Jamming-resistant Multi-radio Multi-channel Opportunistic Spectrum Access in Cognitive Radio Networks

Qian Wang[†], Hai Su[†], Kui Ren[†], and Kai Xing[‡]

[†]Department of ECE, Illinois Institute of Technology, Email: {qian, hai, kren}@ece.wpi.edu

[‡]Dept. of CS, University of Science and Technology of China, Email: kxing@ustc.edu.cn

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 readily be 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 exploiting the same stochastic dynamic decision making process. To address the above concerns, we formulate the problem of anti-jamming multi-channel access in CRNs and solve it as a non-stochastic multiple-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 the sender and receiver to hop to the same set of available channels with high probability in the presence of malicious jamming attacks. 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 simulation are conducted to validate the theoretical analysis and show that the proposed protocol is highly resilient under various jamming scenarios.

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], [10], [11], [19], [22]. 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 level of 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 [22]. In [22], 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, the acknowledge information was used to maintain synchronization between the sender and receiver. In [11], the same authors extended the POMDP framework by considering a multi-channel access problem and prove the optimality of the myopic policy when the total number of channels is two. In [1], the authors proved the optimality of the myopic policy with independent and identically distributed (i.i.d.) positively-correlated channels. In [19], instead of ACKs, a dedicated control channel between the secondary sender and receiver was used for maintaining transceiver synchronization. Upper bounds on the optimal reward were 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 [10], 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 or 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 readily be 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 use the same stochastic dynamic decision making process to jam the channels effectively. In other words, due to the fixed structure of those sensing policies, an jammer can predict which channels the SUs are

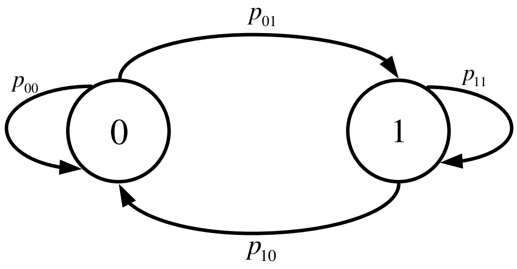


Fig. 1: The Markov channel model.

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) [20], are proposed. However, they are not directly applicable to cognitive radio networks either due to the ad hoc nature of the secondary user network or the primary users' dynamic activities on the spectrum. More specifically, these coordinated hopping approaches rely on some pre-shared secrets (*i.e.*, hopping sequences and/or spreading codes) prior to communication and do not consider the behavior of the PUs. Thus, they inevitably cause high interference to the dynamic PUs. Recently uncoordinated frequency hopping (UFH) schemes are proposed to eliminate the reliance on the pre-shared secrets [15], [17], [18], where 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. Still, significant interference is introduced to the primary user network due to the SUs' random hopping. To address this problem, 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, for the first time, formulate the anti-jamming problem as a non-stochastic MAB problem and develop the online learning based anti-jamming spectrum access protocol for ad hoc cognitive radio networks. The main contributions of this paper are:

1. We first propose an opportunistic spectrum access protocol with unknown traffic statistics for cognitive radio networks and analyze its vulnerability to jamming attacks. We then 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

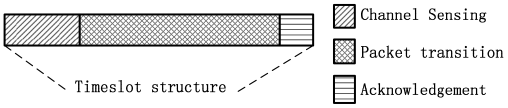


Fig. 2: The structure of a timeslot.

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 $T \to \infty$. 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.

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. Some interesting results are obtained, and it is shown that the proposed algorithm is efficient and highly effective against various jamming attacks.

The rest of the paper is organized as follows: Section II describes the system model, attack model. Section III discusses the related work. Section IV presents an opportunistic spectrum access protocol with unknown traffic statistics and analyzes its vulnerability to jamming attacks. Section V provides a detailed description of a jamming-resistant opportunistic spectrum access protocol. Section VI and VII present the theoretical performance analysis and simulation results, respectively. Finally, Section VIII concludes the paper.

II. Problem Statement

A. System Model

In this paper, we consider a dynamic spectrum access system consisting of a primary user network and a secondary user 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 primary user network, the primary users (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 secondary network, the secondary users (SUs) seek spectrum opportunities among n channels. That is, they reserve a sensing interval in each timeslot to detect the presence of a primary user. Based on the sensing outcome, 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 secondary network without a central controller for coordinating the secondary user network. Each autonomous SU thus aims to maximize it own performance by sensing and accessing the spectrum independently. 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 receiver. The sender and receiver are equipped with $k_s < n$ and $k_r < n$ radios, respectively, enabling them to sense and receive on multiple channels simultaneously at each timeslot. Note that in each timeslot, a secondary user can sense $k_s < n$ and access $k_a \leq k_s$ 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 [17]. In our model, we do not consider message authentication and privacy, which are orthogonal to the problems this work addresses.

B. Adversary Model

Due to different attack philosophies, different attack models will have different levels of effectiveness. In this paper, we consider a general and practical jammer with different jamming strategies. In each timeslot, we assume the jammer is capable of jamming k_j $(k_j < n)$ channels simultaneously. We also assume the jammer will not jam the licensed bands when the primary users are active due to the facts that i) there will be a heavy penalty on the attackers if their identities are known by the primary network and ii) the attackers cannot be too close to the primary users. Therefore, the jammer will also utilize the sensing interval to detect the activity of the primary users 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 secondary users with the limited jamming capability. Specifically, we focus on the following four types of jammers:

(1) *Static jammer*: The static jammer is an oblivious jammer. In each timeslot, he selects the same set of k_j channels to sense and emits jamming signal on the idle channels. The jamming action is made

independent of the sensing history he may have observed.

(2) *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 sense and emits jamming signal on the idle channels. The jamming action is made independent of the sensing history he may have observed.

(3) *Myopic jammer*: The *myopic* jammer is a cognitive jammer running the *myopic* algorithm (the myopic policy will be shown in IV). 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 primary users' 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.

(4) *Adaptive jammer*: Different from a myopic jammer, the adaptive jammer selects the sensing and jamming channels by utilizing an online MAB based learning algorithm (the MAB based learning protocol will be shown in V). Like the sender, he can adjust his sensing and jamming strategies by leveraging the received ACKs from the receiver. The jamming action is made based on the sensing history and the channel occupancy statistics. Note that a clever and reasonable jammer will listen during the ACK transmission interval rather than randomly jamming the ACK packets. Actually, it is very difficult to jam the ACKs as the size of ACK packets are very small.

Note that after the sensing interval, the jammer will make a decision to jam or not in the data transmission interval. We assume that the jammer cannot perform the sensing and jamming operations within the *same* data transmission interval under the appropriately chosen channel hopping rate. Empirical data shows that sensing a channel takes tens of ms [2], [14]. For example, consider a typical sum of channel sensing time t_s and switching time t_w being 10ms [2], for a channel with data rate $B = 10$Mbps, a successful jamming attack on the transmitted packet within the *same* data transmission interval requires the length of packet is at least 10^5 bits. Thus, we can defeat such attack by properly setting the length of the transmission interval (the ACK interval is very small compared to the data transmission interval).

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. Related Work

Opportunistic spectrum access in CRNs In the context of cognitive radio for opportunistic spectrum access, a single-channel access problem within the framework of POMDP is investigated, and myopic policies under both perfect and imperfect sensing cases were first proposed in [22]. In [1], the single-channel access problem with perfect sensing is further analyzed and the optimality of the myopic policy under $p_{11} > p_{01}$ was proved. It has also been shown that if $p_{01} > p_{11}$ the myopic policy remains optimal when the number of channels $n \leq 3$ and the discount factor $\beta \leq 1/2$. In [11], Liu *et al.* extended the POMDP framework by considering a multi-channel access problem. The optimality of the myopic policy was proved for $n = 2$, and the lower and upper bounds on the throughput achieved by the myopic policy were derived. In [19], instead of using ACK, the authors adopted a dedicated control channel between the secondary sender and receiver for transceiver synchronization. When $p_{11} > p_{01}$ for single-channel access, upper bound was derived by assuming that states of all channels are known after sensing. They also considered a parametric model for the distribution of the received signal and developed an algorithm with learning capability. Recently, Liu *et al.* [10] studied a special class of restless multi-armed bandit problems (RMBP), established the indexability and obtained optimal index policy under certain conditions. The proposed policy can be implemented with low complexity and had better performance than myopic policy when channels are not stochastically identical.

