 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.

Manuscript received April 20, 2015; revised November 24, 2015; accepted December 15, 2015. Date of publication December 22, 2015; date of current version April 7, 2016. The work of M. van der Schaar was supported by the NSF under Grant CCF 1524417. Parts of this paper were presented at the International Conference on Communications, London, U.K., June 2015 [1]. The associate editor coordinating the review of this manuscript and approving it for publication was E. Koksal.

S. Amuru and R. M. Buehrer are with the Wireless@VT, Bradley Department of Electrical and Computer Engineering, Virginia Tech, Blacksburg, VA 24061 USA (e-mail: adhiraj@vt.edu; rbuehrer@vt.edu).

C. Tekin is with Bilkent University, Ankara, Turkey (e-mail: cemtekin@ee.bilkent.edu.tr).

M. van der Schaar is with the University of California, Los Angeles, CA 90095-1594 USA (e-mail: mihaela@ee.ucla.edu).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TWC.2015.2510643

TABLE I
COMPARISON BETWEEN RELATED BANDIT WORKS

	Finite armed [28]	Continuous armed [30], [31], [32]	Continuous armed [33]	Contextual [34]	Adversarial [29]	Our work
Regret bounds (function of time)	Logarithmic	Sublinear	Sublinear	Sublinear	Sublinear	Sublinear
Action rewards	i.i.d.	i.i.d.	i.i.d.	i.i.d.	adversarial (worst-case)	i.i.d.
Context space	n/a	n/a	n/a	continuous	n/a	n/a
Similarity (Dis-similarity) metric	n/a	Lipschitz	Hölder	Lipschitz	n/a	Hölder
Action set	finite	continuous	continuous	finite	finite	mixed

As discussed above, none of the previous works considered the learning performance of physical layer jamming strategies in electronic warfare environments where the jammer has limited to no knowledge about the victim transmitter-receiver pair. While several learning algorithms have been proposed in the literature [20]–[32], they are not directly applicable to the jamming problem studied in this paper due to the facts that a) algorithms such as Q-learning [20] do not give any performance guarantees for the jammer's actions, b) the assumptions made in the algorithms (e.g., metric space assumptions in [31], [33]) are not satisfied by the jamming problem studied in this paper, and c) are computationally complex (e.g. tree traversal-based algorithms in [31], [33]). Since we intend to propose jamming algorithms that are practically feasible and can be used in a real time setting, in this paper we make first attempts in developing practically feasible learning algorithms based on the multi-armed bandit (MAB) framework that enable the jammer to learn the optimal physical layer jamming strategies, that were obtained in [12], when the jammer has limited knowledge about the victim.

Although MAB algorithms have been used in the context of wireless communications to address the selection of a wireless channel in either cognitive radio networks [23]–[25] or in the presence of an adversary [26], or antenna selection in MIMO systems [27], these works only consider learning over a finite action set. In contrast, the proposed jamming algorithms in this paper enable the jammer to learn the optimal attack strategies against both static and adaptive victim transmitter-receiver pairs by simultaneously choosing actions from both finite and infinite arm sets (i.e., they can either come from a continuous or a discrete space), that are defined based on the physical layer parameters of the jamming signal. In addition, our algorithms also provide time-dependent (not asymptotic) performance bounds on the jamming performance against static and adaptive victim transmitter-receiver pairs.

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 (for example, a strategy that minimizes the throughput of the victim while using minimum energy) jamming strategy when there is complete knowledge about the victim transmitter-receiver pair, and the cumulative reward achieved by the proposed learning algorithm. Any algorithm with regret that scales sub-linearly in time, will converge to the optimal strategy in terms of the average reward. These regret bounds can also 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. Although the jamming algorithms presented in this paper enable the jammer to learn the optimal attack strategies, we *do not* claim optimality with regards to the regret bounds achieved by the proposed bandit algorithms. We acknowledge the fact that by relying on sophisticated bandit algorithms such as the ones in [31]–[34], the learning rates (regret bounds) may be improved under some regularity conditions. But this analysis is beyond the scope of this paper. The main scope of this work is to present practically feasible learning algorithms that enable the jammer to learn the optimal attack strategies and yet have a reasonable computational complexity and memory requirements. The major differences between our work and the prior work on multi-armed bandit problems (general works that are not related to jamming) are summarized in Table I.

The rest of the paper is organized as follows. We introduce the system model in Section II. The jamming performance against static and adaptive transmitter-receiver pairs is considered in Sections III and IV respectively, where we develop novel learning algorithms for the jammer and present high confidence bounds for its learning performance. Numerical results are presented in Section V where we discuss the learning behavior in both single and multi-user scenarios and finally conclude the paper in Section VI.

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.[1]

[1] Any signal which follows a wireless standard (such as LTE) would have known parameters such as $g(t)$ and T [35].

TABLE II
NOTATIONS USED

Notation	Definition
L, α	Hölder continuity parameters
C_t	Cost function, SER or PER
$\bar{C}$	Average cost function
$\mathfrak{J}$	Set of N_{mod} signaling schemes
$\mathcal{S}$	Set of power level P_J and pulsing ratio ρ
$\mathfrak{J}_t$	Jamming signal at time t
$\mathfrak{J}^*$	Optimal jamming signal
$\mathbf{s}_t$	(P_{J_t}, ρ_t), power and pulsing ratio at time t
$\mathbf{s}^*$	(P_J^*, ρ^*), optimal power and pulsing ratio
$\Delta_i = \bar{C}(\mathfrak{J}^*, \mathbf{s}^*) - \bar{C}(\mathfrak{J}^i, \mathbf{s}^i)$	sub-optimality gap of ith action
M	Discretization parameter for the continuous arms
T	Time period of inner loop in JB - Algorithm 1
n	Total time duration of JB - Algorithm 1

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 $y_k = y(t = kT) = \sqrt{P_x} x_k + \sqrt{P_J} j_k + n_k, \quad k = 1, 2, ..,$ where n_k is the zero-mean additive white Gaussian noise with variance denoted by σ^2. Let SNR $= \frac{P_x}{\sigma^2}$ and JNR $= \frac{P_J}{\sigma^2}$. From [12], the optimal jamming signal shares time between two different power levels one of which is 0 and is hence defined by the on-off/pulsing duration ρ. In other words, the jammer sends the jamming signal $j(t)$ at power level JNR$/\rho$ with probability ρ and at power level 0 (i.e., no jamming signal is sent) with probability $1 - \rho$. For more details on the structure of the jamming signals, please see [12]. While the analysis shown in Sections III and IV assumes coherent reception at the victim receiver (i.e., the jamming signal is coherently received along with the transmitter's signal), we consider the effects of a phase offset between these two signals in Section V. The effects of a timing offset between x and j can also be addressed along similar lines, but is skipped in this paper due to a lack of space. A list of notations used is shown in Table II.

III. Jamming Against a Static Transmitter-Receiver Pair

In this section, we consider scenarios 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 and the average power level belongs to the set JNR $\in$ [JNR$_{\min}$, JNR$_{\max}$].[2] The jamming signal $j(t)$ is defined by the signaling scheme (for example AWGN, BPSK or QPSK) and power level selected at time t. It is shown in [12] that the optimal jamming signal does not have a fixed power level, but instead 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$/\rho$ with probability ρ and at 0 (i.e., no jamming signal is sent) with probability $1 - \rho$. Notice that such pulsed-jamming strategies enable the jammer to cause errors with a low average energy but a high instantaneous energy [12]. Therefore, 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 transmitting. The jammer should learn these optimal physical layer parameters by first 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 where 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. The jammer receives feedback about its actions by observing the acknowledgment /no acknowledgement (ACK/NACK) packets that are exchanged between the transmitter-receiver pair [37]. The average number of NACKs gives an estimate of the PER which can be used to estimate the SER as as $1 - (1 - PER)^{1/N_{sym}}$ where N_{sym} is the number of symbols in one packet (other metrics such as throughput or goodput allowed can also be considered [36]). Remember that the SER and PER are functions of the jammer's actions i.e., the signaling scheme, power level and pulse jamming ratio [12] and thereby allow the jammer to learn about its actions.[3]

B. MAB Formulation

The actions (also called the *arms*) of the mixed MAB are defined by the triplet [Signaling scheme, JNR, ρ]. 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 (or objective) function (feedback metric) $C_t : \{\mathfrak{J}, \mathcal{S}\} \rightarrow \mathbb{R}$ is evaluated by the jammer, where $\mathfrak{J}$ indicates the set of signaling schemes. Since we are interested in finding power efficient

[2] Although we use the variable JNR throughout this paper, it is crucial to notice that the proposed algorithms only need the knowledge of the power with which $j(t)$ is transmitted by the jammer and **do not** need to know the power of the jamming signal as seen at the victim receiver (which depends on the wireless channel whose knowledge is not available to the jammer). There is an unknown but consistent mapping between the jammer's transmit power and JNR. The notation JNR is only used to make the exposition of the Theorems and the algorithms in this paper easier.

[3] Depending on the victim's parameters that the jammer can observe, different cost/reward metrics may be used by the jammer. For example, the jammer can use the following metrics; a) total number of transmissions/re-transmissions b) throughput/data rates [36] or c) power levels employed by the victim which usually increase as the error rates increase and decrease otherwise. In other words, when the jammer cannot observe the ACK/NACK packets exchanged between the victim receiver and its transmitter or if the feedback is erroneous, then alternative metrics must be explored for learning the jamming efficacy.

jamming strategies that maximize the error rate at the victim receiver, we define $C_t = \max(SER_t - SER_{target}, 0)/\text{JNR}_t$ or $\max(PER_t - PER_{target}, 0)/\text{JNR}_t$ where JNR_t indicates the average 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 $\{\mathcal{J} \in \boldsymbol{\mathcal{J}}, \mathbf{s} \in \mathcal{S}\}$ at time t and SER_{target}, PER_{target} are the target error rates that should be achieved by the jammer (achieving a target PER is a common constraint in practical wireless systems [35] and this target is defined *a priori*). The dependence of the cost function on the actions taken is unknown to the jammer *a priori* because it is not aware of a) the victim's transmission strategy, b) the power of the signals x and j at the receiver (the probability of error is a function of these parameters as discussed in [12]) and hence needs to be learned over time in order to optimize the jamming strategy. The jammer does this by trying to maximize C_t as it intends to maximize the error rate at the victim receiver using minimum energy.

When the action set is a continuum of arms, most existing MAB works [30] assume 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 [30]. In this paper, for the first time in a wireless communication setting, we prove that this condition indeed holds true i.e., it is not an assumption but rather an intrinsic (proven) feature of our problem and we show how to evaluate the Hölder continuity parameters for these cost functions. Specifically, Theorem 1 shows that this similarity condition indeed holds true when the cost function is SER and extends it to other commonly used cost functions in wireless scenarios. The result in this Theorem is crucial for deriving the regret and high confidence bounds of the proposed learning algorithm.

Formally, the expected or average cost function $\bar{C}(\mathcal{J}, \mathbf{s}) : \{\boldsymbol{\mathcal{J}}, \mathcal{S}\} \to \mathbb{R}$ is shown to be uniformly locally Hölder continuous with constant $L \in [0, \infty)$, exponent $\alpha \in (0, 1]$ and restriction $\delta > 0$. More specifically, the uniformly locally Hölder continuity condition (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 $0 \leq ||\mathbf{s} - \mathbf{s}'|| \leq \delta$ [38] ($||\mathbf{s}||$ denotes the Euclidean norm of the continuous 2×1 action 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 $\mathcal{J}$. As we will shown next, the algorithms proposed in this paper only require the jammer to know a bound on L and α, since it is not always possible to be aware of the cost function (and its dependence on the actions taken) *a priori*.

