# Dynamic Chinese Restaurant Game: Theory and Application to Cognitive Radio Networks

Chunxiao Jiang, *Member, IEEE,* Yan Chen, *Member, IEEE,* Yu-Han Yang, *Student Member, IEEE,* Chih-Yu Wang, *Member, IEEE,* and K. J. Ray Liu, *Fellow, IEEE*

***Abstract*—Users in a social network are usually confronted with decision making under uncertain network state. While there are some works in the social learning literature on how to construct belief on an uncertain network state, few study has been made on integrating learning with decision making for the scenario where users are uncertain about the network state and their decisions influence with each other. Moreover, the population in a social network can be dynamic since users may arrive at or leave the network at any time, which makes the problem even more challenging. In this paper, we propose a Dynamic Chinese Restaurant Game to study how a user in a dynamic social network learns the uncertain network state and make optimal decision by taking into account not only the immediate utility but also subsequent users' negative influence. We introduce a Bayesian learning based method for users to learn the network state, and propose a Multi-dimensional Markov Decision Process based approach for users to achieve the optimal decisions. Finally, we apply the Dynamic Chinese Restaurant Game to cognitive radio networks and demonstrate from simulations to verify the effectiveness and efficiency of the proposed scheme.**

***Index Terms*—Chinese restaurant game, Bayesian learning, Markov decision process, cognitive radio, game theory.**

## I. Introduction

In a social network [1], users are usually uncertain about the network state when making decisions [2]. For example, when choosing a cloud storage service, users may not know exactly the reliability and effectiveness of each service provider. Besides, users have to consider subsequent others' decisions since overwhelming users sharing one storage service will inevitably increase the waiting time and the blocking rate. Such a phenomenon is known as negative network externality [3], i.e., the negative influence of other users' behaviors on one user's reward, due to which users tend to avoid making the same decisions with others to maximize their own payoffs. Similar problems can be found when it comes to selecting a deal on Groupon website or choosing a WiFi access point in a conference hall. Therefore, how users in a social network learn the network state and make best decisions by predicting the influence of others' possible decisions is an important research issue in the field of social networking.

Although users in a social network only have limited knowledge about the uncertain network state, they can learn from some external information, e.g., other users' experiences, to construct a belief, which is mostly probabilistic, on the uncertain network state. In the social learning literatures [4]-[7], how a user constructs accurate belief through adopting different kinds of learning rules was studied. However, the concept of network externality has not been considered in those traditional social learning works, i.e., they mostly assumed that one user's reward is independent with the actions of subsequent users. In such a case, a user's decision making is purely based on his/her belief without taking into account other users' decisions. As discussed above, the negative network externality is a common phenomenon in social networking and can influence users' rewards and decisions to a large extent. When combining the negative network externality with social learning, users' decision making will inevitably involve the game-theoretic analysis, which analyzes how users' decisions influence each other [8].

In our previous work [9], we proposed a new game called "Chinese Restaurant Game" to study how to involve the strategic decision making into the social learning for the social networking problems with negative network externality. This game concept is originated from Chinese Restaurant Process [10], which is applied in non-parameter learning methods of machine learning to construct the parameters for modeling unknown distributions. In the Chinese Restaurant Game, there are finite tables with different sizes and finite customers sequentially requesting tables for meal. Since customers do not know the exact size of each table, they have to learn the table sizes according to some external information. Moreover, when requesting one table, each customer should take into account the following customers' selections due to the limited dining space in each table, i.e., the negative network externality. Through studying such a Chinese Restaurant Game model, we provided a new general framework for analyzing the strategic learning and predicting behaviors of rational users in a social network. In [11], the applications of Chinese Restaurant Game in various research fields are also discussed.

One assumption in the Chinese Restaurant Game is the fixed population setting, i.e., there is a finite number of customers

Manuscript received April 9, 2013; revised September 27, 2013 and January 7, 2014; accepted January 8, 2014. The associate editor coordinating the review of this paper and approving it for publication was T. Melodia.

C. Jiang is with the Department of Electronic Engineering, Tsinghua University, Beijing 100084, P. R. China. This work was done during his visit at the University of Maryland (e-mail: chx.jiang@gmail.com).

Y. Chen, Y.-H. Yang, C.-Y. Wang, and K. J. R. Liu are with the Department of Electrical and Computer Engineering, University of Maryland, College Park, MD 20742, USA. C.-Y. Wang is also with the Graduate Institute of Communication Engineering, National Taiwan University, Taipei, Taiwan (e-mail: {yan, yhyang, kjrliu}@umd.edu, tomkywang@gmail.com).

This work was partly funded by project 61371079 and 61271267 supported by NSFC China, Postdoctoral Science Foundation funded project.

Digital Object Identifier 10.1109/TWC.2014.030314.130632

choosing the tables sequentially [9]. However, in many real applications, customers may arrive and leave at any time, which results in the dynamic population setting. In such a case, the utilities of customers will change from time to time due to the dynamic number of customers on each table. Considering these problems, in this paper, we extend the Chinese Restaurant Game to the dynamic population setting, where we consider the scenario that customers arrive at and leave the restaurant with a random process. In such a case, each new coming customer not only learns the system state according to the information received and revealed by former customers, but also predicts the following customers' decisions during his/her meal time in order to maximize the utility. With such a dynamic model, our Chinese Restaurant Game framework becomes more general and practical.

The Dynamic Chinese Restaurant Game can be applied to many different fields, such as storage service selection in cloud computing, deal selection on Groupon website in online social networking and WiFi access point selection in wireless networking as discussed at the beginning. In this paper, we will focus on the application to cognitive radio networks [12]. Recently, dynamic spectrum access in cognitive radio networks has been shown to be able to improve the spectrum utilization efficiency, where cognitive devices, called as Secondary Users (SUs), can dynamically access the licensed spectrum, under the condition that the interference to the Primary User (PU) is minimized [13]. In dynamic spectrum access, SUs need to perform spectrum sensing to learn the state of primary channel, and share the available primary channel with other SUs. The more SUs access the same channel, the less throughput can be obtained by each individual SU, i.e., there exists negative network externality. Therefore, the proposed Dynamic Chinese Restaurant Game is an ideal tool for the dynamic spectrum access problems to analyze how SUs learn the state of primary channels and how to access the primary channels by predicting the influence of subsequent SUs' decisions.

The main contributions of this paper are summarized as follows.

1) We propose a Dynamic Chinese Restaurant Game framework to study the social learning problem with negative network externality. Such a framework studies how users in a social network learn the uncertain system state according to the external information and make best decisions to maximize their own expected utilities by considering other users' decisions, which is very general and can be applied to many research areas.
2) Since tables in a restaurant can be reserved, customers have to estimate the table state in order to avoid selecting the reserved tables. We introduce a table state learning method based on Bayesian learning rule, where each customer constructs his/her own belief on the system state according to his/her own signal and the former customer's belief information.
3) When selecting one table for meal, customers not only need to consider immediate utility, but also need to take into account the utility in the future, i.e., considering the subsequent customers' decisions. We formulate the table selection problem as a Multi-dimensional Markov Decision Process (M-MDP) and design a modified value iteration algorithm to find the best strategies.
4) We apply the proposed Dynamic Chinese Restaurant Game to cognitive radio networks and design a Bayesian channel sensing method and M-MDP based channel access scheme. We prove theoretically that there is a threshold structure in the optimal strategy profile for the two primary channel scenario. For multiple primary channel scenario, we propose a fast algorithm with much lower computational complexity while achieving comparable performance.

The rest of this paper is organized as follows. Firstly, the Dynamic Chinese Restaurant Game is formulated in Section II, in which we discuss the Bayesian learning based restaurant state estimation, and introduce an M-MDP model to solve the table selection problem. Then, the application to cognitive radio networks is discussed in details in Section III. Finally, we show simulation results in Section IV and draw conclusions in Section V.

## II. Dynamic Chinese Restaurant Game

In this section, we will introduce the proposed Dynamic Chinese Restaurant Game in details. Specifically, we first discuss how customers learn the restaurant state using Bayesian learning rule, and then focus on how customers make table selection according to the learning result, so as to maximize their own expected utilities during the meal time.

### A. System Model

We consider a Chinese restaurant with $N$ independent tables numbered $1, 2, ..., N$, where each table has finite $L$ seats that can serve finite customers. In our model, all tables in the restaurant are of the same size, i.e. with the same number of seats. The customers are consider as arriving and leaving by Bernoulli process [14], where a single customer arrives with probability $\lambda$ or a single customer leaves with probability $\mu$ in each time slot. As shown in Fig. 1, when arriving at the restaurant, each customer requests for one table to have a meal. Once a customer chooses one specific table, he/she will stay at that table throughout his/her meal time. Moreover, the tables may be reserved in advance and such reserved tables cannot be used to serve new coming customers until the reservation is canceled. We here define the restaurant state $\boldsymbol{\theta} = (\theta_1, \theta_2, ..., \theta_N)$ (all the subscripts mean the table number index in the paper), where $\theta_i \in \{\mathcal{H}_0, \mathcal{H}_1\}$ denotes the state of table $i$, $\mathcal{H}_0$ means the table is available while $\mathcal{H}_1$ means the table is already reserved. Notice that the state of each table $\theta_i$ is time-varying since customers may reserve the table or cancel the reservation at any time.