Multi-armed bandit problem. In classic multi-armed (k-armed) bandit (MAB) problems, a gambler operates *exactly* one machine at each timeslot; all other machines remain frozen. Each operated machine provides a reward drawn from a known distribution associated with that specific machine. The objective of the gambler is to maximize the sum of rewards earned through a sequence of machine operations. Gittins et al. [6] proved that an optimal solution for the this problem is of *index type*. When $m(m < k)$ machines are operated each time and each machine evolves over time even not being operated, the problem becomes a restless multi-armed bandit problem (RMBP). Whittle [21] showed that an optimal solution of the *index type* can also be established in some cases. In this version of the bandit problem, the generation of rewards is assumed to be subject to certain distributions that are known to the gambler. Non-stochastic multi-armed bandit problems are another important version of MAB problems that incorporate an "exploration vs. exploitation" trade-off over an online learning process [3], [4]. The non-stochastic MAB is widely used in solving online shortest path problems, where the decision makers has to choose a path in each round such that the weight of the chosen path be as small as possible [5], [7], [9], [12]. Because the number of possible pathes is exponentially large, the direct application of [4] to the shortest path problem results a too large bound, *i.e.*, dependence on $\sqrt{N}$. The authors in [5], [12] designed algorithms for shortest path problem using the exponentially weighted average predictor and the follow-the-perturbed-leader algorithm. However, the dependence of number of rounds T in their algorithms is much worse than that of [4] (*i.e.*, $O(T^{\frac{2}{3}})$ [5] and $O(T^{\frac{3}{4}})$ [12]). In [7], the authors consider the shortest path problem under partial monitoring model and proposed an algorithm with performance bound that is polynomial in the number of edges. In this paper, we formulate the anti-jamming spectrum sensing and access problem as a non-stochastic MAB problem and analyze it under partial monitoring model [7], where only the rewards of the chosen arms are revealed to the decision maker.

Uncoordinated FHSS anti-jamming communication. The problem of uncoordinated frequency hopping spread spectrum (FHSS) anti-jamming communication has been investigated in recent literature [17], [18]. In [18], the authors proposed an uncoordinated frequency hopping (UFH) scheme based on which messages of Diffie-Hellman key exchange protocol can be delivered in the presence of a jammer. Due to the sender and the receiver's random choices on the sending and receiving channels, the successful reception of fragments is achieved only when the two nodes coincidentally reside at the same channel during the same timeslot. The first work on efficiency study of UFH-based communication is recently proposed in [17], which shows if the sender and the jammer both choose the random strategy, the receiver's best choice would be random strategy.

In this paper, we extend the idea of uncoordinated communication on dynamic spectrum access in cognitive radio networks. Different from previous work where the sender and receiver perform random hopping, we introduce online learning theory into the design of spectrum sensing, access and receiving algorithm in CRNs. The proposed protocol enables the sender and receiver to perform as best as they can and converge to the best strategies as time increases.

IV. Multi-channel Opportunistic Access With Unknown Traffic Statistics

In practice, the primary user's traffic statistics (*i.e.*, transition probabilities and initial belief states) are unknown to the secondary users. In this section, we propose a multi-channel opportunistic spectrum access protocol with unknown traffic statistics. We assume traffic statistics on primary channels are unknown to the secondary users and the communication is jamming-free. Then we analyze the weakness of the protocol under jamming attack due to its 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 secondary user 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], [10], [11], [19], [22]. In this model, the sender chooses a subset of n channels to sense based on its history observations and gains a fixed reward if a channel is sensed idle. The objective of the sender is to maximize the reward that it can gain over a finite or infinite timeslots. It was known that this problem can be solved by a stochastic dynamic programming (SDP) approach [8]. 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 in many applications.

To reduce the computation complexity, a *index policy– myopic policy*, which maximizes the conditional expected reward acquired at t is proposed and explored in recent literature [1], [22]. 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 the the reception of an ACK on channel i or not in timeslot t. Given $a(t)$ and $K_i(t)$, the belief state in timeslot $t+1$ is given by [22]

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

Assume all channel have the same transmission rate B_i(we normalize it as $B_i = 1$), the myopic policy under Ω is defined as

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

Another *index policy* called *Whittle's index policy* is also applied in the dynamic spectrum access and obtained in closed-form (refer to [10] 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 [10] that when channels are stochastically identical, the *myopic policy* and the *Whittle's index policy* are equivalent.

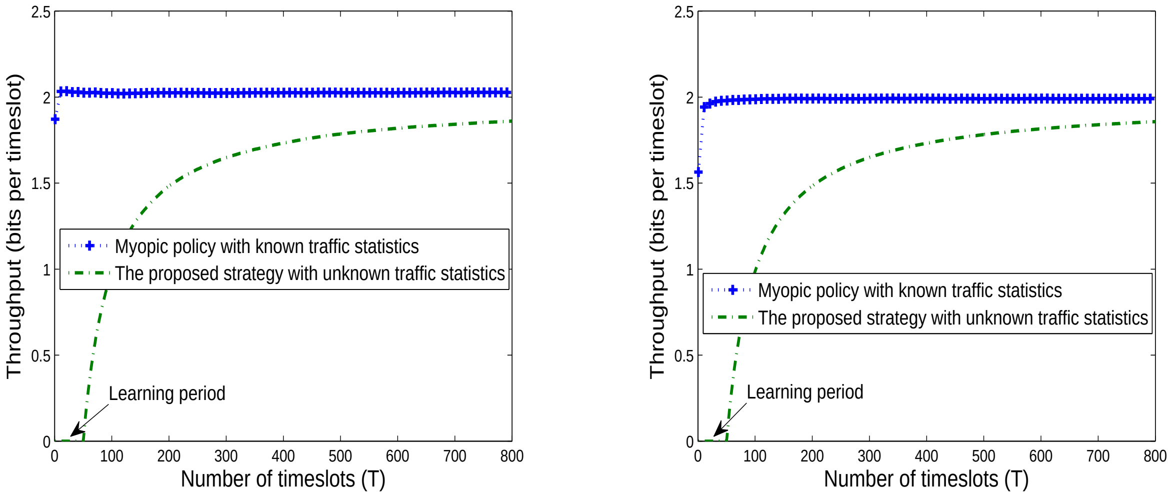
In the above two index policies, their key assumption is that the traffic statistics, *i.e.*, the initial belief vectors $\Omega(0)$ and the order of state transition probabilities (*i.e.*, p_{01}^i is greater or less than p_{11}^i) on all channels are known *a priori* to the SUs. In practice, however, these information may not readily available [13], [19]. To address this problem, we propose a dynamic multi-channel access protocol with online learning capabilities as shown in **Algorithm 1**. The main idea is as follows. The secondary sender and receiver first independently monitor the spectrum for a certain period. Based on the sensing