Theorem 1: For any set of strategies used by the victim and the jammer, the resultant SER is uniformly locally Hölder continuous.

Proof: See Appendix A. In an online setting, the Hölder continuity parameters L and α can be estimated if the jammer has knowledge about the victim's transmission strategy, else a bound on L and α works.

We now give an illustrative example for Theorem 1. Consider the scenario where both the jammer and the victim use BPSK modulated signals. The average SER (first we show for the case when $\rho = 1$ which will be used to prove the result for $\rho \in (0, 1]$) is given by [12]

$$p_e(\text{SNR}, \text{JNR}) = \frac{1}{4}\left[erfc\left(\frac{\sqrt{\text{SNR}} + \sqrt{\text{JNR}}}{\sqrt{2}}\right) + erfc\left(\frac{\sqrt{\text{SNR}} - \sqrt{\text{JNR}}}{\sqrt{2}}\right)\right], \tag{2}$$

where $erfc$ is the complementary error function. To show the Hölder continuity of the above expression, consider JNR_1 and JNR_2 such that $|\text{JNR}_1 - \text{JNR}_2| \leq \delta$, for some $\delta > 0$ (i.e., to consider the case of local Hölder continuity). Then by using the Taylor series expansion of the $erfc$ function and ignoring the higher order terms i.e., $erfc(x) \approx 1 - \frac{2}{\sqrt{\pi}}x + \frac{2}{3\sqrt{\pi}}x^3$, we have

$$\begin{aligned} p_e(\text{SNR}, \text{JNR}_1) - p_e(\text{SNR}, \text{JNR}_2) &\approx \sqrt{\frac{\text{SNR}}{8\pi}}(\text{JNR}_1 - \text{JNR}_2) \\ &\leq \sqrt{\frac{\text{SNR}_{\max}}{8\pi}}(\text{JNR}_1 - \text{JNR}_2), \end{aligned} \tag{3}$$

where $\text{SNR}_{\max}$ relates to the maximum received power level of the victim signal (practical wireless communication devices have limitations on the maximum power levels that can be used). This shows that SER satisfies the Hölder continuity property when $\rho = 1$.

For the case of a pulsed jamming signal i.e., $\rho \in (0, 1]$, the SER is given by $\rho p_e(\text{SNR}, \text{JNR}/\rho) + (1 - \rho)p_e(\text{SNR}, 0)$. The second term is obviously Hölder continuous with respect to the strategy vector $\mathbf{s} = \{\text{JNR}, \rho\}$ for $L_1 = 1, \alpha_1 = 1$. For the first term, consider the probability of error at the strategies $\mathbf{s}_1 = \{\text{JNR}_1, \rho_1\}$ and $\mathbf{s}_2 = \{\text{JNR}_2, \rho_2\}$. To prove the Hölder continuity, we consider the expression $\rho_1 p_e(\text{SNR}, \text{JNR}_1/\rho_1) - \rho_2 p_e(\text{SNR}, \text{JNR}_2/\rho_2) = \left\{\rho_1 p_e\left(\text{SNR}, \frac{\text{JNR}_1}{\rho_1}\right) - \rho_1 p_e\left(\text{SNR}, \frac{\text{JNR}_2}{\rho_1}\right)\right\} + \left\{\rho_1 p_e\left(\text{SNR}, \frac{\text{JNR}_2}{\rho_1}\right) - \rho_2 p_e\left(\text{SNR}, \frac{\text{JNR}_2}{\rho_2}\right)\right\}$. Again, the first term in this expression is Hölder continuous with $L_2 = \sqrt{\frac{\text{SNR}_{\max}}{8\pi}}, \alpha_2 = 1$ which follows from (3). Using the Taylor series for $erfc$ and after some manipulations, the second term in this expression can be written as

$$\begin{aligned} &\rho_1 p_e\left(\text{SNR}, \frac{\text{JNR}_2}{\rho_1}\right) - \rho_2 p_e\left(\text{SNR}, \frac{\text{JNR}_2}{\rho_2}\right) \\ &\leq (\rho_1 - \rho_2)\frac{erfc(\text{SNR})}{2} \\ &\leq \frac{erfc(\text{SNR})}{2}\sqrt{(\text{JNR}_1 - \text{JNR}_2)^2 + (\rho_1 - \rho_2)^2} \\ &\triangleq L_3||\mathbf{s} - \mathbf{s}'||^{\alpha_3}. \end{aligned} \tag{4}$$

Overall, with $L = 3\min(L_1, L_2, L_3)$ and $\alpha = 1$, the SER obtained under pulsed jamming is also Hölder continuous. In general, since the jammer does not know the victim signals' parameters, it is not aware of the exact structure of the SER expression and hence it can use the worst case L and α (across all possible scenarios that may occur in a real time scenario) to account for the Hölder continuity of C_t.

Corollary 1: PER and $\max(PER - PER_{target}, 0)/\text{JNR}$ are Hölder continuous.

Proof: PER can be expressed in terms of the SER. For example, $PER = 1 - (1 - SER)^{N_{sym}}$ when a packet is said to be in error if at least one symbol in the packet is received incorrectly. Since Theorem 1 shows that SER is Hölder continuous, it follows that PER and as a consequence $\max(PER - PER_{target}, 0)/\text{JNR}$ are also Hölder continuous (remember that $\text{JNR} \in [\text{JNR}_{\text{min}}, \text{JNR}_{\text{max}}]$). It is worth noticing that the Hölder continuity parameters L and α depend on the physical layer signaling parameters such as a) the modulation schemes used by the victim and the jammer and b) SNR of the victim signal.

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. Since some dimensions of the joint action set are continuous, and have infinitely many elements, it is not possible to learn the cost function for each of these values, because it will require a certain amount of time to explore each action from these infinite sets, which thereby cannot be completed in finite time. To overcome this, 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 $\text{JNR}_{\text{min}} + (\text{JNR}_{\text{max}} - \text{JNR}_{\text{min}}) * \{1/M, 2/M, \ldots, 1\}$, where M is the *discretization* parameter. The performance of JB will depend on M, hence, we will also compute the optimal value of M in the following sections.

JB, shown in Algorithm 1, divides the entire time horizon n into several rounds with different durations. Within every round (or inner loop, steps $3-8$ of Algorithm 1) of duration T, where T is adaptively changed in the outer loop (steps 1, 2, 9, 10 of Algorithm 1), JB uses a different discretization parameter M to create the discretized joint action set, and learns the best jamming strategy over this set. The operations of JB in one such round is shown in Fig. 1. The discretization M increases with the number of rounds as a function of T. Its value given in line 2 of Algorithm 1 balances the loss incurred due to exploring actions in the discretized set and the loss incurred due to the sub-optimality resulting from the discretization. The various losses incurred and the derivation of the optimal value for M will be explained in detail in Theorem 2. In summary, upon discretization of the continuous arm space, the jammer chooses a) modulation scheme b) power level and c) the pulsing duration by using the UCB1 algorithm shown in Algorithm 2, which is a well known multi-armed bandit algorithm [28]. Therefore, the outer loop of the algorithm adaptively changes the time duration of the inner loop and provides it with the discretization parameter M while the inner loop performs discretization of the arm space and chooses the best arm among these discretized arms by using UCB1.

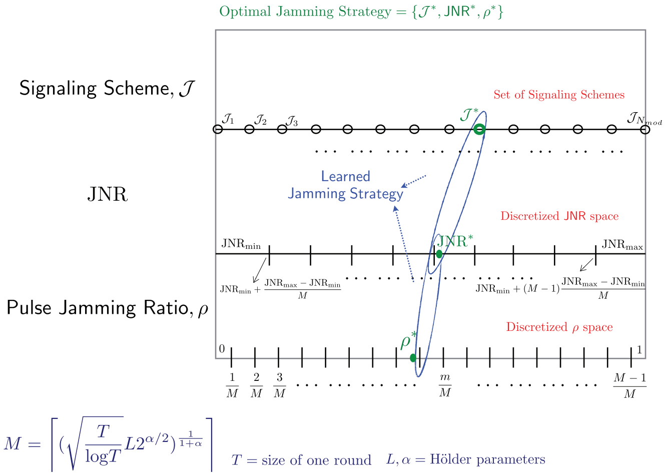
Optimal Jamming Strategy = $\{\mathcal{J}^*, \text{JNR}^*, \rho^*\}$
Signaling Scheme, $\mathcal{J}$
Set of Signaling Schemes
$\mathcal{J}_1$
$\mathcal{J}_2$
$\mathcal{J}_3$
$\mathcal{J}^*$
$\mathcal{J}_{N_{mod}}$
JNR
Learned Jamming Strategy
Discretized JNR space
JNR_{min}
JNR^*
JNR_{max}
$\text{JNR}_{\text{min}} + \frac{\text{JNR}_{\text{max}} - \text{JNR}_{\text{min}}}{M}$
$\text{JNR}_{\text{min}} + (M-1)\frac{\text{JNR}_{\text{max}} - \text{JNR}_{\text{min}}}{M}$
Pulse Jamming Ratio, ρ
Discretized ρ space
0
1
ρ^*
$\frac{1}{M}$
$\frac{2}{M}$
$\frac{3}{M}$
$\frac{m}{M}$
$\frac{M-1}{M}$
$M = \left\lceil (\sqrt{\frac{T}{\log T}} L 2^{\alpha/2})^{\frac{1}{1+\alpha}} \right\rceil$
T = size of one round
L, α = Hölder parameters

Fig. 1. An illustration of learning in one round of JB. It is possible that the optimal strategy denoted by $\{\mathcal{J}^*, \text{JNR}^*, \rho^*\}$ lies out of the set of discretized strategies. In such a case the jammer learns the best discretized strategy, but based on the value of the discretization parameter M, the loss incurred by using this strategy with respect to the optimal strategy can be bounded using the Hölder continuity condition. The value of the discretization M is shown in the figure and Alg. 1.

Algorithm 1. Jamming Bandits (JB)

```
    T←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 [28] with strategy set $\{\text{AWGN}, \text{BPSK}, \text{QPSK}\} \times \{1/M, 2/M, \ldots, 1\} \times \text{JNR}_{\text{min}} + (\text{JNR}_{\text{max}} - \text{JNR}_{\text{min}}) * \{1/M, 2/M, \ldots, 1\}$, where $\times$ indicates the Cartesian product.
 4:     for $t = T, T+1, \ldots, \min(2T-1, n)$ do
 5:         Choose arm $\{\mathcal{J}_t, \mathbf{s}_t\}$ from UCB1 [28]
 6:         Play $\{\mathcal{J}_t, \mathbf{s}_t\}$ and then estimate $C_t(\mathcal{J}_t, \mathbf{s}_t)$ using the ACK/NACK packets
 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
```

Note that JB 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 (popularly known as the doubling trick [30]). The inner loop can use any of the standard finite-armed MAB algorithms such as UCB1 [28], which is shown in Algorithm 2 for completeness.

