LEARNING-STAGE BASED DECENTRALIZED ADAPTIVE ACCESS POLICY FOR DYNAMIC SPECTRUM ACCESS

Marjan Zandi and Min Dong

Department of Electrical Computer and Software Engineering
University of Ontario Institute of Technology, Ontario, Canada
Email: {marjan.zandi, min.dong}@uoit.ca

ABSTRACT

We consider the problem of decentralized online learning and channel access in a cognitive radio network. Based on an existing distributed access policy proposed in [1], named the ρ^{RAND} policy, we propose an adaptive decentralized access policy in which the distributed coordination among secondary users is adjusted at different stages of learning accuracy of the primary network. Specifically, we exploit a "perceived population" by each secondary user to reduce collision events at different learning stages. We design a metric that measures the level of learning accuracy and use that as an indicator to adjust the "perceived population" by each secondary user. Simulations show that our proposed adaptive policy improves the leading constant of the normalized regret and can provide substantial improvement over the ρ^{RAND} policy.

***Index Terms*—** Opportunistic Spectrum Access, Decentralized Multi-Armed Bandit, Cognitive Radio, Adaptive Learning

1. INTRODUCTION

Designing dynamic spectrum access to efficiently utilize the spectrum is one of the main objectives in cognitive radio networks. A hierarchical cognitive radio network consists of licensed primary users for accessing the spectrum and the secondary users who opportunistically use the idle channels that are not occupied by the primary users. Since the channel availability statistics of the primary network are typically unknown to the secondary users (SUs), they rely on limited spectrum sensing to search for idle channels and make decisions based on observation histories for channel access. In designing a distributed policy for spectrum access among SUs, where there is no information exchange or access arrangement among users, the challenges involved not only include online learning of the primary channel statistics using local sensing observations, but also the distributed mechanism to resolve collisions among SUs.

Assume a cognitive radio network with N independent primary channels and M SUs, where $N \geq M$. For centralized scheduling of users' access, the problem can be formulated as the classical Multi-Armed Bandit (MAB) problem [2]–[4]. The throughput loss over time due to learning of the primary channel statistics as compared to the ideal case with known channel statistics is measured by *regret*. The minimum growth of regret over time under an efficient learning algorithm is characterized in [3], and is shown to have a logarithm growth over time. For distributed access by the SUs, the problem formulation can be viewed as the decentralized MAB problem. Motivated by dynamic spectrum access, decentralized policies among multiple players are proposed in [1], [5], [6]. These policies use different mechanisms to achieve "coordination" among SUs to orthogonalize their access to the M-best primary channels. They all achieve logarithmic growth of regret, which are order-optimal. Note that the efficiency of a learning algorithm is measured not only by the asymptotic growth rate of regret, but also by the scaling constant of the growth rate. All aforementioned decentralized policies are order-optimal with a logarithm growth rate of regret. However, they perform differently in terms of the scaling constant. Thus, further improvement should be with respect to the improvement on the scaling constant. In this work, we aim at improving the scaling constant of the growth rate by designing an access policy that is adaptive to different learning stages.

In this paper, we design an adaptive decentralized access policy for spectrum access. In particular, we focus on modifying the ρ^{RAND} policy proposed in [1] which is a very simple distributed learning and access policy requiring least amount of coordination among users. By noticing that the learning accuracy of the primary channels affects the access collision statistics, we adapt the distributed access coordination among SUs at different stages of learning accuracy. Specifically, we exploit a "perceived population" by each SU to reduce collision events at different learning stages. We design a metric that measures the level of learning accuracy and use that as an indicator to adjust the "perceived population" by each SU. Simulations show that our proposed adaptive policy improves the scaling constant of the normalized regret and can provide substantial improvement over the ρ^{RAND} policy.