Algorithm 1 A Dynamic Multi-channel Access Protocol with Unknown Traffic Statistics

```
Input: $n, k_r, k_s, L, T$.
Initialization: The secondary sender (receiver) divides the $n$ available channels into $\lceil \frac{n}{k_s} \rceil$ ($\lceil \frac{n}{k_r} \rceil$) groups.
1: The secondary sender and receiver sense each group of the channels for $L$ timeslots and jointly compute the maximum likelihood estimators for $p_{01}$ and $p_{11}$. $\hat{p}_i^{01} = \frac{A_i^{01}}{A_i^{01}+A_i^{00}}$ and $\hat{p}_i^{11} = \frac{A_i^{11}}{A_i^{11}+A_i^{10}}$, where $A_i^{kl}$ $(k, l \in \{0, 1\})$ is the number of transitions $k$ to $l$ in the training data at channel $i$. The sender and receiver share their transition count information $A_{kl}$ with each other.
2: After $\max\{\lceil \frac{n}{k_s} \rceil, \lceil \frac{n}{k_r} \rceil\}L$ timeslots, the secondary sender and receiver implement the same spectrum sensing and access strategy, i.e., myopic policy or Whittle's index policy. In particular, the sender (receiver) senses those $k_s$ ($k_r$) channels with the highest indices in the sensing interval of each timeslot. If a channel is sensed to be idle, it is accessed.
3: The receiver transmits an ACK to the sender on channel $i$ at the end of each timeslot if it successfully receives a packet on channel $i$.
4: Both the sender and the receiver update their belief vector $\Omega$ according to (1). They also update $A_{kl}$, $\hat{p}_i^{01}$ and $\hat{p}_i^{11}$ if a channel $i$ is selected to access for consecutive two timeslots.
```

results, they obtain a *rough* estimation of the $\{p_{01}^i, p_{11}^i\}$ using maximum likelihood estimators and share with each other the count information, *i.e.*, the number of times each particular transmission happens. Then the sender and receiver will implement the same spectrum sensing and access policy for channel selection. During the communication process, i) the sender and receiver update Ω based on the common ACK information such that transceiver synchronization is maintained, and ii) they continuously refine $\{p_{01}^i, p_{11}^i\}$ based on the sensing results. Actually, we can let only the sender sense the channel, and include the estimated transition probabilities in the packets transmitted to the receiver. In this case, transceiver synchronization is also maintained. Fig. 3 compares the throughput of the proposed learning based spectrum access protocol and that of the one with full knowledge of traffic statistics. It is shown that the proposed dynamic spectrum access protocol with unknown traffic statistics can quickly converges to the greedy approach (*i.e.*, *myopic policy*) with full prior knowledge.

Discussion. It is 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



(a)

(b)

Fig. 3: Performance comparison between the myopic approach and the proposed learning strategy. (a) n=8, $L = 50$, $B_i = 1$, $\{p_{01}^i\}_{i=1}^8 = \{0.1, 0.2, 0.2, 0.4, 0.3, 0.1, 0.1, 0.2\}$, $\{p_{11}^i\}_{i=1}^8 = \{0.7, 0.8, 0.6, 0.8, 0.7, 0.7, 0.6, 0.9\}$. (b)$n$=8, $L = 50$, $B_i = 1$, $\{p_{01}^i\}_{i=1}^8 = \{0.7, 0.8, 0.6, 0.8, 0.7, 0.7, 0.6, 0.9\}$, $\{p_{11}^i\}_{i=1}^8 = \{0.1, 0.2, 0.2, 0.4, 0.3, 0.1, 0.1, 0.2\}$.

statistics or learn them through sensing and estimation, 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) [21]. 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 primary user 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.

V. Anti-jamming 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 primary users 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 [22]. 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 primary users' action on a channel, the sender chooses k_a $(k_a \leq k_s)$ 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 to sense and jam based on the jamming strategy he is inclined to use.

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 primary user's activity 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 primary user'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 the primary users' strategy or activity is s_p. From the receiver's perspective, $s_s \bullet s_p \bullet s_j$ can be looked as as a joint decision made by the sender, the primary user 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 at timeslot t the sender, primary user 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* "$g_{f,t} = 0$" 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, no *reward* is obtained.

Case 2: A packet is received on f. If the packet fails to pass the verification (*i.e.*, jamming based DoS attack), no *reward* is obtained. We use efficient message verification schemes in [17] (*e.g.*, erasure coding combined with short signatures) for packet verification and message reassembly purpose.

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 [16] that accurate *differentiation* of packet errors due to jamming from errors due to weak links can be realized by looking at the received signal strength during bit reception. Here, we do not differentiate 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) [4], 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 [4], [5], [7] to design the opportunistic spectrum access protocol against various jamming scenarios.

We next define some notations used in the following discussion. In each timeslot $t \in \{1, \ldots, T\}$, the sender and receiver independently selects 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 at 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. The important notation used in this paper is summarized in Table I. **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

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

where the maximum is taken over all strategies available to the sender or 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 receiver chooses the best fixed set of channels for message reception. 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

Now we describe our proposed anti-jamming spectrum access protocol as shown in **Algorithm 2**. The algorithm computes two values: $\mathcal{A}^s$ 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. 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 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 channel rewards revealed to the sender and receiver:

$$\begin{aligned} \text{Sender:} \quad w^s_{f,t} &= w^s_{f,t-1} e^{\eta^s g^{s'}_{f,t}}, && (3) \\ \text{Receiver:} \quad w^r_{f,t} &= w^r_{f,t-1} e^{\eta^r g^{r'}_{f,t}}. && (4) \end{aligned}$$

The weight of a strategy is determined by the product of weights of all channels of the strategy and some random factors used for *exploration*:

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

where $g^{s'}_{i,t} = \sum_{f \in i} g^{s'}_{f,t}$ and $g^{r'}_{i,t} = \sum_{f \in i} g^{r'}_{f,t}$. 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 receiver, respectively. R_t is a Bernoulli random variable with $\mathbf{P}\{R_t = 1\} = \varepsilon$.

At the beginning of each timeslot, the sender and 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 [3]. The *covering strategy* $\mathcal{C}^s$ 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 $|\mathcal{C}^r| = \lceil \frac{n}{k_r} \rceil$. Note that we use *rewards* instead of *losses* in both our notations and analysis, as we are interested in the number of successful packet reception attempts instead of delay loss in the shortest path problem [7].

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 spectrum sensor point is usually chosen by letting the operating point be the constraint on the probability of the collision with primary users [22]. (Here for simplicity, we assume the two types of sensing errors *false alarm* probability and *miss detection* probability are the same and denote it as *sensing error probability* τ 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 $\widehat{G}_t^s$ and $\widehat{G}_t^r$ 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 compared to 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. Another benefit of this is to make the sender's strategy more unpredictable to the adversary.

VI. Performance Analysis

Definition 1: An algorithm $\mathcal{A}$ is α-*static* (*adaptive*, respectively) approximation if and only if (1) *Static* (*adaptive*, respectively) optimal solution can transmit a message successfully with high probability (w.h.p) $1 - \frac{1}{l^{\epsilon}}$ in time T, where constant $\epsilon > 0$. (2) Algorithm $\mathcal{A}$ can transmit the message successfully in time αT with the same probability $1 - \frac{1}{l^{\epsilon}}$.

Definition 2: The *regret* of an algorithm $\mathcal{A}$ is the reward difference 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 $G_T^{\mathcal{A}} = \sum_{s=1}^{T} g_{I_s,s} = \sum_{s=1}^{T} \sum_{f \in I_s} g_{f,s}$. The strategy I_s is chosen randomly according to some distribution 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

TABLE I: A summary of important notation.

Symbol	Definition
n	# of orthogonal channels
k_s	# of channels for sending at each timeslot
k_r	# of channels for receiving at each timeslot
k_j	# of jamming channels at each timeslot
l	# of packets for transmission
N^s	# of strategies at the sender side
N^r	# of strategies at the receiver side
I_t^s	sender's chosen strategy at timeslot t
I_t^r	receiver's chosen strategy at timeslot t
i	a strategy in the strategy set
f	channel entry (index) in a strategy vector
$g_{f,t}^s$	sender's reward for channel f at timeslot t
$g_{f,t}^r$	receiver's reward for channel f at timeslot t
$g_{i,t}^s$	sender's reward for strategy i at timeslot t
$g_{i,t}^r$	receiver's reward for strategy i at timeslot t
$G_{i,t}$	reward for strategy i up to timeslot t
$\widehat{G}_t^s$	total rewards over sender's chosen strategies up to timeslot t
$\widehat{G}_t^r$	total rewards over receiver's chosen strategies up to timeslot t
T	# of timeslots (rounds)
$\mathcal{C}^s$	sender's covering set
$\mathcal{C}^r$	receiver's covering set

we discussed above, the secondary sender changes its strategy based on the joint decision made by the primary user, the jammer and the receiver while the secondary receiver changes its strategy based on the joint decision made by the primary user, the jammer and the sender. Due to the probabilistic strategy selection at the sender and the receiver, the jointly decisions for them are different, which results 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 reward of the static optimal strategy for the sender and receiver, respectively.

