Anti-jamming Communication in Cognitive Radio Networks with Unknown Channel Statistics

Qian Wang[†], Kui Ren[†], and Peng Ning[‡]
[†]Department of ECE, Illinois Institute of Technology, Email: {qian, kren}@ece.iit.edu
[‡]Dept. of CS, North Carolina State University, Email: pning@ncsu.edu

***Abstract*—Recently, many opportunistic spectrum sensing and access protocols have been proposed for cognitive radio networks (CRNs). For achieving optimized spectrum usage, existing solutions model the spectrum sensing and access problem as a partially observed Markov decision process (POMDP) and assume that the information states and/or the primary users' (PUs) traffic statistics are known *a priori* to the secondary users (SUs). While theoretically sound, these existing approaches may not be effective in practice due to two main concerns. First, the assumptions they made are not practical, as before the communication starts, PUs' traffic statistics may not be readily available to the SUs. Secondly and more seriously, existing approaches are extremely vulnerable to malicious jamming attacks. A cognitive attacker can always jam the channels to be accessed by leveraging the same statistic information and stochastic dynamic decision making process that the SUs would follow. To address the above concerns, we formulate the problem of anti-jamming multi-channel access in CRNs and solve it as a non-stochastic multi-armed bandit (NS-MAB) problem, where the secondary sender and receiver adaptively choose their arms (*i.e.*, sending and receiving channels) to operate. The proposed protocol enables them to hop to the same set of channels with high probability in the presence of jamming. We analytically show the convergence of the learning algorithms, *i.e.*, the performance difference between the secondary sender and receiver's optimal strategies is no more than $O(\frac{20k}{\sqrt{\varepsilon}}\sqrt{Tn\ln n})$. Extensive simulations are conducted to validate the theoretical analysis and show that the proposed protocol is highly resilient to various jamming attacks.**

I. Introduction

Recently the problem of opportunistic spectrum access (OSA) in cognitive radio networks (CRNs) has received increasing attention due to its potential to improve the spectrum utilization efficiency [1]–[5]. In these spectrum access approaches, the basic principle is the same: individual secondary users (SUs) dynamically search and access the spectrum vacancy to maximize the spectrum utilization while introducing limited interference to the primary users (PUs). To the best of our knowledge, the single-channel sensing and access problem was first investigated under the framework of partially observable Markov decision process (POMDP) in [1]. An myopic sensing policy with a simple round-robin structure was proposed by assuming that a sufficient statistic (*i.e.*, the conditional probability that each channel is idle before sensing starts at time zero) and the order of channel transition probabilities were known to SUs. Under imperfect channel sensing, acknowledgement was used to maintain synchronization between the sender and receiver. In [3], the same authors extended the POMDP framework by considering a multi-channel access problem and proved the optimality of the myopic policy when the total number of channels is two. In [2], the authors proved the optimality of the myopic policy with independent and identically distributed (i.i.d.) positively-correlated channels. In [4], instead of ACKs, a dedicated control channel between the secondary sender and receiver was used for maintaining transceiver synchronization. An upper bound on the optimal reward was derived for the single-channel access by assuming that channels were positively-correlated and all channel states were known after sensing. Recently, the dynamic multi-channel access problem was studied under a special class of restless multi-armed bandit problems (RMBP) in [5], and the proposed *Whittle's index policy* was distinguished from the aforementioned work by achieving near-optimal performance in more general scenarios.

Among these existing protocols, one key assumption made by most of them is that the traffic statistics or the order of the state transition probabilities of all channels are known to the SUs. However, such assumptions may not hold in practice and more seriously, these protocols are not secure in malicious environments. First of all, the PU's traffic statistics (*i.e.*, initial information states and transition probabilities or the order of them) may not be readily available to the SUs prior to the start of sensing. Without *a priori* information on the traffic patterns, those opportunistic spectrum sensing and access protocols cannot work. Moreover, in malicious environments, the attackers can leverage the same statistic information and stochastic dynamic decision making process to jam the channels effectively. In other words, since the structure of those sensing policies is fixed and the channel selection procedure that SUs follow is known, an jammer can predict which channels the SUs are going to use in *each* timeslot and prevent the spectrum from being utilized efficiently.

To cope with jamming attacks, many jamming mitigating protocols, including both frequency hopping spread spectrum (FHSS) and direct-sequence spread spectrum (DSSS) [6], were proposed. These approaches rely on some pre-shared secrets (*i.e.*, hopping sequences and/or spreading codes) prior to communication. However, they are not directly applicable to cognitive radio networks due to the following reasons. First, such coordinated anti-jamming schemes that rely on the pre-sharing of secrets are not applicable in a dynamic SU network since SUs may never meet each other before the start of communication. Second, FHSS requires a wide range of

978-1-4577-1394-1/11/$26.00 ©2011 IEEE

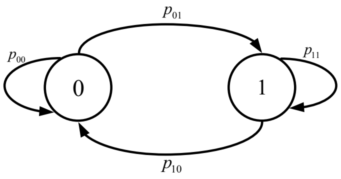


Fig. 1: The Markov channel model.

frequencies, which are not necessarily available in CRNs.

Recently, uncoordinated frequency hopping (UFH) and uncoordinated DSSS (UDSSS) schemes were proposed to eliminate the reliance on the pre-shared secrets [7]–[11]. In UFH, both the sender and receiver hop on randomly selected channels for message transmission without coordination. The successful reception of a packet is achieved when the two nodes reside at the same frequency (channel) during the same timeslot. The major problem with UFH and UDSSS is that they are both very expensive. For UFH, it takes a long time for a sender to transmit a message to a receiver. Thus, it's not practical for CRNs, where the SUs need to finish transmission quickly to yield the channel to the PUs. UDSSS may take less time to transmit a message. However, it is very expensive for the receivers to decode the message. In [12], [13], the problem of defending jamming attacks in cognitive radio networks was investigated using game-theoretic approaches. However, they only explored the single-channel case and assumed that secondary receiver can always communicate with the secondary sender (*i.e.*, they are considered as a single player) and the sensing is perfect.

To address these problems, in this paper we propose a decentralized anti-jamming multi-channel spectrum access protocol for cognitive radio networks, which can accommodate both the environment dynamics and the strategic behaviors of the jammers. To our best knowledge, we are the first to formulate the anti-jamming multi-channel access problem as a non-stochastic MAB problem and develop the online learning based jamming-resistant spectrum access protocol for ad hoc cognitive radio networks. The main contributions of this paper are:

1. We analyze the vulnerability of existing spectrum access protocols under jamming attacks, formulate the anti-jamming problem as a non-stochastic MAB problem and propose the first online adaptive jamming-resistant spectrum access protocol for cognitive radio networks. We analytically show the convergence of the learning algorithms, *i.e.*, the performance difference between the secondary sender and receiver's optimal strategies is no more than $\frac{20k}{\sqrt{\varepsilon}}\sqrt{Tn\ln n}$, where $k = \max\{k_s, k_r\}$, k_r and k_s are the number of channels the receiver and the sender can access simultaneously in each timeslot, and n is the total number of channels. The normalized difference converges to 0 at rate $O(1/\sqrt{T})$ as $T \rightarrow \infty$. We also show that the proposed algorithms can be implemented efficiently with time complexity $O(k_r nT)$ and space complexity $O(k_r n)$ for the receiver, with time complexity $O(k_s nT)$ and space complexity $O(k_s n)$ for the sender.

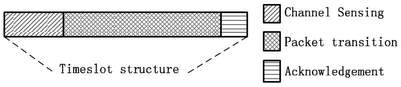


Fig. 2: The structure of a timeslot.

2. We also present a thorough quantitative performance characterization of the proposed scheme. The performance is evaluated by analyzing a practical metric–the expected time for message delivery with *high* probability. We derive the approximation factors for both *static* optimal and *adaptive* optimal strategies. We also perform an extensive simulation study to validate our theoretical results. It is shown that the proposed algorithm is efficient and highly effective against various jamming attacks.