2. NETWORK MODEL

Assume M SUs independently searching for the idle channels among the N channels licensed to a slotted primary network in a cognitive radio network. We assume that $M \leq N$. The availability state of the i^{th} channel in the primary network at slot n is denoted as $X_i(n)$, where $X_i(n) = 1$, if channel i is available at time slot n, and 0 otherwise. The availability statistic for each channel i in the primary network is assumed to evolve as an i.i.d. Bernoulli random process over n, with the mean $\theta_i \in [0, 1]$, *i.e.,* $X_i(n) \sim$ Bernoulli(θ_i), $\forall n$, where $\theta_i = E[X_i(n)]$, for $i = 1, \cdots, N$. We assume θ_i's are distinct to each other and are unknown to the SUs. We denote the mean channel availability vector as $\boldsymbol{\theta} \triangleq [\theta_1, \theta_2, ..., \theta_N]^T$. We assume M is known to all SUs, and channel sensing at all SUs is perfect. At the beginning of each slot n, each SU selects a channel to sense, and if available, it will access the channel. Using the sensing outcome and observation history, the SUs learn the mean channel availability $\boldsymbol{\theta}$ over time.

For distributed spectrum access among SUs, we assume a collision model for multiple access: If more than one SU access the same channel, it will result in failed transmissions and zero

throughput for all users involved. Otherwise, the sole SU accessing a primary channel will receive a unit throughput.

The total throughput obtained up to slot n using a learning and access policy is given by $\sum_{i=1}^{N}\sum_{j=1}^{M}\theta_i E[S_i^j(n)]$, where $S_i^j(n)$ denotes the number of times, up to current slot n, that SU j is the only user to sense channel i. In an ideal scenario, where $\boldsymbol{\theta}$ is known to users, the total throughput up to slot n is $n\sum_{k=1}^{M}\theta_{k^*}$, where k^* represents index of the k^{th} highest element in $\boldsymbol{\theta}$. For any learning and access policy, regret is a metric used to measure the throughput loss of a given policy due to learning. It is defined as the difference over throughput between the ideal scenario and a given policy, *i.e.,*

$$R(n,\boldsymbol{\theta},M) \triangleq n\sum_{k=1}^{M}\theta_{k^*} - \sum_{i=1}^{N}\sum_{j=1}^{M}\theta_i E[S_i^j(n)]. \quad (1)$$

Considering the above model, the design objective is to device a decentralized policy that minimizes the regret, with no exchange of information among the SUs. For a distributed access policy, each SU will select a channel to access based on its own estimate of the mean availability of channels.

3. DECENTRALIZED SPECTRUM ACCESS POLICIES

The UCB1 algorithm proposed in [7] is a sample-mean based index policy for the single user learning and access. The existing decentralized policies proposed in [1], [5], [6] are considered as the extensions of the UCB1 algorithm to the distributed case. In UCB1 algorithm, channels are ranked at each SU using a statistic called g-statistic. Let $T_i^j(n)$ denote the number of times that the SU j senses channel i up to time slot n. If SU j selects channel i to sense at time slot n, then it obtains the value of $X_i(n)$ and records this value as $X_i^j(T_i^j(n))$. Let $\mathbf{X}_i^j(n) \triangleq [X_i^j(1),\cdots,X_i^j(T_i^j(n))]^T$ be the vector holding the sensing observation history of channel i up to time slot n at SU j. Using $\mathbf{X}_i^j(n)$, SU j estimates θ_i of channel i at time n as

$$\hat{\theta}_i^j(T_i^j(n)) \triangleq \frac{1}{T_i^j(n)}\sum_{k=1}^{T_i^j(n)} X_i^j(k). \quad (2)$$

The g-statistic at SU j for channel i is defined as

$$I_i^j(n) \triangleq \hat{\theta}_i^j(T_i^j(n)) + \sqrt{\frac{2\log n}{T_i^j(n)}}. \quad (3)$$