Remark 1: Although the proposed JB algorithm is similar in spirit to the CAB1 algorithm in [30], the Theorems and the associated proofs in this paper are specific for the scenarios studied in this paper and also specific for the UCB1 algorithm considered in our proposed JB algorithm. CAB1 algorithm only considers a single unknown parameter learning scenario and cases where the cost function is assumed to be Lipschitz. In contrast we consider an algorithm which exploits the Hölder

Algorithm 2. Upper confidence bound-based MAB algorithm - UCB1

```
Initialization: Play each arm once
Loop:
Use signaling scheme $\mathcal{J}$, power JNR, pulse jamming ratio $\rho$, which maximizes $\hat{C}(\mathcal{J}, \underbrace{\mathrm{JNR}, \rho}_{\mathbf{s}}) + \sqrt{\frac{2\log t}{u_{\mathcal{J},\mathbf{s}}}}$ where $t$ is the time duration since the start of the algorithm, $u_{\mathcal{J},\mathbf{s}}$ is the number of times the arm $\{\mathcal{J}, \mathbf{s}\}$ has been played and $\hat{C}(\mathcal{J}, \underbrace{\mathrm{JNR}, \rho}_{\mathbf{s}})$ is the estimated average reward obtained from this arm.
```

(more general than Lipschitz) continuity in the continuous parameter space in addition to learning the discrete parameter. However, when the value of M matches the discretization $\lceil \left(\frac{T}{\log T}\right)^{\frac{1}{2\alpha+1}} \rceil$ proposed in [30], then the jamming performance would be similar as the jammer's action space would be the same.

Remark 2: CKL-UCB is a recently proposed bandit-based algorithm for learning discrete and continuous parameters that satisfy a Lipschitz similarity metric [32]. It was shown to outperform other popular bandit algorithms such as zooming [31] and HOO [33]. However, via simulations (these results are not presented here due to a lack of space), we observed that the CKL-UCB algorithm [31] does not work well for the current problem because a) an problem-dependent optimization problem must be solved in order to evaluate the regret bound, which therefore does not allow the ability to discretize the continuous arm space based on the regret bounds (which is done in our paper in Theorem 2), b) a Lipschitz continuity metric condition is assumed for the case of discrete arms, which does not hold true for the problems studied in our paper as the error rate metric is discontinuous across various modulation schemes [13], and c) these issues force to learn the continuous and the discrete actions separately which degrades the performance of the jammer because the joint impact of the modulation scheme, power and the pulsing ratio account for the optimal jamming strategies.

Remark 3: In [31], [33], optimal MAB algorithms have been proposed to learn in continuous action spaces. However, these algorithms cannot be directly extended to the jamming problem due to a) the mixed action setting considered in this paper, b) the error rate metric considered in this work does not satisfy the proerties of a metric space (due to a lack of space, this proof is skipped in this paper), and c) computational complexity. The mixed action setting forces to consider seperate instantiations of the algorithms in [31], [33] for each discrete action (in this case, the modulation scheme) which therefore significantly increases the computational complexity and the memory requirements of the learning algorithms. Specifically, the computational complexity of the tree-based algorithms in [33] are $\mathcal{O}(N_{mod}n^2)$ in the nth round (or the nth time instant) and the memory requirement is $\mathcal{O}(N_{mod}n)$. For the modified-HOO proposed in [33], the complexity is $\mathcal{O}(N_{mod}n)$ at the same memory requirement. It is mentioned in [33] that the algorithms in [31] have a complexity higher than $\mathcal{O}(N_{mod}n^2)$. However, JB is a practically feasible algorithm that enables the jammer to learn the optimal jamming strategies in real time at a reasonable computational complexity $\mathcal{O}(N_{mod}\frac{n}{\log n}^{\frac{1}{1+\alpha}})$ and memory requirement $\mathcal{O}(N_{mod}\frac{n}{\log n}^{\frac{1}{1+\alpha}})$ at round n (note that this is significantly less compared to the algorithms in [31] and [33]).

D. Upper Bound on the Regret

For the proposed algorithm, 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 $R_n = \mathbf{E}\left[\sum_{t=1}^{n} \left(C_t(\mathcal{J}^*, \mathbf{s}^*) - C_t(\mathcal{J}_t, \mathbf{s}_t)\right)\right]$, where the expectation is over all the possible strategies that can be chosen by the proposed algorithm. Here we present an upper bound on the cumulative regret that is incurred by the jammer when it uses Algorithm 1 to minimize regret or in other words maximize the cost/objective function.

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: See Appendix B.

Remark 4: 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 more 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. Since the proposed algorithms and hence their regret bounds are dependent only on L and α, which are in turn a function of the various signal parameters such as the modulation schemes used by the victim and the jammer, the wireless channel model i.e., AWGN channel, Rayleigh fading channel etc, the proposed algorithms can be extended to a wide variety of wireless scenarios by only changing these parameters. The exact values of L and α need not be known in these cases (because the jammer may not have complete knowledge of the wireless channel conditions), the worst case L and α (as shown in the BPSK example below Theorem 1) can be used in the proposed JB algorithm.

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. We first present the one-step confidence bounds i.e., the instantaneous regret and later show the confidence level obtained on the cumulative regret over n time steps.

The sub-optimality gap Δ_i of the ith arm $\{\mathcal{J}^i, \mathbf{s}^i\}$ (recall that $i \in [1, N_{mod}M^2]$), 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 its sub-optimality gap exceeds a threshold based on the required jamming confidence level. Let $u_i(t)$ denote the total number of times the ith arm, which is sub-optimal, 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 all i in the set of sub-optimal arms denoted by $\mathcal{U}_>$.

Theorem 3: (i) Let $\delta = 2 \times 2^{\frac{3\alpha+2}{2(1+\alpha)}} L^{\frac{1}{1+\alpha}} \left(\frac{\log T}{T}\right)^{\frac{\alpha}{2(1+\alpha)}}$ and M be defined as in Algorithm 1. 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, $P\left(\bar{C}(\mathcal{J}^*, \mathbf{s}^*) - \bar{C}(\mathcal{J}_t, \mathbf{s}_t) > \delta\right) \leq 2(N_{mod} + M^2)t^{-4}$.
(ii) We also have

$$E[|U(T)|] \leq \sum_{t=1}^{T} P(\text{a sub-optimal arm } i \in \mathcal{U}_> \text{ is chosen at } t)$$
$$\leq 8 \sum_{i\in\mathcal{U}_>} \left(\frac{\log T}{\Delta_i^2}\right) + \left(1 + \frac{\pi^2}{3}\right)|\mathcal{U}_>|,$$

which means that our confidence bounds hold in all except logarithmically many time slots in expectation.

Proof: See Appendix C in the longer version of this paper [39] for the proof.

Remark 5: A lower bound on the sub-optimality gap i.e., $\Delta_{\min} = \min_{i\in\mathcal{U}_>} \Delta_i$, can be used to approximately estimate $U(T)$. For instance, in a wireless setting when SER is used as the cost function, if the jammer is aware of the smallest tolerable error in SER that is allowed, then it can approximately evaluate $U(T)$. A detailed discussion on how the jammer can estimate $U(T)$ is given in Appendix C in the longer version of this paper [39].

Remark 6 ***A note on $u_i(t)$:*** Let the victim transmit a BPSK modulated signal with $SNR = 25$ dB. Let $JNR = 20$ dB and $T = 500000$. The jammer intends to learn the optimal jamming scheme and the pulsing ratio. From our previous results, [12], [13], it is known that the maximum symbol error rate achievable when the jammer uses AWGN is 0.053 and when it uses BPSK it is 0.126 and that BPSK is the optimal strategy. Thus in this case $\Delta_{AWGN} = 0.073$ which indicates the sub-optimality gap for AWGN. Therefore, we have that $\frac{8\log T}{\Delta_{AWGN}^2} = 19700$, which in other words indicates that at most 19700 out of $T = 500000$ time slots (approximately 4% of the total time) is necessary to differentiate between the AWGN (sub-optimal) and BPSK (optimal) jamming schemes (remember, a time slot is typically on the order of micro seconds in typical wireless standards). By performing such calculations, the jammer can build confidence on the required number of time slots necessary to learn the optimal jamming strategy.

Corollary 3: The one-step regret converges to zero in probability i.e.,

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

Theorem 3 can be used to achieve desired confidence levels about the jamming performance, which is particularly important in military settings. In order 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 V).

Theorem 4: For any signaling scheme $\mathcal{J}$ chosen by the jammer, $P\left(\sum_{t=1}^{T}(\bar{C}(\mathcal{J}, \mathbf{s}^*) - \bar{C}(\mathcal{J}, \mathbf{s}_t)) > \left(\frac{8}{3\epsilon}\left(\frac{T}{\log T}\right)^{\frac{4}{1+\alpha}}\right)^{1/3}\right)$ $< \epsilon, \forall\, \epsilon > 0$.

Proof: See the longer version of this paper [39] for the proof. Using Theorem 4, a confidence bound on the overall cumulative regret defined as $\sum_{t=1}^{T}[\bar{C}(\mathcal{J}^*, \mathbf{s}^*) - \bar{C}(\mathcal{J}_t, \mathbf{s}_t)]$ can be directly obtained as discussed in [39]. This bound indicates the overall confidence acquired by the jammer. The regret performance of JB will be discussed in more detail via numerical results in Section V.

Theorem 5: Let $\delta = 2 \times 2^{\frac{5\alpha+4}{2(1+\alpha)}} L^{\frac{1}{1+\alpha}} \left(\frac{\log T}{T}\right)^{\frac{\alpha}{2(1+\alpha)}}$ and M be defined as in JB. 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 $\bar{C}(\mathcal{J}_t, \mathbf{s}_t)$, the expected reward of strategy $(\mathcal{J}_t, \mathbf{s}_t)$ selected by the jammer at time t.

Proof: See the longer version of this paper [39] for the proof. Theorem 5 presents a high confidence bound on the estimated cost function of any strategy used by the jammer. Such high confidence bounds (Theorems 3–5) will enable the jammer to make decisions on the jamming duration and jamming budget, which is explained below with an example. Again, this is a worst case bound and the proposed algorithm performs much better than predicted by the bound as will be discussed in detail in Section V.

Remark 7: Fig. 2 summarizes the importance and usability of Theorems 3 and 5 in realtime wireless communication 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 victim 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. In the case of rateless codes, a message of length N is encoded into an

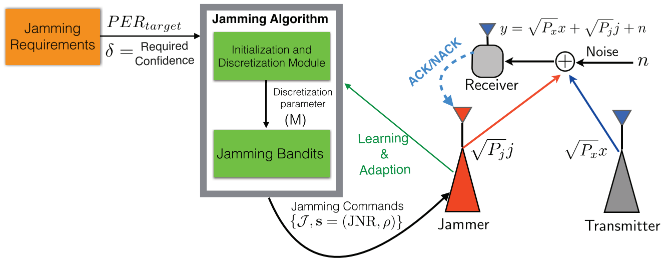
Jamming Requirements
PER_{target}
$\delta =$ Required Confidence
Jamming Algorithm
Initialization and Discretization Module
Discretization parameter (M)
Jamming Bandits
Learning & Adaption
Jamming Commands $\{\mathcal{J}, \mathbf{s} = (\text{JNR}, \rho)\}$
ACK/NACK
$y = \sqrt{P_x}x + \sqrt{P_j}j + n$
Receiver
Noise
n
$\sqrt{P_j}j$
$\sqrt{P_x}x$
Jammer
Transmitter

Fig. 2. Using Theorems 3 and 5 in a real time jamming environment.

infinitely long new message sequence of length $\hat{N} >> N$ (for example, by using random linear combinations) out of which any N are linearly independent. Upon successfully receiving N such messages, the entire message can be recovered. Under such scenarios, the high confidence bounds help the jammer to decide the number of packets/time instants to jam successfully in order to disrupt the wireless link between the transmitter-receiver pair.