The proposed Dynamic Chinese Restaurant Game is to formulate the problem that how a new arriving customer selects a table. For each customer, his/her action set is $\mathcal{A} = \{1, 2, ..., N\}$, i.e., choosing one table from all $N$ tables. Note that we only consider pure strategies in this paper. Let us define the grouping state when the $j$th customer arrives, $\boldsymbol{G}^j = (g_1^j, g_2^j, ..., g_N^j)$ (all the superscripts mean the customer index in the paper), where $g_i^j \in \{0, 1, \ldots, L\}$ stands for the number of customers in table $i$. Assuming that the $j$th

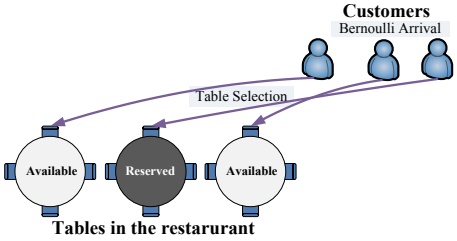


Fig. 1. System model of the Chinese Restaurant Game.

customer finally chooses table $i$, his/her utility function can be given by $U\left(\theta_i^j, g_i^j\right)$, where $\theta_i^j$ denotes the state of table $i$ and $g_i^j$ denotes the number of customers choosing table $i$ during the $j$th customer's meal time in table $i$. Note that the utility function is a decreasing function in terms of $g_i^j$, which can be regarded as the characteristic of negative network externality since the more subsequent customers join table $i$, the less utility the $j$th customer can achieve.

As discussed above, the restaurant state $\boldsymbol{\theta}$ is changing with time. For the $j$th new arriving customer, he/she may not know the exact reservation state of each table $\theta_i^j$. Nevertheless, customers can estimate the state through some external information such as advertisement and reviews. Therefore, we assume that the customers may have an initial prior distribution of the state $\theta_i$ for each table, which is denoted as $\boldsymbol{b}^0 = \{b_i^0 | b_i^0 = \Pr(\theta_i = \mathcal{H}_0), \forall i \in 1, 2, ..., N\}$. Moreover, each customer can receive a signal $\boldsymbol{s}^j = \{s_i^j, \forall i \in 1, 2, .., N\}$ generated from a predefined distribution $f(s_i|\theta_i)$. Such signals can be regarded as the observation (estimation) of the restaurant state by customers. Note that not all the customers necessarily have the initial belief since they can observe the previous customer's belief as the initial belief. Moreover, since the customers do not exactly know the reservation state of each table, they may select the tables which are already reserved and only receive 0 utility.

### *B. Bayesian Learning for the Restaurant State*

In this subsection, we discuss how customers estimate the restaurant state with some external information. Since the restaurant state $\boldsymbol{\theta}$ is time-varying, customers have to learn each $\theta_i$ before making table selection to avoid the reserved tables. As discussed above, each customer receives a signal about the restaurant state. Besides, customers also receive former customers' reviews about the restaurant, i.e., previous customer's belief. With these collected information, we can use Bayesian learning model to update the belief on the current restaurant state.

Here, we first introduce the concept of belief to describe customers' uncertainty about the state of the tables. The belief $b_i^j$ denotes $j$th customer's belief on the state of table $i$. It is assumed that each customer reveals his/her beliefs after making the table selection. Unlike the previous static Chinese Restaurant Game model where signals are revealed instead of beliefs, the restaurant state $\theta$ is changing with time in this dynamic model. In such a case, for customer $j$, signals $s^{j-2}, s^{j-3}, \ldots$ are of less and less use for him/her to construct belief. Moreover, belief $b^{j-1}$ contains more information than signal $s^{j-1}$, which is more useful for the following customers' beliefs calculation. Therefore, each customer's belief on table $i$ is learned from former customer's belief $b_i^{j-1}$, his/her own signal $s_i^j$ and the conditional distribution $f(s_i|\theta_i)$, which can be defined as

$$\boldsymbol{b}^j = \{b_i^j | b_i^j = \Pr(\theta_i^j = \mathcal{H}_0 | b_i^{j-1}, s_i^j, f), \forall i \in 1, 2, ..., N\}. \quad (1)$$

From the definition above, we can see that the belief $b_i^j \in [0, 1]$ is a continuous parameter. In a practical system, it is impossible for a customer to reveal his/her continuous belief using infinite data bits. Therefore, we quantize the continuous belief into $M$ belief levels $\{\mathbb{B}_1, \mathbb{B}_2, ..., \mathbb{B}_M\}$, which means that if we have $b_i^j \in [\frac{k-1}{M}, \frac{k}{M}]$, then $B_i^j = \mathbb{B}_k$. Since each customer can only reveal and receive the quantized belief, the former customer's quantized belief $\boldsymbol{B}^{j-1}$ is first mapped into a belief $\widehat{\boldsymbol{b}}^{j-1}$ according to the rule that if $B_i^{j-1} = \mathbb{B}_k$ then $\hat{b}_i^{j-1} = \frac{1}{2}\left(\frac{k-1}{M} + \frac{k}{M}\right)$. Note that the mapping belief $\hat{b}_i^{j-1}$ here is not the former customer's real continuous belief $b_i^{j-1}$. Then, $\widehat{\boldsymbol{b}}^{j-1}$ is combined with the signal $\boldsymbol{s}^j$ to calculate the continuous belief $\boldsymbol{b}^j$. Finally, $\boldsymbol{b}^j$ is quantized into the belief $\boldsymbol{B}^j$. Thus, the learning process for the $j$th customer can be summarized as $\boldsymbol{B}^{j-1} \xrightarrow{\text{Mapping}} \widehat{\boldsymbol{b}}^{j-1} \xrightarrow{\boldsymbol{s}^j} \boldsymbol{b}^j \xrightarrow{\text{Quantize}} \boldsymbol{B}^j$.

In the learning process, the most important step is how to calculate current belief $\boldsymbol{b}^j$ according to current signal $\boldsymbol{s}^j$ and the former customer's belief $\widehat{\boldsymbol{b}}^{j-1}$, which is a classical social learning problem. Based on the approaches to belief formation, social learning can be classified as Bayesian learning [5] and non-Bayesian learning [7]. Bayesian learning refers that rational individuals use Bayes' rule to form the best estimation of the unknown parameters, such as the restaurant state in our model, while non-Bayesian learning requires individuals to follow some predefined rules to update their beliefs, which inevitably limits the rational customers' optimal decision making. Since customers in our Dynamic Chinese Restaurant Game are assumed to be fully rational, they will adopt Bayesian learning rule to update their beliefs on the

$$\Pr\left(\theta_i^j = \mathcal{H}_0 | \hat{b}_i^{j-1}\right) = \Pr(\theta_i^j = \mathcal{H}_0 | \theta_i^{j-1} = \mathcal{H}_0)\hat{b}_i^{j-1} + \Pr(\theta_i^j = \mathcal{H}_0 | \theta_i^{j-1} = \mathcal{H}_1)(1 - \hat{b}_i^{j-1}), \quad (3)$$

$$\Pr\left(\theta_i^j = \mathcal{H}_1 | \hat{b}_i^{j-1}\right) = \Pr(\theta_i^j = \mathcal{H}_1 | \theta_i^{j-1} = \mathcal{H}_0)\hat{b}_i^{j-1} + \Pr(\theta_i^j = \mathcal{H}_1 | \theta_i^{j-1} = \mathcal{H}_1)(1 - \hat{b}_i^{j-1}). \quad (4)$$

$$b_i^j = \frac{\left(\Pr(\theta_i^j = \mathcal{H}_0 | \theta_i^{j-1} = \mathcal{H}_0)\hat{b}_i^{j-1} + \Pr(\theta_i^j = \mathcal{H}_0 | \theta_i^{j-1} = \mathcal{H}_1)(1 - \hat{b}_i^{j-1})\right) f(s_i^j | \theta_i^j = \mathcal{H}_0)}{\sum_{l=0}^{1}\left(\Pr(\theta_i^j = \mathcal{H}_l | \theta_i^{j-1} = \mathcal{H}_0)\hat{b}_i^{j-1} + \Pr(\theta_i^j = \mathcal{H}_l | \theta_i^{j-1} = \mathcal{H}_1)(1 - \hat{b}_i^{j-1})\right) f(s_i^j | \theta_i^j = \mathcal{H}_l)}. \quad (5)$$

restaurant state $\boldsymbol{b}^j = \{b_i^j\}$ as

$$b_i^j = \frac{\Pr\big(\theta_i^j = \mathcal{H}_0 | \hat{b}_i^{j-1}\big) f(s_i^j | \theta_i^j = \mathcal{H}_0)}{\sum_{l=0}^{1} \Pr\big(\theta_i^j = \mathcal{H}_l | \hat{b}_i^{j-1}\big) f(s_i^j | \theta_i^j = \mathcal{H}_l)}, \tag{2}$$

where $\Pr(..|\widehat{b}_i^{j-1})$ stands for the probability given the belief of the $(j-1)$th customer. For example, $\Pr(\theta_i^j = \mathcal{H}_0 | \widehat{b}_i^{j-1})$ stands for, given the $(j-1)$th customer's belief, i.e., $\Pr(\theta_i^{j-1} = \mathcal{H}_0) = \widehat{b}_i^{j-1}$, what the probability $\Pr(\theta_i^j = \mathcal{H}_0)$ is. Note that (2) is based on the fact that when given the exact state $\theta_i^j$, the signal observed by current customer, $s_i^j$, is independent of the last customer's belief $\hat{b}_i^{j-i}$.