It will be used as an index for SU j to rank the channels. In the single user case ($M=1$), using the above index, the user selects the channel with the highest index at time n. For multiple SUs, each user computes its own index vector $\mathbf{I}^j(n) \triangleq [I_1^j(n),\cdots,I_N^j(n)]^T$ based on its own observation history. Then, each user will select the channel with the k^{th} highest ranking in $\mathbf{I}^j(n)$ to access. Distributed learning policies [1], [5], [6] propose different mechanisms for access coordination to ensure that the SUs choose different channels but within the first M-highest indexed channels. Among these policies, the ρ^{RAND} policy in [1] is a very simple distributed policy, requiring the least amount of coordination among users and it is order-optimal. We aim to modify the ρ^{RAND} policy using adaptive learning to improve the performance. We briefly review the ρ^{RAND} policy below:

1) *Select channel to sense and access*: At slot n, each SU j obtains its ranking vector $\mathbf{I}^j(n)$. It then selects the r_j^{th} best channel among the M-best channels to sense, where r_j is drawn from a uniform distribution: $r_j \overset{i.i.d.}{\sim} \text{Uniform}(M)$. Let $\sigma(r_j,\mathbf{I}^j(n))$ be the channel index of the r_j^{th} highest rank in $\mathbf{I}^j(n)$. If the channel is available, then SU j accesses the channel.
2) *Reselect channel under collision*: Each SU j uses an acknowledgement for collision feedback. SU j will redraw its rank $r_j \sim \text{Uniform}(M)$ only if there is collision in the previous transmission. Otherwise, it will keep using the rank r_j generated previously to determine which channel to access.

4. AN ADAPTIVE LEARNING POLICY BASED ON PERCEIVED POPULATION

To measure the efficiency of a learning algorithm, we need to consider both the asymptotic growth rate of regret, denoted as $r(n)$, and the scaling constant of the growth rate, $\lim_{n\to\infty} R(n,\boldsymbol{\theta},M)/r(n)$. It has been shown in [2] that an efficient learning algorithm for centralized MAB problems should have a logarithm growth rate of regret. All aforementioned decentralized policies are order-optimal with a logarithm growth rate of regret. What is unclear is how they perform in terms of the scaling constant. This differentiates the performance among these existing decentralized algorithms. Note that all the existing proposed policies rely on the exact knowledge of the secondary network population M to resolve collisions among SUs. We aim at improving the scaling constant of the growth rate by designing a learning policy that exploits a "perceived population" by each SU.

Define $U_j(n)$ to be a "perceived population" at SU j, *i.e.,* what the user perceives to be the current population. The user will use this parameter in determining the primary channel to access. The "perceived population" is adaptive over time as a function of M: $U_j(n) = f(M,n)$. Note that using $U_j(n) \neq M$ for learning and access is equivalent to having an inaccurate estimate of M of the secondary network population, and in the long run, will lead to a linear growth rate of regret [8]. However, in the short time, it can improve the performance by reducing collision events. Fig. 1 shows an example of the impact of the population overestimation on the performance of the ρ^{RAND} policy, where $U_j(n) = M+1, \forall j,n$. We see an improvement of the regret during the transient behavior at the early time slots.

To understand this behavior, we note that each SU learns the mean channel availabilities $\boldsymbol{\theta}$ over time for access decision in a decentralized fashion. There are two types of events contribute to the regret $R(n,\boldsymbol{\theta},M)$: 1) *Not choosing M-best channels*: the channel i that SU j accesses has the mean availability θ_i that is not among the top M highest ones in $\boldsymbol{\theta}$; 2) *Collision among SUs*: distributed access results in collision and thus unsuccessful transmissions for all colliding users. Although the two types of events are correlated, the coordination mechanism in a decentralized access policy directly affects the type 2 event. Note that the SUs are more prone to collision at the beginning. This is because the estimate of the mean channel availability $\hat{\theta}_i^j(T_i^j(n))$ in (2) is very inaccurate, resulting in the channel ranking in $\mathbf{I}^j(n)$ varies over time. In other words, the k^{th} highest ranking in $\mathbf{I}^j(n)$ maps to different channel indexes more frequently. For SUs j_1 and j_2 selecting the channels which have the k_1^{th} and k_2^{th} ranks among M-highest values in $\mathbf{I}^j(n)$, they may collide in the next time slot, even though they do not collide in the current slot. At this stage, if we relax the constraint of selecting among the M-best channels to