For instance, when $M = 15$, we have at large time t, $\delta > 0.01$, i.e., $P(SER^* - \hat{SER}_t > 0.01) = 0$, where SER^* is the optimal average SER achievable and $\hat{SER}_t$ 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). As an example, consider the case when packets of length 100 symbols are exchanged and that a packet is said to be in error only when there are more than 10 errors in the packet. Thus, in order to jam 100 packets successfully the jammer needs to affect at least 463 packets on an average if SER^* (which corresponds to $PER = 0.2167$) was achievable. However, since it can only achieve $\hat{SER} = 0.065$ i.e., $\hat{PER} = 0.1153$, it has to jam at least 865 packets on an average to have sufficient confidence regarding its jamming performance. The jammer can accordingly plan its energy budget/jamming duration etc. by using such knowledge.

F. Improving Convergence via Arm Elimination

When the number of signaling schemes that the jammer can choose from is large or when α is small (i.e., it is difficult to separate the arms that are close to each other), then the learning speed using JB can be relatively slow. We now present an algorithm to improve the learning rate and convergence speed of JB under such scenarios. In order to achieve this, Algorithm 1 is modified to use the UCB-Improved algorithm [41] inside the inner loop of JB instead of UCB1. The UCB-Improved algorithm eliminates sub-optimal arms (that are evaluated in terms of the mean rewards and the confidence intervals), in order to avoid exploring the sub-optimal arms (which is important in electronic warfare scenarios). The modified algorithm and the associated UCB-Improved algorithm are shown in Algorithms 3 and 4 respectively.

Algorithm 3. Jamming Bandits with Arm Elimination

```
T←1
while $T \leq n$ do
    Initialize UCB-Improved [41] algorithm with the strategy set $\{\text{AWGN,BPSK,QPSK}\} \times \{1/M, 2/M, \ldots, 1\} \times \text{JNR}_{\min} + (\text{JNR}_{\max} - \text{JNR}_{\min}) * \{1/M, 2/M, \ldots, 1\}$, where × indicates the Cartesian product.
    for $t = T, T+1, \ldots, \min(2T-1, n)$ do
        Use the UCB-Improved [41] MAB Algorithm to eliminate sub-optimal arms
    end for
    $T \leftarrow 2T$
end while
```

Algorithm 4. UCB-Improved

```
Input the set of arms $A$ and time horizon $T$
$\tilde{\Delta}_0 = 0$, $B_0 = A$
for rounds $m = 0, 1, 2, \ldots, \frac{1}{2}\log_2 \frac{T}{e}$ do
    Arm Selection
    If $|B_m| > 1$, choose each arm in $B_m$ for $n_m = \left\lceil \frac{2\log(T\tilde{\Delta}_m^2)}{\tilde{\Delta}_m^2} \right\rceil$
    Else choose the remaining arm until time $T$
    Arm Elimination
    Delete arm $i$ in the set $B_m$ for which $\left(\bar{C}_i + \sqrt{\frac{\log(T\tilde{\Delta}_m^2))}{2n_m}}\right) < \max_{j \in B_m} \left(\bar{C}_j - \sqrt{\frac{\log(T\tilde{\Delta}_m^2))}{2n_m}}\right)$ to obtain the set of new arms $B_{m+1}$; $\bar{C}_i$ is the average cost incurred by playing arm $i$.
    Reset $\tilde{\Delta}_m$ : $\tilde{\Delta}_{m+1} = \tilde{\Delta}_m/2$.
end for
```

To obtain the value of M i.e., the discretization for JNR and ρ, we used numerical optimization tools to solve $TL\left(\frac{2}{M^2}\right)^{\frac{\alpha}{2}} - \left(\sqrt{M^2T}\frac{\log(M^2\log(M^2))}{\sqrt{\log(M^2)}}\right) = 0$. See the longer version of this paper [39] for more details. Later in Section V, we show the benefits of using this algorithm via numerical simulations. The regret bounds can be derived along similar lines to Theorems 1–5 by using the properties of the UCB-Improved algorithm [41].

IV. Learning Jamming Strategies Against a Time-Varying User

In this section, we consider scenarios where the victim transmitter-receiver pair can choose their strategies in a time-varying manner.[4] We specifically consider two scenarios

[4] The model considered in this formulation is different from the *adversarial scenarios* studied in the context of MAB algorithms [29]. In the adversarial bandit cases, the adversary (or the victim in this current context) observes the action of the jammer and then assigns a reward function either based on the jammers' current action or on the entire history of jammers' actions. However, in the current scenario we assume that the user picks a strategy in an i.i.d manner independent of the jammer. Considering learning algorithms in adversarial scenarios is reserved for future work.

a) when the victim changes its strategies in an i.i.d. fashion and b) when the victim is adapting its transmission strategies to overcome the interference seen in the wireless channel.[5] The worst case jammer's performance can be understood by considering a victim that changes its strategies in an i.i.d. fashion. For example, such i.i.d. strategies are commonly employed in a multichannel wireless system where the victim can randomly hop onto different channels (either in a pre-defined or an un-coordinated fashion [42]) to probabilistically avoid jamming/interference. The randomized strategies chosen by the victim can confuse the jammer regarding its performance. For instance, if the jammer continues using the same strategy irrespective of the victim's strategy, then the jammers' performance will be easily degraded. However, if the jammer is capable of anticipating such random changes by the victim and learns the jamming strategies, then it can disrupt the communication irrespective of the victims' strategies.

We assume that the victim can modify its power levels and the modulation scheme to adapt to the wireless environment (the most widely used adaption strategy [40]). Again we allow the jammer to learn the optimal jamming strategy by optimizing the 3 actions, namely signaling scheme, JNR and ρ as before. The jammer has to learn its actions without any knowledge regarding the victim's strategy set and any possible distribution that the victim may employ to choose from this strategy set. We use Algorithm 1 and not Algorithm 3 to address such dynamic scenarios because eliminating arms in such a time-varying environment may not always be beneficial. For example, a certain arm might not be good against one strategy used by the victim but might be the optimal strategy when the victim changes its strategy.

While the regret bounds presented below assume that the victim employs a random unknown distribution over its strategy set and chooses its actions in an i.i.d. manner (also referred to as stochastic strategies) i.e., scenario (a) mentioned earlier, we discuss the jammer's performance against any strategy (i.e., without any predefined distribution over the strategies, for example, increase the power levels when the PER increases) employed by the victim (which includes scenario (b)) in Section V.

1) Upper Bound on the Regret: Let $\{p_i\}_{i=1}^{|\mathcal{P}|}$ denote the probability distribution with which the victim selects its strategies in an i.i.d manner, from a set consisting of $|\mathcal{P}|$ number of possible strategies. The jammer is not aware of this distribution chosen by the victim and needs to learn the optimal strategy by repeatedly interacting with the victim. The regret under such scenarios is defined as $R_n = \mathbf{E}\left[\sum_{t=1}^{n}\left(C_t(\mathcal{J}^*, \mathbf{s}^*) - C_t(\mathcal{J}_t, \mathbf{s}_t)\right)\right]$, where the expectation is over the random strategies chosen by the jammer as well as the victim (which is different from the formulation in Section III). Thus, the above expression can be re-written as $R_n = \mathbf{E}\left[\sum_{t=1}^{n}\sum_{i=1}^{|\mathcal{P}|} p_i \left(C_t^i(\mathcal{J}^*, \mathbf{s}^*) - C_t^i(\mathcal{J}_t, \mathbf{s}_t)\right)\right]$, with C_t^i indicating the cost function when the victim uses strategy i with probability p_i and the expectation is now taken only over the strategies chosen by the jammer.

Theorem 6: The regret of JB when the victim employs stochastic strategies is $\mathcal{O}(N_{mod} n^{\frac{\alpha+2}{2(\alpha+1)}} (\log n)^{\frac{\alpha}{2(\alpha+1)}})$.

Proof: See the longer version of this paper [39] for the proof. This is an upper bound on the cumulative regret incurred by JB under such stochastic scenarios. Similar to the regret incurred by JB in Theorem 1, the regret under stochastic cases also converges to 0 as $\mathcal{O}(n^{\frac{-\alpha}{2(\alpha+1)}} (\log n)^{\frac{\alpha}{2(\alpha+1)}})$. The one step confidence bounds similar to Theorems 3-5 can be derived even in this case but are skipped due to lack of space.

Remark 8: When the victim is adapting its strategies based on the error rates observed over a given time duration (as is typically done in practical wireless communication systems), we show that by employing sliding-window based algorithms, the jammer can effectively track the changes in the victim and jam it in a power efficient manner. This is discussed more in detail in the next section.

V. 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 [12]. It is assumed that the victim and the jammer send 1 packet with 10000 symbols at any time t. 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 SNR, 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 $\{AWGN, BPSK, QPSK\}$[6] [12] i.e., $N_{mod} = 3$.

A. Fixed User Strategy

The jammer uses SER or PER inflicted at the victim receiver (estimated using the ACK and NACK packets) as feedback to learn the optimal jamming strategy. We first consider a scenario where the JNR is fixed and the jammer can optimize its jamming strategy by choosing the optimal signaling scheme $\mathcal{J}^*$ and the associated pulse jamming ratio ρ^*. These results enable comparison with previously known results obtained via an optimization framework with full knowledge about the victim as discussed in [12]. Note that unlike [12], the jammer here does not know the signaling parameters of the victim signal, and hence it cannot solve an optimization problem 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.

Figs. 3–6 show the results obtained in this setting (fixed SNR, modulation scheme for the victim and fixed JNR). For a fair

[5] While the victim is not entirely adaptive against the jammers' strategies, it is adaptive in the sense that it can choose from a set of strategies to overcome the jamming/interference effects. For example, it can be adaptive based on the PER seen at the victim receiver. This scenario is discussed in detail in Section V.

[6] It is very easy to extend the results in this paper and [12] to PAM and QAM signals of any constellation size.

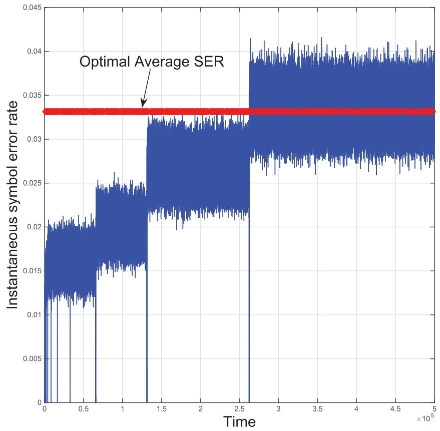

Fig. 3. Instantaneous SER achieved by the JB algorithm when JNR $= 10$ dB, SNR $= 20$ dB and the victim uses BPSK.

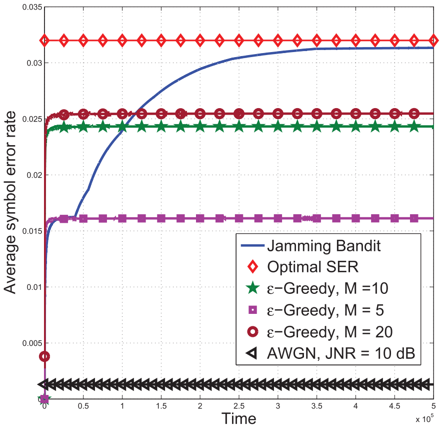

Fig. 4. Average SER achieved by the jammer when JNR $= 10$ dB, SNR $= 20$ dB and the victim uses BPSK. The jammer learns to use BPSK with $\rho = 0.078$ using JB. The learning performance of the ϵ-greedy learning algorithm with various discretization factors M is also shown.