II. Problem Statement

A. System Model

In this paper, we consider a dynamic spectrum access system consisting of a primary user (PU) network and a secondary user (SU) network. We assume the spectrum is divided into n channels, each of which evolves independently (*i.e.*, the channels statistics are not necessarily the same for the n channels) and has the same bandwidth. In the PU network, PUs occupy and vacate the spectrum following a discrete-time Markov process (MDP). As shown in Fig. 1, channel i transits from busy state ("0") to idle state ("1") with probability p_{01} and stays in idle state ("1") with probability p_{11}. In the SU network, SUs seek spectrum opportunities among n channels. That is, they reserve a sensing interval in each timeslot to detect the presence of a PU. Based on the sensing outcomes, they will take the opportunity to access the currently idle channels, and vacate the spectrum whenever PUs reclaim them. We also assume that at the end of the timeslot, the receiver sends an acknowledgement (ACK) to the sender on the channel where a packet transmission is successful. The basic timeslot structure is illustrated in Fig. 2.

We focus on an ad hoc SU network without a central controller for coordinating the SUs. Each autonomous SU thus aims to maximize its own performance by sensing and accessing the spectrum independently [1]. We assume that the traffic statistics (*i.e.*, p_{01} and p_{11}) are not available to SUs. For ease of illustration, we term one pair of communicating SUs as the sender and the receiver. The sender and the receiver are equipped with $k_s < n$ and $k_r < n$ radios, respectively, enabling them to access multiple channels simultaneously in each timeslot. Note that in each timeslot, a SU senses $k_s < n$ and accesses $k_a \leq k_s$ channels sequentially.

We also assume that at the receiver side, efficient message verification schemes (*e.g.*, erasure coding combined with short signatures) are used for packet verification and message reassembly purpose [8]. In our model, we do not consider message authentication and privacy, which are orthogonal to the problems this work addresses.

B. Adversary Model

In this paper, we consider a general and practical jammer with different jamming strategies. In each timeslot, we assume the jammer is capable of sensing and jamming k_j ($k_j < n$) channels simultaneously. We also assume the jammer will not jam the licensed bands when PUs are active due to the facts that i) there will be a heavy penalty on the attackers if their identities are known by the PU network [14] and ii) the attackers may not be too close to the PUs in some scenarios (e.g., in a PU network formed by TV towers). Therefore, the jammer will also utilize the sensing interval to detect the activity of the PUs and jam the idle channels based on the sensing outcomes. Assume the jammer knows the whole spectrum access protocol, his objective then is to prevent the spectrum from being utilized efficiently by the legitimate SUs with the limited jamming capability. Specifically, we focus on the following four types of jammers:

Static jammer: The static jammer is an oblivious jammer. In each timeslot, he selects the same set of k_j channels to emit jamming signals on the channels. The jamming action is made independent of the sensing history he may have observed.

Random jammer: The random jammer is also an oblivious jammer. In each timeslot, he selects a set of k_j channels uniformly at random to emit jamming signals on the channels. The jamming action is made independent of the sensing history he may have observed.

Myopic jammer: The *myopic* jammer is a cognitive jammer running the *myopic* algorithm. He senses all the channels for a certain time and makes an estimation of the traffic statistics. He then makes use of the myopic policy to predict the PUs' channel occupancy pattern and emits jamming signal on the most likely idle channels. The jamming action is made based on the sensing history and the channel occupancy statistics. The myopic policy will be discussed in Section III.

Adaptive jammer: Different from a myopic jammer, the adaptive jammer selects the sensing and jamming k_j channels based on his sensing history and past observations. In this paper, we will focus on an adaptive jammer utilizing multi-armed bandit (MAB) based learning algorithm (the MAB based learning protocol will be shown in section IV). He can adjust his sensing and jamming strategies by leveraging the outcomes of jamming. In other words, we assume that the jammer knows whether he succeeds in jamming the transmitting channels (where both the sender and the receiver reside on in a timeslot) for all the past timeslots. We consider this powerful jammer for performance comparison purpose. Also note that a clever and reasonable jammer will listen during the ACK transmission interval rather than randomly jamming the ACK packets. In fact, it is very difficult to jam the ACKs as the size of ACK packets could be very small.

In this paper, our goal is to develop decentralized anti-jamming spectrum access protocols for an ad hoc cognitive radio network. With unknown spectrum traffic statistics, the proposed protocol should enable the SUs to independently search for spectrum opportunities while accommodating both the traffic statistics and the jamming strategies.

III. Jamming Vulnerability of Existing Multi-channel Opportunistic Access Protocols

In this section, we analyze the weakness of the existing multi-channel opportunistic spectrum access protocols under jamming attacks due to their *deterministic* feature, which motivates us to develop a *probabilistic* spectrum sensing and access approach in the next section. For ease of illustration, in the following we consider a SU network with a single sender-receiver pair, but the same ideas can also be applied and extended to a multi-user setting.

Many spectrum sensing and access policies have been proposed for *jamming-free* cognitive radio networks [1]–[5]. In these models, the sender chooses a subset of n channels to sense based on its past observations and gains a fixed reward if a channel is sensed idle. The objective of the sender is to maximize the rewards that it can gain over a finite or infinite number of timeslots. It was known that this problem can be solved by a stochastic dynamic programming (SDP) approach [15]. The SDP algorithm proceeds backward in time and at every stage t determines an optimal decision rule by quantifying the effect of every decision on the current and future conditional expected rewards. Although it provides a powerful methodology for stochastic optimization, the *backward induction* procedure of SDP is computationally expensive.

To reduce the computation complexity, an *index policy–myopic policy*, which maximizes the conditional expected reward acquired at t–was proposed and explored in recent literature [1], [2]. This policy concentrates only on the present and completely ignores the future. So *myopic approaches* are suboptimal in general. It has also been shown that a sufficient statistic or the *information state* of the system for optimal decision making is given by the belief vector $\Omega(t) = [\omega_1(t), \omega_2(t), \ldots, \omega_n(t)]$, where $\omega_i(t)$ is the conditional probability that channel i is idle in timeslot t. A sensing action $a(t)$ denotes the k_s channels to be sensed in timeslot t. Let $K_i(t) \in \{0, 1\}$ denote whether an ACK on channel i is received or not in timeslot t. Given $a(t)$ and $K_i(t)$, the belief state in timeslot $t+1$ is given by [1]

$$\omega_i(t+1) = \begin{cases} p_{11}^i, & i \in a(t), K_i(t) = 1 \\ p_{01}^i, & i \in a(t), K_i(t) = 0 \\ \omega_i(t)p_{11}^i + (1-\omega_i(t))p_{01}^i, & i \notin a(t) \end{cases} \qquad (1)$$

Assume all channels have the same transmission rate B_i, the myopic policy under Ω is defined as

$$\hat{a}(t) = \arg\max_{a(t)} \sum_{i \in a(t)} \omega_i(t) B_i. \qquad (2)$$

Another *index policy* called *Whittle's index policy* was also applied in the dynamic spectrum access and obtained in closed-form (refer to [5] for the explicit expressions for *Whittle's index*). Similarly, *Whittle's index policy* is implemented by sensing k_s channels with the largest indices in each timeslot. Its optimality is lost in general due to the strict constraint of sensing exactly k_s for all t, but even so the *Whittle's index policy* has the near optimal performance. It has also been shown in [5] that when channels are stochastically identical, the *myopic policy* and the *Whittle's index policy* are equivalent.

Vulnerability to Jamming Attacks. In the above two index policies, their key assumption is that the traffic statistics, *i.e.*, the initial belief vectors $\Omega(0)$ and the order of state transition probabilities (*i.e.*, p_{01}^i is greater or less than p_{11}^i) on all channels are known *a priori* to the SUs. In practice, however, these statistics may not be readily available [4]. It is also worth noting that all the above policies or protocols only work well in non-malicious environments. An essential problem with these protocols is that the channel selection approach is *deterministic*, *i.e.*, the channel hopping is predictable. An intelligent jammer, which knows the traffic statistics of all channels or learns them through sensing and estimation by observing all channels, can leverage these information to obtain the *same* myopic/Whittle's index of all channels. Since the index policies always choose the first k_s channels with largest indices for sensing and accessing, the jammer can use the same dynamic decision process to perform effective jamming attacks. In the worst case, the communication can be completely jammed as the jammer maintains the same updates for channel "index" as SUs in each timeslot.