Due to the probabilistic strategy selections, the secondary sender and receiver are not synchronized at 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 $O(1/\sqrt{T})$

Algorithm 2 An Anti-jamming Multi-channel Access Protocol with Unknown Traffic Statistics

```
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 $w^s_{i,0} = 1$ $(w^r_{i,0} = 1)$ $\forall i \in [1,N]$, 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 $t = 1, 2, \dots, T$
    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 $g^{s'}_{f,t}$ $(g^{r'}_{f,t})$ $\forall f \in [1,n]$ 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
```

as $T \to \infty$, with probability at least $1 - \delta$. By using dynamic programming, the sensing and access algorithm has time complexity $O(k_s n T)$ and space complexity $O(k_s n)$. The receiving algorithm has time complexity $O(k_r n T)$ and space complexity $O(k_r n)$.

Proof:

We first prove that at the receiver side, with probability at least $1-\delta$, the *regret* $G^{max}_T(r) - G^{\mathcal{A}^r}_T$ 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 $T \geq \max\{\frac{k_r}{n}\ln\frac{n}{\delta}, 4n\ln n\}$.

Now We introduce some notations for performance analysis: $G_{i,T} = \sum_{t=1}^{T} g_{i,t}$ and $G'_{i,T} = \sum_{t=1}^{T} g'_{i,t}$ for all $1 \leq i \leq N$, where $G_{i,T}$ ($G'_{i,T}$) denotes the total gain (virtual gain, respectively) of strategy i in T timeslots, and $G_{f,T} = \sum_{t=1}^{T} g_{f,t}$ and $G'_{f,T} = \sum_{t=1}^{T} g'_{f,t}$ for all $1 \leq f \leq n$, where $G_{f,T}$ ($G'_{f,T}$) denotes the total gain (virtual gain, respectively) on channel f in T timeslots. The relation between gain and virtual gain is derived as follows.

The proof is applicable for any fixed f. For any $u > 0$ and $c > 0$, by the Chernoff bound, we have $\mathbb{P}[G_{f,T} > G'_{f,T} + u] \leq e^{-cu}\mathbb{E}[e^{c(G_{f,T}-G'_{f,T})}]$. Let $u = \ln\frac{n}{\delta}/\beta$ and $c = \beta$, we get $e^{-cu}\mathbb{E}[e^{c(G_{f,T}-G'_{f,T})}] = \frac{\delta}{n}\mathbb{E}[e^{\beta(G_{f,T}-G'_{f,T})}]$. So it suffices to prove that $e^{\beta(G_{f,T}-G'_{f,T})} \leq 1$ for all T. Let $Z_t = e^{\beta(G_{f,t}-G'_{f,t})}$. By showing that $\mathbb{E}[Z_t] \leq Z_{t-1}$ for all $t \geq 2$ and $\mathbb{E}[Z_1] \leq 1$, It suffices to prove that for any $\delta \in (0,1)$, $0 \leq \beta < 1$ and $1 \leq f \leq n$,

$$\mathbb{P}[G_{f,T} > G'_{f,T} + \frac{1}{\beta}\ln\frac{n}{\delta}] \leq \frac{\delta}{n} \tag{11}$$

Note that, in the following proofs, we use a superscript in η, γ, β to differentiate the sender and receiver. However, for ease of exposition, we do not differentiate the other notations since they are independent in the proofs for the sender and the receiver.

Now We prove the bound of regret by using the quantity $\ln\frac{W_T}{W_0}$ as following. First of all, we have the lower bound by definitions $\ln\frac{W_T}{W_0} = \ln\sum_{i=1}^{N} e^{\eta^r G'_{i,T}} - \ln N \geq \eta^r \max_{1\leq i\leq N} G'_{i,T} - \ln N$. Then we derive the upper bound as follows: $\eta^r g'_{i,t} = \eta^r \sum_{f\in i} g'_{f,t} \leq \eta^r \sum_{f\in i} \frac{1+\beta^r}{q_{f,t}} \leq \frac{\eta^r k_r (1+\beta^r)|\mathcal{C}|}{\gamma^r} \leq 1$, where the second inequality follows because $q_{f,t} \geq \frac{\gamma^r}{|\mathcal{C}|}$ for all f by definition.

Using the fact that $e^x \leq 1 + x + x^2$ for all $x \leq 1$, for all $t = 1, 2, \cdots, T$ we have $\ln\frac{W_t}{W_{t-1}} = \ln\sum_{i=1}^{N} \frac{w_{i,t-1}}{W_{t-1}} e^{\eta^r g'_{i,t}} \leq \ln(\sum_{i=1}^{N} \frac{w_{i,t-1}}{W_{t-1}}(1+\eta^r g'_{i,t} + (\eta^r)^2 g'^2_{i,t})) \leq \ln(1+\sum_{i=1}^{N} \frac{p_{i,t}}{1-\gamma^r}(\eta^r g'_{i,t} + (\eta^r)^2 g'^2_{i,t})) \leq \frac{\eta^r}{1-\gamma^r}\sum_{i=1}^{N} p_{i,t} g'_{i,t} + \frac{(\eta^r)^2}{1-\gamma^r}\sum_{i=1}^{N} p_{i,t} g'^2_{i,t}$. The above inequalities hold using the fact that $\sum_{i=1}^{N} p_{i,t} \leq 1-\gamma^r$ and inequality $\ln(1+x) \leq x$ for all $x > -1$.

Let $\mathcal{N}$ denote the strategy set $\{1, \ldots, N\}$. On the one hand, we have $\sum_{i=1}^{N} p_{i,t} g'_{i,t} = \sum_{i=1}^{N} p_{i,t} \sum_{f\in i} g'_{f,t} = \sum_{f=1}^{n} g'_{f,t} \sum_{i\in\mathcal{N}:f\in i} p_{i,t} = \sum_{f=1}^{n} g'_{f,t} q_{f,t} = g_{I_t,t} + n\beta^r$. On the other hand, $\sum_{i=1}^{N} p_{i,t} g'^2_{i,t} = \sum_{i=1}^{N} p_{i,t}(\sum_{f\in i} g'_{f,t})^2 \leq \sum_{i=1}^{N} p_{i,t} k_r \sum_{f\in i} g'^2_{f,t} = k_r \sum_{f=1}^{n} g'^2_{f,t} \sum_{i\in\mathcal{N}:f\in i} p_{i,t} = k_r \sum_{f=1}^{n} g'^2_{f,t} q_{f,t} \leq k_r(1+\beta^r)\sum_{f=1}^{n} g'_{f,t}$, which holds the fact that $g'_{f,t} \leq \frac{1+\beta^r}{q_{f,t}}$ (**Note** that for clearly differentiating the *regret* bounds for the sender and the receiver, in the derivation we loose the bounds a little bit by choosing k_r instead of $\min\{k_r, k_s\varepsilon(1-\tau), n-k_j\}$.). Therefore, $\ln\frac{W_t}{W_{t-1}} \leq \frac{\eta^r}{1-\gamma^r}(g_{I_t,t} + n\beta^r) + \frac{(\eta^r)^2 k_r (1+\beta^r)}{1-\gamma^r}\sum_{f=1}^{n} g'_{f,t}$.

Summing for $t = 1, \cdots, T$, we have the following inequality $\ln\frac{W_T}{W_0} \leq \frac{\eta^r}{1-\gamma^r}(\widehat{G}_T + n\beta^r T) + \frac{(\eta^r)^2 k_r(1+\beta^r)}{1-\gamma^r}\sum_{f=1}^{n} G'_{f,T} \leq \frac{\eta^r}{1-\gamma^r}(\widehat{G}_T + n\beta^r T) + \frac{(\eta^r)^2 k_r (1+\beta^r)}{1-\gamma^r}|\mathcal{C}| \max_{1\leq i\leq N} G'_{i,T}$ Note that $\widehat{G}_T$ is the expected total gain of our algorithm in T time slots. Combining the upper bound with the lower bound, we have $\widehat{G}_T \geq (1-\gamma^r -$

$\eta^r k_r(1+\beta^r)|\mathcal{C}|)\max_{1\le i\le N} G'_{i,T} - \frac{1-\gamma^r}{\eta^r}\ln N - n\beta^r T$.

