Master's Thesis

Greedy Confidence Bound Techniques for Restless Multi-armed Bandit Based Cognitive Radio

Restless Multi-armed Bandit에 기반한 인지 라디오시스템을 위한 Greedy 신뢰도 분석 기법

August 2013

Graduate School of Seoul National University
Department of Electrical Engineering and Computer Science
Dong Shuyan
董 书炎

Greedy Confidence Bound Techniques for Restless Multi-armed Bandit Based Cognitive Radio

Restless Multi-armed Bandit에 기반한 인지 라디오시스템을 위한 Greedy 신뢰도 분석 기법

Advisor: Jungwoo Lee

Submitting a master's thesis of Electrical Engineering and Computer Science

June 2013

Graduate School of Seoul National University
Department of Electrical Engineering and Computer Science

Dong Shuyan
董 书炎

Confirming the master's thesis written by _____

June 2013

Chair_____

Vice Chair_____

Examiner_____

Abstract

Greedy Confidence Bound Techniques for Restless Multi-armed Bandit Based Cognitive Radio

The electromagnetic radio spectrum is a natural resource, the use of which by transmitters and receivers is licensed by governments. The underutilization of the electromagnetic spectrum leads us to think in terms of spectrum holes, where a spectrum hole is a band of frequencies assigned to a primary user, but, at a particular time and specific geographic location, the band is not being utilized by that user. Spectrum utilization can be improved significantly by making it possible for a secondary user (who is not being serviced) to access a spectrum hole unoccupied by the primary user at the right location and the time in question.

Cognitive radio is viewed as a novel approach for improving the utilization of a precious natural resource: the radio electromagnetic spectrum. The cognitive radio, built on a software-defined radio, is defined as an intelligent wireless communication system that is aware of its environment and uses the methodology of understanding-by-building to learn from the environment and adapt to statistical variations in the input stimuli, with two primary objectives in mind: Highly reliable communication whenever and wherever needed; Efficient utilization of the radio spectrum.

Multi-armed bandit (MAB) problems are a class of sequential resource allocation problems, which has fundamental conflict between a strategy yielding high present reward and a strategy sacrificing present gain for better future reward. In a multi-armed bandit problem, there are multiple (N) arms which

generate stochastic reward, and a player seeks a policy to select multiple $K \geq 1$ arms in order to maximize the expected total reward over multiple time-slots.

A particularly challenging variant of MAB problems is the restless multi-armed bandit problem (RMAB), in which all arms evolve as Markov chains. Even in the Bayesian case, where the parameters of the Markov chains are known, this problem is difficult to solve, and has been proved to be PSPACE hard. We consider more challenging non-Bayesian RMAB problems, in which the parameters of the Markov chain are further assumed to be unknown a priori. We demonstrate our approach on a practical problem related to dynamic spectrum sensing for cognitive radio applications. If the primary user occupancy on each channel is modeled as an identical but independent Markov chain with unknown parameters, we obtain a non-Bayesian RMAB. Our main contribution in this work is that we develop an efficient new multi-channel spectrum sensing algorithm for unknown dynamic channels based on the two-slot greedy confidence bound algorithm (Two-slot GCB), which combines the Markov Chain parameter estimation and the channels sensing simultaneously and provides a new analysis which shows that UCB algorithms have the regret rate of ln(t). And finally we give out another solution for the accessing policy for Cognitive Radio of unconstrained continuous time Markov chain channel model.

Keywords: Cognitive Radio, MAB, Markov chain, Greedy Algorithm, Two-slot GCB

Student number: 2011-24072

Contents

List of Figures

Chapter 1

Introduction

1.1 Cognitive Radio (CR)

The electromagnetic radio spectrum is a natural resource, the use of which by transmitters and receivers is licensed by governments. In November 2002, the Federal Communications Commission (FCC) published a report prepared by the Spectrum-Policy Task Force, aimed at improving the way in which this precious resource is managed in the United States [1]. The task force was made up of a team of high-level, multidisciplinary professional FCC staff—economists, engineers, and attorneys—from across the commission's bureaus and offices. Among the task force major findings and recommendations, the second finding on page 3 of the report is rather revealing in the context of spectrum utilization:

"In many bands, spectrum access is a more significant problem than physical scarcity of spectrum, in large part due to legacy command-and-control regulation that limits the ability of potential spectrum users to obtain such access."

Indeed, if we were to scan portions of the radio spectrum including the revenue-rich urban areas, we would find that [2]-[4]:

1) Some frequency bands in the spectrum are largely unoccupied most of the time;
2) Some other frequency bands are only partially occupied;
3) The remaining frequency bands are heavily used.

The underutilization of the electromagnetic spectrum leads us to think in terms of spectrum holes, for which we offer the following definition [2]:

A spectrum hole is a band of frequencies assigned to a primary user, but, at a particular time and specific geographic location, the band is not being utilized by that user.

Spectrum utilization can be improved significantly by making it possible for a secondary user (who is not being serviced) to access a spectrum hole unoccupied by the primary user at the right location and the time in question. Cognitive radio [5], [6] is viewed as a novel approach for improving the utilization of a precious natural resource: the radio electromagnetic spectrum. The cognitive radio, built on a software-defined radio, is defined as an intelligent wireless communication system that is aware of its environment and uses the methodology of understanding-by-building to learn from the environment and adapt to statistical variations in the input stimuli, with two primary objectives in mind:

- Highly reliable communication whenever and wherever needed;
- Efficient utilization of the radio spectrum.

In this thesis, we focus on the second aspect, especially the channel accessing policy of secondary users.

1.2 MAB problem

Multi-armed bandit (MAB) problems are a class of sequential resource allocation problems, which has fundamental conflict between a strategy yielding high present reward and a strategy sacrificing present gain for better future reward [7]. In a multi-armed bandit problem, there are multiple (N) arms which generates stochastic reward, and a player seeks a policy to select multiple (K >= 1) arms in order to maximize the expected total reward over multiple time-slots. MAB problems can be generally classified into Bayesian and non-Bayesian

problems. The model becomes Bayesian when the statistical model/parameters of the reward process for each are known, and it becomes non-Bayesian when they are unknown. In the case of non-Bayesian MAB problems, the objective is to design an arm selection policy that minimizes regret, which is defined as the gap between the expected reward that can be achieved by a genie that knows the parameters, and the reward from the given policy.

A particularly challenging variant of MAB problems is the restless multi-armed bandit problem (RMAB) [8], in which all arms evolve as Markov chains. Even in the Bayesian case, where the parameters of the Markov chains are known, this problem is difficult to solve, and has been proved to be PSPACE hard [9]. One approach to this problem has been Whittle's index, which is asymptotically optimal under certain regimes. However it does not always exist. Even when it does, it is not easy to compute. It is recently shown that non-trivial tractable classes of RMAB where computable Whittle's index exists have been identified [10].

1.3 Combination

We consider more challenging non-Bayesian RMAB problems, in which the parameters of the Markov chain are further assumed to be unknown a priori. Our main contribution in this work is providing a novel approach that can produce higher reward than the UCB algorithms [11],[12] for arms with a markov chain reward process, and provides a new analysis which shows that UCB algorithms have the regret rate of ln(t). We demonstrate our approach on a practical problem related to dynamic spectrum sensing for cognitive radio applications. We consider a scenario where a secondary user must select one of N channels to sense at each time slot to maximize its expected reward.

If the primary user occupancy on each channel is modeled as an identical but independent Markov chain with unknown parameters, we obtain a non-Bayesian RMAB. Note that for RMAB with a known model staying with the best arm is suboptimal. Thus, the sub-linear regret is not appropriate because the deviation

from the maximum average reward can be arbitrarily large. In this thesis, we use accumulated reward instead of regret as the performance measure. We develop an efficient new multi-channel spectrum sensing algorithm for unknown dynamic channels based on the two-slot greedy confidence bound algorithm (Two-slot GCB), which combines the Markov Chain parameter estimation and the channels sensing simultaneously.

Chapter 2

MAB algorithms with a Markov Chain Reward Process

2.1 The multi-armed bandit problem

In probability theory, the multi-armed bandit problem is the problem a gambler faces at a row of slot machines, sometimes known as "one-armed bandits", when deciding which machines to play, how many times to play each machine and in which order to play them. When played, each machine provides a random reward from a distribution specific to that machine. The objective of the gambler is to maximize the sum of rewards earned through a sequence of lever pulls.