From a theoretical perspective, the above *index policies* are established based on the stochastic model of the channel statistics. For example, the Whittle's index policy is developed for the restless multi-armed bandit problems (RMBP) [16]. Since the evolvement of information state (belief vector) is known, the players (sender and receiver) can compute ahead of time exactly what payoffs (rewards) will be received from each arm (channel). However, when jamming occurs, the channel statistics caused by the PU cannot reflect the true state (idle or busy) of the channel, and the rewards associated with each arm may not be modeled by a stationary distribution. Hence, the existing *deterministic* dynamic spectrum access protocols are vulnerable to jamming attacks. As will be shown in the next section, we propose a *probabilistic* spectrum access protocol that is resistant to various jamming attacks and can accommodate the special characteristics of cognitive radio networks.

IV. Jamming-resistant Opportunistic Spectrum Access

In this section, we show that the anti-jamming spectrum access problem can be formulated as a non-stochastic multi-armed bandit problem. We then propose an efficient and jamming-resistant multi-channel access protocol for ad hoc cognitive radio networks.

A. Non-stochastic Multi-armed Bandit Problem Formulation

As discussed above, the Whittle's index policy is established under the assumption that the sender can compute ahead of time exactly what rewards will be obtained from each channel. Hence, this class of stochastic MAB problems are optimization problems. Our proposed spectrum protocol is motivated by the fact that, under jamming, no statistical assumptions can be made about the transition of information state and generation of rewards. Thus, the transceivers need to keep *exploring* the best set of channels for transmission as i) jammer may dynamically adjust his strategy and ii) the PUs occasionally occupy and free the channels. At the same time, the transceivers also need to *exploit* the previously chosen best channels as too much exploration will potentially underutilize them. The problem is thus the one balancing between *exploitation* and *exploration*, rather than optimization.

We consider an anti-jamming game among a secondary sender, a secondary receiver and a jammer. The channel states (idle or busy) are not directly observable before the sensing action is made [1]. During the sensing interval of each timeslot, the sender chooses k_s to sense, where the sensing action is made based on all the past decisions and observations. As the sensing outcome could be busy or idle due to the PUs' actions on a channel, the sender chooses k_a ($k_a \le k_s$) idle channels to access. The access action results in a reward at the end of this timeslot. At the receiver side, the receiver independently chooses k_r channels to receive, where action is also made based on all the past decisions and observations. The receiver also receives a reward at the end of this timeslot. During the same timeslot, the jammer chooses k_j channels to sense and jam based on the jamming strategy he uses.

The objective is to choose the sensing, access and receiving actions in each timeslot to maximize the total expected rewards over T timeslots. To further formalize the problem, we consider a vector space $\{0,1\}^n$ and number the available transmitting channels from 1 to n. The sensing strategy space for the sender is set as $S_s \subseteq \{0,1\}^n$ of size $\binom{n}{k_s}$, and the receiver's receiving strategy is set as $S_r \subseteq \{0,1\}^n$ of size $\binom{n}{k_r}$. If the f-th channel is chosen for sending or receiving, the value of the f-th ($f \in \{1,\ldots,n\}$) entry of a vector (or strategy) is 1; 0 otherwise. The jamming strategy space for the jammer is set as $S_j \subseteq \{0,1\}^n$ of size $\binom{n}{k_j}$. For technical convenience, in this case, the value 0 in the f-th entry denotes that the f-th channel is jammed; the value 1 in the f-th entry denotes that the f-th channel is unjammed. The PUs' activities on the channels can be denoted as a vector $s_p \in \{0,1\}^n$, where the value 1 denotes the channel is idle and the value 0 denotes the channel is busy.

During each timeslot, the three parties choose their own respective strategies s_s, s_r, and s_j, where $s_s \in S_s$, $s_r \in S_r$ and $s_j \in S_j$. On the sender side, he receives a reward on channel f if an ACK is successfully received on f. From the perspective of the receiver, rewards (successful receptions) are determined by i) its choice of strategies, ii) the sender's accessing strategies, iii) the dynamics of PU's occupying/vacating the channels and iv) the jammer's choices of jamming strategies. It is easy to see that the sender and receiver's accumulated rewards over T timeslots are the same.

During a certain timeslot t, assume PUs' strategy or activity is s_p. From the receiver's perspective, $s_s \bullet s_p \bullet s_j$ can be considered as a joint decision made by the sender, the PU and the jammer, where $\bullet$ denotes the multiplication of corresponding entries in s_s, s_p and s_j (Note it is not a dot product.). We say that in timeslot t the sender, PU and jammer jointly introduce a *reward* "$g_{f,t} = 1$" for channel f if the value of the f-th entry of $s_s \bullet s_p \bullet s_j$ is 1; a *reward* "$g_{f,t} = 0$" (*i.e.*, no reward is obtained) otherwise. Whether the receiver can obtain the reward depends on the state of the channel f it has *chosen* for packet reception:

Case 1: No packet is received on f. In this case, no *reward* is obtained.

Case 2: A packet is received on f. We use efficient message verification schemes in [8] (*e.g.*, erasure coding combined with short signatures) for packet verification and message reassembly purpose. This is used to defend against jamming based DoS attack. If the packet fails to pass the verification, no *reward* is obtained.

Case 3: A packet is received on f. If jamming/collision is detected on the received packet, no *reward* is obtained. Real experiments have shown in [17] that by looking at the received signal strength during bit reception, accurate *differentiation* of packet errors caused by jamming and those caused by weak links can be realized. Here, we do not differentiate between packet jamming and collision as they both cause interference to the legitimate packets. For simplicity, we do not consider packet coding. So the jammed or collided packets are discarded, resulting in no reward.

Case 4: A packet is received on f. If no jamming is detected, a *reward* 1 is obtained.

Therefore, after choosing a strategy s_r, the reward is revealed to the receiver if and only if f is chosen as a receiving channel. It is obvious that this problem is a non-stochastic MAB problem (NS-MAB) [18], where each channel is considered as *an arm*. Each channel is associated with an arbitrary and unknown sequence of *rewards*. The sender and the receiver can obtain the corresponding rewards on a channel if they choose that channel for sending or receiving. In this paper, we will use online learning algorithms developed under NS-MAB problems [18]–[20] to design the opportunistic spectrum access protocol against various jamming attacks.

We next define some notations used in the following discussion. In each timeslot $t \in \{1, \ldots, T\}$, the sender and receiver independently select a strategy I_t from the strategy sets. We write $f \in i$ if channel f is **chosen** in strategy i, *i.e.*, the value of the fth entry of i is 1. Note I_t denotes a particular strategy chosen in timeslot t, and i denotes a general strategy in the strategy set. The total rewards of a strategy i during timeslot t is $g_{i,t} = \sum_{f \in i} g_{f,t}$, and the cumulative rewards up to timeslot t of each strategy i is $G_{i,t} = \sum_{s=1}^{t} g_{i,s} = \sum_{f \in i} \sum_{s=1}^{t} g_{f,s}$. The total rewards over all **chosen** strategies up to timeslot t is $\widehat{G}_t = \sum_{s=1}^{t} g_{I_s,s} = \sum_{s=1}^{t} \sum_{f \in I_s} g_{f,s}$, where the strategy I_t is chosen randomly according to some distribution over the strategy set. Note that in the following discussions, we use a superscript to differentiate sender from receiver. To quantify the performance, we study the *regret* over T timeslots of the game

$$\begin{cases} \text{On the sender side:} & \max_{i \in S_s} G_{i,T} - \widehat{G}_T^s, \\ \text{On the receiver side:} & \max_{i \in S_r} G_{i,T} - \widehat{G}_T^r, \end{cases}$$

where the maximum is taken over all strategies available to the sender or the receiver. The *regret* is defined as the accumulated rewards *difference* over T timeslots between the proposed strategy and the optimal *static* one in which the sender or the receiver chooses the best fixed set of channels for message reception in the presence of jamming. In other words, the *regret* is the difference between the number of successfully received packets using the proposed algorithm and that using the best fixed solution.