among the U-best channels, where $U > M$, it potentially decreases the collision among the SUs and also increases the chance of selecting one of the true M-best channels. This is equivalent to use a larger "perceived population" $U_j(n) > M$ at SU j. As demonstrated in Fig. 1, by allowing larger perceived population, the regret improves at early time slots.

However, as the learning of $\boldsymbol{\theta}$ improves over time, the channel ranking in $\mathbf{I}^j(n)$ becomes more accurate and stable. Once two users select two different channels for access, they are likely to stay on the respective selected channels and remain collision free. In this case, using larger perceived population will increase the probability of selecting the channels outside of the M-best channels and hence has a negative impact on the throughput. Therefore, at this stage, it is necessary to use the true population as the "perceived population" for each SU.

Based on this analysis on the transient and long-term behavior, we propose an adaptive learning algorithms which adapt the "perceived population" $U_j(n)$ at each SU j to the different stages of learning of primary channel statistics to improve both the short-term and long-term regrets.

The main challenge in designing the adaptive learning algorithm is to determine the switching point for $U_j(n)$. We propose a thresholding method in determining $U_j(n)$. Let O_M be the set of indexes of the true M-best channels,

$$O_M = \left\{ i_m : \theta_{i_m} \in \{\theta_{(1)}, \cdots, \theta_{(M)}\},\ 1 \le m \le M \right\} \quad (4)$$

where $\{\theta_{(i)}\}$ is the ordered statistics of $\{\theta_i\}$ with $\theta_{(1)} > \cdots > \theta_{(N)}$. Let $\widehat{O}_M^j(n)$ denote the set of indexes of the empirical M-best channels for SU j at time slot n,

$$\widehat{O}_M^j(n) = \Big\{ i_m : \hat{\theta}_{i_m}^j(T_i^j(n)) \in \left\{ \hat{\theta}_{(1)}^j(T_i^j(n)), \cdots, \hat{\theta}_{(M)}^j(T_i^j(n)) \right\},\ 1 \le m \le M \Big\}. \quad (5)$$

Now we denote $\delta_W^j(n)$ as the average number of estimated M-best channels in common during a window period W for SU j given by

$$\delta_W^j(n) = \frac{\sum_{i=1}^{W} \left| \widehat{O}_M^j(n) \cap \widehat{O}_M^j(n-i) \right|}{W}, \quad n \ge W \quad (6)$$

where $0 \le \delta_W^j(n) \le M$. Denote the normalized version of $\delta_W^j(n)$ as $\bar{\delta}_W^j(n) = \delta_W^j(n)/M$. Denote $\Delta^j(n)$ as the cumulative moving average of $\bar{\delta}_W^j(n)$ as

$$\Delta^j(n) = \frac{\sum_{n'=W}^{n} \bar{\delta}_W^j(n')}{n - W} \quad (7)$$

where we have $0 \le \Delta^j(n) \le 1$. This quantity can be computed recursively based on the current $\bar{\delta}_W^j(n)$ and previous $\Delta^j(n-1)$ without the need to store all the history data as

$$\Delta^j(n) = \frac{1}{n-W}\bar{\delta}_W^j(n) + \frac{(n-W-1)}{n-W}\Delta^j(n-1), \quad \text{for } n \ge W. \quad (8)$$

As the estimate of $\boldsymbol{\theta}$ improves over time, the difference between $\widehat{O}_M^j(n)$ and O_M reduces. Thus, $\Delta^j(n)$ indicates the level of accuracy of the empirical M-best channels to the true M-best channels. In other words, the metric $\Delta^j(n)$ provides a measure of the learning accuracy over time.