In practice, multi-armed bandits have been used to model the problem of managing research projects in a large organization, like a science foundation or a pharmaceutical company. Given its fixed budget, the problem is to allocate resource among the competing projects, whose properties are only partially known now but may be better understood as time passes.

In the early versions of the multi-armed bandit problem, the gambler has no initial knowledge about the levers. The crucial tradeoff the gambler faces at each trial is between "exploitation" of the lever that has the highest expected payoff and "exploration" to get more information about the expected payoffs of the other levers [13].

R_1 R_2 R_K

Figure 1: Slot machines

The multi-armed bandit (or just bandit for short) can be seen as a set of real distributions $B = \{R_1, \dots, R_K\}$, each distribution being associated with the rewards delivered by one of the K levers. Let $\mu_1, \dots, \mu_K$ be the mean values associated with these reward distributions. The gambler iteratively plays one lever per round and observes the associated reward. The objective is to maximize the sum of the collected rewards. The bandit problem is formally equivalent to a one-state Markov decision process. The regret ρ(T) after T rounds is defined as the difference between the reward sum associated with an optimal strategy and the sum of the collected rewards[14]:

$$\rho(T) = T\mu^* - \sum_{t=1}^{T} r_t \,, \mu^* = max_k\{\mu_k\}$$

Where μ^* is the maximal reward mean, and μ_k is the reward at time t. A strategy whose average regret per round ρ/T tends to zero with probability 1 when the number of played rounds tends to infinity is a zero-regret strategy. Intuitively, zero-regret strategies are guaranteed to converge to an optimal strategy, not necessarily unique, if enough rounds are played.

2.2 UCB algorithm for MAB

Upper Confidence Bound (UCB) Algorithm [11] is very simple but effective for MAB problem with reward generated independently for each

$$m_i(t) + \sqrt{\frac{2 * \ln(t)}{t_i}}$$

time slot (here we use time slot instead of round). And it can also be applied for MAB problem with a Markov Chain Reward process, where the generated reward for each time slot is related. For each time slot, the bandit arm we decide to play is chosen through calculating the following value of each arm shown as below:

Where $m_i(t)$ is channel i's mean reward by time t, and t_i is the times channel i is played by time t.

It's intuitive to choose the channel with the highest mean reward so $m_i(t)$ in the equation acts as the exploitation part of the channels. But we also need an exploration part to guarantee the judgment of the best channel (with highest mean reward) is right. So the second part of the equation $\sqrt{\frac{2*\ln(t)}{t_i}}$ acts as the exploration part.

2.3 A simple induction for UCB's regret rate of log(t)

Suppose there're just two arms 1 and 2, with mean reward m_1, m_2 and $m_1 > m_2$. The arm choosing criteria is calculated as below:

$$ucb_1(t) = m_1(t-1) + \sqrt{\frac{2 * \ln(t)}{k_1(t-1)}} \qquad (3\text{-}1)$$

$$ucb_2(t) = m_2(t-1) + \sqrt{\frac{2 * \ln(t)}{k_2(t-1)}}$$

Where m(t) is the reward mean at time t, k(t) is the number of plays of the arm after t time slots.

Suppose at time t^* (large enough), α is arm 1's played times percentage:

$$\begin{aligned} k_1(t^*) &= \alpha t^* \\ k_2(t^*) &= (1-\alpha)t^* \\ ucb_1(t^*) &= ucb_2(t^*) \end{aligned} \tag{3-2}$$

Then we can rewrite Equation (3-1):

$$\begin{aligned} ucb_1(t^*) &= ucb_2(t^*) \\ &= m_1(t^*-1) + \sqrt{\frac{2*\ln(t^*)}{\alpha t^*}} \\ &= m_2(t^*-1) + \sqrt{\frac{2*\ln(t^*)}{(1-\alpha)t^*}} \end{aligned} \tag{3-3}$$

Furthermore we get:

$$(m_1(t^*-1) - m_2(t^*-1))^2 = \frac{2*\ln(t^*)}{t^*} \ \frac{1}{1-\alpha} \ (1-\sqrt{\frac{1-\alpha}{\alpha}})^2 \qquad (3-4)$$

The regret $\rho(t^*)$ by time t^* is:

$$\begin{aligned} \rho(t^*) &= m_1 t^* - \left(m_1 t^* \alpha + m_2 t^* (1-\alpha)\right) \\ &= t^* (1-\alpha)(m_1 - m_2) \end{aligned} \tag{3-5}$$

From Equation (3-4) we have:

$$t^* (1-\alpha)(m_1(t^*-1) - m_2(t^*-1)) = \frac{(1-\sqrt{\frac{1-\alpha}{\alpha}})^2 \ln(t^*)}{m_1(t^*-1) - m_2(t^*-1)} \qquad (3-6)$$

As $t \to \infty$, we know $t^* \to \infty$ and $\alpha \to 1$ (explained later), then:

$$\rho(t^*) = t^* (1-\alpha)(m_1 - m_2)$$

$$= \frac{2*\ln(t^*)}{m_1 - m_2} \left(1 - \sqrt{\frac{1-\alpha}{\alpha}}\right)^2$$

$$= C\ln(t^*) \tag{3-7}$$

Where $C = \frac{(1-\sqrt{\frac{1-\alpha}{\alpha}})^2}{m_1 - m_2} = \frac{1}{m_1 - m_2}$ is a constant.

Here we suppose we have just 2 arms, in fact, we can do the same deduction for arbitrary number of arms by taking the other arms except the best one as an equivalent arm with an average mean, then the calculation still holds for these two special arms. Now we can conclude that the regret rate of UCB algorithm increases with a rate $\ln(t)$.

From the mathematic deduction, we notice that the $\ln(t)$ part in Equation (3-1) remains unchanged. So we can replace $\ln(t)$ with other terms.

From Equation (3-5) and (3-7) we have:

$$1 - \alpha = \frac{2*\ln(t^*)}{t^*} \quad \frac{1}{(m_1 - m_2)^2} \left(1 - \sqrt{\frac{1-\alpha}{\alpha}}\right)^2 \tag{3-8}$$

From Equation (3-8), as $t^* \to \infty$, $\alpha \to 1$ with a converging speed $\frac{\ln(t^*)}{t^*}$.

Then we have the idea to replace $\ln(t)$ with other terms which have a faster converging rate. But that doesn't mean the faster it converges, the higher total reward will be, since it should be guaranteed enough time to explore the arms. Obviously, $\frac{\ln(\ln(t))}{t}$ converges to 0 much faster than $\frac{\ln(t)}{t}$. For simplicity, the changed UCB algorithm by replacing $\ln(t)$ with $\ln(\ln(t))$ is named as Log-UCB algorithm and similarly we have an Exp-UCB by replacing $\frac{\ln(t)}{t}$ with e^{-t}.

Here are some simulation results for comparing UCB, Log-UCB and Exp-UCB algorithm under I.I.D arm reward distribution. The simulation environment settings are 10 arms, the number of players varies from 1 to 10 and each arm generates its reward according to a uniform distribution.