B. The Proposed Anti-jamming Spectrum Access Protocol

In this section, we describe our MAB-based algorithm for frequency hopping, whose performance is asymptotically optimal. The main difficulty of our problem is the tradeoff between *exploration* and *exploitation*. Our algorithm needs to keep *exploring* the best set of channels for transmission since the jammer may dynamically adjusts his strategy, which changes the channel qualities. If the strategy given by our algorithm is not dynamic, the performance will be severely affected by a clever jammer. We will show this in our simulation. At the same time, our algorithm also needs to *exploit* the current best strategy since exploring is risky at the price of performance. Too much exploring will also affect the algorithm's performance.

Now we present our proposed anti-jamming spectrum access protocol as shown in **Algorithm 1**. The algorithm comprises two subalgorithms: $\mathcal{A}^s$ on the sender side and $\mathcal{A}^r$ on the receiver side. The basic idea is as follows: In each timeslot, the sender chooses the "best" channels to sense, obtaining sensing results: busy or idle. It transmits on the sensed idle channels, obtaining ACK from the receiver. Under perfect sensing, receiving no ACK means a channel is jammed or the receiver is not receiving on the same channel. The sender then adjusts its sensing channels in the next timeslot based on the above information. On the receiver side, it adjusts its receiving channels based on the results of packet verification and jamming detection.

Let N^s and N^r denote the total number of strategies at the sender side and the receiver side, respectively. As shown in the algorithm, each strategy is assigned a strategy weight, and each channel is assigned a channel weight. During each timeslot, the channel weight is dynamically adjusted based on the virtual channel rewards revealed to the sender and the receiver:

$$\text{Sender:} \quad w_{f,t}^s = w_{f,t-1}^s e^{\eta^s g_{f,t}^{s\prime}}, \tag{3}$$

$$\text{Receiver:} \quad w_{f,t}^r = w_{f,t-1}^r e^{\eta^r g_{f,t}^{r\prime}}. \tag{4}$$

It is easy to see that the increase of the virtual channel rewards leads to larger channel weights.

Since a strategy is determined by all channels, we define the weight of a strategy as the product of the weights of all

Algorithm 1 An MAB based Anti-jamming Multi-channel Access Protocol.

```
Input: $n, k_r, k_s, T, \varepsilon \in (0,1], \delta \in (0,1), \beta^s, \beta^r \in (0,1], \gamma^s, \gamma^r \in (0,1/2], \eta^s, \eta^r > 0$.
Initialization: The secondary sender (receiver) sets initial channel weight $w^s_{f,0} = 1$ ($w^r_{f,0} = 1$) $\forall f \in [1,n]$, initial hopping strategy weight $w^s_{i,0} = 1$ ($w^r_{i,0} = 1$) $\forall i \in [1, N^s]$ or $[1, N^r]$, and initial total strategy weight $W^s_0 = N^s = \binom{n}{k_s}$ ($W^r_0 = N^r = \binom{n}{k_r}$).
For timeslot $t = 1, 2, \ldots, T$
    The sender selects a sensing strategy $I^s_t$ at random according to its strategy's probability distribution $p^s_{i,t}$ $\forall i \in [1, N^s]$ and the receiver selects a receiving strategy $I^r_t$ at random according to its strategy's probability distribution $p^r_{i,t}$ $\forall i \in [1, N^r]$, with $p^s_{i,t}$ and $p^r_{i,t}$ computed following Eqs. (9) and (10).
    The sender and receiver compute the probability $q^s_{f,t}$ and $q^r_{f,t}$ $\forall f \in [1,n]$ as $q^s_{f,t} = \sum_{i: f\in i} p^s_{i,t}$ and $q^r_{f,t} = \sum_{i: f\in i} p^r_{i,t}$, respectively.
    The sender transmits a packet if and only if the channel is sensed to be idle. At the receiver side, once a packet is received on channel $f$, the receiver performs verification and jamming detection. If the packet passes the check, an ACK is transmitted back to the sender on $f$ at the end of the timeslot.
    The sender calculates the channel reward $g^s_{f,t}$ $\forall f \in I^s_t$ based on the sensing results and ACK information. The receiver calculates the channel reward $g^r_{f,t}$ $\forall f \in I^r_t$ based on the outcomes of signature verification and jamming detection. With the revealed rewards $g_{f,t}$, the sender and receiver further compute the virtual channel rewards $g^{s'}_{f,t}$ and $g^{r'}_{f,t}$ $\forall f \in [1,n]$, respectively, following Eqs. (7) and (8).
    The sender updates the channel weight $w^s_{f,t}$ and strategy weight $w^s_{i,t}$ following Eqs. (3) and (5), respectively. The receiver updates all the channel weight $w^r_{f,t}$ and strategy weight $w^r_{i,t}$ following Eqs. (4) and (6), respectively. They also update the total strategy weight as $W^s_t = \sum_{i=1}^{N^s} w^s_{i,t}$ and $W^r_t = \sum_{i=1}^{N^r} w^r_{i,t}$.
End
```

channels:

$$\begin{aligned} \text{Sender:} \quad w^s_{i,t} &= \Pi_{f\in i} w^s_{f,t} = w^s_{i,t-1} e^{\eta^s g^{s'}_{i,t}}, && (5) \\ \text{Receiver:} \quad w^r_{i,t} &= \Pi_{f\in i} w^r_{f,t} = w^r_{i,t-1} e^{\eta^r g^{r'}_{i,t}}, && (6) \end{aligned}$$

where $g^{s'}_{i,t} = \sum_{f\in i} g^{s'}_{f,t}$ and $g^{r'}_{i,t} = \sum_{f\in i} g^{r'}_{f,t}$.

Here, the reason to estimate reward for each channel first instead of estimating rewards for each strategy directly is that the reward of each channel can provide useful information about the other unchosen strategies containing the same channels. The parameter β is to control the bias in estimating the channel reward $g^{s'}_{f,t}$ and $g^{r'}_{f,t}$, which are computed as:

$$\text{Sender:} \quad g^{s'}_{f,t} = \begin{cases} \frac{g^s_{f,t}+\beta^s}{\varepsilon q^s_{f,t}} R_t & \text{if } f \in I^s_t, \\ \frac{\beta^s}{\varepsilon q^s_{f,t}} R_t & \text{oththerwise,} \end{cases} \tag{7}$$

$$\text{Receiver:} \quad g^{r'}_{f,t} = \begin{cases} \frac{g^r_{f,t}+\beta^r}{q^r_{f,t}} & \text{if } f \in I^r_t, \\ \frac{\beta^r}{q^r_{f,t}} & \text{oththerwise,} \end{cases} \tag{8}$$

where $q^s_{f,t}$ and $q^r_{f,t}$ are channel f's probability distributions computed by the sender and the receiver, respectively. R_t is a Bernoulli random variable with $\mathbf{P}\{R_t = 1\} = \varepsilon$. The definition of virtual rewards can be explained as follows: if a reward 1 is received on a channel which is less likely to be accessed, we will increase the virtual reward of this channel so as to increase its weight.