Applying (11), we can have that, with probability at least $1-\delta$, $\widehat{G}_T \ge (1-\gamma^r-\eta^r k_r(1+\beta^r)|\mathcal{C}|)(\max_{1\le i\le N} G_{i,T} - \frac{k_r}{\beta^r}\ln\frac{n}{\delta}) - \frac{1-\gamma^r}{\eta^r}\ln N - n\beta^r T$. Here, we used the fact $1-\gamma^r-\eta^r k_r(1+\beta^r)|\mathcal{C}| > 0$ which follows the assumptions of the theorem.

By doing some transpositions and using the following fact $\max_{1\le i\le N} G_{i,T} \le Tk_r$, we have $\max_{1\le i\le N} G_{i,T} - \widehat{G}_T \le (\gamma^r+\eta^r(1+\beta^r)k_r|\mathcal{C}|)Tk_r + (1-\gamma^r-\eta^r(1+\beta^r)k_r|\mathcal{C}|)\frac{k_r}{\beta^r}\ln\frac{n}{\delta} + +\frac{1-\gamma^r}{\eta^r}\ln N + n\beta^r T$ with probability at least $1-\delta$. Let $K = \min\{k_s, n-k_j, k_r\}$. Since $\hat{G_T} = KT - \hat{L_T}$ and $\max_{1\le i\le N} G_{i,T} = KT - \min_{1\le i\le N} L_{i,T}$, we have

$$\hat{L_T} \le KT(\gamma^r+\eta^r(1+\beta^r)k_r|\mathcal{C}|) + (1-\gamma^r-\eta^r(1+\beta^r)k_r|\mathcal{C}|)\min_{1\le i\le N} L_{i,T}$$
$$+(1-\gamma^r-\eta^r(1+\beta^r)k_r|\mathcal{C}|)\frac{k_r}{\beta^r}\ln\frac{n}{\delta} + \frac{1-\gamma^r}{\eta^r}\ln N + n\beta^r T$$

with probability $1-\delta$. Simplify above inequality, we can get

$$\hat{L_T} - \min_{1\le i\le N} L_{i,T} \le k_r T\gamma^r + 2\eta^r T k_r n + \frac{k_r}{\beta^r}\ln\frac{n}{\delta} + \frac{1-\gamma^r}{\eta^r}k_r\ln n + n\beta^r T$$

with probability $1-\delta$

Setting $\beta^r = \sqrt{\frac{k_r}{nT}\ln\frac{n}{\delta}}$ *and* $\gamma^r = 2\eta^r k_r|\mathcal{C}|$, we can get $\max_{1\le i\le N} G_{i,T} - \widehat{G}_T \le 4\eta^r T {k_r}^2|\mathcal{C}| + \frac{\ln N}{\eta^r} + 2\sqrt{k_r nT\ln\frac{n}{\delta}}$ which holds with probability $1-\delta$ if $T \ge \frac{k_r}{n}\ln(\frac{n}{\delta})$. Finally, using the facts $|\mathcal{C}| = \lceil\frac{n}{k_r}\rceil$ and $N \le n^{k_r}$. and setting $\eta^r = \sqrt{\frac{\ln N}{4k_r^2 T|\mathcal{C}|}}$, we prove that $\max_{1\le i\le N} G_{i,T} - \widehat{G}_T \le 6k_r\sqrt{Tn\ln n}$ with probability $1-\delta$.

Similarly, at the sender side we first show the connection between the true and the estimated cumulative rewards. The only difference is that the computation of estimated channel rewards is involved with a random variable ϵ. We prove that 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 $T \ge \max\{\frac{k_s\ln^2\frac{2n}{\delta}}{\varepsilon n\ln n}, \frac{n\ln\frac{2n}{\delta}}{k_s}, 4n\ln n\}$. Finally, as $G_T^{\mathcal{A}^s} = G_T^{\mathcal{A}^r}$, $|G_T^{max}(s) - G_T^{max}(r)| \le 6k_r\sqrt{Tn\ln n} + 14k_s\sqrt{\frac{Tn\ln n}{\varepsilon}} \le \frac{20k}{\sqrt{\varepsilon}}\sqrt{Tn\ln n}$, 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 $\min\{k_r, k_s\varepsilon(1-\tau), n-k_j\}$. Hence, sensing error probability τ does not appear in the final results.

We next show that by using dynamic programming both the sender's sensing and access algorithm and the receiver's receiving algorithm can be efficiently implemented with time complexity which is linear to n and k_s (k_r). We prove it for the receiver side, and the proofs for the sender side is similar.

In the proposed algorithm, step 1 and 2 are time consuming since the total number of possible strategies is $N = O(n^{k_r})$. In this proof, we show that the time complexity can be reduced by using dynamic programming. Let $S(\bar{f}, \bar{k})$ denote the strategy set in which each strategy chooses $\bar{k}$ channels from channel $\bar{f}, \bar{f}+1, \cdots, n$. We also use $\bar{S}(\bar{f}, \bar{k})$ to denote the strategy set in which each strategy chooses $\bar{k}$ channels from channel $1, 2, \cdots, \bar{f}$. We define $W_t(\bar{f}, \bar{k}) = \sum_{i \in S(\bar{f}, \bar{k})} \prod_{f \in i} w_{f,t}$ and $\bar{W}_t(\bar{f}, \bar{k}) = \sum_{i \in \bar{S}(\bar{f}, \bar{k})} \prod_{f \in i} w_{f,t}$. Note $W_t(\bar{f}, \bar{k}) = W_t(\bar{f}+1, \bar{k}) + w_{\bar{f},t} W_t(\bar{f}+1, \bar{k}-1)$ and $\bar{W}_t(\bar{f}, \bar{k}) = \bar{W}_t(\bar{f}-1, \bar{k}) + w_{\bar{f},t} W_t(\bar{f}-1, \bar{k}-1)$, which implies both $W_t(\bar{f}, \bar{k})$ and $\bar{W}_t(\bar{f}, \bar{k})$ can be computed in time $O(k_r n)$ (letting $W_t(\bar{f}, 0) = 1$, $W(n+1, \bar{k}) = \bar{W}(0, \bar{k}) = 0$) by dynamic programming for all $1 \le \bar{f} \le n$ and $1 \le \bar{k} \le k_r$.

In step 1, a strategy should be drawn from N strategies. Instead of drawing a strategy, we choose channel for the strategy one by one. Assume we make decision on each channel one by one in increasing order of their indices, *i.e.*, we first decide whether channel 1 should be chosen or not, and channel 2, and so on. For any channel f, if $k \le k_r$ channels has been chosen in channel $1, \cdots, f-1$, we choose channel f with probability $\frac{w_{f,t-1} W_{t-1}(f+1, k_r-k-1)}{W_{t-1}(f, k_r-k)}$ and we do not choose channel f with probability $\frac{W_{t-1}(f+1, k_r-k)}{W_{t-1}(f, k_r-k)}$. Let $w(f) = w_{f,t-1}$ if channel f is chosen in the strategy i; 0 otherwise. $w(f)$ is the weight of f in the total weight of the strategy. In our algorithm, $w_{i,t-1} = \prod_{f=1}^{n} w(f)$. Let $c(f) = 1$ if channel f is chosen in the strategy i; 0 otherwise. $\sum_{f=1}^{\bar{f}} c(f)$ denotes the number of channels chosen among channels $1, 2, \cdots, \bar{f}$ in strategy i. In this implementation, the probability that a strategy i is chosen is $\prod_{\bar{f}=1}^{n} \frac{w(\bar{f}) W_{t-1}(\bar{f}+1, k_r - \sum_{f=1}^{\bar{f}} c(f))}{W_{t-1}(\bar{f}, k_r - \sum_{f=1}^{\bar{f}-1} c(f))} = \frac{\prod_{\bar{f}=1}^{n} w(\bar{f})}{W_{t-1}(1, k_r)} = \frac{w_{i,t-1}}{W_{t-1}}$. The probability is exactly same as that in Algorithm **??**, which implies the correctness of this implementation.