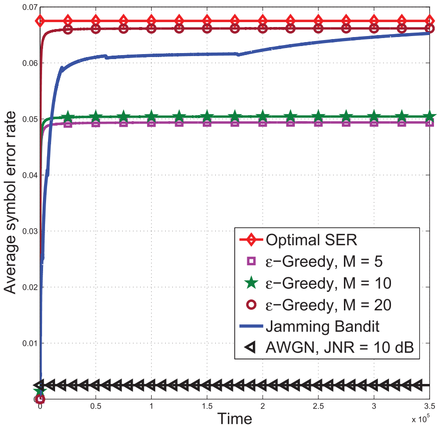

Fig. 5. Learning the optimal jamming strategy when JNR $= 10$ dB, SNR $= 20$ dB and the victim uses QPSK modulation scheme. The jammer learns to use QPSK signaling scheme with $\rho = 0.087$.

comparison with [12], we initially assume that the jammer can directly estimate the SER inflicted at the victim receiver. We will shortly discuss the more practical setting in which the jammer can only estimate PER. In all these figures, it is seen that the jammers' performance converges to that of the optimal jamming strategies [12]. For example, in Figs. 3 and 4, when the victim transmitter-receiver pair exchange a BPSK modulated signal at SNR $= 20$ dB, the jammer learns to use BPSK signaling at JNR $= 10$ dB and $\rho = 0.078$ which is in agreement with the results presented in [12].

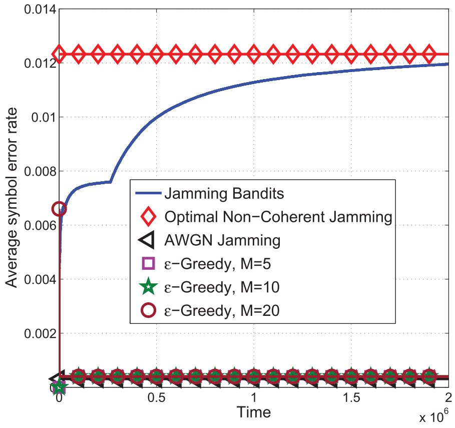

Fig. 6. Average SER achieved by the jammer when JNR $= 10$ dB, SNR $= 20$ dB and the victim uses BPSK and there is a phase offset between the two signals. The jammer learns to use BPSK with $\rho = 0.051$ using JB. The learning performance of the ϵ-greedy learning algorithm with various discretization factors M is also shown.

Fig. 3 shows the instantaneous learning performance of the jammer in terms of the SER achieved by using the JB algorithm. The variation in the achieved SER after convergence is only due to the wireless channel. The time instants at which the SER varies a lot, i.e., the dips in SER seen in these results are due to the exploration phases performed when a new value of discretization i.e., M is chosen by the algorithm (recall from Algorithm 1 that for every round the discretization M is re-evaluated). Fig. 4 shows the average SER attained by this learning algorithm. Also shown in Fig. 4 is the performance of the the ϵ-greedy learning algorithm [28] with exponentially decreasing exploration probability $\epsilon(t) = \epsilon^{\frac{t}{10}}$ (initial exploration probability is taken to be 0.9) and various discretization factors M. In the ϵ-greedy learning algorithm, the jammer explores (i.e., it tries new strategies) with probability $\epsilon(t)$ and exploits (i.e., uses the best known strategy that has been tried thus far) with probability $1 - \epsilon(t)$. It is seen that unless the optimal discretization factor M is known (so that the optimal strategy is one among the possible strategies that can be chosen by the ϵ-Greedy algorithm), the ϵ-greedy algorithm performs significantly worse in comparison to JB.

Similar results were observed in the case of QPSK signaling as seen in Fig. 5. Notice that while the ϵ-greedy algorithm with discretization $M = 20$ did not achieve satisfactory results in the BPSK signaling scenario, it achieved close to optimal results in the QPSK signaling scenario as seen in Fig. 5. Thus, the performance of the ϵ-greedy algorithm highly depends on M, and it can be sub-optimal if M is chosen incorrectly. It is easy to see that if the jammer cannot use QPSK signaling to jam the victim in this scenario, then the jamming performance would be limited as clearly described in [12]. However, in our learning setting it is not possible to know the optimal M *a priori*. Also,

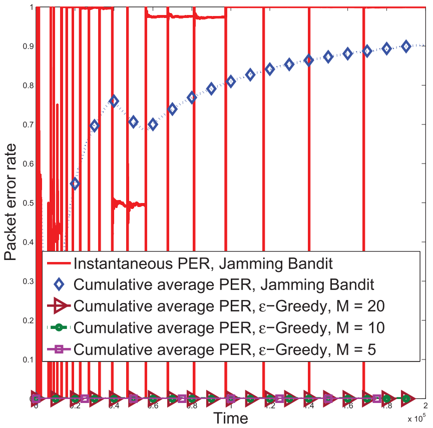

Fig. 7. Average PER inflicted by the jammer at the victim receiver, SNR $=$ 20 dB, victim uses BPSK and JNR $=$ 10 dB. The jammer learns to use BPSK signaling scheme with $\rho = 0.23$.

the performance of AWGN jamming (which is the most widely used jamming signal [16], [40] when the jammer is not intelligent) is significantly lower than the performance of JB. The algorithms behave along similar lines in a non-coherent scenario where there is a random unknown phase offset between the jamming and the victim signals, as seen in Fig. 6. The jammer learned to use BPSK signaling at $\rho = 0.051$ while the optimal jamming signal derived in [12] indicates that $\rho^* = 0.06$ when $JNR = 10$ dB and $SNR = 20$ dB.

Now that we have established the performance of the proposed learning algorithm by comparing with previously known results, we now consider the performance of the learning algorithm in terms of the PER which is a more relevant and practical metric to be considered in wireless environments. Further, it is also easy for the jammer to estimate PER by observing the ACKs/NACKs exchanged between victim receiver and transmitter via the feedback channel [37].[7] Fig. 7 shows the learning performance of various algorithms in terms of the average PER 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. 4. 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 sub-optimal value of M) achieve a $PER = 0$ based on the SER results in Fig. 4. Even in this case, JB outperforms traditional jamming techniques that use AWGN or the ϵ-greedy learning algorithm.

We next consider the cost function as $\max(0, (PER(t) - 0.8)/\text{JNR}(t))$ (the cost function remains to be Hölder continuous and is bounded in [0, 1]) to ensure that we choose only those strategies which achieve at least 80% PER (remember, the jammer intends to maximize this cost/objective function) while concurrently minimizing the energy used. Fig. 8 compares the learning performance of JB with respect to the optimal strategy and Fig. 9 shows the confidence levels as predicted by the

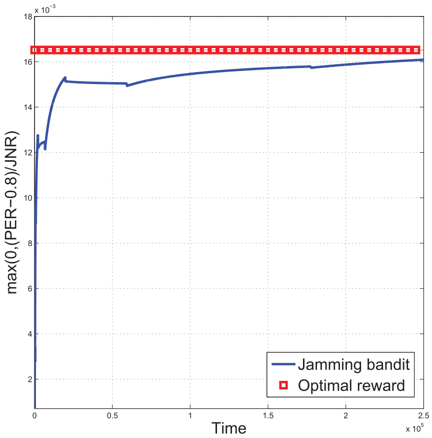

Fig. 8. Average reward obtained by the jammer against a BPSK modulated victim, $SNR = 20$ dB. The optimal reward is obtained via grid search with discretization $M = 100$.

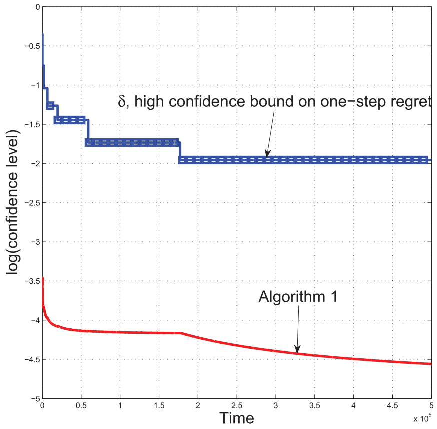

Fig. 9. Confidence level (optimal reward-achieved reward) predicted by Theorem 3 and that achieved by JB.

one-step regret bound in Theorem 3 and that achieved by JB. 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. 9 are due to change in the discretization M as shown in Algorithm 1. As mentioned before, the algorithm performs much better than predicted by the high confidence bound (evidenced by a lower value of δ).

Fig. 10 shows the learning results obtained by using Algorithm 3 i.e., JB uses the UCB-Improved algorithm in the inner loop instead of the UCB1 algorithm. It shows the learning performance of Algorithms 1 and 3 in one inner loop iteration when $T = 10^5$ (i.e., for one value of discretization M evaluated as shown in Algorithm 1). It is seen that the Algorithm 3 converges faster in comparison to the earlier approach as the algorithm eliminates sub-optimal arms and thereby only exploits the best jamming strategy. Even in this case the jammer learned to use BPSK signaling scheme against a BPSK-modulated victim signal. Further notice that the algorithm converges in about 10000 time steps in this case as opposed to > 50000 time steps using JB. Recall that in the simulations we assume that one packet is sent every time instant and hence in order to obtain reliable estimates of the performance of each jamming strategy, the jammer requires about 10000 time instants.

[7] In this paper, we assume that the feedback channel via which the jammer observes the ACK/NACK packets is error free. However, if there are errors in this feedback metric, then the jammer must resort to alternative feedback metrics as described earlier in Section III.

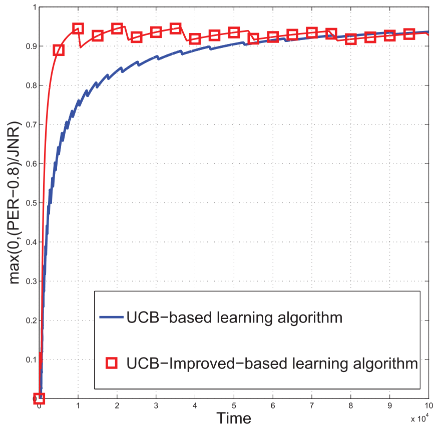

Fig. 10. Learning the jamming strategies by using arm-elimination. The victim uses BPSK with SNR $= 20$ dB. The jammer learned to use BPSK with JNR $= 15$ dB and $\rho = 0.22$.

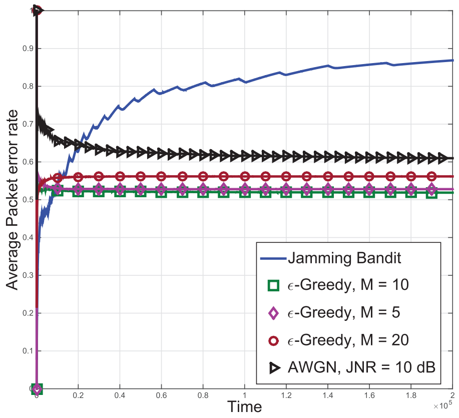

Fig. 11. Learning jammers' strategy against a stochastic user. The victim transmitter-receiver pair use a uniformly random signaling scheme that belongs to the set {BPSK,QPSK} and random power level in the range [0, 20] dB.