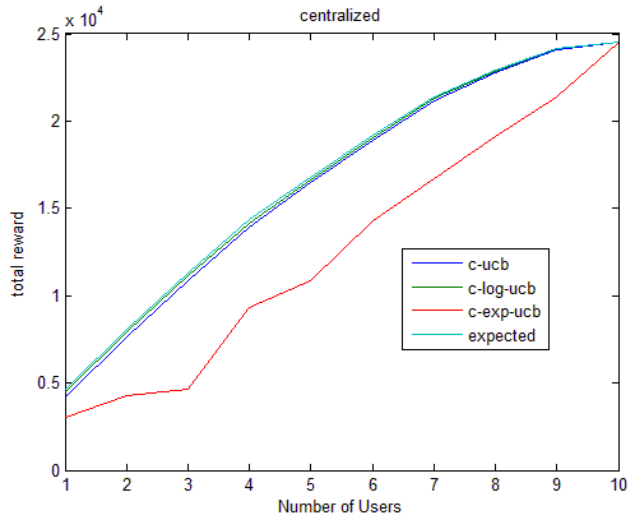

Figure 2: Centralized various UCB algorithms

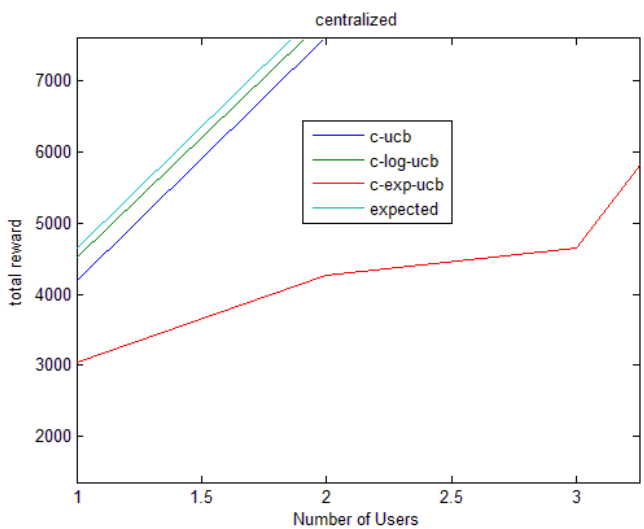

Figure 3: Zoom-in of Figure 2

In the simulation, expected means if there are N arms played for each time slot then the best N arms among the total K arms are played, namely the maximum reward. Judging from Figure 2, Log-UCB do outperforms UCB in aspect of total reward and the Exp-UCB's total reward is quite poor. The total rewards are relevant to both exploitation part and exploration part in the various UCB equations. If it converges to some arm too fast that means too less exploration, it may probably lead a biased estimation about the best arm.

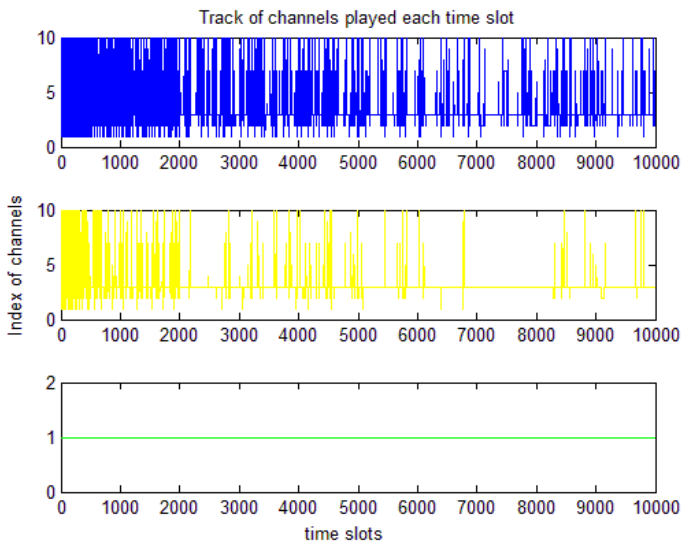

Figure 4: Track of channels played each time slot

Through Figure 4, we can see that for most of the time slots both UCB and Log-UCB stays on arm 4 (there's almost a horizontal line), which is the best arm among the 10 arms. Exp-UCB has too fast a converging speed and it mistook arm 1 as the best arm and stays in that arm (keeps exploiting that arm), due to too less exploration of the arms. The line of Log-UCB, representing exploitation of the best arm, is more obvious than that of UCB, meaning that in spite of less exploration but a faster converging into the best arm can still guarantee more exploitation and consequently a higher total reward.

2.4 CR with Markov Chain channel model and Greedy algorithm

2.4.1 Problem Modeling

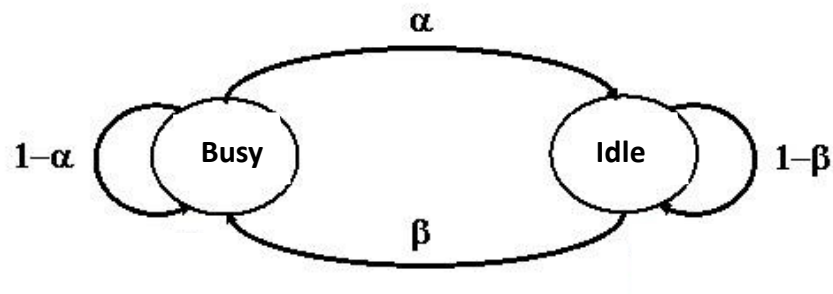

Figure 5: Markov chain model

The channels in Cognitive Radio communication system are viewed as the arms in MAB problem with two states: Busy (0) & Idle (1). The reward in MAB problem here refers to the channel utilization, namely when the channel chosen to access is busy the secondary user can get no reward (reward=0) while if it's idle then the user can get a reward (reward=1). The channel accessing decision problem in CR now is formed as a MAB problem, we aim to design a channel accessing algorithm to maximize the total reward.

2.4.2 Belief Vector based Greedy Algorithm

When the state When the state transition possibilities of each channel are pre-known and channels are independent, MAB problem with Markov Chain Reward Process can be specified as a Partially Observable Markov Decision Problem (POMDP), while Belief Vector based Greedy Algorithm [15] acts as an optimal solution to this problem.

Belief vector $\lambda(t) = [\lambda_1(t), \lambda_2(t), \lambda_3(t), \dots, \lambda_N(t)]$, means the user's expectation of each arm's reward in the next slot, which is updated according to:

$$\lambda_i(t) = \begin{cases} \lambda_i(t-1)*(1-\beta_i)+(1-\lambda_i(t-1))*\alpha_i, & \text{if a(t)} \neq \text{i} \\ 1-\beta_i, & \text{if a(t)=i,}\Theta_a\text{(t)=1} \\ \alpha_i, & \text{if a(t)=i,}\Theta_a\text{(t)=0} \end{cases}$$

a(t) is the index of the chosen channel to be played at time t; Θ_a(t) is the state of channel a(t). This equation calculates the idle possibility for each arm according to whether the channel is observed or not and whether the observed channel is Idle or Busy.

2.5 UCB algorithm for CR Markov Chain channel model

UCB algorithm can be applied directly to the MAB problem developed from CR Markov chain channel model even without any information about the channel state transition parameters because it can always view the two states Markov Chain as an equivalent binary distribution. Meanwhile, it is also due to its special dealing of the actual Markov Chain as a binary distribution, it cannot make use of the observations of the channels to make a more profitable decision of which channel to access. Its information about the channel's mean reward is more a historical long time estimation instead of real time estimation, namely we can utilize the current time slot's channel state observation result to estimate the possibility of this channel to be idle for the next time slot, that's the property of Markov Chain which is ignored by UCB algorithm.

2.6 Two and One slot algorithm for CR Markov Chain channel model

2.6.1 Two-slot Algorithm

Since Belief Vector based Greedy Algorithm is optimal to POMDP, we want to modify this algorithm so that it can still performs well without the requirement of the Markov state transition possibility. The intuitive idea is that we can calculate the belief vector for two slots $\lambda_i^{(2)}(t)$ instead of one, namely we calculate the idle possibility of each channel in the next two slots. Since each time we access one channel and stay in that channel for two slots, the observations can be used to do the channel state transition possibility estimation. Furthermore, if the current access channel is also chosen in last time then we can observe one more state transition to make a better estimation. Inspired by the exploitation and exploration parts in UCB algorithms, we also introduce the part $\sqrt{\frac{2*\ln(t)}{t_i}}$ to act as the channel exploration; finally we come to the following channel accessing algorithm *Two-slot Greedy Confidence Bound Algorithm* (TGCB):

$$\lambda_i^{(2)}(t) + 0.5 * m_i + \sqrt{\frac{2 * \ln(t)}{t_i}}$$

The updating of $\lambda_i^{(2)}(t)$ in TGCB is a little different from that of $\lambda_1(t)$ in greedy algorithm, shown as below:

$$\begin{cases} \text{if a(t-2)=i,} \\ \quad \text{if } \Theta_a\text{(t-1)=1, } \lambda_i(t) = 1-\beta_i; \\ \quad \text{if } \Theta_a\text{(t-1)=0, } \lambda_i(t) = \alpha_i; \\ \text{if a(t-2)} \neq \text{i,} \\ \quad \lambda_i(t-1) = \lambda_i(t-2)*(1-\beta_i) + (1-\lambda_i(t-2))*\alpha_i, \\ \quad\quad \lambda_i(t) = \lambda_i(t-1)*(1-\beta_i) + (1-\lambda_i(t-1))*\alpha_i; \end{cases}$$

The coefficient of 0.5 of the term m_i is an empirical parameter from simulation results.