As discussed in the system model, the state of each table is varying with time. Here, we define the state transition probability as $\Pr(\theta_i^j = \mathcal{H}_0 | \theta_i^{j-1} = \mathcal{H}_0)$, which represents the probability that table $i$ is currently available when the $j$th customer arrives given the condition that table $i$ was available when the $(j-1)$th customer arrived. Similarly, we have $\Pr(\theta_i^j = \mathcal{H}_1 | \theta_i^{j-1} = \mathcal{H}_0)$, $\Pr(\theta_i^j = \mathcal{H}_0 | \theta_i^{j-1} = \mathcal{H}_1)$ and $\Pr(\theta_i^j = \mathcal{H}_1 | \theta_i^{j-1} = \mathcal{H}_1)$. In such a case, a customer can calculate the items $\Pr\big(\theta_i^j = \mathcal{H}_0 | \hat{b}_i^{j-1}\big)$ and $\Pr\big(\theta_i^j = \mathcal{H}_1 | \hat{b}_i^{j-1}\big)$ in (2) using (3) and (4), where the four state transition probabilities are as prior information for customers.

To summarize, for the $j$th customer, the belief updating process for table $i$ is $B_i^{j-1} \xrightarrow{\text{Mapping}} \hat{b}_i^{j-1} \xrightarrow{\text{Bayesian}+s_i^j} b_i^j \xrightarrow{\text{Quantize}} B_i^j$, where the Bayesian learning from $\hat{b}_i^{j-1}$ and $s_i^j$ to $b_i^j$ is (5).

### C. Multi-dimensional MDP Based Table Selection

In this subsection, we investigate the table selection game by modeling it as a Markov Decision Process (MDP) problem [15]. In this game, each customer selects a table after restaurant state learning, with the objective of maximizing his/her own expected utility during the meal time in the restaurant. To achieve this goal, rational customers not only need to consider the immediate utility, but also need to predict the following customers' selections. In our model, customers arrive by Bernoulli process and make the table selection sequentially. When making the decision, one customer is only confronted with current grouping information $\boldsymbol{G}^j$ and belief information $\boldsymbol{B}^j$. In order to take into account customers' expected utility in the future, we use Bellman equation to formulate a customer's utility and use MDP model to formulate this table selection problem. In the traditional MDP problem, a player can adjust his/her decision when the system state changes. However, in our system, once choosing a table, a customer cannot adjust his/her decision even if the system state has already changed. Therefore, traditional MDP cannot be directly applied here. To solve this problem, we propose a Multi-dimensional MDP (M-MDP) model, and a modified value iteration method to derive the best response (strategy) for each customer.

*1) System State:* To construct the MDP model, we first define the system state and verify the Markov property of the state transition. Let the quantized belief $\boldsymbol{B} = (B_1, B_2, ..., B_N) \in \{1, 2, \ldots, M\}^N$ be the belief state. Thus, we can define the system state $S$ as the belief state $\boldsymbol{B}$ with the grouping state $\boldsymbol{G} = (g_1, g_2, ..., g_N) \in \{0, 1, \ldots, L\}^N$, i.e., $S = (\boldsymbol{B}, \boldsymbol{G})$, where the finite state space is $\mathcal{X} = \Big(\{1, 2, \ldots, M\}^N \times \{0, 1, \ldots, L\}^N\Big)$. Note that the system state is defined at each time slot. When the $j$th customer arrives at the restaurant, the system state he/she encounters is $S^j = (\boldsymbol{B}^j, \boldsymbol{G}^j)$. In such a case, with multiple customers arriving sequentially, the system states at different arrival time $\{S^1, S^2, ...S^j, ...\}$ form a stochastic process. In our learning rule, only the $(j-1)$th customer's belief is used to update the $j$th customer's belief. Therefore, $\boldsymbol{B}^j$ depends only on $\boldsymbol{B}^{j-1}$. Moreover, since customers arrive by Bernoulli process, the grouping state $\boldsymbol{G}^j$ is also memoryless. In such a case, we can verify that $\{S^1, S^2, ...S^j, ...\}$ is a Markov process.

*2) Belief State Transitions:* Note that a customer's belief transition is independent with his/her action, and is only related to the state of the tables, as well as the Bayesian learning rule. Here, we define the belief state transition probability as $P\big(\boldsymbol{B}^j | \boldsymbol{B}^{j-1}\big)$. Since all tables are independent with each other, we have

$$P\Big(\boldsymbol{B}^j \big| \boldsymbol{B}^{j-1}\Big) = \prod_{i=1}^{N} P\big(B_i^j | B_i^{j-1}\big), \tag{6}$$

where $P\big(B_i^j | B_i^{j-1}\big)$ is the belief state transition probability of table $i$. In such a case, there is an $M \times M$ belief state transition matrix for each table, which can be derived according to the Bayesian learning rule. To find $P\big(B_i^j = \mathbb{B}_q | B_i^{j-1} = \mathbb{B}_p\big)$, with the quantized belief $B_i^{j-1} = \mathbb{B}_p$, we can calculate the corresponding mapping belief $\hat{b}_i^{j-1} = \frac{1}{2}\big(\frac{p-1}{M} + \frac{p}{M}\big)$. Then, with $B_i^j = \mathbb{B}_q$, we can have the value interval of $b_i^j = [\frac{q-1}{M}, \frac{q}{M}]$. Thus, the belief state transition probability can be computed by

$$P\big(B_i^j = \mathbb{B}_q | B_i^{j-1} = \mathbb{B}_p\big) = \int_{\frac{q-1}{M}}^{\frac{q}{M}} P(b_i^j | \hat{b}_i^{j-1}) db_i^j. \tag{7}$$

where $P(b_i^j | \hat{b}_i^{j-1})$ can be calculated by (5).

*3) Actions and System State Transitions:* The finite action set for customers is the $N$ table set, i.e., $\mathcal{A} = \{1, 2, ..., N\}$. Let $a \in \mathcal{A}$ denote a new customer's action under the system state $S = (\boldsymbol{B}, \boldsymbol{G})$. Let $P\big(S' = (\boldsymbol{B}', \boldsymbol{G}') \big| S = (\boldsymbol{B}, \boldsymbol{G}), a\big)$ denote the probability that action $a$ in state $S$ will lead to state

$$P\big(\boldsymbol{G}' = (g_1, g_2, ..., g_i + 1, ..., g_N) | \boldsymbol{G} = (g_1, g_2, ..., g_i, ..., g_N), a = i\big) = \lambda, \tag{9}$$

$$P\big(\boldsymbol{G}' = (g_1, g_2, ..., g_j + 1, ..., g_N) | \boldsymbol{G} = (g_1, g_2, ..., g_i, ..., g_N), a = i\big) = 0, \quad (\forall j \neq i), \tag{10}$$

$$P\big(\boldsymbol{G}' = (g_1, g_2, ..., g_i - 1, ..., g_N) | \boldsymbol{G} = (g_1, g_2, ..., g_i, ..., g_N)\big) = g_i \mu, \quad \big(\forall i \in \{1, 2, \ldots, N\}\big), \tag{11}$$

$$P\big(\boldsymbol{G}' = \boldsymbol{G} | \boldsymbol{G} = (g_1, g_2, ..., g_i, ..., g_N)\big) = 1 - \lambda - \sum_{i=1}^{N} g_i \mu. \tag{12}$$

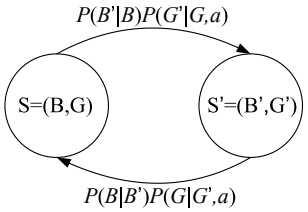


Fig. 2. Illustration of state transition.