Note in this implementation, we do not maintain the total weight of each strategy $w_{i,t}$. So we cannot compute $q_{f,t}$ as we described in step 2 of our algorithm. The probability $q_{f,t}$ can be computed within $O(n)$ as follows $q_{f,t} = (1-\gamma) \frac{\sum_{k=0}^{k_r-1} \bar{W}_{t-1}(f-1, k) w_{f,t-1} W_{t-1}(f+1, k_r-k-1)}{W_{t-1}(1, k_r)}$
$+ \gamma \frac{|\{i \in \mathcal{C} : f \in i\}|}{|\mathcal{C}|}$ for each round.

■

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 can guarantee the message can be 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 what kinds of packets have been received successfully. Therefore, in our protocol, every time the sender want to send a packet, he will pick up a "new" 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.

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 $(1+\delta+c\epsilon)l - 6k_r\sqrt{(1+\delta+\epsilon)Tn\ln n}$. 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 $(1+\epsilon)l$. 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 $l \geq 36\frac{n^3 \ln n K(1+c\epsilon)}{k_s(n-k_j)(c-1)^2\epsilon^2}$, 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 $(1+\epsilon)l$. So the probability to reconstruct the message is at least $1 - \frac{1}{l^\epsilon}$. ■

Discussion. Notice that the parameters β, η and γ is 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. Compute ϵ such that $1 - \frac{1}{l^\epsilon} = P$. Find a feasible constant $c > 1$ such that $l = 36(1 + \delta + c\epsilon)\underline{k_r} n \ln n/(c-1)^2\epsilon^2$. The total time of transmission will be $T = (1 + \delta + c\epsilon)l/(\underline{k_r}\frac{k_s\varepsilon(1-\tau)}{n}\frac{n-\overline{k_j}}{n})$. Theorem 2 can guarantee that the receiver will obtain the message with probability at least P.

VII. 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 served as the theoretical efficiency upper bound in our simulation.

In our simulation, the sender uses MAB-based channel sensing and access strategy and the receiver uses MAB-based channel receiving strategy; the primary user dynamically occupies and vacates the spectrum obeying certain traffic statistics (we assume $p^i_{11} > p^i_{01}$); 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 primary user dynamically uses the spectrum and the receiver chooses MAB-based strategy. Without loss of generality, we assume the sender and 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. 4 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 $p^{11}_i > p^{01}_i$. Fig. 4 (a), (c), (e), (g) under different jamming strategies, *static opt* and *adaptive opt* always remain close to each other, especially when static jamming is adopted. That implies that the primary user's dynamics lead to a seemly "static " channel availabilities from the secondary user's perspective, so the *adaptive opt* cannot gain much more than the *static opt*. 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 $T = 250$ for the receiver to obtain the whole message 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 ACKs) 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 less number of received packets, and the larger sensing probability will help to improve the performance as the sender can refine his strategy distributions with the sensing results. In Fig. 7, 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 reduces when τ increases. However, it is *surprised* to find that when adaptive and myopic jamming occurs the system performance improves as τ increases. This phenomenon can be explained by the fact that larger sensing error probability can help to ”disrupt” the adaptive and myopic jammers' prediction on the available channels.

In Fig. 8, Fig. 9 and Fig. 10, we use the setting "mab myo dyn mab" as an example to show how the parameter n and l affect the system performance. Fig. 8 shows that when l increases (*i.e.*, from 10 to 30), the expected time to received the message w.h.p. increases correspondingly. On the other hand, different values of n will also affect performance as T increases. For example, see the circle point in Fig. 8 and Fig. 9. When $T < 180$, the case of $n = 8$ gives the best performance; After $T > 180$, the case of $n = 10$ outperforms that of $n = 8$; When the time reaches $T = 240$, the case of $n = 14$ outperforms the case of $n = 8$ and it gives the best performance after $T = 320$. That means that it is better to choose a small n when the message size is short; a larger n is preferred when the message size is relatively large. However, it dose not imply that the larger n will always give the best performance. As shown in Fig. 10, when n increases from 12 to 14, the performance gain is very small, and when n further increases to $n = 16$, the performance is deteriorated. This is because the use of a large n also makes it

T Jamming stra.		800	1200	1600	2000	2400	2800	3200	3600	4000
Static	$\tau = 0.1, \epsilon = 1$	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	$\tau = 0.1, \epsilon = 1$	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	$\tau = 0.1, \epsilon = 1$	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	$\tau = 0, \epsilon = 1$	**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 II: Convergence of the secondary sender and receiver's strategy probability distributions. The Euclidian distance of the two parties' strategy probability distribution is measured under $p_i^{11} > p_i^{01}$, $n = 8$.

difficult for the sender and receiver to hop to the same set of channels.

B. Convergence Evaluation

As T increases, the sender and receiver will converge to their static optimal strategies through the online learning, respectively. In Section VI, 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 $w_{f,t}^s$ and $w_{f,t}^r$ as the closeness of them indicates that they are approaching the best strategies. In Table II, we show the Euclidian distance of the two parties' strategy probability distribution under different jamming scenarios with $p_i^{11} > p_i^{01}$, $n = 8$.

The bold numbers in the table indicate the start of distance decrease at certain time. In general, it is shown that the sender and receiver's perceptions about the channels converges the fastest under static jamming, and the worst performance is obtained in the case of myopic jamming; The performance 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. That 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.

VIII. Conclusion

In this paper, we study the design of anti-jamming mechanism in cognitive radio networks. We formulate 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 sender and receiver to hop to the same set of available channels with high probability. 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 simulation are conducted to validate the theoretical analysis and show that the proposed protocol is very effective and resilient against various jamming attacks.

References

[1] 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*, 55(9):4040–4050, 2009.

[2] W. Arbaugh. Improving the latency of the probe phase during 802.11 handoff. online at www.umiacs.umd.edu/partnerships/ltsdocs/Arbaug_talk2.pdf.

[3] 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*, pages 322–331, 1995.

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

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

[6] J. C. Gittins. Bandit processes and dynamic allocation indices. *Journal of the Royal Statistical Society Series B Methodological*, 41:148–177, 1979.

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

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

[9] A. Kalai and S. Vempala. Efficient algorithms for online decision problems. In *Proc. of COLT'03*, pages 26–40, 2003.

[10] K. Liu and Q. Zhao. A restless bandit formulation of multi-channel opportunistic access: Indexablity and index policy. *IEEE Transactions on Information Theory*, 56(11):5547–5567, 2010.

[11] K. Liu, Q. Zhao, and B. Krishnamachari. Dynamic multichannel access with imperfect channel state detection. *IEEE Transactions on Signal Processing*, 58(5):2795–2808, 2010.

[12] H. B. McMahan and A. Blum. Online geometric optimization in the bandit setting against an adaptive adversary. In *COLT*, 2004.

[13] O. Mehanna, A. Sultan, and H. E. Gamal. Cognitive mac protocols for general primary network models. *CoRR*, abs/0907.4031, 2009.

[14] T. Shu and M. Krunz. Throughput-efficient sequential channel sensing and probing in cognitive radio networks under sensing errors. In *MobiCom'09*, pages 37–48, 2009.

[15] 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.

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

[17] M. Strasser, C. Pöpper, and S. Capkun. Efficient uncoordinated fhss anti-jamming communication. In *Prob. of ACM MobiHoc'09*, July 2009.