TGCB is quite similar to UCB algorithms and in fact it is indeed equivalent to UCB algorithms: for the channel accessed, the mean reward (namely the possibility to be idle) will be the possibility of transiting from Idle to Idle if the channel's state is idle and be the possibility of transiting from Busy to Idle if the state is busy; for those channels not accessed, the mean reward will be the statistic mean reward. Namely, only those channels observed can provide real time estimation about the next time slot while those channels not observed can only make use of their average mean reward to do the estimation.

2.6.2 One-slot Algorithm

One-slot Greedy Confidence Bound Algorithm (OGCB), which is almost the same with TGCB except that it just calculate one slot belief vector (updating process is the same with Greedy algorithm) and how it completes

$$\lambda_i + 0.5 * m_i + \sqrt{\frac{2 * ln(t)}{t_i}}$$

parameter estimation. It has an expression as below:

In the simulation, we tract the channel chosen to access for each time slot and find that the user will stay in the same channel for several slots in series, which can be used to do the transition possibility estimation.

2.6.3 Combination of Two-slot and One-slot algorithms

The difference between Two-slot and One-slot algorithms mainly lies in the updating calculation of the belief vector, is there any possibility that we can unify them into one algorithm?

In fact, there do exit some method to combine these two algorithms. During each calculation for selection of channels to access, firstly we calculate the belief vector for One-slot and Two-slot, and then we compare these two vectors and choose the larger one, the slot is chosen accordingly. Here below is the mathematic expression:

$$\lambda(t) = [\lambda_1(t), \lambda_2(t), \lambda_3(t), \dots, \lambda_N(t)]: \ \text{One} - \text{slot}$$

$$\lambda^{(2)}(t) = \left[\lambda_1{}^{(2)}(t), \lambda_2{}^{(2)}(t), \lambda_3{}^{(2)}(t), \dots, \lambda_N{}^{(2)}(t)\right]: \ \text{Two} - \text{slot}$$

After the calculation, we sort $\lambda(t)$ and $\lambda^{(2)}(t)$ in a descending order into $\lambda'(t)$ and $\lambda'^{(2)}(t)$. If one secondary user chooses L ($L \leq N$) channels to access, then we decide how many slots to access the channels by comparing:

$$A = \sum_{i=1}^{L} \lambda_i'(t) \ \ and \ B = \sum_{i=1}^{L} \frac{1}{2} \left[\lambda_i'(t) + \lambda_i{}'^{(2)}(t)\right]$$

There are two possible combination versions by changing the conditions for comparing. Firstly the slot number (I define this parameter "recur") for calculating the belief vector is set to be 2. One possible condition is that recur can have single way changing: once $A \geq B$, recur will be set to be 1 and no more changed, namely turned into a pure One-slot algorithm ; The other possible condition is that the value of recur is set according to the comparing result of A and B. We label the combined algorithm for the first condition as "12slot-s" and for the second condition as "12slot-d" in the simulation results.

2.7 Simulation

2.7.1 Single User Single Play

First we simulate for the case “Sing User Sing Play”, the simulation environment settings are: 10 channels, 10000 time slots, one secondary user chooses one channel to access. Here we compare Greedy algorithm, Two-slot algorithm, One-slot algorithm, UCB algorithm and Log-UCB algorithm in terms of their total rewards and the tracking of channels accessed for each time slot.

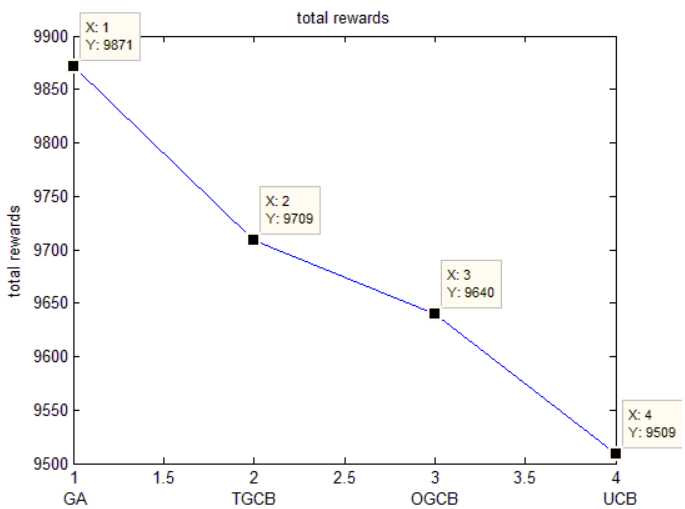

Figure 6: Single User Single Play

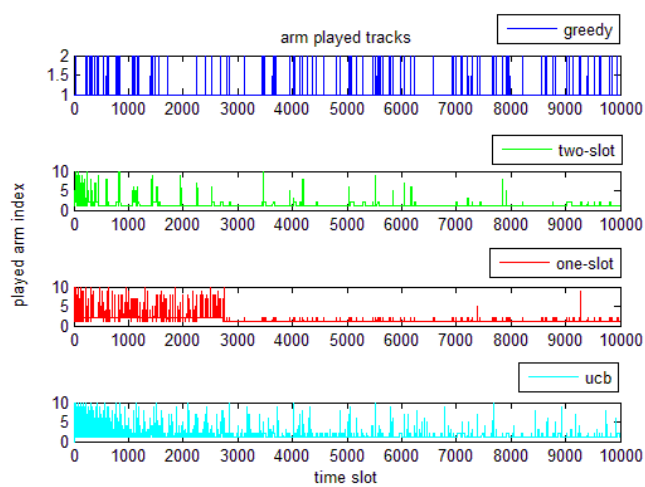

Figure 7: Arm played Tracks

From Figure 6, we can see that in terms of total rewards, Greedy algorithm gains the highest reward, followed by Two-slot algorithm, One-slot algorithm and UCB in series. From Figure 7, which shows for each time slot, which channel is accessed by each of the four algorithms, we can see that Two-slot algorithm's channel access behavior is more similar with that of Greedy algorithm compared to One-slot and UCB, meaning a closer total reward with Greedy algorithm.

2.7.2 Single User Multi play

Then we simulate the situation where there are multiple secondary users and each user can only access one channel at a time. The simulation environment settings are: 5 channels, 10000 time slots, one secondary user chooses multiple channels to access (from 1 to 5 channels).