B. Jamming Performance Against an Adaptive Victim

We first assume that the victim employs a uniform distribution over its strategy set i.e., it chooses uniformly at random (at every time instant) a power level in the range $[\mathrm{SNR}_{\min}, \mathrm{SNR}_{\max}]$ and the modulation scheme from the set {BPSK,QPSK}. The performance of JB when the victim employs such a stochastic strategy is shown in Fig. 11. Again, the superior performance of the bandit-based learning algorithm when compared to the traditionally used AWGN jamming and naive learning algorithms such as ϵ-Greedy is proved from these results.[8]

When the victim changes its strategy rapidly, JB cannot track the changes perfectly as seen in Fig. 12 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) [43]. Specifically, we use the concept of drifting [43] to adapt to the victim's strategy. In this algorithm, each round i (which is of T time steps, where $T = 2^i$) is divided it into several frames each of W time instants. Within each frame, the first $W/2$ time steps, are termed as the passive slot and the second $W/2$ time instants are termed as the active slot. In the first frame, both the slots will be taken to be active slots. Each passive slot overlaps with the active slot of the previous frame. If time t belongs to active slot of frame w, then actions are taken as per the UCB1 indices evaluated in this particular frame w. However, if it belongs to the passive slot of frame w, which is taken to overlap with the active slot of frame $w - 1$, then it takes actions as per the indices of the frame $w - 1$, but updates the UCB1 indices so that it can be used in frame w. Specifically, at the start of every frame w, the counters and mean reward estimates are all reset to zero and when actions are taken in the passive slot of frame w, these counters and reward estimated are updated so as to be used in the active slot. Thus when the algorithm enters the active slot of frame w, it already has some observations using which it can exploit without wasting time in the exploration phase. Such splitting of the time horizon will enable the jammer to quickly adapt to the victim's varying strategies. Please see [43] for more details on the drifting algorithm. Specifically, we consider the drifting algorithm with a window length $W = 25000$.

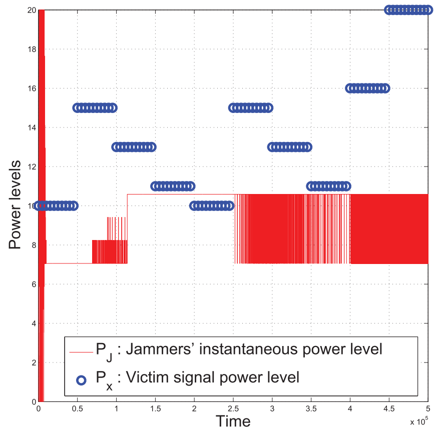

Fig. 12. Learning against a victim with time-varying strategies. The figure shows the power levels adaptation by the jammer and that used by the victim.

Fig. 13 shows the jammers' power level adaption when the victim is randomly varying its power levels across time and the jammer employs the drifting algorithm in conjunction with JB. The dips seen at regular intervals in Fig. 13 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. 7, 9 in comparison to other jamming techniques. While Fig. 13 considered the case when the victim changes its power levels randomly, the jammer can also easily track the victim when it employs commonly used adaptation strategies such as increasing the power levels when PER increases and vice versa. These results successfully illustrate

[8] Model-free learning algorithms such as Q-Learning and SARSA [22] cannot be employed in the scenarios considered in this paper because it is assumed that the jammer cannot observe any of the environment parameters such as the victim's modulation scheme and power levels. However, the performance of the learning algorithms can be improved when such additional information is available, which is typically the case in optimization-based algorithms.

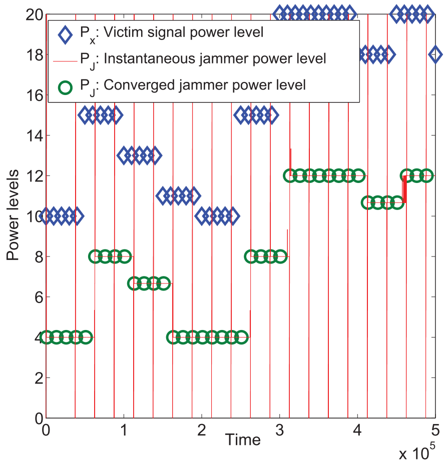

Fig. 13. Learning against a victim with time-varying strategies. The figure shows the power level adaptation by the jammer using a drifting algorithm and that used by the victim.

the adaptive capabilities of the proposed learning algorithms that can overcome the difficulties faced by JB as shown in Fig. 12.

C. Multiple Victims

In this subsection, we consider a case when the jammer uses an omnidirectional antenna and intends to jam two victims in a network. Interesting scenarios arise in this scenario because the jammer has to optimize its jamming strategy based on the PER of both the victims. For example, when both the victims use BPSK, the jammer will learn to use BPSK signaling scheme but the power level at which it should jam depends on the relative power levels of both the victims. Several factors such as path loss, shadowing etc. akin to practical wireless systems can be introduced into this problem, but we are mainly interested in understanding the learning performance of the jammer. Hence we ignore these physical layer parameters and assume that both the victims are affected by the jamming signal with the same JNR. The jammer considers the mean packet error rate seen at both these victims as feedback with target mean $PER = 0.8$, in order to learn the performance of its actions.

Fig. 14 shows the learning performance of the jammer against 2 users that employ BPSK signaling at different power levels. It is seen that the jammer learns to use BPSK signaling as well (since BPSK is optimal to be used against BPSK signaling as discussed in [12]). Similar learning results were achieved when both the users employ QPSK signaling. Fig. 15 shows the learning performance when one user uses QPSK and and the other user uses BPSK. It was observed that when the victim with BPSK has higher power than QPSK victim, the jammer learns to use the BPSK jamming signal and vice versa. This again agrees with previous results which show that BPSK (QPSK) is better to jam a BPSK (QPSK) signal. Also, the learning algorithm performs comparably well to the optimal strategy obtained by performing an extensive grid search over the complete set of strategies. Fig. 16 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, such as power control schemes, in which the victim increases its power level until it reaches a maximum so

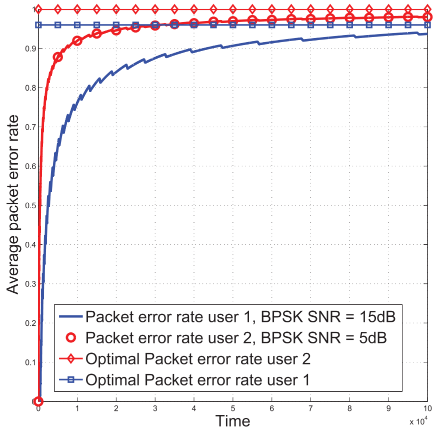

Fig. 14. PER achieved by the jammer against 2 users, user 1 uses BPSK at 15 dB and user 2 sends BPSK at 5 dB. The jammer learns to use BPSK signal with power 13 dB and $\rho = 0.46$.

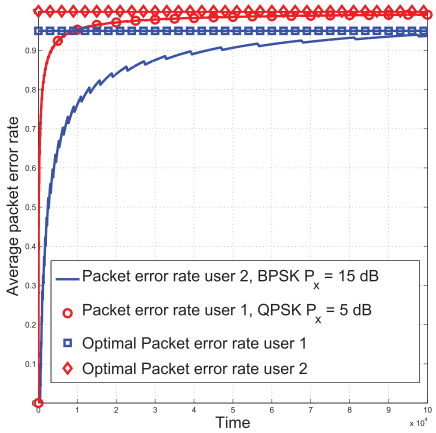

Fig. 15. PER achieved by the jammer against 2 users, user 1 sends QPSK at 5 dB and user 2 sends BPSK at 15 dB. The jammer learns to use BPSK signal with power 11.25 dB and $\rho = 0.25$.

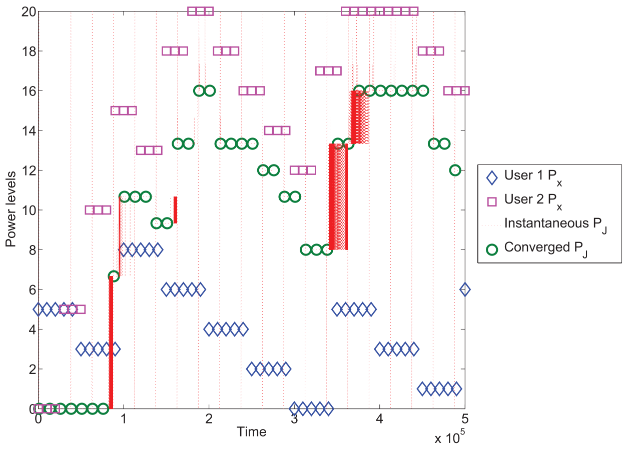

Fig. 16. PER achieved by the jammer against 2 stochastic users in the network. Both the users employ BPSK signaling scheme. The jammer learns to use the BPSK signaling scheme to achieve power efficient jamming strategies and also tracks the changes in the users' strategies.

as to overcome interference). Although each victim has a different adaption cycle (specifically, victim 1 changes its power levels based on the performance history over the past 50000 time instants and victim 2 adapts its power levels over a window of size 30000 time instants), the jammer is capable of tracking these changes in a satisfactory manner.

By using a weighted PER metric rather than a mean PER metric, the jammer can prioritize jamming one set of transmit-receive pairs against the others. Several other multiple victim cases can easily be considered by using this framework. For example, by allowing the jammer to choose the direction of jamming as another action, the jammer can prioritize jamming only the transmit-receive pairs in a given direction rather than spread all its power uniformly across all directions. However, such improved jamming techniques will only come at the expense of more knowledge about the location of the users, users' behavior etc. Nevertheless, it is worth appreciating the applicability of the proposed algorithms to a wide variety of electronic warfare-type scenarios.

D. A Note on the Assumptions

In this paper, we assumed that the jammer is aware of $g(t)$ and T since these parameters are typically defined *a priori* in wireless standards. However, if they are unknown, then the jammer may treat them as additional parameters and learn using JB. For example, the possible pulse shapes that the jammer may use include square pulse, root-raised cosine pulse, and raised cosine pulse. Therefore the jammer can treat this as an additional discrete arm and learn using JB. Since this parameter is learned from a finite space, it will not result in a time order change in the regret.

We also assumed that the jammer is synchronized with the victim receiver. Note that this is possible in practical wireless systems such as LTE because they use signals like PSS and SSS for synchronization between the victim receiver and the transmitter. When the jammer encounters these signals, it can also synchronize with the victim. For results in the paper, where we take SER as the cost function for the jammer, the jammer is assumed to be symbol-synchronous with the victim receiver. However, these results were only shown to prove the validity and performance of our algorithm in comparison to the theoretical optimal results obtained in our previous work [13]. For rest of the results, when PER is considered as the cost function (evaluated using ACK/NACK), then the jammer only needs to be synchronous with the victim on a per-packet basis. Similar assumptions have been made in the past [20]–[26]. However, if the jammer is not synchronized, then the jamming performance is degraded as discussed in [13].

VI. Conclusion

In this paper, we proved that 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. Learning algorithms based on the multi-armed bandit framework were developed to optimally jam the victim transmitter-receiver pairs. The learning algorithms are capable of learning the optimal jamming strategies in both coherent and non-coherent scenarios where the jamming signal and the victim signal are either phase synchronous or asynchronous with each other. Also, the rate of learning is faster in comparison to commonly used reinforcement learning algorithms. These algorithms are capable of tracking the different strategies used by multiple 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 any time instant.