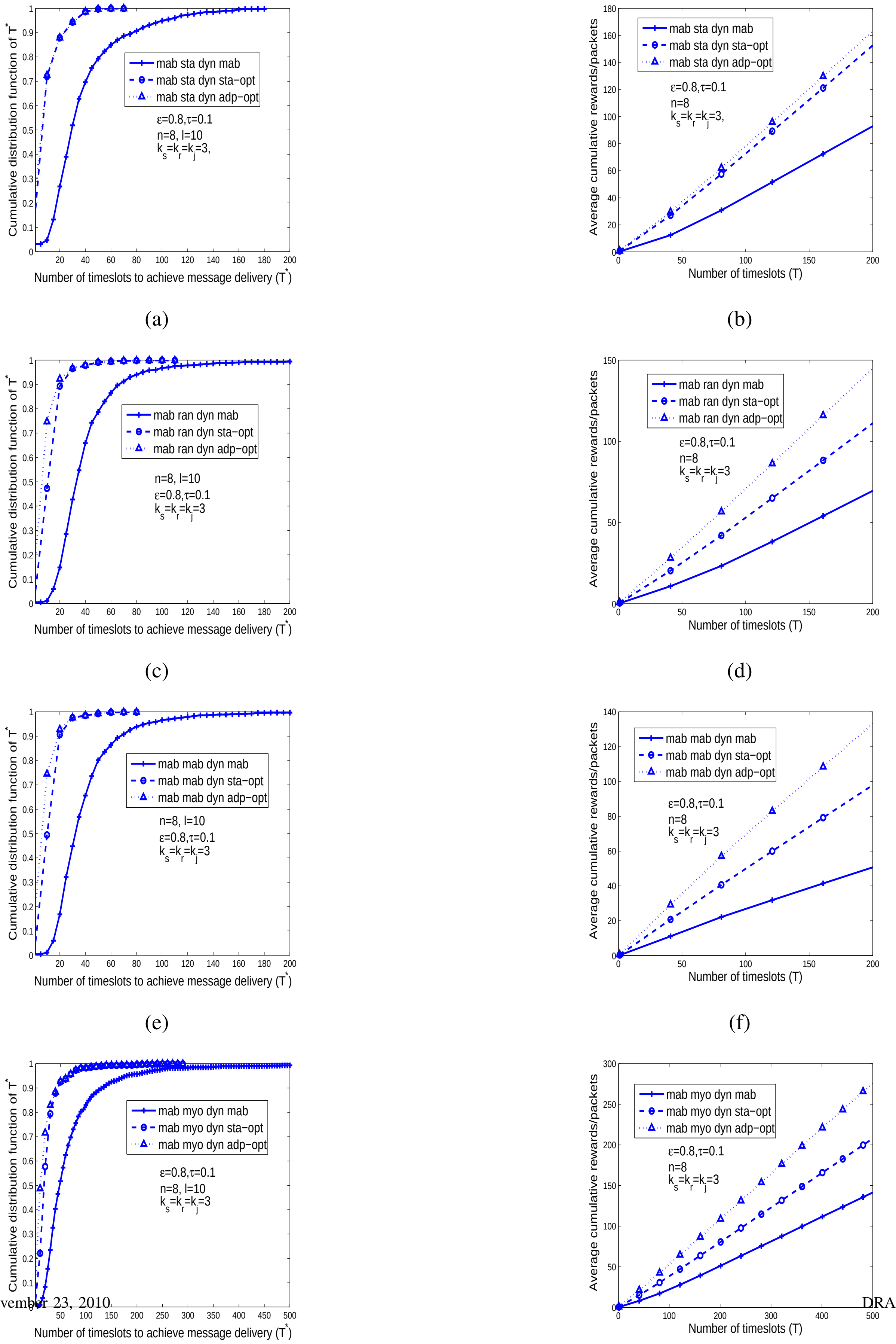
[18] M. Strasser, C. Pöpper, S. Capkun, and M. Cagalj. Jamming-resistant key establishment using uncoordinated frequency hopping. In *Proc. of IEEE Security and Privacy*, May 2008.

[19] J. Unnikrishnan and V. V. Veeravalli. Algorithms for dynamic spectrum access with learning for cognitive radio. *IEEE Transactions on Signal Processing*, 58(2):750–760, 2010.

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

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

[22] 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*, 25(3):589–600, 2007.


Cumulative distribution function of T*
Number of timeslots to achieve message delivery (T*)
mab sta dyn mab
mab sta dyn sta−opt
mab sta dyn adp−opt
ε=0.8,τ=0.1
n=8, l=10
k_s=k_r=k_j=3,
Average cumulative rewards/packets
Number of timeslots (T)
mab sta dyn mab
mab sta dyn sta−opt
mab sta dyn adp−opt
ε=0.8,τ=0.1
n=8
k_s=k_r=k_j=3,
mab ran dyn mab
mab ran dyn sta−opt
mab ran dyn adp−opt
n=8, l=10
ε=0.8,τ=0.1
k_s=k_r=k_j=3
mab ran dyn mab
mab ran dyn sta−opt
mab ran dyn adp−opt
ε=0.8,τ=0.1
n=8
k_s=k_r=k_j=3
mab mab dyn mab
mab mab dyn sta−opt
mab mab dyn adp−opt
n=8, l=10
ε=0.8,τ=0.1
k_s=k_r=k_j=3
mab mab dyn mab
mab mab dyn sta−opt
mab mab dyn adp−opt
ε=0.8,τ=0.1
n=8
k_s=k_r=k_j=3
mab myo dyn mab
mab myo dyn sta−opt
mab myo dyn adp−opt
ε=0.8,τ=0.1
n=8, l=10
k_s=k_r=k_j=3
mab myo dyn mab
mab myo dyn sta−opt
mab myo dyn adp−opt
ε=0.8,τ=0.1
n=8
k_s=k_r=k_j=3
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
20
40
60
80
100
120
140
160
180
200
50
150
250
300
350
400
450
500

(a)

(b)

(c)

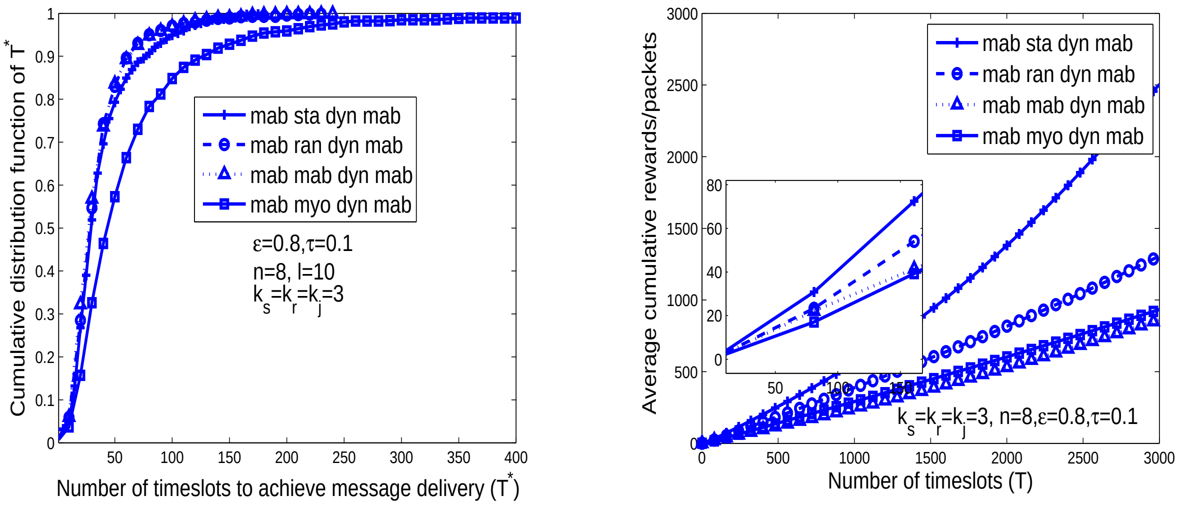
(d)

(e)

(f)