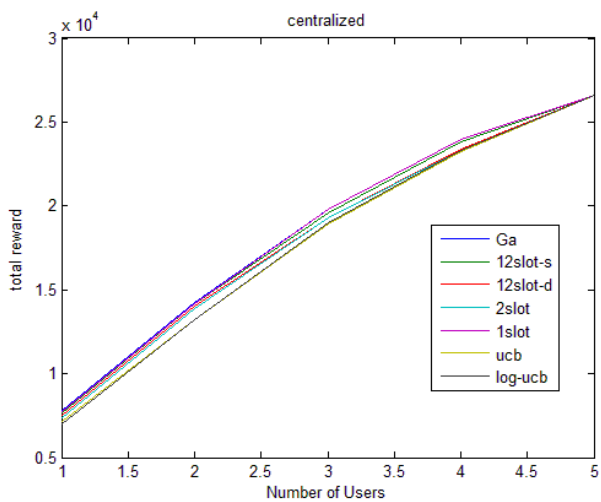

Figure 8: Single User Multi play

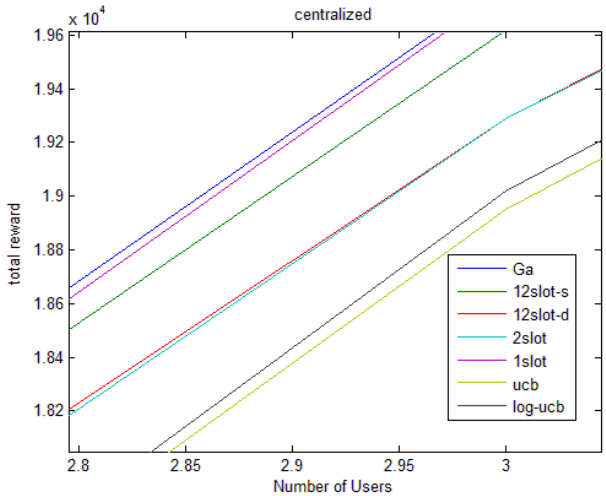

Figure 9: Zoom-in of Figure 4

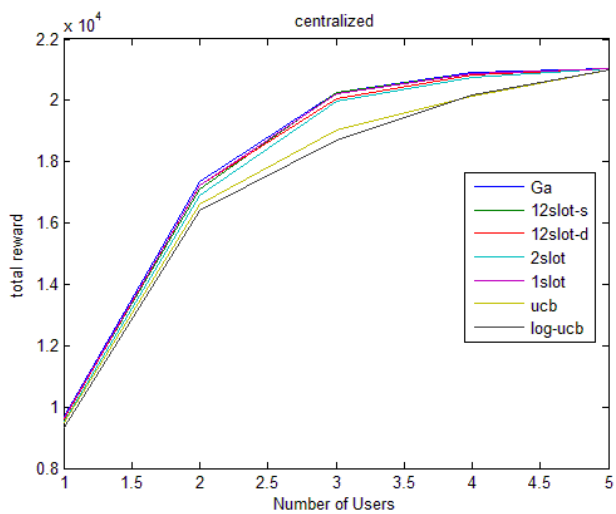

Figure 10: Single User Multi Play (Low state transition probability)

From Figure 8 and Figure 10, we can conclude that the 12slot-s algorithm performs similar to 1slot algorithm; and12slot-d algorithm performs similar to 2slot. In section 2.6.3, we distinguish two combination conditions to get 12slot-s algorithm and 12slot-d algorithm. For 12slot-s, once the parameter recur changes it will hold the same value of 1, then it is just the same with the One-slot algorithm, which means the total reward difference just depends on when $A \geq B$. While for 12slot-d algorithm, the parameter recur changes

according to the comparison result of A and B. The parameter recur has more chances to be set to be 2, which makes this algorithm has an intrinsic similar performance as Two-slot algorithm. All of the algorithms containing the greedy part in the selection calculation of channel accessing gain a higher total reward than UCB algorithms and the differences among these algorithms depend on both the number of channels to access and the transition probabilities. The slower the channels shift from one state to another, the larger total reward difference would be. This phenomenon can be explained as: if the state changing probabilities are very small, suppose the secondary user currently chooses the best channel to access and find it to be busy, from the angle of UCB algorithms, it would probably ignore the fact that for next time slot the best channel would stay in busy state (0 reward) with a very large probability and still chooses the best channel to access for next time slot while for those algorithms containing a greedy belief part, they would probably choose channels with higher probabilities to be idle (1 reward) for next time slot.

The situation, single user multi play is equal to the situation of centralized multi user single play.

2.7.3 Decentralized Multi User Single Play

Before we begin the simulation, we should specify some details on decentralized settings and collision models for Multi-user (U secondary users & N channels).

To ensure fairness and make multi users work in a decentralized manner, we introduce the following settings into the algorithms:

Initially each user will be assigned an index (index=1,2,3,…,U) representing their channel accessing priority (if the user get the index 2 that means it can only access the second best channel according to its own estimation of the channels' order for the current time). After each round of channel accessing (round=2 slots for Two-slot algorithm and round=1 slot for the rest algorithms), each user increase their index by 1. In the next round, only those users with an index not great than N (the number of channels) can access the channels.

Collision happens when more than one user choose the same channel to access and under this situation, we randomly choose one user from those users choosing the same channel to get the reward and only this user can update its parameters.

Expected reward means if we know each channel's mean reward, we would always access the best K channels, K=U (the number of users) when U<=N (the number of channels) otherwise K=N. It is used as a baseline to compare the performances of these algorithms.

The simulation environment settings are: 3 channels, 10000 time slots, the number of secondary users vary from 1 to 6 and the channel access priority keeps for just one round.