The value of $\Delta^j(n)$ will be tested against thresholds $\{\tau_k\}$ to determine the switching point for $U_j(n)$, where $k = 1, \cdots, K$ and K indicates the total number of switching points used. We summarize the main steps of the modified ρ^{RAND} policy with adaptive learning and access, named Rand-ALC(K), below. Detailed description is shown in Algorithm 1.

1) Start with $U_j(n) = M + K$.
2) Compute $\Delta^j(n)$: At every time slot n, each SU j obtain $\Delta^j(n)$.
3) Update $U_j(n)$: If $\Delta^j(n) \ge \tau_k$, then the SU j set $U_j(n) = U_j(n-1) - 1$, where τ_k is the current threshold used, and $1 \le k \le K$; otherwise, keep $U_j(n) = U_j(n-1)$.
4) Run the ρ^{RAND} policy (randomized access over the $U_j(n)$-best channel)

As we will see in simulations, using Rand-ALC(1) with $K = 1$ and single threshold τ is already effective in improving the throughput performance and regret.

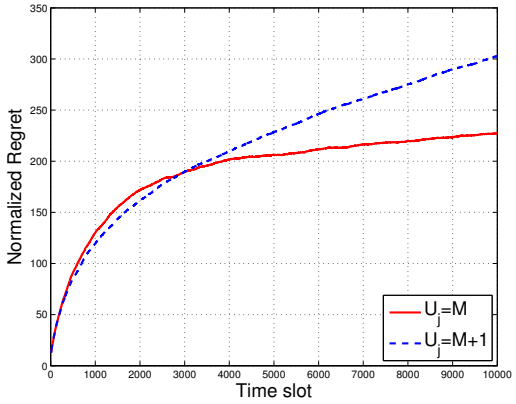

Fig. 1. Normalized regrets $\frac{R(n,\boldsymbol{\theta},M)}{\log n}$ under the ρ^{RAND} policy using "perceived population" U_j, $\boldsymbol{\theta} = [0.1, 0.2, ..., 0.9]$, M = 4, N = 9.

5. SIMULATION RESULTS

In this section, we present the simulation results obtained by the proposed policy. We assume a cognitive radio network with $N = 9$ channels and $M = 4$ SUs. The channel availability $X_i(n)$ follows i.i.d. Bernoulli random process, for $i = 1, \cdots, N$.

To demonstrate how the metric $\Delta^j(n)$ reflects the level of learning accuracy, in Fig. 2, we plot the trajectory of the averaged $\Delta^j(n)$ over time. We set the mean channel availability randomly as $\boldsymbol{\theta} = [0.3, 0.34, 0.5, 0.6, 0.67, 0.91, 0.2, 0.8, 0.7]^T$, and window size $W = 10$. We fix $U_j(n)$ value over time, and let each user implements the ρ^{RAND} policy with the "perceived population" $U_j(n)$, where $U_j(n) = M$, $M+1$, or $M+2$. As we see, the trajectory of averaged $\Delta^j(n)$ shows two stages of learning at different rates, the initial learning with much faster rate of improvement, and then switched to a slower learning speed. In addition, we see that the rate of learning is not sensitive to the variation of $U_j(n)$.