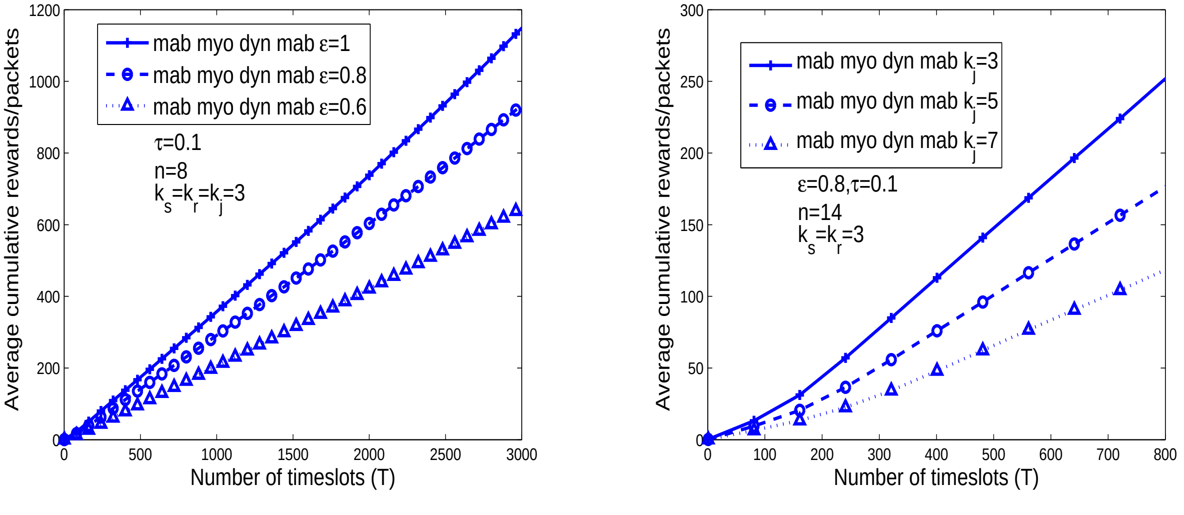
(g)

(h)



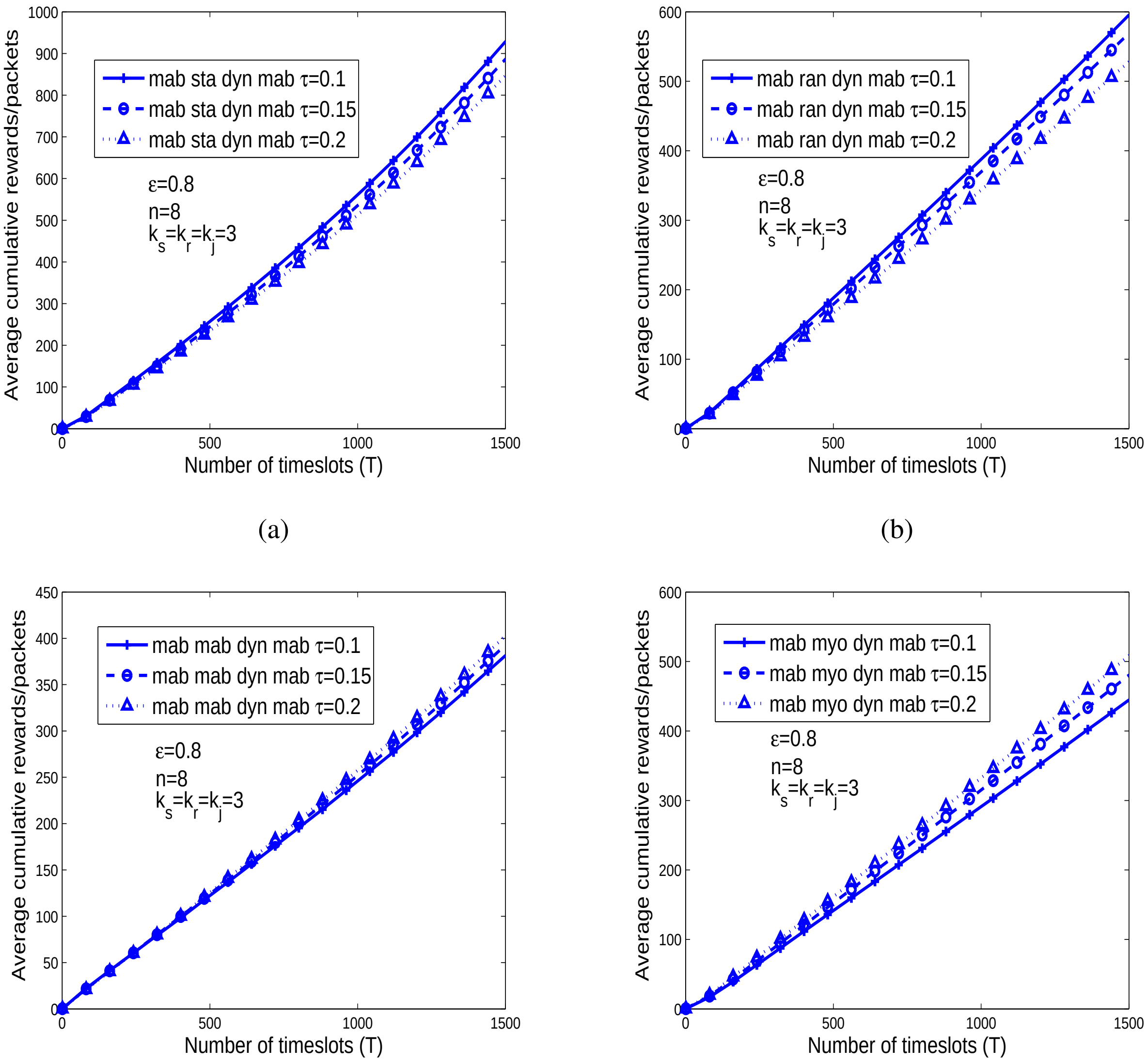
(a) (b)

Fig. 5: The comparisons of the 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".



(a)

(b)

(c)

(d)

Fig. 7: The effects of sensing error probability τ on the system performance.

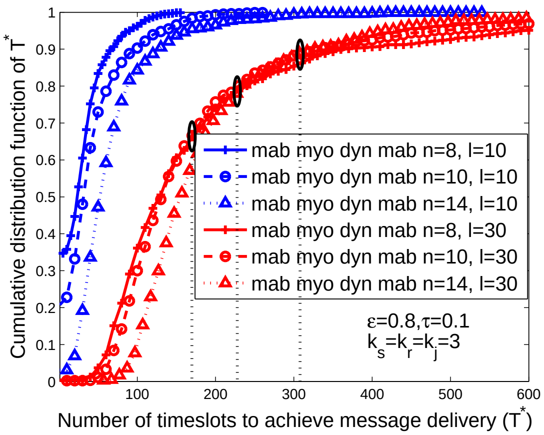


Fig. 8: The effects of n and l on the system performance with respect to the CDF of the expected time to achieve message delivery.

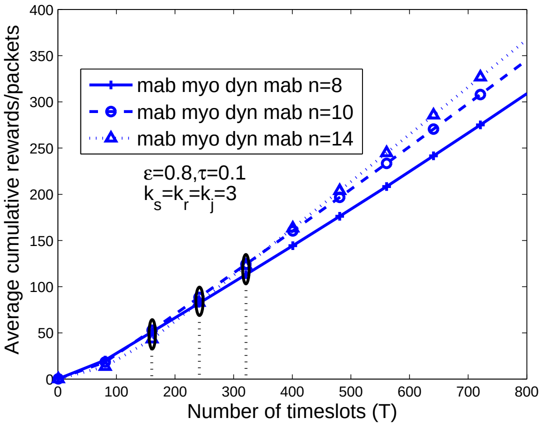


Fig. 9: The effect of n on the system performance with respect to the average cumulative rewards/packets.

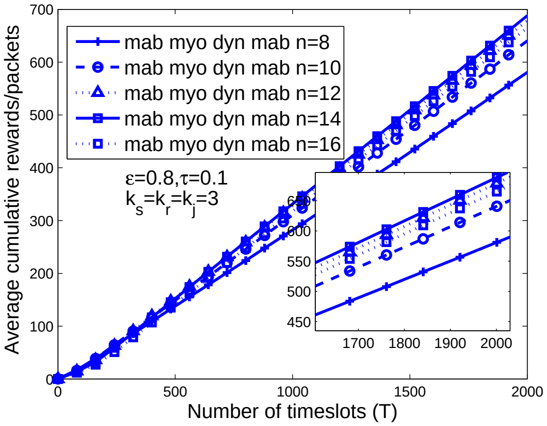


Fig. 10: The effect of n on the system performance with respect to the average cumulative rewards/packets.