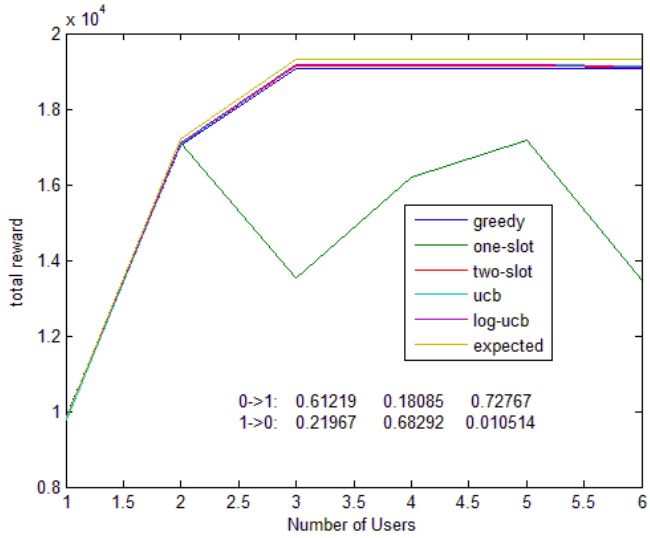

Figure 11: (1) Decentralized multi user single play priority 1

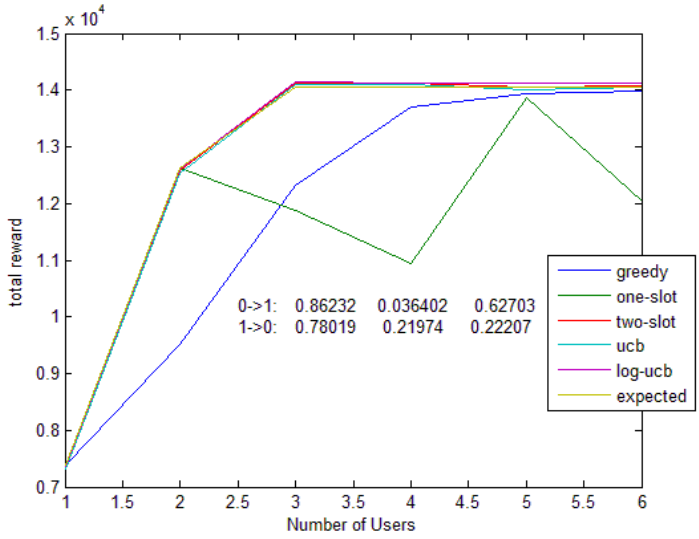

Figure 12: (2) Decentralized multi user single play priority 1

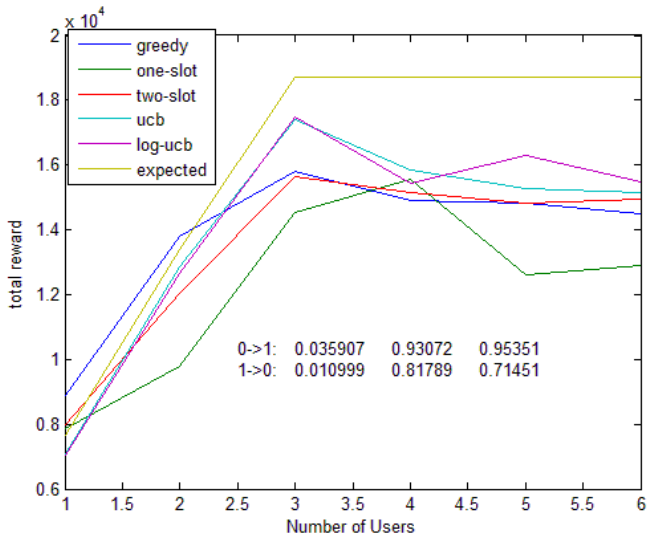

Figure 13: (3) Decentralized multi user single play priority 1

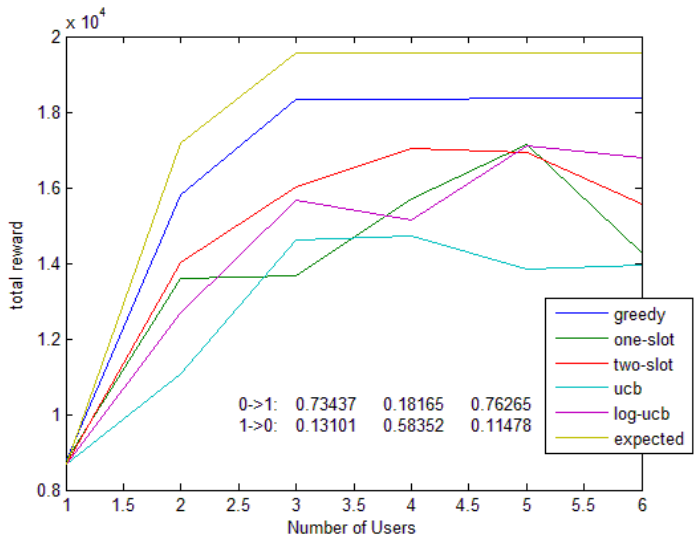

Figure 14: Decentralized multi user single play priority infinity

Under decentralized case, these algorithms could still perform much worse compared to centralized case (in centralized situation, all the total rewards are larger than the expected mean reward), which we could see from Figure 11.

Judging from figure 12, we can see that both UCB algorithms and Two-slot algorithm are better even than greedy algorithm while One-slot algorithm has a very bad performance.

From figure 13, we can see that UCB algorithms perform better than those algorithms containing greedy belief vectors.

We provide the following explanations to explain the simulation results:

First, each secondary user keeps its own channel ranking order, which will cause lots of collisions when they choose channels to access according to their assigned accessing order. That's the reason why all the algorithms under decentralized condition cannot gain a better total reward than that of centralized condition.

Secondly, the exploration part in UCB algorithms has a higher weight than that in algorithms containing greedy belief vector, which will ensure a more accurate ranking of channels and less collisions. This may explain why UCB algorithms can gain higher total rewards than greedy algorithm, Two-slot algorithm and One-slot algorithm.

Thirdly, Each secondary user can only keep channel accessing priority for just one round, which makes One-slot algorithm can't work because One-slot algorithm relies on its intrinsic property to stay in the same channel for a series of time slots to make channel transition probability estimation.

Then we try to set the priority keeping time to be infinity, we get the simulation result of Figure 14. From Figure 14, we can see that the change degrade UCB algorithms quite a lot due to the fact that if a secondary user can only access the second best channel then it can hardly achieve a right ordered ranking of channels and this ranking won't be revised until the weight of exploration part is larger than that of the exploitation part which keeps decreasing due to their wrong estimation about the channels. The reason for that both greed algorithm and Two-slot algorithm's performances are not affected is

due to that their estimation about the ranking is unrelated to the priority keeping time.

Chapter 3

Un-slotted channel access policy

This chapter explains how the mathematic expression for Markov channel collision probability constrained accessing policy is developed and two general policies are presented based on the two general solutions to this math problem, which either utilizes expectation or variance to avoid the requirement of knowing the channel idle distributions.

And we develop another model for collision time constrained channel accessing policy based on renewal theory.

3.1 Continuous-time Markov channel model

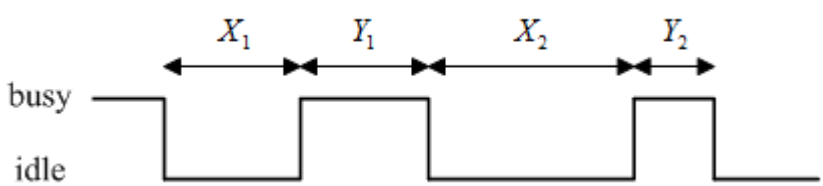

Figure 15: Continuous-time Markov channel model

In this model we use random variable $X_1, X_2, \ldots$ and $Y_1, Y_2, \ldots$to represent the length of channel being idle/busy, with mean value l_I and l_B， and we use 0/1 to represent the sample result of idle/busy.

3.2 Channel access policy without distribution constraints

3.2.1 Channel access behavior and mathematic modeling

Suppose the transmission of secondary users is slotted with a length l_s, $l_s \leq l_I, l_s \leq l_B$ and $l_s \leq l_p$， where l_p is the packet length of the primary user. We don't require the synchronization between primary users (PU) and secondary users (SU) but suppose that SU always have packets to transmit. When SU sense the channel idle, they will choose to transmit with some possibility or to simply back-off some time.

There could be two kinds of collisions between primary users and secondary users:

Type A:

When PU comes back to the channel while SU haven't finished their transmission, collision type A will occur under the proposition that SU can't carry out transmission and channel sensing simultaneously. Even with perfect channel sensing, this type of collision is not avoidable except that SU can predict the exact coming back time of PU, as a consequence SU chooses not to transmit even the channel sensing result is idle.

Type B:

When channel sensing is imperfect, SU may mistake the busy channel condition as idle and transmit, which causes extra collisions with PU [16].

Parameter table

Symbol	Definition
L_I (L_B)	the sojourn time of PU idle (busy) state
l_I (l_B)	average PU idle (busy) time
α	percentage of PU idle time
$f(\cdot)$ ($F(\cdot)$)	PDF (CDF) of PU idle time
N_p (N_c)	number of (collided) PU packets in a PU busy time
n_p (n_c)	mean of N_p (N_c)
l_p (l_s)	the packet length of PU (SU)
p_c	collision probability perceived by the PU
p_c^A (p_c^B)	type-A (B) collision probability
η	collision probability constraint
N_c^A (N_c^B)	number of type-A (B) collisions in a PU busy time
n_c^A (n_c^B)	mean of N_c^A (N_c^B)
Γ_s	SU throughput performance
G_s	SU successful transmission time
q	SU transmission probability
τ_s (f_s)	sensing time (sampling frequency)
P_m (P_f)	probability of missed detection (false alarm)
σ_p^2 (σ_n^2)	power of PU signal (noise)
t	amount of time elapsed since latest PU idle state
k (m)	Index of PU idle-busy periods (SUs)
β, μ, κ, σ	parameters of PU idle time distribution

The portion of channel being idle is $\alpha = \frac{I_I}{I_I + I_B}$, which is also the upper bound of how large a portion SU can make use of the channel. The collision possibility observed by PU is defined as the percentage of transmitted packages collided with SU:

$$p_c = \lim_{K\to\infty} \frac{\sum_{k=0}^{K} N_c(k)}{\sum_{k=0}^{K} N_p(k)} \le \eta \qquad (3\text{-}1)$$

G_s represents the random variable of successfully transmitted time in one idle-busy period of PU. The throughput of SU Γ_s is defined as the portion of successfully transmitted time percentage in one idle-busy period. Then:

$$\Gamma_s = \frac{E[G_s]}{I_I + I_B} \qquad (3\text{-}2)$$

Usually, optimal strategy makes decisions based on current and historical action information, so we express the access strategy as a function of channel's current and historical values, note $h \in [0, T]$ as the current time, $\Lambda(h) = \{\tau \mid \phi(\tau) = Busy, \tau \le h\}$ as the history observations, $\phi(\tau)$ represents the observation of time τ .The access policy $\pi = [q(h, \Lambda(h)) : h \in [0, T]]$ determines the possibility $q(h, \Lambda(h))$ to transmit at each time, obviously $q(h, \Lambda(h)) = 0$ when $\phi(h) = Busy$。

Let $\tau_{\max}(h)$ be the latest time one channel busy interval ends by time h, t means time difference:

$$\tau_{\max}(h) = \max_{\tau\le h}\{\tau : \phi(\tau) = Busy\} \qquad (3\text{-}3)$$

$$t = h - \tau_{\max}(h) \tag{3-4}$$

Now policy π can be simply expressed as $\pi = [q(t_0), q(t_1), \ldots]$, $t_0, t_1, \ldots$ is the possible value t may be. For simplicity, we use $\phi(t)$ replace $\phi(h) = \phi(\tau_{\max}(h) + t)$。

Under perfect sensing only collision type A can occur，due to $l_s \leq l_p$, so at most one packet collision happens，so to make collision threshold effective, it should：

$$\eta < \frac{1}{n_p}$$

(if $\eta \geq \frac{1}{n_p}$, then the secondary user can just transmit whenever it senses the channel being idle)

Let $Z(k)$ be the event that primary user comes back during $[t_k, t_k + l_s)$, then during one idle-busy period the expectation of collision times n_c:

$$\begin{aligned} n_c &= \sum_{k=0}^{\infty} 1 \cdot \Pr[Z(k), SU \text{ transmits at time } t_k] \\ &= \sum_{k=0}^{\infty} q(t_k) \cdot \Pr[Z(k)] \end{aligned} \tag{3-5}$$

$$, \Pr[Z(k)] = \Pr[t_k \leq L_I \leq t_k + l_s]$$

SU's throughput $\Gamma_s(\pi)$:

$$\Gamma_s(\pi) = \frac{\sum_{k=1}^{\infty} (\sum_{l=0}^{k-1} l_s q(t_l)) \Pr[Z(k)]}{l_B + l_I} \tag{3-6}$$

$\sum_{l=0}^{k-1} l_s q(t_l)$ means before PU comes back during $[t_k, t_k + l_s)$, the

expectation of packets SU has already transmitted.