Appendix A
Proof of Theorem 1

For the system model in Section II, the average probability of error at the victim receiver that uses a maximum likelihood (ML) detector (since it is assumed that the victim transmit-receive pair is not aware of the presence of the jammer) is given by

$$\begin{aligned} &p_e\left(j, \text{SNR}, \text{JNR}\right) \\ &\quad = 1 - \int_x \int_\Omega f_N\left(y - \sqrt{\text{SNR}}x - \sqrt{\text{JNR}}j\right) f_X(x) dy dx, \end{aligned} \tag{5}$$

where Ω indicates the ML decision region for x. For instance, when the signal levels are $\pm A$, $\Omega = \mathsf{real}\,(y) < 0$ when $x = -A$ and $\Omega = \mathsf{real}\,(y) > 0$ when $x = +A$. In the above equation, the received signal normalized by the noise power σ^2 is considered. Further, f_X indicates the distribution of the signal x (described by the modulation scheme used by the victim) and f_N indicates the additive white Gaussian noise distribution. For a pulsed jamming signal with pulsing ratio ρ, the SER is given by $\rho p_e(j, \text{SNR}, \frac{\text{JNR}}{\rho}) + (1-\rho) p_e(j, \text{SNR}, 0)$. We first establish the Hölder continuity of $p_e\,(j, \text{SNR}, \text{JNR})$ which can then be used to prove the Hölder continuity of p_e for pulsed jamming scenarios.

In order to prove that SER i.e., p_e is uniformly locally Lipschitz, we show that $|p_e(j, \text{SNR}, \text{JNR}_1) - p_e(j, \text{SNR}, \text{JNR}_2)| \leq L|\text{JNR}_1 - \text{JNR}_2|^\alpha$ for some $L > 0$ and $\alpha > 0$. Using (5) we can bound $|p_e(j, \text{SNR}, \text{JNR}_1) - p_e(j, \text{SNR}, \text{JNR}_2)|$ as shown in (6). Thus it is sufficient to show that $|f_N(y - \sqrt{\text{SNR}}x - \sqrt{\text{JNR}_2}j) -$

$$\begin{aligned} &|p_e(j, \text{SNR}, \text{JNR}_1) - p_e(j, \text{SNR}, \text{JNR}_2)| \\ &\quad = \left| \int_x \int_\Omega \left[f_N\left(y - \sqrt{\text{SNR}}x - \sqrt{\text{JNR}_2}j\right) - f_N\left(y - \sqrt{\text{SNR}}x - \sqrt{\text{JNR}_1}j\right)\right] f_X(x) dy dx \right| \\ &\quad \leq \int_x \int_\Omega \left[\left| f_N\left(y - \sqrt{\text{SNR}}x - \sqrt{\text{JNR}_2}j\right) - f_N\left(y - \sqrt{\text{SNR}}x - \sqrt{\text{JNR}_1}j\right)\right| \right] f_X(x) dy dx. \end{aligned} \tag{6}$$

$f_N(y - \sqrt{\mathrm{SNR}}x - \sqrt{\mathrm{JNR}_1}j)| \le L'|\mathrm{JNR}_1 - \mathrm{JNR}_2|^{\alpha'}$ for some $L' > 0$ and $\alpha' > 0$ which follows from the definition of f_N as it is the probability density function (pdf) of the noise signal n. We briefly show it below for completeness. Since we already normalized the signal by σ^2, the pdf of n is now given by the zero mean unit variance Gaussian distribution.

$$\begin{aligned}
&\left|f_N\left(y - \sqrt{\mathrm{SNR}}x - \sqrt{\mathrm{JNR}_2}j\right) - f_N\left(y - \sqrt{\mathrm{SNR}}s - \sqrt{\mathrm{JNR}_1}j\right)\right| \\
&= \left|\frac{1}{\sqrt{2\pi}}\left[\exp\left(-\left(y - \sqrt{\mathrm{SNR}}x - \sqrt{\mathrm{JNR}_2}j\right)^2\right)\right.\right. \\
&\qquad \left.\left. - \exp\left(-\left(y - \sqrt{\mathrm{SNR}}x - \sqrt{\mathrm{JNR}_1}j\right)^2\right)\right]\right| \\
&\approx \frac{1}{\sqrt{2\pi}}\left[\left|\left(\sqrt{JNR_1} - \sqrt{JNR_2}\right)j\right|\right], \qquad (7)
\end{aligned}$$

where the last approximation is obtained by ignoring the higher order terms since we only consider the cases where $|\mathrm{JNR}_1 - \mathrm{JNR}_2| \le \delta$ i.e., local Hölder continuity. Then, for a given jamming signal j, (7) can be bounded as

$$\begin{aligned}
\frac{\left[\left|\left(\sqrt{JNR_1} - \sqrt{JNR_2}\right)j\right|\right]}{\sqrt{2\pi}} &\le \sqrt{\frac{\mathrm{JNR}_2}{2\pi}}\left|\left(\sqrt{1 + \frac{\delta}{\mathrm{JNR}_2}} - 1\right)\right| \\
&\approx \sqrt{\frac{\mathrm{JNR}_2}{2\pi}}\left|\left(1 + \frac{\delta}{2\mathrm{JNR}_2} - 1\right)\right| \\
&\le \sqrt{\frac{1}{2\pi\mathrm{JNR}_{\min}}\delta} \\
&= \sqrt{\frac{1}{2\pi\mathrm{JNR}_{\min}}(\mathrm{JNR}_1 - \mathrm{JNR}_2)}, \qquad (8)
\end{aligned}$$

which proves that argument inside the integral in [(6), shown at the bottom of the previous page] is uniformly locally Lipschitz with $L' = \sqrt{\frac{1}{2\pi\mathrm{JNR}_{\min}}}$ and $\alpha' = 1$. In the above proof we used the fact that $|j| \le 1$ for standard signaling schemes that are employed by the jammer (for the AWGN jamming signal, the SER is obtained by using a Gaussian distribution with variance $1 + \mathrm{JNR}$ i.e., a slightly different approach when compared to (5) is taken and by following the above sequence of arguments, Hölder continuity can be proved even in this case). Using (8), the overall SER i.e., $p_e(j, \mathrm{SNR}, \mathrm{JNR})$ is also uniformly locally Hölder continuous. By following the same steps, the Hölder continuity for the pulsed jamming cases can also be proved. An example for the Hölder continuity in the pulsed jamming case is shown in Section III.

Appendix B
Proof of Theorem 2

Since the set of signaling schemes is discrete, we first obtain the regret bound for a particular signaling scheme $\mathcal{J}$. It is easy to see that the overall regret bound is a scaled version (by N_{mod}) of the regret achievable for a single signaling scheme. Since the time horizon of the inner loop of Algorithm 1 is T, we first show that the regret incurred by the inner loop is $\mathcal{O}(\sqrt{M^2 T\log(T)})$. Since the overall time horizon is generally unknown, the algorithm is run for several rounds of time steps on the order of 2^i as shown in Algorithm 1 and the regret bounds for the overall algorithm can be achieved by using the doubling trick [38].

The upper bound on the overall regret incurred by Algorithm 1 can be obtained by upper bounding $\sum_{t=1}^{T}\left(\bar{C}(\mathcal{J}, \mathbf{s}^*) - \bar{C}(\mathcal{J}, \mathbf{s}_t)\right)$, where $\bar{C}$ indicates the average cost function and $\mathbf{s}^*$ is the best strategy for a given signaling scheme $\mathcal{J}$ and $\mathbf{s}_t$ is the actual strategy chosen at time t. For ease of presentation, $\mathcal{J}$ is ignored in the rest of the proof. We obtain the regret bound in two steps by rewriting it as

$$\begin{aligned}
\sum_{t=1}^{T}\left(\bar{C}(\mathbf{s}^*) - \bar{C}(\mathbf{s}_t)\right) &= \sum_{t=1}^{T}\left(\bar{C}(\mathbf{s}^*) - \bar{C}(\mathbf{s}')\right) \\
&\quad + \sum_{t=1}^{T}\left(\bar{C}(\mathbf{s}') - \bar{C}(\mathbf{s}_t)\right), \qquad (9)
\end{aligned}$$

where $\mathbf{s}' \in \{1/M, 2/M, \ldots, 1\} \times \mathrm{JNR}_{\min} + (\mathrm{JNR}_{\max} - \mathrm{JNR}_{\min}) * \{1/M, 2/M, \ldots, 1\}$ is the strategy nearest (in terms of the Euclidean distance) to $\mathbf{s}^*$. Then we have $||\mathbf{s}' - \mathbf{s}^*|| = \sqrt{(\mathrm{JNR}' - \mathrm{JNR}^*)^2 + (\rho' - \rho^*)^2} \le \sqrt{\frac{2}{M^2}}$ based on the discretization of the continuous arms set in Algorithm 1.

For the first term in the above equation, by using the Hölder continuity properties of the average cost function $\bar{C}$, we have

$$\begin{aligned}
\mathbf{E}\left(\sum_{t=1}^{T} C_t(\mathbf{s}^*) - C_t(\mathbf{s}')\right) &= \sum_{t=1}^{T}\left(\bar{C}(\mathbf{s}^*) - \bar{C}(\mathbf{s}')\right) \\
&\le TL\left(\frac{2}{M^2}\right)^{\alpha/2}. \qquad (10)
\end{aligned}$$

We now bound the second term $\mathbf{E}\left(\sum_{t=1}^{T} C_t(\mathbf{s}') - C_t(\mathbf{s}_t)\right) = \sum_{t=1}^{T}\left(\bar{C}(\mathbf{s}') - \bar{C}(\mathbf{s}_t)\right)$. Due to the discretization technique used in Algorithm 1, this problem is equivalent to a standard MAB problem with M^2 arms [28]. In order to bound (10), we define two sets of arms: near-optimal arms and sub-optimal arms. We set $\Delta = \sqrt{M^2\log(T)/T}$ and say that an arm is sub-optimal in this case, if its regret incurred is greater than Δ and near-optimal when its regret is less than Δ. Thus, for a near-optimal arm, even when that arm is selected at all time steps, the contribution to regret will be at most $T\Delta$. In contrast for a sub-optimal arm, the contribution to the regret when it is selected can be large. Since we use the UCB1 algorithm, it can be shown that the sub-optimal arms will be chosen only $\mathcal{O}(\log(T)/\Delta(\mathbf{s})^2)$ times ($\Delta(\mathbf{s})$ is the regret of the strategy $\mathbf{s}$) [28], before they are identified as sub-optimal. Thus the regret for these sub-optimal arms is on the order of $\mathcal{O}(\log(T)/\Delta)$ since $\Delta(\mathbf{s}) > \Delta$. From these arguments the second term in (9) can be upper bounded as

$$\mathbf{E}\left(\sum_{t=1}^{T} C_t(\mathbf{s}_t) - C_t(\mathbf{s}')\right) \le \mathcal{O}\left(\sqrt{M^2 T\log(T)}\right). \qquad (11)$$

Using (10) and (11), and setting $M = \lceil(\sqrt{\frac{T}{\log(T)}}L2^{\alpha/2})^{\frac{1}{1+\alpha}}\rceil$ (this is obtained by matching the regret bounds shown in (10) and (11), the regret for any given signaling scheme is given by $\mathcal{O}(\sqrt{M^2 T \log(T)})$. By noting the fact that the jammer can choose from N_{mod} possible signaling schemes and using the value of M, the doubling trick, and summing the regret over all inner loop iterations of Algorithm 1, the regret over the entire time horizon n can be expressed as $\mathcal{O}(N_{mod} n^{\frac{\alpha+2}{2(\alpha+1)}} (\log n)^{\frac{\alpha}{2(\alpha+1)}})$.

References

[1] S. Amuru, C. Tekin, M. van der Schaar, and R. M. Buehrer, "A systematic learning method for optimal jamming," in *Proc. Int. Conf. Commun. (ICC)*, Jun. 2015, pp. 2822–2827.

[2] A. D. Wyner, "The wire-tap channel," *Bell Syst. Tech. J.*, vol. 54, no. 8, pp. 1335–1387, Jan. 1975.