$S'$. As shown in Fig. 2, since a customer's belief transition is independent with his/her action, we have

$$P\big(S'=(\boldsymbol{B}',\boldsymbol{G}')|S=(\boldsymbol{B},\boldsymbol{G}),a\big)=P\big(\boldsymbol{B}'|\boldsymbol{B}\big)P\big(\boldsymbol{G}'|\boldsymbol{G},a\big), \quad (8)$$

where $P\big(\boldsymbol{G}'|\boldsymbol{G},a\big)$ is the system grouping state transition probability. Suppose that current grouping state is $\boldsymbol{G}=(g_1,g_2,...,g_N)$, since a new customer arrives with probability $\lambda$, given the action of the arriving customer is table $i$, i.e., $a=i$, we have the arriving transition probabilities in (9) and (10). When no customer arrives, but some customer leaves the restaurant at state $G$, we have the leaving transition probability in (11), where $\mu$ is the leaving probability of customers, $\lambda$ and $\mu$ are normalized such that $\lambda+NL\mu\leq 1$ since $g_i\leq L$ and $\lambda+\sum_{i=1}^{N}g_i\mu\leq\lambda+NL\mu\leq 1$ according to (12). In such a case, the system state transition probabilities $P(S'|S)$ form an $\Big(M(L+1)\Big)^N\times\Big(M(L+1)\Big)^N$ state transition matrix when given action $a$. Note that (9-12) are based on the assumption that the system time is discretized into small time slots and customers arrive and leave by Bernoulli process. During each time slot, a single user arrives with probability $\lambda$ or a single user leaves with probability $\mu$. There is no multiple customers leaving the same table or multiple customers leaving different tables. This model is also called as "sampled-time approximation to a Markov process" as in [14]. Under this model, the state transition from one time slot to the next can only be increasing 1 customer, decreasing 1 customer, or keeping unchanged.

*4) Expected Utility:* The immediate utility of a customer in table $i$ at system state $S$ is

$$U_i(S)=\hat{b}_i\cdot R_i\big(g_i\big), \quad (13)$$

where $\hat{b}_i$ is the mapping belief of $B_i$ and $R_i$ is a decreasing function with respect to the number of customers in table $i$, $g_i$. In general, each customer will stay at the selected table for a period of time, during which the system state may change. Therefore, when making the table selection, the customer should not only consider the immediate utility, but also take into account the future utilities. In the MDP model [15], Bellman equation is defined as a user's long-term expected payoff with the form as

$$V(S_0,a_0)=\max_{\{a_t\}_{t=0}^{\infty}}U(S_0,a_0)+\sum_{t=1}^{\infty}\beta^t U(S_t,a_t), \quad (14)$$

where the first term is the immediate utility of current state $S_0$, the second term is the expected utilities of the future states beginning from the initial state $S_0$, and $\beta^t$ is a discount factor series which ensures the summation is bounded. Bellman equation is usually written by a recursive form as follow

$$V(S)=\max_{a_S}U(S,a_S)+\beta\sum_{S'\in\mathcal{X}}P(S'|S,a_S)V(S'), \quad (15)$$

where $S'$ represents all possible next states of $S$ and $P(S'|S)$ is the transition probability. From the definition of Bellman equation, we can see that it not only considers the immediate utility, but also takes into account the future utilities, which is highly accord with the customer's expected utility in our Chinese Restaurant Game. Therefore, we can define a customer's expected utility at table $i$, $V_i(S)$, based on Bellman equation by

$$V_i(S)=U_i(S)+(1-\mu)\sum_{S'\in\mathcal{X}}P_i(S'|S)V_i(S'), \quad (16)$$

where $(1-\mu)$ is the discount factor, which can be regarded as the probability that the customer keeps staying at the selected table since $\mu$ is the departure probability. $P_i(S'|S)$ is the state transition probability defined as

$$P_i\big(S'=(\boldsymbol{B}',\boldsymbol{G}')|S=(\boldsymbol{B},\boldsymbol{G})\big)=P\big(\boldsymbol{B}'|\boldsymbol{B}\big)P_i\big(\boldsymbol{G}'|\boldsymbol{G}\big), \quad (17)$$

where $P\big(\boldsymbol{B}'|\boldsymbol{B}\big)$ is the belief state transition probability, and $P_i\big(\boldsymbol{G}'|\boldsymbol{G}\big)$ is the grouping state transition probability conditioned on that customers in table $i$ still stay at table $i$ in the next state $S'$, which is different with $P\big(\boldsymbol{G}'|\boldsymbol{G}\big)$ in (9-12). Note that $P_i\big(\boldsymbol{G}'|\boldsymbol{G}\big)$ is closely related to the new arriving customer's action. Suppose that the new customer's action $a_S=k$, i.e., choosing table $k$ at state $S$, we have the arriving transition probability in (18). For the leaving transition probability, since we have considered the discount factor $(1-\mu)$ in the future utility, i.e., the customer will not leave the restaurant, thus

$$P_i\big(\boldsymbol{G}'=(g_1,g_2,...,g_k+1,...,g_N)|\boldsymbol{G}=(g_1,g_2,...,g_k,...,g_N)\big)\;=\;\lambda. \quad (18)$$

$$P_i\big(\boldsymbol{G}'=(g_1,g_2,...,g_i-1,...,g_N)|\boldsymbol{G}=(g_1,g_2,...,g_i,...,g_N)\big)\;=\;(g_i-1)\mu, \quad (19)$$

$$P_i\big(\boldsymbol{G}'=(g_1,g_2,...,g_{i'\neq i}-1,...,g_N)|\boldsymbol{G}=(g_1,g_2,...,g_{i'\neq i},...,g_N)\big)\;=\;g_{i'}\mu,\quad \big(\forall i'\in\{1,2,\ldots,N\}\big), \quad (20)$$

$$P_i\big(\boldsymbol{G}'=\boldsymbol{G}|\boldsymbol{G}=(g_1,g_2,...,g_N)\big)\;=\;1-\lambda-\Big(\sum_{i=1}^{N}g_i-1\Big)\mu. \quad (21)$$

$$\begin{bmatrix}V_1(S)\\V_2(S)\\\vdots\\V_N(S)\end{bmatrix}=\begin{bmatrix}U_1(S)\\U_2(S)\\\vdots\\U_N(S)\end{bmatrix}+(1-\mu)\begin{bmatrix}\mathbf{P}_1(S'|S)&\mathbf{0}&\ldots&\mathbf{0}\\\mathbf{0}&\mathbf{P}_2(S'|S)&\ldots&\mathbf{0}\\\vdots&\vdots&\ddots&\vdots\\\mathbf{0}&\mathbf{0}&\ldots&\mathbf{P}_N(S'|S)\end{bmatrix}\begin{bmatrix}\mathbf{V}_1(S')\\\mathbf{V}_2(S')\\\vdots\\\mathbf{V}_N(S')\end{bmatrix}. \quad (22)$$

**Algorithm 1** Modified Value Iteration Algorithm for Multi-dimensional MDP Problem.

• Given tolerance $\eta_1$ and $\eta_2$, set $\epsilon_1$ and $\epsilon_2$.
• Initialize $\{V_i^{(0)}(S) = 0, \forall S \in \mathcal{X}\}$ and randomize
  $\pi = \{a_S, \forall S \in \mathcal{X}\}$.
**while** $\epsilon_1 > \eta_1$ or $\epsilon_2 > \eta_2$ **do**
  **for** all $S \in \mathcal{X}$ **do**
    • Calculate $\mathbf{P}_i(S'|S), \forall i \in \{1, 2, \ldots, N\}$ using $\pi$ and (17-21).
    • Update $\mathbf{V}_i^{(n+1)}(S), \forall i \in \{1, 2, \ldots, N\}$ using (22).
  **end for**
  **for** all $S \in \mathcal{X}$ **do**
    • Update $\pi^\star = \{a_S\}$ using (23).
  **end for**
  • Update the parameter $\epsilon_1$ by $\epsilon_1 = \|\pi - \pi^\star\|_2$.
  • Update the parameter $\epsilon_2$ by $\epsilon_2 = \left\|\mathbf{V}_i^{(n+1)}(S) - \mathbf{V}_i^{(n)}(S)\right\|_2$.
  • Update the strategy file $\pi = \pi^\star$.
**end while**
• The optimal strategy profile is $\pi^\star$.

we have (19) and (20), where the item $(g_i - 1)$ is because the grouping at table $i$, $g_i$, already includes this customer who will not leave the table at state $S'$. (21) is the staying probability. In such a case, we can have an multi-dimensional expected utility function set as (22), where $\mathbf{P}_i(S'|S) = \left[P_i(S'|S)|\forall S' \in \mathcal{X}\right]$ and $\mathbf{V}_i(S'|S) = \left[V_i(S'|S)|\forall S' \in \mathcal{X}\right]^T$.

*5) Best Strategy:* The strategy profile $\pi = \{a_S|\forall S \in \mathcal{X}\}$ is a mapping from the state space to the action space, i.e., $\pi : \mathcal{X} \to \mathcal{A}$. Due to the selfish nature, each customer will choose the best strategy to maximize his/her own expected utility. Here, we first give the definition of Nash equilibrium in the Dynamic Chinese Restaurant Game.