$\Gamma_s(\pi)$ increases as l_s decreases, then as $l_s \to 0$, turn the sum-equations into integrations we get:

$$p_c(\pi) = \frac{n_c}{n_p} = \frac{1}{n_p}\int_0^\infty f(\tau)q(\tau)d\tau \tag{3-7}$$

$$\Gamma_s(\pi) = \frac{\int_0^\infty f(t)\int_0^t q(\tau)d\tau dt}{l_B + l_I} \tag{3-8}$$

So now the channel access problem can be modeled as a mathematic problem as above[16].

3.2.2 One general channel access policy

One general solution is $\hat{\pi} = [\hat{q}(t) : t = 0, 1, \cdots]$:

$$\hat{q}(t) = \begin{cases} n_p\eta, & \text{if } \Phi(t) = \text{Idle}, \\ 0, & \text{otherwise.} \end{cases} \tag{3-9}$$

We can easily get:

$$p_c = \eta, G_s(\hat{\pi}) = n_p\eta l_I,$$
$$\Gamma_s(\hat{\pi}) = n_p\eta\frac{l_I}{l_I + l_B} = n_p\eta\alpha \tag{3-10}$$

This general policy can guarantee a throughput which is optimal for exponential distribution Markov channel, while this channel situation is the worst channel condition for secondary users.

3.2.3 Another general channel access policy

We start from the mathematic expressions and try to find another solution for this problem and develop a policy based on it.

$$\begin{cases} p_c(\pi) = \dfrac{n_c}{n_p} = \dfrac{1}{n_p}\int_0^{\infty} f(\tau)q(\tau)d\tau \\ \Gamma_s(\pi) = \dfrac{\int_0^{\infty} f(t)\int_0^{t} q(\tau)d\tau dt}{l_B + l_I} \end{cases} \qquad (3\text{-}11)$$

The previous solution makes use of expectation to avoid the requirement of knowing the exact function of the distribution while we know besides the expectation; we can also make use of variance which can also spare us from that difficulty.

Based on the thinking above，we can express $q(t)$ as a linear function of t：

$$q(t) = 2at + b \qquad (3\text{-}12)$$

Since linear function can't converge as time goes to infinity, so this requires the channel idle time should be upper bounded, which is also reasonable for some heavy loaded channels:

$$\begin{cases} p_c(\pi) = \dfrac{1}{n_p}\int_0^{L_{\max}} f(t)\,(2at + b)dt \\ \qquad = \dfrac{1}{n_p}(2al_I + b) \le \eta \\ \Gamma_s(\pi) = \dfrac{\int_0^{L_{\max}} f(t)\,(at^2 + bt)dt}{l_B + l_I} \\ \qquad = \dfrac{\int_0^{L_{\max}} f(t)\,[a(t - l_I)^2 + (b + 2al_I)t - al_I^2]dt}{l_B + l_I} \\ \qquad = \dfrac{al_\delta + (b + 2al_I)l_I - al_I^2}{l_B + l_I}, \\ l_\delta \text{ is the variance of idle time} \end{cases} \qquad (3\text{-}13)$$

Now we need to decide the value of a, b in $q(t) = 2at + b$. Take

$b = n_p\eta - 2a1_I$ into $\Gamma_s(\pi)$ we get:

$$\Gamma_s(\pi) = \frac{a(1_\delta - 1_I^2) + n_p\eta 1_I}{1_B + 1_I} = \frac{a(1_\delta - 1_I^2)}{1_B + 1_I} + \frac{n_p\eta 1_I}{1_B + 1_I} \quad (3\text{-}14)$$

It is noticed that the right part of this throughput is right the previous general solution's throughput.

If we take a, b as

$$a = 1_\delta - 1_I^2$$
$$b = n_p\eta - 21_I(1_\delta - 1_I^2) \quad (3\text{-}15)$$

This policy can always guarantee a better throughput than the previous one.

While $q(t)$ is a possibility, then

$1 \geq q(t) = 2at + b \geq 0$, $t \in [0, L_{\max}]$, so take a, b according to (3-15) will not always be valid.

○. When $1_\delta - 1_I^2 \geq 0$, $n_p\eta - 21_I(1_\delta - 1_I^2) \geq 0$, we can set a, b according to (3-15) ;

○. When $1_\delta - 1_I^2 < 0$, $n_p\eta - \frac{1_I}{L_{\max}} > 0$, then set a, b

$$b = n_p\eta - \frac{1_I}{L_{\max}}$$
$$a = \frac{\frac{1_I}{L_{\max}} - n_p\eta}{2L_{\max}} \quad (16)$$

Although this policy can achieve a throughput not lower than previous policy, but knowing exactly when the channel becomes idle is very vital since the possibility to transmit is closely related with the time while for the previous policy, the exactness of idle starting time just decrease the throughput a little (I already work out the exact mathematic expression for this) but doesn't make it unworkable.

3.3 Another model for collision time constraint

We develop a model based on renewal theory [17] to describe the collision time constrained channel access problem. We suppose secondary user can know exactly when the channel becomes idle. We define success transmit factor α and the uncollided transmit factor β when collision happens, both of them are positive and $\alpha \geq \beta$. The channel idle distribution is $f(t)$, l_I and l_B are the mean value of channel idle and busy，the collision possibility constraint is η。

Let τ represent the time primary user plans to stay in this channel right after the channel is sensed idle, and S represent the successful transmission time，then its expectation is：

$$\begin{aligned} E[S] &= E[S \mid \text{PU is absent during } \tau] \\ &\quad + E[S \mid \text{PU comes back during } \tau] \\ &= \tau \int_{\tau}^{\infty} f(t)dt + \int_{0}^{\tau} t f(t)dt \end{aligned} \tag{3-17}$$

Collision time T：

$$E[T] = \int_{0}^{\tau} f(t)\,(\tau - t)dt \tag{3-18}$$

According to the collision time constraint:

$$\frac{E[T]}{l_I + l_B} = \frac{\int_0^{\tau} f(t)\,(\tau - t)dt}{l_I + l_B} \le \eta \qquad (19)$$

Express throughput W as:

$$E[W] = \alpha \cdot E[S \mid \text{PU is absent during } \tau] + \beta \cdot E[S \mid \text{PU comes back during } \tau]$$

Then the channel access problem can be expressed as:

$$\max_{\substack{s.t. \\ E[T] \le \eta(l_I + l_B)}} \{E[W]\} \qquad (20)$$

We show here an example:

suppose $f(t)$ is a uniformly distribution over $(0, 2l_I)$,

$E[W] = \alpha\tau(1 - \frac{\tau}{2l_I}) + \beta\frac{\tau^2}{4l_I}$, we set $\beta = 0$ （it means retransmit all the packet even some of them are well received）， then

$E[W] = \alpha(-\frac{1}{2l_I}(\tau - l_I)^2 + \frac{l_I}{2})$。Under the constraint

$E[T] \le \eta(l_I + l_B)$, we have $\tau \le \sqrt{4l_I\eta(l_I + l_B)}$.

That means we can choose $\tau = \sqrt{4l_I\eta(l_I + l_B)}$ to achieve the theoretical throughput.