At the beginning of each timeslot, the sender and the receiver choose their own strategies based on certain probability distributions $p^s_{i,t}$ and $p^r_{i,t}$, which are computed as:

$$p^s_{i,t} = \begin{cases} (1-\gamma^s)\frac{w^s_{i,t-1}}{W^s_{t-1}} + \frac{\gamma^s}{|\mathcal{C}^s|} & i \in \mathcal{C}^s \\ (1-\gamma^s)\frac{w^s_{i,t-1}}{W^s_{t-1}} & \text{otherwise} \end{cases} \tag{9}$$

$$p^r_{i,t} = \begin{cases} (1-\gamma^r)\frac{w^r_{i,t-1}}{W^r_{t-1}} + \frac{\gamma^r}{|\mathcal{C}^r|} & i \in \mathcal{C}^r \\ (1-\gamma^r)\frac{w^r_{i,t-1}}{W^r_{t-1}} & \text{otherwise} \end{cases} \tag{10}$$

The introduction of γ^s and γ^r is to ensure that $p^s_{i,t} \geq \frac{\gamma^s}{|\mathcal{C}^s|}$ and $p^r_{i,t} \geq \frac{\gamma^r}{|\mathcal{C}^r|}$ so that a mixture of exponentially weighted average distribution and uniform distribution can be used [21]. The *covering strategy set* $\mathcal{C}^s$ and $\mathcal{C}^r$ are defined to ensure that each channel/frequency is sampled sufficiently often. The covering set has the property that for each channel f, there is a strategy i in the covering set such that $f \in i$. Since there are totally n channels and each strategy includes k_s or k_r channels, we have $|\mathcal{C}^s| = \lceil \frac{n}{k_s} \rceil$ and $|\mathcal{C}^r| = \lceil \frac{n}{k_r} \rceil$.

Discussion. In the above protocol, the receiver does not sense in each timeslot since the sender and the receiver do not have the same sensing results due to the potential sensing errors. In practice, the operating point of the spectrum sensor is set as the probability of the collision with PUs [1]. There are two types of sensing errors: *false alarm* probability and *miss detection* probability. Without loss of generality, we use τ to denote the *sensing error probability* in the following discussion and analysis. To eliminate the information asymmetry, the sender and receiver thus rely on the common ACK information to compute rewards and update the strategy's probability distribution. This design leads to two observations: i) the accumulated rewards $\widehat{G}^s_t$ and $\widehat{G}^r_t$ are equal; ii) the sender and receiver are not perfectly synchronized. To measure the performance of the system, we should evaluate how close the sender and receiver's strategies are as T increases. This is equivalent to saying that how well the learning based algorithm proceeds to maximize the throughput.

As a final point on the proposed anti-jamming spectrum access protocol, we note that the sensing process consumes more energy than reception, *i.e.*, it is costly to obtain the sensing results. Thus, we introduce a Bernoulli random variable with $\mathbf{P}\{R_t = 1\} = \varepsilon$ on the sender side. That means the sender will sense the channel with probability ε and it may remain silent in some timeslots without transmitting any packets.

V. Performance Analysis

Definition 1: An algorithm $\mathcal{A}$ is *α-static* (or *α-adaptive*) approximation of the *static* (or *adaptive*) optimal solution if and only if it can transmit the message successfully in time αT with high probability (w.h.p) $1 - \frac{1}{l^\epsilon}$ when the *static* (or *adaptive*) optimal solution can transmit the same message successfully with the same probability $1-\frac{1}{l^{\epsilon}}$ in time T, where constant $\epsilon > 0$.

Definition 2: The *regret* of an algorithm $\mathcal{A}$ is the difference between the accumulated rewards using the *static* optimal strategy and that using $\mathcal{A}$ over T timeslots, *i.e.,* $G_T^{max} - G_T^{\mathcal{A}}$, where $G_T^{max} = \max_{i \in S} G_{i,T} = \max_{i \in S} \sum_{f \in i} \sum_{s=1}^{T} g_{f,s}$ and $G_T^{\mathcal{A}} = \sum_{s=1}^{T} g_{I_s,s} = \sum_{s=1}^{T} \sum_{f \in I_s} g_{f,s}$.

Here, the strategy I_s is chosen randomly in each timeslot over strategy set S. We will write G^{max} instead of G_T^{max} whenever the value of T is clear from the context. Note that for two algorithms $\mathcal{A}_1$ and $\mathcal{A}_2$ running along the same time line, their G^{max}s are usually different. As we discussed earlier, the sender changes its strategy based on the joint decision made by the PU, the jammer and the receiver while the receiver changes its strategy based on the joint decision made by the PU, the jammer and the sender. Due to the probabilistic strategy selection at the sender and the receiver, the joint decisions for them are different, which result in the different static optimal strategies at two sides. In the following discussion, we will write $G_T^{max}(s)$ and $G_T^{max}(r)$ to denote the rewards of the *static* optimal strategies for the sender and the receiver, respectively.

Due to the probabilistic strategy selections, the secondary sender and receiver are not synchronized in each timeslot. We next show the sender's sensing strategy and the receiver's receiving strategy will both converge to their own optimal strategies. The following theorem measures how close their optimal strategies are as $T \to \infty$.

Theorem *1:* The normalized reward distance $\frac{1}{T}(G_T^{max}(s) - G_T^{max}(r))$ converges to 0 at rate $O(1/\sqrt{T})$ as $T \to \infty$, with probability at least $1-\delta$. By using dynamic programming, the sensing and access algorithm in Algorithm 1 has time complexity $O(k_s nT)$ and space complexity $O(k_s n)$. The receiving algorithm has time complexity $O(k_r nT)$ and space complexity $O(k_r n)$.

Proof: The proof of the theorem is based on Theorem 1 of [20] with necessary modifications and extensions required to accommodate for the anti-jamming problem. Due to space limitation, we only sketch the general idea for the proof here. We first prove that at the receiver side, with probability at least $1-\delta$, the *regret* $G_T^{max}(r) - G_T^{\mathcal{A}^r}$ is at most $6k_r\sqrt{Tn\ln n}$, while $\beta^r = \sqrt{\frac{k_r}{nT}\ln\frac{n}{\delta}}$, $\gamma^r = 2\eta^r n$ and $\eta^r = \sqrt{\frac{\ln n}{4Tn}}$ and $T \geq \max\{\frac{k_r}{n}\ln\frac{n}{\delta}, 4n\ln n\}$. Then we prove that at the sender side, with probability at least $1-\delta$, the *regret* $G_T^{max}(s) - G_T^{\mathcal{A}^s}$ is at most $14k_s\sqrt{\frac{Tn\ln n}{\varepsilon}}$, while $\beta^s = \sqrt{\frac{k_s}{nT\varepsilon}\ln\frac{2n}{\delta}}$, $\gamma^s = \frac{2\eta^s n}{\varepsilon}$ and $\eta^s = \sqrt{\frac{\varepsilon\ln n}{4Tn}}$ and $T \geq \max\{\frac{k_s\ln^2\frac{2n}{\delta}}{\varepsilon n\ln n}, \frac{n\ln\frac{2n}{\delta}}{k_s}, 4n\ln n\}$. Finally, as $G_T^{\mathcal{A}^s} = G_T^{\mathcal{A}^r}$, $|G_T^{max}(s) - G_T^{max}(r)| \leq 6k_r\sqrt{Tn\ln n} + 14k_s\sqrt{\frac{Tn\ln n}{\varepsilon}} \leq \frac{20k}{\sqrt{\varepsilon}}\sqrt{Tn\ln n}$, where $k = \max\{k_s, k_r\}$. Thus, $\frac{1}{T}(G_T^{max}(s) - G_T^{max}(r)) \to 0$ at rate $O(1/\sqrt{T})$ as $T \to \infty$. **Note** that for clearly differentiating the *regret* bounds for the sender and the receiver, during derivation we loose the bounds a little bit by choosing k_r and k_s instead of $\min\{k_r, k_s\varepsilon(1-\tau), n-k_j\}$, ε. Hence, sensing error probability τ does not appear in the final results.