*Definition 1: A strategy profile $\pi^\star$ is a Nash equilibrium of the Dynamic Chinese Restaurant Game, if and only if, when all customers adopt $\pi^\star$, for each new arriving customer, his/her utility of adopting any other strategy profile $\pi \neq \pi^\star$ is always no more than that of adopting $\pi^\star$.*

From *Definition 1*, we can see that the utility of each customer can be damaged if he/she unilaterally deviates from the Nash equilibrium. Suppose that one customer arrives at the restaurant with system state $S = \big(\boldsymbol{B}, \boldsymbol{G} = (g_1, g_2, ..., g_i, ..., g_N)\big)$, his/her best strategy can be defined as

$$a_S = \underset{i\in\{1,2,...,N\}}{\operatorname{argmax}} \Big\{V_i\big(\boldsymbol{B}, \boldsymbol{G} = (g_1, ..., g_i + 1, ..., g_N)\big)\Big\}. \tag{23}$$

Since the strategy profile satisfying (22) and (23), denoted by $\pi^\star$, maximizes every arriving customer's utility, $\pi^\star$ is a Nash equilibrium of the proposed game.

*6) Modified Value Iteration Algorithm:* As discussed at the beginning of Section II-C, although the table selection problem of Chinese Restaurant Game can be modeled as an MDP problem, it is different from the traditional MDP problem that the customer cannot adjust action even if the system state changes. In traditional MDP problem, there is only one Bellman equation associated with each system state, and the optimal strategy is directly obtained by optimizing the Bellman equation. In our Multi-dimensional MDP problem, there is a set of Bellman equations as shown in (22) and the optimal strategy profile should satisfy (22) and (23) simultaneously. Therefore, the traditional dynamic programming method in [16] cannot be directly applied. To solve this problem, we design a modified value iteration algorithm.

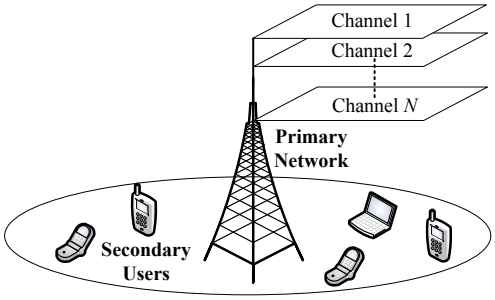


Fig. 3. System model of the cognitive radio network.

Given an initial strategy profile $\pi$, the conditional state transition probability $\mathbf{P}_i(S'|S)$ can be calculated by (17-21), and thus the conditional expected utility $\mathbf{V}_i(S)$ can be found by (22). Then, with $\mathbf{V}_i(S)$, the strategy profile $\pi$ can be updated again using (23). Through such an iterative way, we can finally find the optimal strategy $\pi^\star$. In Algorithm 1, we summarize the proposed modified value iteration algorithm for the Multi-dimensional MDP problem.

## III. Application to Cognitive Radio Networks

In this section, we study the application of the proposed Dynamic Chinese Restaurant Game in cognitive radio networks. In a cognitive radio network, SUs can opportunistically utilize the PU's licensed spectrum bands without harmful interference to the PU. The SUs who intend to access the primary channel should first perform spectrum sensing to check whether the PU is absent, which is known as "Listen-before-Talk" [17]. In order to counter the channel fading and shadowing problem, cooperative spectrum sensing technology was proposed recently, in which SUs share their spectrum sensing results with each other [18]. After spectrum sensing, each SU chooses one primary channel to access for data transmission. However, traditional cooperative sensing schemes simply combine all SUs' sensing results while ignoring the structure of sequential decision making [19], especially in a dynamic scenario where SUs arrive and leave stochastically. Moreover, the negative network externality has not been considered in the previous channel access methods [20].

The spectrum sensing and access in cognitive radio networks can be ideally modeled as a Dynamic Chinese Restaurant Game, where the tables are the primary channels which may be reserved by the PU, and customers are the SUs who are seeking available channel. With the proposed Dynamic Chinese Restaurant Game, how a SU utilizes other SUs' sensing results to learn the primary channel state can be regarded as how a customer learns the table state, while how a SU chooses a channel to access by predicting subsequent SUs' decisions can be formulated as how a customer selects a table. Although the spectrum sensing and access problem has also been modeled using game theory as in [21]-[23], the SUs' sequential decision making structure has not been well investigated. In the following, we will discuss in details how to apply the proposed Dynamic Chinese Restaurant Game to cognitive radio networks.

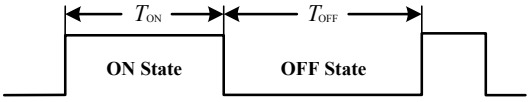


Fig. 4. ON-OFF primary channel.

### A. System Model

*1) Network Entity:* As shown in Fig. 3, we consider a primary network with $N$ independent primary channels. The PU has priority to occupy the channels at any time, while SUs are allowed to access the channel under the condition that the PU's communication QoS is guaranteed. Mixed underlay and overlay spectrum sharing are adopted in our model, which means SUs should detect PUs' existences and interference to the PUs should also be minimized [24]. We denote the primary channel state as $\boldsymbol{\theta} = \{\theta_1, \theta_2, ..., \theta_N\}$ and $\theta_i \in \{\mathcal{H}_0, \mathcal{H}_1\}$, where $\mathcal{H}_0$ is the hypothesis that the PU is absent and $\mathcal{H}_1$ means the PU is present.

For the secondary network, SUs arrive and depart by Bernoulli process with probability $\lambda$ and $\mu$, respectively. All SUs can independently perform spectrum sensing using energy detection method. Here, we use a simple binary model on the spectrum sensing result, where $s_i^j = 1$ if the $j$th SU detects some activity on channel $i$ and $s_i^j = 0$ if no activity is detected on channel $i$. In such a case, the detection and false-alarm probability of channel $i$ can be expressed as $P_i^d = \Pr(s_i = 1|\theta_i = \mathcal{H}_1)$ and $P_i^f = \Pr(s_i = 1|\theta_i = \mathcal{H}_0)$, which are considered as common priors for all SUs. Moreover, we assume that there is a log-file in the server of the secondary network, which records each SU's channel belief and channel selection result. Through querying this log-file, the new coming SU can obtain current grouping state information, i.e., the number of SUs in each primary channel, as well as the former SU's belief on the channel state.

*2) ON-OFF Primary Channel Model:* For the PU's behavior in the primary channel, we model it as a general alternating ON-OFF renewal process. The ON state means the channel is occupied by the PU, while the OFF state is the "spectrum hole" which can be freely accessed by SUs, as shown in Fig. 4. This general ON-OFF switch model can be applied in the scenario when SUs have no knowledge about the PU's exact communication mechanism [25]. Let $T_{\text{ON}}$ and $T_{\text{OFF}}$ denote the length of the ON state and OFF state, respectively. According to different types of the primary services (e.g., digital TV broadcasting or cellular communication), $T_{\text{ON}}$ and $T_{\text{OFF}}$ statistically satisfy different types of distributions. Here we assume that $T_{\text{ON}}$ and $T_{\text{OFF}}$ are independent and satisfy exponential distributions with parameter $r_1$ and $r_0$, respectively [26], denoted by $f_{\text{ON}}(t)$ and $f_{\text{OFF}}(t)$ as follows:

$$\begin{cases} T_{\text{ON}} \sim f_{\text{ON}}(t) = \frac{1}{r_1} e^{-t/r_1}, \\ T_{\text{OFF}} \sim f_{\text{OFF}}(t) = \frac{1}{r_0} e^{-t/r_0}. \end{cases} \tag{24}$$

In such a case, the expected lengths of the ON state and OFF state are $r_1$ and $r_0$ accordingly. These two parameters $r_1$ and $r_0$ can be effectively estimated by a maximum likelihood estimator [27]. Such an ON-OFF behavior of the PU is a combination of two Poisson process, which is a renewal process [28]. The renewal interval is $T_p = T_{\text{ON}} + T_{\text{OFF}}$ and the distribution of $T_p$, denoted by $f_p(t)$, is

$$f_p(t) = f_{\text{ON}}(t) * f_{\text{OFF}}(t), \tag{25}$$

where the symbol "$*$" represents the convolution operation.

### B. Bayesian Channel Sensing

In this subsection, we discuss how SUs estimate the primary channel state using Bayesian learning rule. Let us define the continuous belief of the $j$th SU on the state of channel $i$ as $b_i^j = \Pr(\theta_i^j = \mathcal{H}_0)$, and the quantized belief as $B_i^j \in \{\mathbb{B}_1, \mathbb{B}_2, ..., \mathbb{B}_M\}$, where $B_i^j = \mathbb{B}_k$ if $b_i^j \in \left[\frac{k-1}{M}, \frac{k}{M}\right]$. Since all primary channels are assumed to be independent, the learning processes of these channels are also independent. In such a case, for channel $i$, the $j$th SU can receive the belief of former SU choosing channel $i$, $B_i^{j-1}$, and his/her own sensing result, $s_i^j$. As discussed in Section II-B, the learning process is $B_i^{j-1} \xrightarrow{\text{Mapping}} \hat{b}_i^{j-1} \xrightarrow{\text{Bayesian}+s_i^j} b_i^j \xrightarrow{\text{Quantize}} B_i^j$, where $\hat{b}_i^{j-1} = \left(\frac{1}{k-1} + \frac{1}{k}\right)/2$ when $B_i^{j-1} = \mathbb{B}_k$, and $b_i^j$ can be derived according to (5) using Bayesian learning rule as (26).