In Fig. 3, we compare the normalized regret $R(n, \boldsymbol{\theta}, M)/\log n$ under the proposed Rand-ALC(1) policy and the ρ^{RAND} policy. We also compare them with Rand-ALC$^{\text{gen}}$(1), a genie-aided policy where we use normalized regret curve under fixed $U_j(n) = M$ and $U_j(n) = M+1$ (e.g. in Fig. 1) to find the switching time n_{sw}

Algorithm 1 : Rand-ALC(K) policy for SU j

1) **Input**:
   n: Current time slot
   M: Number of SUs
   N: Number of channels
   T: Horizon length
   W: Window Size
   τ_k, $k = 1, \cdots, K$: Threshold
   $U_j(n)$: Perceived population of SU j at time slot n
   $\Delta^j(n)$: Level of learning accuracy of the empirical to the true M-best channels
2) **Init**: Sense each channel once
   $n \leftarrow N+1$, $U_j(n) \leftarrow M+K$, $\Delta^j(n) \leftarrow 0$, $k \leftarrow K$;
3) **Start Loop** $n \leftarrow n+1$
   i) Update $U_j(n)$: If $\frac{\Delta^j(n)}{M} \geq \tau_k$, then $U_j(n) = U_j(n-1) - 1$, $k = k-1$; otherwise, $U_j(n) = U_j(n-1)$;
   ii) Run ρ^{RAND} policy (randomized access over $U_j(n)$-best channel);
   iii) Obtain $\widehat{O}^j_M(n)$: Set of indexes of empirical M-best channels for SU j at time slot n;
   iv) Compute $\delta^j_W(n)$ as in (6), and compute $\bar{\delta}^j_W(n)$;
   v) Update $\Delta^j(n)$ as in (8).
   Stop Loop when $n = T$.

for $U_j(n)$ from $M+1$ to M to produce a lower regret. The same $\boldsymbol{\theta}$ value as in Fig. 2 is used in Fig. 3. We see that, our proposed policy with threshold $\tau = 0.98$ substantially outperforms the ρ^{RAND} policy in both transient and long-term behavior. Over 30% improvement is seen in long-term normalized regret, which indicates the improved scaling constant of the growth of regret. The performance of our proposed policy also tracks that of the genie-aided Rand-ALC$^{\text{gen}}$(1) policy very closely.

Similar to the experiment above, Fig. 4 shows the normalized regret under a different mean channel availability statistics, where $\boldsymbol{\theta} = [0.51, 0.52, \cdots, 0.59]^T$, *i.e.,* very similar mean statistics among the channels. As can be seen, our proposed policy again substantially outperforms the ρ^{RAND} policy (20% improvement) and approaches the genie-aided Rand-ALC$^{\text{gen}}$(1) policy.

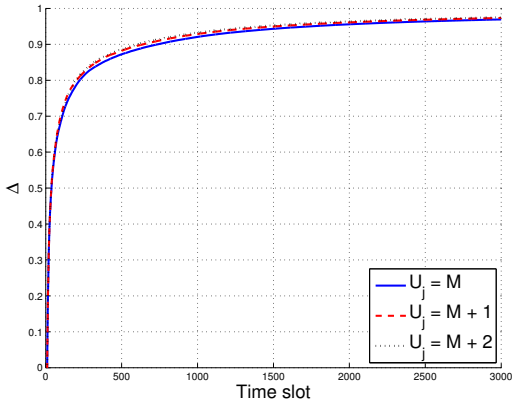

Fig. 2. Average $\Delta^j(n)$ vs. time slot n. ($W = 10$, $\boldsymbol{\theta} = [0.3, 0.34, 0.5, 0.6, 0.67, 0.91, 0.2, 0.8, 0.7]$, M = 4, N = 9)

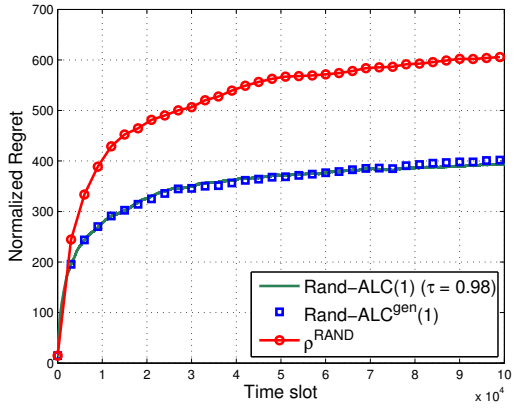

Fig. 3. Normalized regrets $\frac{R(n,\boldsymbol{\theta},M)}{\log n}$ vs. time slot n. ($\boldsymbol{\theta} = [0.3, 0.34, 0.5, 0.6, 0.67, 0.91, 0.2, 0.8, 0.7]$, M = 4, N = 9)