According to Theorem 3 in [20], we can exploit the internal structure of strategy selection process in **Algorithm 1** to reduce the space and time complexities. By using dynamic programming the proposed algorithms $\mathcal{A}^s$ and $\mathcal{A}^r$ can be efficiently implemented with time and space complexities linear to n and k_s (or k_r). ■

Due to the large message size, the message for transmission should be divided into small fragments or packets to fit the length of the timeslots. Since the transmission process is not reliable (*e.g.*, data packets may be jammed), and the sender and receiver are not perfectly synchronized, the proposed algorithms cannot guarantee the message is delivered in certain time with probability 100%. So we next consider the expected time usage such that a message could be delivered with *high* probability. Here *high* probability means the probability tends to 1 when total number of packets tends to infinite. Since the sender can get ACKs from the receiver, he knows which packets have been received successfully. Therefore, in our protocol, every time the sender wants to send a packet, he will pick up a "new" one that has not been received. Assume a message M is divided into l packets $M_1, M_2, \cdots, M_l$ with the same size, *i.e.*, $|M_i| = |M|/l$ for all $1 \leq i \leq l$. All l packets of message M must be received before the message M can be reassembled. We have the following theorems:

Theorem *2:* When $l \geq 36(1+c\epsilon)k_r n\ln n/(c-1)^2\epsilon^2$, our algorithm is $(1+c\epsilon)$-static approximation for any constant $c > 1$.

Proof: When receiving $(c+\epsilon)l$ packets, the probability p that at least $(c-1)l+1$ kinds of packets are not received is around $p \leq \binom{cl}{l-1}(\frac{l-1}{cl})^{(c+\epsilon)l}$. According to Stirling's approximation we have $e(\frac{n}{e})^n \leq n! \leq e(\frac{n+1}{e})^{n+1}$, we get $p \leq \frac{cl+1}{e^2}(\frac{c}{c-1})^{(c-1)l+1}c^{l-1}\frac{1}{c^{(c+\epsilon)l}} \leq l^{\epsilon}$ when $\epsilon l \geq \frac{\ln(cl+1)}{\ln c}$. Therefore, the probability that at least l different kinds of packets have been received is at least $1-\frac{1}{l^{\epsilon}}$.

To reconstruct the message with high probability, it is necessary to collect at least l packets in time T. In time $(1+c\epsilon)T$, our algorithm will collect at least $(1+c\epsilon)l - 6k_r\sqrt{(1+c\epsilon)Tn\ln n}$. When $l \geq 36(1+c\epsilon)k_r n\ln n/(c-1)^2\epsilon^2$, the number of packets is no less than $(c+\epsilon)l$. Therefore, the probability that the message can be reconstructed successfully is at least $1-\frac{1}{l^{\epsilon}}$ which finishes the proof. ■

Theorem *3:* When $l \geq 36\frac{n^3\ln nK(1+c\epsilon)}{k_s\varepsilon(1-\tau)(n-k_j)(c-1)^2\epsilon^2}$, our algorithm is $\frac{n^2\min\{k_s,k_r,n-k_j\}}{k_s k_r(n-k_j)}(1+c\epsilon)$-adaptive approximation for any constant $c > 1$, where $K = \min\{k_r, k_s\varepsilon(1-\tau), n-k_j\}$, ε is the probability of sensing a channel and τ is the sensing error probability.

Proof: In each timeslot, the sender chooses k_s channel and sense each channel with probability ε. Thus, the total number of channels to be sensed X is binomial distributed with parameters k_s and ε. The expected value of X is $k_s\varepsilon$. Assume τ is the sensing error probability, the adaptive optimal solution get KT packets in T time in expectation where $K = \min\{k_r, k_s\varepsilon(1-\tau), n-k_j\}$. We know that it is necessary to collects at least l packets to reconstruct the message with high probability, which implies $KT \geq l$. On the other hand, since the static optimal solution collect $k_r \frac{k_s \varepsilon(1-\tau)}{n} \frac{n-k_j}{n}$ in expectation each round. Therefore, in time $\frac{n^2}{k_r k_s \varepsilon(1-\tau)(n-k_j)} K(1+c\epsilon)T$, our algorithm collects at least $K(1+c\epsilon)T - 6k_r \sqrt{\frac{n^2}{k_r k_s \varepsilon(1-\tau)(n-k_j)} K(1+c\epsilon)Tn \ln n}$ packets. When $l \geq 36 \frac{n^3 \ln n \min\{k_s \varepsilon(1-\tau), k_r, n-k_j\}(1+c\epsilon)}{k_s \varepsilon(1-\tau)(n-k_j)(c-1)^2 \epsilon^2}$, the above formula is no less than $(c+\epsilon)l$. So the probability to reconstruct the message is at least $1 - \frac{1}{l^\epsilon}$. ■

Discussion. Notice the parameters β, η and γ are determined by the transmission time T. Here we discuss how to choose a feasible T for our algorithm. In our protocol, the sender will determine T and **encode** it in *each* packet. After receiving the first packet, the receiver knows the parameters T and runs our algorithm. Given quality requirement P, which denotes the probability that the receiver can receive the message, the sender can decide a feasible T as follows. The sender first estimates a lower bound $\underline{k_r}$ for k_r and a upper bound $\overline{k_j}$ for k_j. Then it computes ϵ such that $1 - \frac{1}{l^\epsilon} = P$ and finds a feasible constant $c > 1$ such that $l = 36(1+c\epsilon)\underline{k_r} n \ln n/(c-1)^2\epsilon^2$. The total time of transmission will be $T = (1+c\epsilon)l/(\underline{k_r} \frac{k_s \varepsilon(1-\tau)}{n} \frac{n-\overline{k_j}}{n})$. Theorem 2 shows the receiver will obtain the message with probability at least P.

VI. Simulation Studies

In this section, we conduct extensive simulations to demonstrate the performance of our proposed anti-jamming multi-channel access protocol under various jamming attacks. We also compare the performance of our proposed approach with that of the receiver's *static* optimal strategy and *adaptive* optimal strategy. The *static opt* is the best fixed strategy chosen to maximize the number of received packets/total rewards over T timeslots. The *adaptive opt*, which constantly chooses the best strategy in each timeslot and obtains maximized number of received packets, is actually infeasible in reality, and hence serves as the theoretical upper bound for efficiency.

In our simulation, the sender uses MAB-based channel sensing and access strategy and the receiver uses MAB-based receiving strategy; PU dynamically occupies and vacates the spectrum obeying certain traffic statistics (we assume $p_{11}^i > p_{01}^i$); the jammer chooses from four strategies (as defined in section II-B): static, random, myopic and adaptive/MAB-based jamming. We use a four-element tuple to denote the four parties' respective strategies in a particular simulation scenario, *e.g.*, "mab sta dyn mab" denotes that the sender chooses MAB-based strategy, the jammer chooses static jamming strategy, the PU dynamically uses the spectrum and the receiver chooses MAB-based strategy. Without loss of generality, we assume the sender and the receiver have the same number of antennas with $k_s = k_r = 3$. We vary the strategies of the jammer to study the average number of received packets when T increases and the cumulative distribution function (CDF) of the expected time to reach message delivery T^*. We also vary the jammer's jamming capability (k_j) and the total number of orthogonal frequencies n, sensing probability ϵ and sensing error probability τ to study the impact of parameter selections on the performance of the proposed scheme. We show that, the proposed protocol is asymptotically optimal regardless of the jamming strategies. Finally, we measure the statistical distance of the sender and receiver's strategy probability distributions to show their convergence as T increases.