To compute belief $b_i^j$, we need to first derive the primary channel state transition probabilities in (26). Since the primary channel is modeled as an ON-OFF process, the channel state transition probability depends on the time interval between the $(j-1)$th and $j$th SUs' arrival time, $t^j$. Note that the $t^j$ can be directly obtained from the log-file in the server. For notation simplicity, in the following, we will use $P_{00}(t^j)$, $P_{01}(t^j)$, $P_{10}(t^j)$ and $P_{11}(t^j)$ to denote $\Pr(\theta_i^j = \mathcal{H}_0|\theta_i^{j-1} = \mathcal{H}_0)$, $\Pr(\theta_i^j = \mathcal{H}_1|\theta_i^{j-1} = \mathcal{H}_0)$, $\Pr(\theta_i^j = \mathcal{H}_0|\theta_i^{j-1} = \mathcal{H}_1)$ and $\Pr(\theta_i^j = \mathcal{H}_1|\theta_i^{j-1} = \mathcal{H}_1)$, respectively, where $P_{01}(t^j) = 1 - P_{00}(t^j)$ and $P_{11}(t^j) = 1 - P_{10}(t^j)$.

The close-form expression for $P_{01}(t^j)$ can be derived using the renewal theory as follow [29]

$$P_{01}(t^j) = \frac{r_1}{r_0 + r_1}\left(1 - e^{-\frac{r_0 + r_1}{r_0 r_1} t^j}\right). \tag{27}$$

Thus, we can have $P_{00}(t^j)$ as

$$P_{00}(t^j) = 1 - P_{01}(t^j) = \frac{r_1}{r_0 + r_1}\left(\frac{r_0}{r_1} + e^{-\frac{r_0 + r_1}{r_0 r_1} t^j}\right). \tag{28}$$

Similarly, the close-form expression for $P_{11}(t^j)$ can also be obtained by the renewal theory as follows.

*Lemma 1:* $P_{11}(t)$ satisfies the renewal equation given by

$$P_{11}(t) = r_1 f_{\text{ON}}(t) + \int_0^t P_{11}(t-w) f_p(w) dw, \tag{29}$$

where $f_{\text{ON}}(t)$ is the probability density function (*p.d.f*) of the ON state's length given in (24) and $f_p(t)$ is the *p.d.f* of the PU's renewal interval given in (25).

$$b_i^j = \frac{\left(\Pr(\theta_i^j = \mathcal{H}_0|\theta_i^{j-1} = \mathcal{H}_0)\hat{b}_i^{j-1} + \Pr(\theta_i^j = \mathcal{H}_0|\theta_i^{j-1} = \mathcal{H}_1)(1 - \hat{b}_i^{j-1})\right)\Pr(s_i^j|\theta_i^j = \mathcal{H}_0)}{\sum_{l=0}^{1}\left(\Pr(\theta_i^j = \mathcal{H}_l|\theta_i^{j-1} = \mathcal{H}_0)\hat{b}_i^{j-1} + \Pr(\theta_i^j = \mathcal{H}_l|\theta_i^{j-1} = \mathcal{H}_1)(1 - \hat{b}_i^{j-1})\right)\Pr(s_i^j|\theta_i^j = \mathcal{H}_l)}. \tag{26}$$

*Proof:* See Appendix A. ■

By solving (29) in *Lemma 1*, we can obtain the close-form expression for $P_{11}(t^j)$ given by

$$P_{11}(t^j) = \frac{r_0}{r_0+r_1}\Big(\frac{r_1}{r_0} + e^{-\frac{r_0+r_1}{r_0 r_1}t^j}\Big). \tag{30}$$

Then, we can have $P_{10}(t_i)$ as

$$P_{10}(t^j) = 1 - P_{11}(t^j) = \frac{r_0}{r_0+r_1}\Big(1 - e^{-\frac{r_0+r_1}{r_0 r_1}t^j}\Big). \tag{31}$$

By substituting (27-28) and (30-31) into (26), we can calculate the $j$th SU's belief $b_i^j$ with the corresponding sensing results $s_i^j = 1$ and $s_i^j = 0$ by (32) and (33), respectively. For simplicity, in the following, we denote (32) as $b_i^j|_{s_i^j=1} = \phi(\hat{b}_i^{j-1}, t_i, s_i^j = 1)$, and denote (33) as $b_i^j|_{s_i^j=0} = \phi(\hat{b}_i^{j-1}, t_i, s_i^j = 0)$ .

### C. Belief State Transition Probability

In this subsection, we will discuss how to calculate the belief state transition probability matrix of each channel, i.e., $\Pr(B_i^j = \mathbb{B}_q | B_i^{j-1} = \mathbb{B}_p)$. The belief state transition probability can be obtained according to the learning rules $B_i^{j-1} \xrightarrow{\text{Mapping}} \hat{b}_i^{j-1} \xrightarrow{\text{Bayesian}+s_i^j} b_i^j \xrightarrow{\text{Quantize}} B_i^j$. Note that $\hat{b}_i^{j-1} = \frac{1}{2}\big(\frac{p-1}{M} + \frac{p}{M}\big)$ if $B_i^{j-1} = \mathbb{B}_p$, and $b_i^j \in [\frac{q-1}{M}, \frac{q}{M}]$ if $B_i^j = \mathbb{B}_q$. In such a case, the belief state transition probability can be calculated by

$$\begin{aligned}&\Pr(B_i^j = \mathbb{B}_q | B_i^{j-1} = \mathbb{B}_p)\\ &= \int_{\frac{q-1}{M}}^{\frac{q}{M}} \Pr\Big(b_i^j \Big| \hat{b}_i^{j-1} = \frac{1}{2}\Big(\frac{p-1}{M} + \frac{p}{M}\Big)\Big) db_i^j.\end{aligned} \tag{34}$$

According to (32) and (33), we have $b_i^j = \phi\Big(\hat{b}_i^{j-1} = \frac{1}{2}\big(\frac{p-1}{M} + \frac{p}{M}\big), t^j, s_i^j\Big)$. Therefore, the belief state transition probability can be re-written by (35), where the second equality follows the assumption that the arrival interval of two SUs $t^j$ obeys exponential distribution with parameter $\lambda$ and is independent with the belief. To calculate (35), we need to derive $\Pr(s_i^j|\hat{b}_i^{j-1})$, which represents the distribution of the $j$th SU's received signal when given the $(j-1)$th SU's belief. Note that given current channel state $\theta_i^j$, signal $s_i^j$ is independent with belief $\hat{b}_i^{j-1}$. Thus, $\Pr(s_i^j|\hat{b}_i^{j-1})$ can be calculated as follows:

$$\begin{aligned}\Pr(s_i^j|\hat{b}_i^{j-1}) = {} & f(s_i^j|\theta_i^j = \mathcal{H}_0)\Pr(\theta_i^j = \mathcal{H}_0|\hat{b}_i^{j-1}) +\\ & f(s_i^j|\theta_i^j = \mathcal{H}_1)\Pr(\theta_i^j = \mathcal{H}_1|\hat{b}_i^{j-1}).\end{aligned} \tag{36}$$

Moreover, given the previous channel state $\theta_i^{j-1}$, current state $\theta_i^j$ is also independent with the former SU's belief $\hat{b}_i^{j-1}$. In such a case, $\Pr(\theta_i^j = \mathcal{H}_0|\hat{b}_i^{j-1})$ in (36) can be obtained as:

$$\begin{aligned}\Pr(\theta_i^j = \mathcal{H}_0|\hat{b}_i^{j-1}) &= \Pr(\theta_i^j = \mathcal{H}_0|\theta_i^{j-1} = \mathcal{H}_0)\hat{b}_i^{j-1} +\\ &\quad \Pr(\theta_i^j = \mathcal{H}_0|\theta_i^{j-1} = \mathcal{H}_1)(1-\hat{b}_i^{j-1}),\\ &= P_{00}(t^j)\hat{b}_i^{j-1} + P_{10}(t^j)(1-\hat{b}_i^{j-1}).\end{aligned} \tag{37}$$

Similarly, for $\Pr(\theta_i^j = \mathcal{H}_1|\hat{b}_i^{j-1})$, we have

$$\Pr(\theta_i^j = \mathcal{H}_1|\hat{b}_i^{j-1}) = P_{01}(t^j)\hat{b}_i^{j-1} + P_{11}(t^j)(1-\hat{b}_i^{j-1}). \tag{38}$$

By substituting (37-38) into (36), the conditional distribution of the signal can be obtained as (39) and (40).