Chapter 4

Conclusions

This thesis is mainly on applying Multi-Armed Bandit (MAB) problem to model Cognitive Radio's channel accessing behavior with Markov chain channel model. First we introduce Upper Confidence Bound (UCB) algorithms to solve I.I.D MAB problems which can achieve a regret rate of log(t) and we give out a simple analysis for why the regret rate is log(t). By replacing the log(t) part, which drives the player to do the exploration, we get Log-UCB and Exp-UCB and by simulation we can see the effects of converging speed with the stability and total rewards of that algorithm.

Then we introduce the Cognitive Radio Markov chain channel model and its optimal solution, Greedy algorithm, when the channel transition parameters are pre-known and channels are independent to each other. And inspired from the exploration part and exploitation part in UCB algorithm, we develop an algorithm named *Two-slot Greedy Confidence Bound Algorithm* (TGCB) which contains both greedy algorithm and UCB algorithm without need of channel transition parameters due to the fact that each time the secondary user access the channel would stay in that channel for two time slots. By making use of the observation of that two time slots, the user can do channel transition parameter estimations. After noticing the intrinsic property of UCB algorithm that the user would choose to stay in the same channel for several time slots in series, we get *One-slot Greedy Confidence Bound Algorithm* (OGCB) that will make use of this property to do the channel transition parameter estimation. Finally after adding some comparing condition we can combine TGCB and OGCB. And we simulate to compare the properties of these algorithms under 3 cases: Single User Single Play, Single User Multi Play (equal to Centralized

Multi User Single Play) and Decentralized Multi User Single Play. We give out simulation analysis in detail in section 2.7.

In centralized cases, the simulation can show us that all of the algorithms containing the greedy part in the selection calculation of channel accessing can gain a higher total reward than UCB algorithms and the differences among these algorithms depend on both the number of channels to access and the transition probabilities. The slower the channels shift from one state to another, the larger total reward difference would be.

In decentralized cases, we first give the details about the rules how each secondary user can access the channels (especially when the number of secondary users is larger than that of channels). We find that UCB algorithms can gain a much better total reward due to their heavily weighted exploration part in their expression which can ensure a more accurate channel ranking order so that much less collisions (if more than one secondary user choose to access the same channel) would occur.

개 요

Restless Multi-armed Bandit에 기반한 인지 라디오시스템을 위한 Greedy 신뢰도 분석 기법

인지 라디오는 제한된 스펙트럼 리소스를 활용하는 효율적인 기술이다. 이러한 소프트웨어 기반 인지 라디오 시스템은 무선 통신환경이 가지고 있는 통계적인 특성을 인식 하고 그것에 기반하여 스스로를 적응시키는 동작을 수행한다. 인지 라디오 시스템이 추구하는 목표는 크게 신뢰도 높은 통신을 언제 어디서나 구축하는것과 스펙트럼을 효율적으로 사용하는 두가지로 요약할수있다.

연속성이있는 자원의 효율적인 할당문제는 MAB로 알려진 알고리즘적 방식을 통해 모델링하여 반복적인 실험을 통해 해를 구할수있다. MAB 문제에서는 현재의 적절한 보상을 달성하기위한 전략과 현재의 이득을 어느정도 포기하고 미래에 예측가능한 이득을 증진시키는 두가지 전략을 적절히 배합하여 최적의 해를 찾는다. 일반적인 MAB 관련 문제에서는 확률적인 보상을 생성하는 다중 Arm 또는 이에 상응하는 등가의 미리 정의된 동작이 있으며, 플레이어는 예상되는 총 보상을 극대화하기 위한 K개의 Arm을 연속적으로 선택하는 정책을 찾는다.

이 논문에서는 보다 도전적인 문제인 비 베이지안 RMAB 문제를 다룬며 이때 Markov chain의 파라미트들은 미리 알려지지 않았다고 가정된다. 또한 이를 이용해 인지 라디오 시스템을 위한 실용적인 동적 스펙트럼 센싱에에대한 연구를 진행한다. 만약 채널 사용에 있어 우선순위를 갖는 1차 사용자의 채널 사용 패턴이 동일하고 독립적인

Markov chain으로 모델링 할때 이를 비 베이지안 RMAB문제로 다룰수있다. 본 논문의 기여도는 통계적 특성이 알려지지 않은 동적 채널에서 2시간 슬롯 Greedy 신뢰도 범위 알고리즘을 (Two slot GCB) 이용해 효율적인 다중 채널 스펙트럼 센싱 알고리즘을 개발한데에 있다. 이는 Markov 파라미터 추정기법과 채널 센싱기법이 결합된 형태를 띄고 있으며 이를 통해 기존의 UCB 알고리즘의 Regret rate이 ln(t) 을 가짐을 보일수있다. 마지막으로 본 연구를 통해 제한조건이 없는 연속 시간 Markov chain 채널 모델을 적용한 인지 라디오 시스템에서의 채널 접근 정책을 제안한다.

Keywords: 인지 라디오, MAB, Markov chain, Greedy Algorithm, Two-slot GCB

Student Number: 2011-24072

Bibliography

[1] Federal Communications Commission, "*Spectrum Policy Task Force*", Rep. ET Docket no. 02-135, Nov. 2002

[2] P. Kolodzy et al., "*Next generation communications: Kickoff meeting*", in Proc. DARPA, Oct. 17, 2001

[3] M. McHenry, "*Frequency agile spectrum access technologies*", in FCC Workshop Cogn. Radio, May 19, 2003

[4] G. Staple and K. Werbach, "*The end of spectrum scarcity*", IEEE Spectrum, vol. 41, no. 3, pp. 48-52, Mar. 2004

[5] J. Mitola et al., "*Cognitive radio: Making software radios more personal*", IEEE Pers. Commun., vol. 6, no. 4, pp. 13-18, Aug. 1999

[6] J. Mitola, "*Cognitive radio: An integrated agent architecture for soft-ware defined radio*", Doctor of Technology, Royal Inst. Technol. (KTH), Stockholm, Sweden, 2000

[7] A. Mahajan and D. Teneketzis, "*Multi-armed bandit problems"*, University of Michigan, Ann Arbor, MI, USA, in Foundations and Applications of Sensor Management, A. O. Hero III, D. A. Castanon, D. Cochran and K. Kastella, (Editors), Springer-Verlag, 2007

[8] P. Whittle, "*Restless bandits: activity allocation in a changing World"*, Journal of Applied Probability, vol. 25, 1988

[9] C. H. Papadimitriou and J. N. Tsitsiklis, "The complexity of optimal queueing network control", Mathematics of Operations Research, vol. 24, no. 2, 1999

[10] K. Liu and Q. Zhao, "*Indexability of restless bandit problems and Optimality of Whittle index for dynamic multichannel access*", IEEE Trans. on Info. Theory, vol. 56, no. 11, 2010

[11] L. DaCosta, A. Fialho, M. Schoenauer, and M. Sebag, "*Adaptive operator selection with dynamic multi-armed bandits*", July 12-16, 2008, Atlanta, Georgia, USA

[12] W. Jouini, D. Ernst, C. Moy, and J. Palicot, "*Multi-armed bandit based policies for cognitive radio's decision making issues*", the 3rd International Conference on Signals, Circuits and Systems, Jerba, Tunisia, 6-8 November 2009

[13] *Multi-armed bandit,* From Wikipedia, the free encyclopedia

[14] W. Dai , Y. Gai, B. Krishnamachari, Q. Zhao, "*The non-bayesian restless multi-armed bandit: a case of near-logarithmic regret*", the IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP 2011, May 22-27, 2011

[15] Q. Zhao, L. Tong, A. Swami, and Y. Chen, "*Decentralized Cognitive mac for opportunistic spectrum access in ad hoc networks: a pomdp Framework*", IEEE J. ON SEL. AREAS IN COMM., Vol. 25, No. 3, April 2007.

[16] Xin Liu, Zhi Ding, Senhua Huang, "Optimal Transmission Strategies for Dynamic Spectrum Access in Cognitive Radio Networks", IEEE Transactions on Mobile Computing, vol. 8, No. 12, Dec. 2009

[17] Renewal theory, From Wikipedia, the free encyclopedia