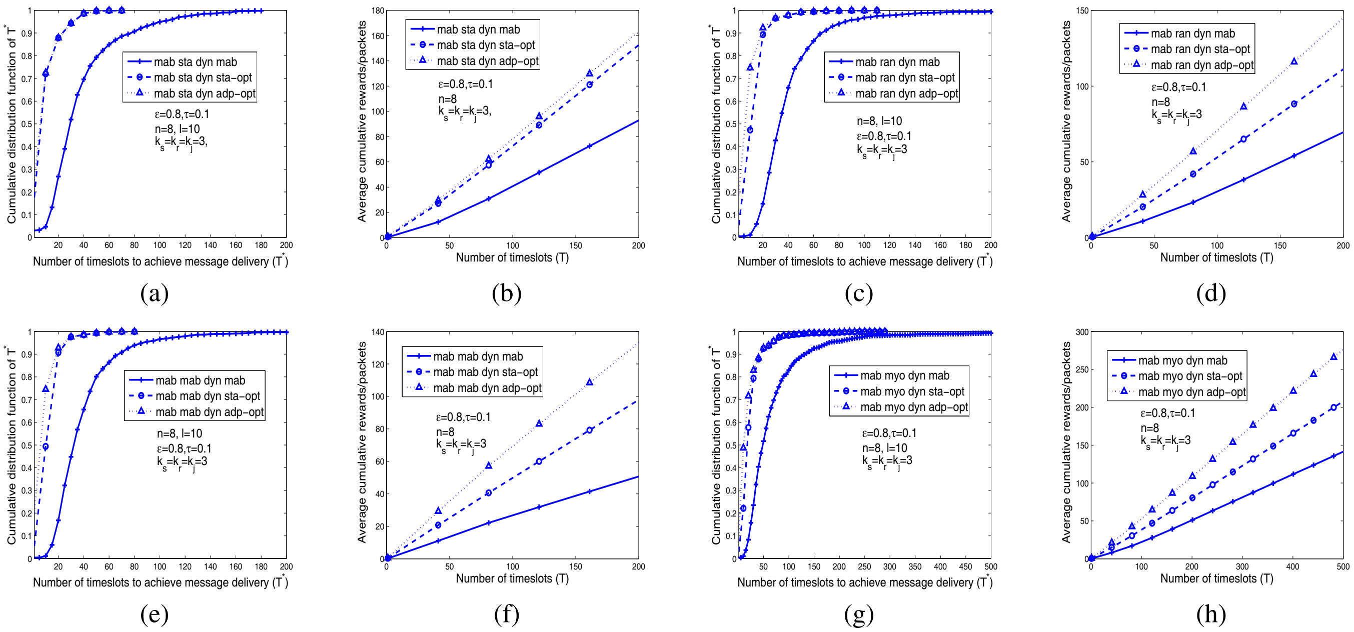
A. Message Delivery with High Probability and Average Cumulative Received Packets

Fig. 3 shows (i) the average number of received packets versus T and (ii) the CDF of the expected time to achieve message delivery under different strategy settings given $l = 10$, $k_j = 3$, $n = 8$ and $p_i^{11} > p_i^{01}$. Fig. 3 (a), (c), (e), (g) show that under different jamming strategies, *static opt* and *adaptive opt* always remain close to each other, especially when static jamming is adopted. This implies that PU's dynamics lead to a seemly "static" channel availabilities from SU's perspective, so the *adaptive optimal* strategy cannot gain much more than the *static optimal* strategy. The comparisons of different jamming strategies on the system performance are shown in Fig. 5. In Fig. 5 (a), it shows that when the jammer chooses static, random or MAB-based jamming strategies and the number of packets is relatively small (*e.g.*, $l = 10$), the message can be successfully received with high probability before $T = 150$. In the case of myopic jamming, it is required at least $T = 250$ timeslots to achieve message delivery with high probability. However, as shown in Fig. 5 (b), when T further increases (*i.e.*, after 150 timeslots), the adaptive jammer using MAB-based algorithm causes almost the same performance deterioration as myopic jamming due to his active learning. The main reason why the myopic and adaptive jamming are the most effective jamming strategies is that they can make use of the system information (*e.g.*, traffic statistics or feedback information) to adjust their strategies.

Fig. 6 (a) and (b) show the effects of sender's sensing probability ϵ and jammer's jamming capability k_j on the system performance, respectively. As expected, the larger k_j will lead to fewer number of received packets, and the larger sensing probability will help improve the performance as the sender can refine his strategy distributions with the sensing results. In Fig. 4, we evaluate the effect of sensing error probability τ on the system performance. It is shown that, in the case of static jamming or random jamming, the average number of cumulative received packets decreases when τ increases. However, it is *surprising* to find that when adaptive and myopic jamming occurs the system performance improves as τ increases. This phenomenon can be explained as follows: although larger sensing error probability will affect the throughput performance, it can help "disrupt" the adaptive and myopic jammers' predictions on the available channels.

B. Convergence Evaluation

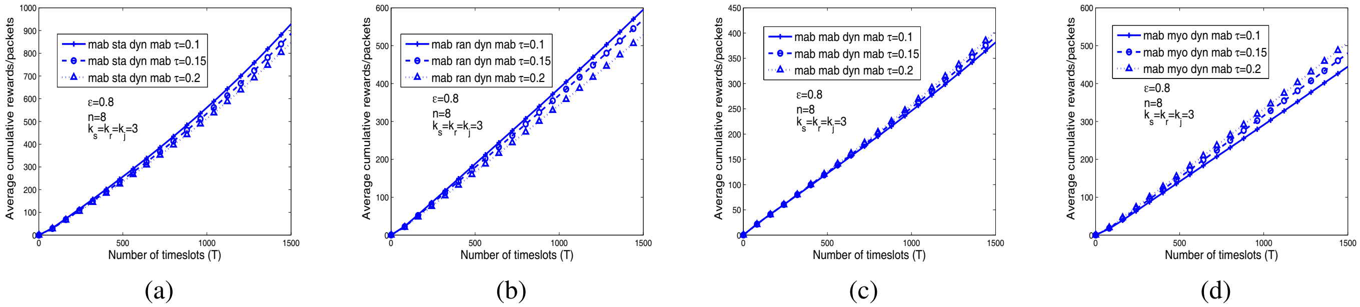
As T increases, the sender and the receiver will converge to their static optimal strategies through online learning,



(a) (b) (c) (d)

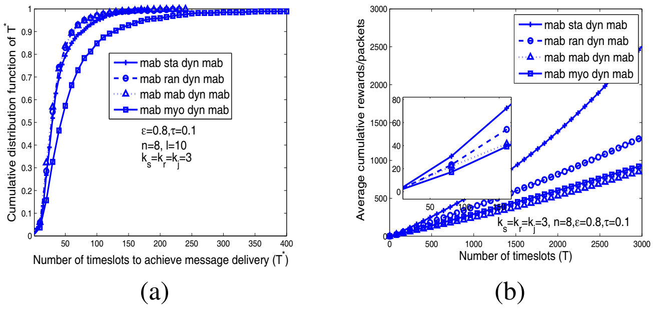
(e) (f) (g) (h)

Fig. 3: Average number of received packets vs. the number of timeslots (T) and CDF of expected time to achieve message delivery under different strategy settings with $p_i^{11} > p_i^{01}$.



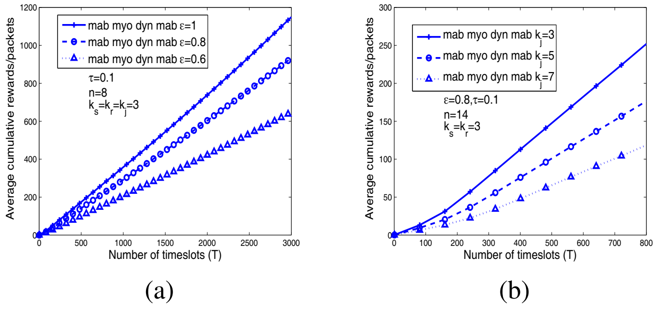
(a) (b) (c) (d)

Fig. 4: The effects of sensing error probability τ on the system performance.



(a) (b)

Fig. 5: The comparisons of different jamming strategies on the system performance.



(a) (b)

Fig. 6: The effects of sensing probability ϵ and jamming capability k_j on the system performance under “mab myo dyn mab”.

respectively. In Section V, we show that the normalized reward difference $\frac{1}{T}(G_T^{max}(s) - G_T^{max}(r))$ converges to 0 at rate $O(1/\sqrt{T})$ as $T \to \infty$. Therefore, we can measure the statistical distance between $q_{f,t}^s$ and $q_{f,t}^r$ (they are both vectors of length n) as the closeness of them indicates that the two parties are approaching the best strategies. In Table I, we show the Euclidean distance of the two parties' channel probability distributions under different jamming scenarios with $p_i^{11} > p_i^{01}$, $n = 8$.