Finally, with (39-40), we can calculate the belief transition probability matrix using (35).

### D. Channel Access: Two Primary Channels Case

In this subsection, we discuss the case where there are two primary channels. In such a case, the system state $S = (B_1, B_2, g_1, g_2)$, where $B_1$ and $B_2$ are beliefs of two channels, $g_1$ and $g_2$ are numbers of SUs in two channels. We define the immediate utility of SUs in channel $i$, $U(B_i, g_i)$, as

$$U(B_i, g_i) = \hat{b}_i R(g_i) = \hat{b}_i \log\left(1 + \frac{\text{SNR}}{(g_i-1)\text{INR}+1}\right), \tag{41}$$

where $\hat{b}_i$ is the mapping of quantized belief $B_i$, SNR is the average signal-noise-ratio of the SUs and INR is the average interference-noise-ratio.

---

$$b_i^j|_{s_i^j=1} = \frac{\Big(r_0 e^{\frac{r_0+r_1}{r_0 r_1}t^j} - r_0 + (r_1+r_0)\hat{b}_i^{j-1}\Big)P_i^f}{\Big(r_0 e^{\frac{r_0+r_1}{r_0 r_1}t^j} - r_0 + (r_1+r_0)\hat{b}_i^{j-1}\Big)P_i^f + \Big(r_1 e^{\frac{r_0+r_1}{r_0 r_1}t^j} + r_0 - (r_1+r_0)\hat{b}_i^{j-1}\Big)P_i^d}, \tag{32}$$

$$b_i^j|_{s_i^j=0} = \frac{\Big(r_0 e^{\frac{r_0+r_1}{r_0 r_1}t^j} - r_0 + (r_1+r_0)\hat{b}_i^{j-1}\Big)(1-P_i^f)}{\Big(r_0 e^{\frac{r_0+r_1}{r_0 r_1}t^j} - r_0 + (r_1+r_0)\hat{b}_i^{j-1}\Big)(1-P_i^f) + \Big(r_1 e^{\frac{r_0+r_1}{r_0 r_1}t^j} + r_0 - (r_1+r_0)\hat{b}_i^{j-1}\Big)(1-P_i^d)}. \tag{33}$$

$$\begin{aligned}\Pr(B_i^j = \mathbb{B}_q|B_i^{j-1} = \mathbb{B}_p) &= \iint_{\frac{q-1}{M} \le \phi\left(\hat{b}_i^{j-1} = \frac{1}{2}(\frac{p-1}{M}+\frac{p}{M}), t^j, s_i^j\right) \le \frac{q}{M}} \Pr\big(t^j, s_i^j|\hat{b}_i^{j-1}\big) dt^j ds^j,\\ &= \int_{\frac{q-1}{M} \le \phi\left(\hat{b}_i^{j-1} = \frac{1}{2}(\frac{p-1}{M}+\frac{p}{M}), t^j, s_i^j = 0\right) \le \frac{q}{M}} \lambda e^{-\lambda t^j}\Pr(s_i^j = 0|\hat{b}_i^{j-1})dt^j + \int_{\frac{q-1}{M} \le \phi\left(\hat{b}_i^{j-1} = \frac{1}{2}(\frac{p-1}{M}+\frac{p}{M}), t^j, s_i^j = 1\right) \le \frac{q}{M}} \lambda e^{-\lambda t^j}\Pr(s_i^j = 1|\hat{b}_i^{j-1})dt^j.\end{aligned} \tag{35}$$

$$\Pr(s_i^j = 0|\hat{b}_i^{j-1}) = \big(1-P_i^f\big)\Big(P_{00}(t^j)\hat{b}_i^{j-1} + P_{10}(t^j)(1-\hat{b}_i^{j-1})\Big) + \big(1-P_i^d\big)\Big(P_{01}(t^j)\hat{b}_i^{j-1} + P_{11}(t^j)(1-\hat{b}_i^{j-1})\Big), \tag{39}$$

$$\Pr(s_i^j = 1|\hat{b}_i^{j-1}) = P_i^f\Big(P_{00}(t^j)\hat{b}_i^{j-1} + P_{10}(t^j)(1-\hat{b}_i^{j-1})\Big) + P_i^d\Big(P_{01}(t^j)\hat{b}_i^{j-1} + P_{11}(t^j)(1-\hat{b}_i^{j-1})\Big). \tag{40}$$

According to (22), the expected utility functions of two channels can be written as

$$V_1(S) = U(B_1, g_1) + (1-\mu)\sum_{S'\in\mathcal{X}} P_1(S'|S)V_1(S'), \quad (42)$$

$$V_2(S) = U(B_2, g_2) + (1-\mu)\sum_{S'\in\mathcal{X}} P_2(S'|S)V_2(S'), \quad (43)$$

where $P_1$ and $P_2$ are the state transition probabilities conditioned on the event that SUs stay in the channels they have chosen. According to (17-21), we can summarize $P_1$ and $P_2$ as (44) and (45), where $\mathbf{1}(a_S)$ is an indicator function defined by

$$\mathbf{1}(a_S) = \begin{cases} 1 & \text{if } a_S = 1, \text{ i.e., selecting channnel 1,} \\ 0 & \text{if } a_S = 2, \text{ i.e., selecting channnel 2.} \end{cases} \quad (46)$$

According to (23), we can have the best strategy $a_S$ for SUs arriving with system state $S = (B_1, B_2, g_1, g_2)$ as follows:

$$a_S = \begin{cases} 1, & V_1(B_1,B_2,g_1+1,g_2) \geq V_2(B_1,B_2,g_1,g_2+1), \\ 2, & V_1(B_1,B_2,g_1+1,g_2) < V_2(B_1,B_2,g_1,g_2+1). \end{cases} \quad (47)$$

Thus, with (41-47), we can find the optimal strategy profile $\pi^\star = \{a_S, \forall S \in \mathcal{X}\}$ using the modified value iteration method in Algorithm 1. In the following, we will show that when given the beliefs of two channel, there exists a threshold structure in the optimal strategy profile $\pi^\star$.

*Lemma 2:* The value function $V_1$ and $V_2$ updated by Algorithm 1 have the quality that for any $g_1 \geq 0$ and $g_2 \geq 1$,

$$V_1(B_1, B_2, g_1, g_2) \geq V_1(B_1, B_2, g_1+1, g_2-1), \quad (48)$$

$$V_2(B_1, B_2, g_1, g_2) \leq V_2(B_1, B_2, g_1+1, g_2-1). \quad (49)$$

*Proof:* See Appendix B. ∎

*Lemma 2* shows that given the beliefs of two channels, $V_1$ is non-decreasing and $V_2$ is non-increasing along the line of $g_1+g_2 = m, \forall m \in \{0, 1, ..., 2L\}$. Based on *Lemma 2*, we will show the threshold structure in the optimal strategy profile $\pi^\star$ by *Theorem 1*.

*Theorem 1:* For the two-channel case, given the belief state, the optimal strategy profile $\pi^\star = \{a_S\}$ derived from the modified value iteration algorithm has threshold structure as follows:

$$\text{If } a_{S=(B_1,B_2,g_1,g_2)} = 1, \text{then } a_{S=(B_1,B_2,g_1-g',g_2+g')} = 1. \quad (50)$$

$$\text{If } a_{S=(B_1,B_2,g_1,g_2)} = 2, \text{then } a_{S=(B_1,B_2,g_1+g',g_2-g')} = 2. \quad (51)$$

*Proof:* According to *Lemma 2*, we can have

$$V_1(B_1, B_2, g_1+1, g_2) - V_2(B_1, B_2, g_1, g_2+1) \geq V_1(B_1, B_2, g_1+2, g_2-1) - V_2(B_1, B_2, g_1+1, g_2), \quad (52)$$

which shows that the difference of $V_1$ and $V_2$ is non-decreasing along $g_1 + g_2 = m, \forall m \in \{0, 1, ..., 2L\}$. In such a case, on one hand, if $V_1(B_1, B_2, g_1+1, g_2) \leq V_2(B_1, B_2, g_1, g_2+1)$, i.e., $a_{S=(B_1,B_2,g_1,g_2)} = 2$, then for any $g' > 0$, $V_1(B_1, B_2, g_1+g'+1, g_2-g') \leq V_2(B_1, B_2, g_1+g', g_2-g'+1)$, i.e., $a_{S=(B_1,B_2,g_1+g',g_2-g')} = 2$. On the other hand, if $V_1(B_1, B_2, g_1+1, g_2) \geq V_2(B_1, B_2, g_1, g_2+1)$, i.e., $a_{S=(B_1,B_2,g_1,g_2)} = 1$, then for any $g' > 0$, $V_1(B_1, B_2, g_1-g'+1, g_2+g') \geq V_2(B_1, B_2, g_1-g', g_2+g'+1)$ which means $a_{S=(B_1,B_2,g_1-g',g_2+g')} = 1$. Therefore, we can conclude that if $a_{S=(B_1,B_2,g_1,g_2)} = 1$, then the upper left of line $g_1+g_2 = m$ will be all 1, and if $a_{S=(B_1,B_2,g_1,g_2)} = 2$, then the lower right of line $g_1 + g_2 = m$ will be all 2. Thus, there exists some threshold on the line of $g_1 + g_2 = m$. ∎

Note that the optimal strategy profile $\pi^\star$ can be obtained off-line and the profile can be stored in a table in advance. We can see that for some fixed belief state, the number of system states is $(L+1)^2$, which means the corresponding strategy file has $(L+1)^2$ strategies. With the proved threshold structure on each line $g_1 + g_2 = m, \forall m \in \{0, 1, \ldots, 2L\}$, we just need to store the threshold point on each line. In such a case, the storage of the strategy profile can be reduced from $\mathcal{O}(L^2)$ to $\mathcal{O}(2L)$.