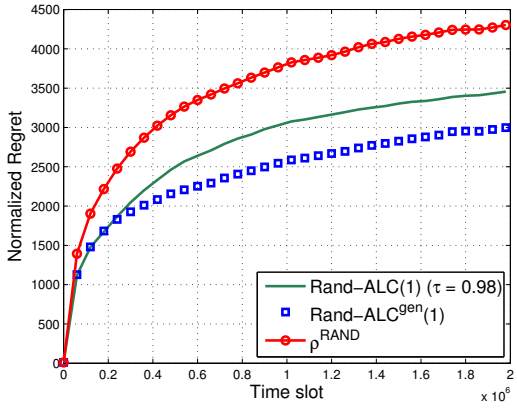

Fig. 4. Normalized regrets $\frac{R(n,\boldsymbol{\theta},M)}{\log n}$ vs. time slot n ($\boldsymbol{\theta} = [0.51, 0.52, ..., 0.59]$, M = 4, N = 9).

6. CONCLUSION

We consider the problem of decentralized online learning and channel access in a cognitive radio network. Based on the existing distributed access policy, the ρ^{RAND} policy, we propose an adaptive decentralized access policy Rand-ALC(K). It adjusts the distributed coordination mechanism among SUs by adaptively changing the "perceived population" at each SU to reduce collisions at different learning accuracy stages. We design a metric that measures the level of learning accuracy and use that as an indicator to adjust the "perceived population" by each SU. Simulations show that our proposed adaptive policy improves the scaling constant of the normalized regret and can provide substantial improvement over the ρ^{RAND} policy.

7. REFERENCES

[1] A. Anandkumar, N. Michael, K. Tang, and A. Swami, "Distributed algorithms for learning and cognitive medium access with logarithmic regret," *IEEE J. Select. Areas Commun.*, vol. 29, no. 4, pp. 731–745, Apr. 2011.

[2] T. L. Lai and H. Robbins, "Asymptotically efficient adaptive allocation rules," *Advances in Applied Mathematics*, vol. 6, pp. 4–22, 1985.

[3] V. Anantharam, P. Varaiya, and J. Walrand, "Asymptotically efficient allocation rules for the multiarmed bandit problem with multiple plays- part i: I.i.d. rewards," *IEEE Trans. Autom. Control*, vol. 32, pp. 968–976, Nov. 1987.

[4] ——, "Asymptotically efficient allocation rules for the multiarmed bandit problem with multiple plays- part ii: Markovian rewards," *IEEE Trans. Autom. Control*, vol. 32, pp. 977–982, Nov. 1987.

[5] K. Liu and Q. Zhao, "Distributed learning in multi-armed bandit with multiple player," *IEEE Trans. Signal Processing*, vol. 58, no. 11, pp. 5665–5681, Nov. 2010.

[6] Y. Gai, B. Krishnamachari, and M. Hsieh, "Decentralized online learning algorithms for opportunistic spectrum access," in *Proc.IEEE Global Telecommn. Conf. (GLOBECOM)*, Dec. 2011.

[7] P. Auer, N. Cesa-Bianchi, and P. Fisher, "Finite-time analysis of the multiarmed bandit problem," *Machine learning*, vol. 47, pp. 235–256, 2002.

[8] M. Zandi and M. Dong, "Distributed opportunistic spectrum access with unknown population," in *Proc. IEEE International Conference on Communications in China (ICCC)*, Aug. 2012.