The bold numbers in the table indicate the start of distance

Timeslots T Jamming strategy		800	1200	1600	2000	2400	2800	3200	3600	4000
Static jamming	$\tau = 0.1, \epsilon = 1$	0.0472	**0.0583**	0.0560	0.0425	0.0284	0.0198	0.0147	0.0096	0.0025
	$\tau = 0.1, \epsilon = 0.8$	0.0591	0.0800	**0.0883**	0.0817	0.0651	0.0480	0.0360	0.0238	0.0110
Random jamming	$\tau = 0.1, \epsilon = 1$	0.0348	0.0429	0.0493	0.0543	**0.0565**	0.0563	0.0546	0.0518	0.0485
	$\tau = 0.1, \epsilon = 0.8$	0.0407	0.0518	0.0617	0.0687	0.0741	**0.0778**	0.0762	0.0740	0.0697
Adaptive jamming	$\tau = 0.1, \epsilon = 1$	0.0377	0.0465	0.0521	0.0553	**0.0563**	0.0555	0.0532	0.0504	0.0478
	$\tau = 0.1, \epsilon = 0.8$	0.0446	0.0574	0.0669	0.0737	0.0777	**0.0796**	0.0793	0.0772	0.0749
Myopic jamming	$\tau = 0, \epsilon = 1$	**0.0103**	0.0077	0.0051	0.0037	0.0026	0.0020	0.0019	0.0019	0.0017
	$\tau = 0.1, \epsilon = 1$	0.0262	0.0308	0.0357	0.0390	0.0416	0.0440	0.0450	0.0465	0.0474
	$\tau = 0.1, \epsilon = 0.8$	0.0295	0.0375	0.0439	0.0493	0.0542	0.0580	0.0604	0.0624	0.0645

TABLE I: Convergence of the secondary sender and receiver's strategy probability distributions. The Euclidean distance of the two parties' channel probability distribution is measured under $p_i^{11} > p_i^{01}$, $n = 8$.

decrease at certain time. In general, it is shown that the sender and the receiver's perceptions about the channels converge the fastest under static jamming, and the worst performance is obtained in the case of myopic jamming. The performances under the random and adaptive jamming are almost the same. The sensing probability ϵ and sensing error probability τ also have a great effect on the performance, especially when myopic jamming occurs. As shown, when $\epsilon = 1$ and $\tau = 0$, the distance decreases since $T = 800$. However, when the τ increases, it requires a long time for the distance to decrease. This implies that in face of a powerful jammer such as myopic jammer, it would be better to choose a spectrum sensor with high sensing accuracy.

VII. Conclusion

In this paper, we studied the design of anti-jamming mechanism in cognitive radio networks. We formulated the anti-jamming multi-channel access problem in CRNs as a non-stochastic multiple-armed bandit (NS-MAB) problem, where the secondary sender and receiver adaptively choose their sending and receiving channels in each timeslot to maximize the throughput. The proposed protocol enables the secondary sender and receiver to hop to the same set of available channels with high probability. We analytically showed the convergence of the learning algorithms, *i.e.*, the performance difference between the secondary sender and receiver's optimal strategies is no more than $O(\frac{20k}{\sqrt{\varepsilon}}\sqrt{Tn\ln n})$. Extensive simulation were conducted to validate the theoretical analysis and show that the proposed protocol is very effective and resilient against various jamming attacks.

Acknowledgment

This work was partially supported by the US National Science Foundation under grants CNS-0831963, CNS-1117811, and CNS-1016260.

References

[1] Q. Zhao, L. Tong, A. Swami, and Y. Chen, "Decentralized cognitive mac for opportunistic spectrum access in ad hoc networks: A pomdp framework," *IEEE JSAC*, vol. 25, no. 3, pp. 589–600, 2007.

[2] S. H. A. Ahmad, M. Liu, T. Javidi, Q. Zhao, and B. Krishnamachari, "Optimality of myopic sensing in multi-channel opportunistic access," *IEEE Transactions on Information Theory*, vol. 55, no. 9, pp. 4040–4050, 2009.

[3] K. Liu, Q. Zhao, and B. Krishnamachari, "Dynamic multichannel access with imperfect channel state detection," *IEEE Transactions on Signal Processing*, vol. 58, no. 5, pp. 2795–2808, 2010.

[4] J. Unnikrishnan and V. V. Veeravalli, "Algorithms for dynamic spectrum access with learning for cognitive radio," *IEEE Transactions on Signal Processing*, vol. 58, no. 2, pp. 750–760, 2010.

[5] K. Liu and Q. Zhao, "A restless bandit formulation of multi-channel opportunistic access: Indexablity and index policy," *IEEE Transactions on Information Theory*, vol. 56, no. 11, pp. 5547–5567, 2010.

[6] A. J. Viterbi, *CDMA: Principles of Spread Spectrum Communication*. Addison Wesley, 1995.

[7] M. Strasser, C. Pöpper, S. Capkun, and M. Cagalj, "Jamming-resistant key establishment using uncoordinated frequency hopping," in *Proc. of IEEE Security and Privacy*, May 2008.

[8] M. Strasser, C. Pöpper, and S. Capkun, "Efficient uncoordinated FHSS anti-jamming communication," in *Prob. of ACM MobiHoc'09*, July 2009.

[9] D. Slater, P. Tague, R. Poovendran, and B. J. Matt, "A coding-theoretic approach for efficient message verification over insecure channels," in *Proc. of ACM WISEC'09*. ACM, 2009.

[10] A. Liu, P. Ning, H. Dai, and Y. Liu, "USD-FH: Jamming-resistant wireless communication using frequency hopping with uncoordinated seed disclosure," in *Proc. of MASS'10*, 2010.

[11] Y. Liu, P. Ning, H. Dai, and A. Liu, "Randomized differential DSSS: Jamming-resistant wireless broadcast communication," in *Proc. of IEEE INFOCOM'10*, 2010.

[12] H. Li and Z. Han, "Dogfight in spectrum: Combating primary user emulation attacks in cognitive radio systems, part i: Known channel statistics," *IEEE Transactions on Wireless Communications*, vol. 9, no. 11, pp. 3566–3577, 2010.

[13] ——, "Dogfight in spectrum: Combating primary user emulation attacks in cognitive radio systems - part ii: Unknown channel statistics," *IEEE Transactions on Wireless Communications*, vol. 10, no. 1, pp. 274–283, 2011.

[14] B. Wang, Y. Wu, K. J. R. Liu, and T. C. Clancy, "A stochastic anti-jamming game in cognitive radio networks," *IEEE JSAC*, 2011, to appear.

[15] A. O. Hero, D. A. Castan, D. Cochran, and K. Kastella, *Foundations and Applications of Sensor Management*. Springer Publishing Company, Incorporated, 2007.

[16] P. Whittle, "Restless bandits: activity allocation in a changing world," *Journal of Applied Probability*, vol. 25A, pp. 287–298, 1988.

[17] M. Strasser, B. Danev, and S. Čapkun, "Detection of reactive jamming in sensor networks," in *ACM Transactions on Sensor Networks (TOSN)*. ACM, 2010.

[18] P. Auer, N. Cesa-Bianchi, Y. Freund, and R. E. Schapire, "The non-stochastic multiarmed bandit problem," *SIAM J. Comput.*, 2002.

[19] B. Awerbuch and R. D. Kleinberg, "Adaptive routing with end-to-end feedback: distributed learning and geometric approaches," in *Proc. of ACM STOC'04*, 2004, pp. 45–53.

[20] A. György, T. Linder, G. Lugosi, and G. Ottucsák, "The on-line shortest path problem under partial monitoring," *J. Mach. Learn. Res.*, 2007.

[21] P. Auer, N. Cesa-Bianchi, Y. Freund, and R. E. Schapire, "Gambling in a rigged casino: The adversarial multi-arm bandit problem," in *Proc. of IEEE FOCS'95*, 1995, pp. 322–331.