[3] I. Csiszar and J. Korner, "Broadcast channels with confidential messages," *IEEE Trans. Inf. Theory*, vol. IT-24, no. 3, pp. 339–348, May 1978.

[4] Y. Liang, H. V. Poor, and S. Shamai (Shitz), "Information theoretic security," *Found. Trends Commun. Inf. Theory*, vol. 5, no. 7, pp. 355–580, 2008.

[5] T. Basar, "The Gaussian test channel with an intelligent jammer," *IEEE Trans. Inf. Theory*, vol. IT-29, no. 1, pp. 152–157, Jan. 1983.

[6] M. Medard, "Capacity of correlated jamming channels," in *Proc. Allerton Conf. Commun. Control Comput.*, Monticello, IL, USA, 1997, pp. 1043–1052.

[7] A. Kashyap, T. Basar, and R. Srikant, "Correlated jamming on MIMO Gaussian fading channels," *IEEE Trans. Inf. Theory*, vol. 50, no. 9, pp. 2119–2123, Sep. 2004.

[8] A. Mukherjee and A. L. Swindlehurst, "Jamming games in the MIMO wiretap channel with an active eavesdropper," *IEEE Trans. Signal Process.*, vol. 62, no. 1, pp. 82–91, Jan. 2013.

[9] Y. O. Basciftci, C. E. Koksal, and F. Ozguner, "To obtain or not to obtain CSI in the presence of hybrid adversary," in *Proc. IEEE Int. Symp. Inf. Theory*, Istanbul, Turkey, Jan. 2013, pp. 2865–2869.

[10] M. Azizoglu, "Convexity properties in binary detection problems," *IEEE Trans. Inf. Theory*, vol. 42, no. 4, pp. 1316–1321, Jul. 1996.

[11] 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.

[12] S. Amuru and R. M. Buehrer, "Optimal jamming strategies in digital communications-impact of modulation," in *Proc. Global Commun. Conf.*, Austin, TX, USA, Dec. 2014, pp. 1619–1624.

[13] S. Amuru and R. M. Buehrer, "Optimal jamming against digital modulation," *IEEE Trans. Inf. Forensics Security*, vol. 10, no. 10, pp. 2212–2224, Oct. 2015.

[14] K. Dabcevic, A. Betancourt, L. Marcenaro, and C. S. Regazzoni, "A fictitious play-based game-theoretical approach to alleviating jamming attacks for cognitive radios," in *Proc. IEEE Int. Conf. Acoust. Speech Signal Process. (ICASSP)*, Florence, Italy, May 2014, pp. 8158–8162.

[15] Y. E. Sagduyu, R. A. Berry, and A. Ephremides, "Jamming games in wireless networks with incomplete information," *IEEE Commun. Mag.*, vol. 49, no. 8, pp. 112–118, Aug. 2011.

[16] R. McEliece and W. Stark, "An information theoretic study of communication in the presence of jamming," in *Proc. Int. Conf. Commun.*, 1981, pp. 45.3.1–45.3.5.

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

[18] H. I. Volos and R. M. Buehrer, "Cognitive engine design for link adaptation: An application to multi-antenna systems," *IEEE Trans. Wireless Commun.*, vol. 9, no. 9, pp. 2902–2913, Sep. 2010.

[19] H. I. Volos and R. M. Buehrer, "Cognitive radio engine training," *IEEE Trans. Wireless Commun.*, vol. 11, no. 11, pp. 3878–3889, Nov. 2012.

[20] 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.

[21] Y. Wu, B. Wang , K. J. R. Liu, and T. C. Clancy, "Anti-jamming games in multi-channel cognitive radio networks," *IEEE J. Sel. Areas Commun.*, vol. 30, no. 1, pp. 4–15, Jan. 2012.

[22] Y. L. Gwon, S. Dastangoo, C. E. Fossa, and H. T. Kung, "Competing mobile network game: Embracing antijamming and jamming strategies with reinforcement learning," in *Proc. Commun. Netw. Security*, Washington, DC, USA, Oct. 2013, pp. 28–36.

[23] C. Tekin and M. Liu, "Online learning in opportunistic spectrum access: A restless bandit approach," in *Proc. Int. Conf. Comput. Commun.*, Shanghai, China, Apr. 2011, pp. 2462–2470.

[24] S. Ahmad, M. Liu, T. Javidi, Q. Zhao, and B. Krishnamachari, "Optimality of myopic sensing in multichannel opportunistic access," *IEEE Trans. Inf. Theory*, vol. 55, no. 9, pp. 4040–4050, Sep. 2009.

[25] Y. Gai, B. Krishnamachari, and R. Jain, "Learning multiuser channel allocations in cognitive radio networks: A combinatorial multi-armed bandit formulation," in *Proc. New Front. Dyn. Spectr. (DYSPAN)*, Singapore, Apr. 2010, pp. 1–9.

[26] 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.

[27] 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.

[28] P. Auer, N. Cesa-Bianchi, and P. Fischer, "Finite-time analysis of the multi-armed bandit problem," *Mach. Learn.*, vol. 47, no. 2, pp. 235–256, May 2002.

[29] P. Auer, N. Cesa-Bianchi, Y. Freund, and R. E. Schapire, "The nonstochastic multi-armed bandit problem," *SIAM J. Comput.*, vol. 32, no. 1, pp. 48–77, Jan. 2002.

[30] R. Kleinberg, "Nearly tight bounds for the continuum-armed bandit problem," in *Proc. Neural Inf. Proc. Syst.*, 2004, pp. 201–208.

[31] R. Kleinberg *et al.*., "Multi-armed bandits in metric spaces," in *Proc. Symp. Theory Comput. (STOC)*, 2008, pp. 681–690.

[32] S. Magureanu *et al.*, "Lipschtiz bandits: Regret lower bounds and optimal algorithms," in *Proc. Conf. Learn. Theory (COLT)*, 2014, pp. 975–999.

[33] S. Bubeck *et al.*, "Online optimization in X-armed bandits," in *Proc. Neural Inf. Proc. Syst. (NIPS)*, 2008, pp. 201–208.

[34] A. Slivkins, "Contextual bandits with similarity information," in *Proc. Conf. Learn. Theory (COLT)*, 2011, pp. 2533–2568.

[35] 3GPP, "Evolved universal terrestrial radio access (E-UTRA); LTE physical layer-general description (Release 8)," TS 36.201, Nov. 2007.

[36] S. Amuru and R. M. Buehrer, "Optimal jamming using delayed learning," in *Proc. Mil. Commun. Conf.*, Baltimore, MD, USA, Oct. 2014. pp. 1528–1533.

[37] Z. Serceki and M. Wilhoyte, "Method for determining packet error rate of wireless LAN stations," *U.S. Patent 200 400 761 38A1*, Apr. 22, 2004.

[38] J.-Y. Audibert, R. Munos, and C. Szepesvari, "Exploration-exploitation trade-off using variance estimates in multi-armed bandits," *Theor. Comput. Sci.*, vol. 410, no. 19, pp. 1876–1902, Apr. 2009.

[39] S. Amuru, C. Tekin, M. van der Schaar, and R. M. Buehrer, "Jamming bandits," in arXiv preprint arXiv:1411.3652, Nov. 2014.

[40] R. A. Poisel, *Introduction to Communication Electronic Warfare Systems*. Norwood, MA, USA: Artech House, 2008.

[41] P. Auer and R. Ortner, "UCB revisited: Improved regret bounds for the stochastic multi-armed bandit problem," *Period. Math. Hung.*, vol. 61, pp. 55–65, 2010.

[42] M. Strasser, S. Čapkun, and M. Čagalj, "Jamming-resistant key establishment using uncoordinated frequency hopping," in *Proc. IEEE Symp. Security Privacy*, Oakland, CA, USA, May 2008, pp. 64–78.

[43] C. Tekin, L. Canzian, and M. van der Schaar, "Context adaptive big data stream mining," in *Proc. Allerton Conf. Commun. Control Comput.*, Monticello, IL, USA, Oct. 2014, pp. 483–490.

SaiDhiraj Amuru (S'12–M'15) received the B.Tech. degree in electrical engineering from the Indian Institute of Technology Madras, Chennai, India, and the Ph.D. degree in electrical and computer engineering from Virginia Tech, Blacksburg, VA, USA, in 2009 and 2015, respectively. From 2009 to 2011, he was with Qualcomm, India, as a Modem Engineer. He visited the Networks, Economics, Communication Systems, Informatics, and Multimedia Research Laboratory, University of California, Los Angeles, CA, USA, in 2014. His research interests include cognitive radio, statistical signal processing, and online learning.

Cem Tekin received the B.Sc. degree in electrical and electronics engineering from the Middle East Technical University, Ankara, Turkey, the M.S.E. degree in electrical engineering: systems, the M.S. degree in mathematics, and the Ph.D. degree in electrical engineering: systems from the University of Michigan, Ann Arbor, MI, USA, in 2008, 2010, 2011, and 2013, respectively. He is an Assistant Professor of Electrical and Electronics Engineering at Bilkent University, Ankara, Turkey. From February 2013 to January 2015, he was a Postdoctoral Scholar with the University of California, Los Angeles, CA, USA. His research interests include machine learning, multiarmed bandit problems, data mining, multiagent systems, and game theory. He was the recipient of the University of Michigan Electrical Engineering Departmental Fellowship in 2008, and the Fred W. Ellersick Award for the best paper in MILCOM 2009.

Mihaela van der Schaar is Chancellor's Professor of Electrical Engineering at the University of California, Los Angeles, CA, USA. She holds 33 granted U.S. patents. She is also the Founding and Managing Director of the UCLA Center for Engineering Economics, Learning, and Networks. Her research interests include game theory, network science, data science, machine learning, and medical informatics. She was the recipient of an NSF CAREER Award (2004), the Okawa Foundation Award (2006), the IBM Faculty Award (2005, 2007, and 2008), and several best paper awards, including the 2011 IEEE Circuits and Systems Society Darlington Best Paper Award.

R. Michael Buehrer (S'89–M'91–SM'04) joined Virginia Tech, Blacksburg, VA, USA, from Bell Labs as an Assistant Professor with the Bradley Department of Electrical and Computer Engineering in 2001. He is currently a Professor of Electrical Engineering and is the Director of Wireless@Virginia Tech, a comprehensive research group focusing on wireless communications. In 2009, he was a Visiting Researcher at the Laboratory for Telecommunication Sciences (LTS), a federal research laboratory, which focuses on telecommunication challenges for national defense. His research focus was in the area of cognitive radio with a particular emphasis on statistical learning techniques. He has authored or coauthored over 50 journal and approximately 150 conference papers and holds 11 patents in the area of wireless communications. His research interests include position location networks, iterative receiver design, dynamic spectrum sharing, cognitive radio, communication theory, multiple-input multiple-output (MIMO) communications, intelligent antenna techniques, ultra wideband, spread spectrum, interference avoidance, and propagation modeling. His work has been funded by the National Science Foundation, the Defense Advanced Research Projects Agency, Office of Naval Research, and several industrial sponsors. Currently, he is an Associate Editor for the IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS and the IEEE WIRELESS COMMUNICATIONS LETTERS. He was formerly an Associate Editor for the IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGIES, the IEEE TRANSACTIONS ON COMMUNICATIONS, the IEEE TRANSACTIONS ON SIGNAL PROCESSING, and the IEEE TRANSACTIONS ON EDUCATION. In 2003, he was named Outstanding New Assistant Professor by the Virginia Tech College of Engineering. He was a corecipient of the Fred W. Ellersick MILCOM Award for the best paper in the unclassified technical program in 2010, and the Dean's Award for Teaching Excellence in 2014.