**Algorithm 2** Fast Algorithm for the Multi-channel Case.

```
if N is even then
    while N > 1 do
        • Randomly divide the N primary channels into N/2 pairs.
        for all N/2 pairs do
            • Select one channel from each pair according to Algorithm 1.
        end for
        • N = N/2.
    end while
end if
if N is odd then
    while N > 1 do
        • Randomly divide the N primary channels into
          (N − 1)/2 pairs and one channel.
        for all (N − 1)/2 pairs do
            • Select one channel from each pair according to Algorithm 1.
        end for
        • N = (N − 1)/2 + 1.
    end while
end if
```

### *E. Channel Access: Multiple Primary Channels Case*

In this subsection, we discuss the case where there are multiple primary channels. Although the optimal strategy profile of the multi-channel case can also be obtained using

$$P_1(S'|S) = P\Big((B_1', B_2')|(B_1, B_2)\Big) \cdot \begin{cases} \mathbf{1}(a_S)\lambda & \text{if } S' = (B_1', B_2', g_1+1, g_2), \\ \big(1-\mathbf{1}(a_S)\big)\lambda & \text{if } S' = (B_1', B_2', g_1, g_2+1), \\ (g_1-1)\mu & \text{if } S' = (B_1', B_2', g_1-1, g_2), \\ g_2\mu & \text{if } S' = (B_1', B_2', g_1, g_2-1), \\ 1-\lambda-(g_1+g_2-1)\mu & \text{if } S' = (B_1', B_2', g_1, g_2), \end{cases} \quad (44)$$

$$P_2(S'|S) = P\Big((B_1', B_2')|(B_1, B_2)\Big) \cdot \begin{cases} \mathbf{1}(a_S)\lambda & \text{if } S' = (B_1', B_2', g_1+1, g_2), \\ \big(1-\mathbf{1}(a_S)\big)\lambda & \text{if } S' = (B_1', B_2', g_1, g_2+1), \\ g_1\mu & \text{if } S' = (B_1', B_2', g_1-1, g_2), \\ (g_2-1)\mu & \text{if } S' = (B_1', B_2', g_1, g_2-1), \\ 1-\lambda-(g_1+g_2-1)\mu & \text{if } S' = (B_1', B_2', g_1, g_2). \end{cases} \quad (45)$$

Algorithm 1, the computation complexity grows exponentially in terms of the number of primary channels $N$. Besides, the storage and retrieval of the strategy profile are also challenging when the number of system states exponentially increases with $N$. Therefore, it is important to develop a fast algorithm for the multi-channel case.

Suppose the channel number $N$ is even, we can randomly divide these $N$ primary channels into $N/2$ pairs. For each pair, SUs can choose one channel using the threshold strategy in *Theorem 1*. Then, SUs can further divide the selected $N/2$ channels into $N/4$ pairs and so on so forth. In such a case, SUs can finally select one suboptimal channel to access. On the other hand, if the channel number $N$ is odd, the suboptimal channel can be selected by a similar way. With such an iterative dichotomy method, a SU can find one suboptimal primary channel only by $\log N$ steps and the complexity of each step is same with that of the two-channel case. This fast algorithm is summarized in Algorithm 2. In the simulation section, we will compare the performance of this fast algorithm with the optimal algorithm using modified value iteration.

### F. Analysis of Interference to the PU

Since mixed underlay and overlay spectrum sharing are used in this paper, it is crucial to compute the interference to the PU and evaluate the impact on the PU's data transmission. In our system, the primary channel is based on ON-OFF model and SUs cannot be synchronous with the PU. In such a case, they may fail to discover the PU's recurrence when transmitting packets in the primary channel, which may cause interference to the PU [30].

As long as there are SUs in the primary channel, interference may occur to the PU. Therefore, we define the interference probability of channel $i$, $P_{Ii}$, as the probability that the number of SUs in this channel is non-zero. Given a strategy profile $\boldsymbol{\pi} = \{a_S\}$, the system state transition probability matrix $\boldsymbol{P}_s = \{P(S'|S), \forall S' \in \mathcal{X}, \forall S \in \mathcal{X}\}$ can be obtained according to (8-12). With $\boldsymbol{P}_s$, we then can derive the stationary distribution of the Markov chain, $\boldsymbol{\sigma} = \{\sigma(\boldsymbol{B}, \boldsymbol{G})\}$, by solving $\boldsymbol{\sigma}\boldsymbol{P}_s = \boldsymbol{\sigma}$. In such a case, the interference probability $P_{Ii}$ can be calculated by

$$P_{Ii} = 1 - \sum_{\boldsymbol{B}} \sum_{\boldsymbol{G}\setminus g_i} \sigma\Big(\boldsymbol{B}, \boldsymbol{G} = (g_1, g_2, ..., g_i = 0, ..., g_N)\Big). \quad (53)$$

If there is no interference from SUs, the PU's instantaneous rate is $\log(1 + \text{SNR}_p)$, where $\text{SNR}_p$ is the Signal-to-Noise Ratio of primary signal at the PU's receiver. On the other hand, if the interference occurs, the PU's instantaneous rate is $\log\left(1 + \frac{\text{SNR}_p}{\text{INR}_p + 1}\right)$, where $\text{INR}_p$ is the Interference-to-Noise Ratio of secondary signal received by the PU. Therefore, the PU's average data rate $R_i$ in channel $i$ can be calculated by

$$R_{pi} = (1 - P_{Ii}) \log\left(1 + \text{SNR}_p\right) + P_{Ii} \log\left(1 + \frac{\text{SNR}_p}{\text{INR}_p + 1}\right). \quad (54)$$

## IV. Simulation Results

In this section, we conduct simulations to evaluate the performance of proposed scheme in cognitive radio networks.

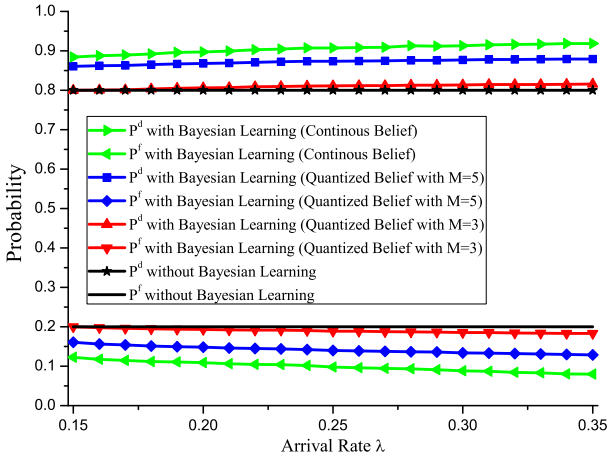


Fig. 5. Detection and false-alarm probability.

Specifically, we evaluate the performance of channel sensing and access, as well as the interference to the PU.

### A. Bayesian Channel Sensing

In this simulation, we evaluate the performance of channel sensing with Bayesian learning. We first generate one primary channel based on the ON-OFF model, and the channel parameters are set to be $r_0 = 55$s and $r_1 = 50$s, respectively. Then, a number of SUs with some arrival rate $\lambda$ sequentially sense the primary channel and construct their own beliefs by combining the sensing result with the former SU's belief. In Fig. 5, we compare the detection and false-alarm probabilities between channel sensing with Bayesian learning based on continuous belief, sensing with Bayesian learning based on quantized belief (belief level $M = 5$ and 3), and sensing without Bayesian learning under different arrival rate $\lambda$. Overall, the detection probability is enhanced and false-alarm probability decreases when the Bayesian learning is adopted. We can see that there are some performance loss due to the quantization operation of the beliefs and setting more belief levels can achieve less loss. Moreover, we can see that with Bayesian learning, the larger the arrival rate $\lambda$, the higher detection probability and the lower the false-alarm probability. This is because a larger $\lambda$ means a shorter arrival interval between two SUs, and thus the former SU's belief information is more useful for current SU's belief construction.

### B. Channel Access of Two Primary Channel Case

In this subsection, we evaluate the performance of the proposed Multi-dimensional MDP model, as well as the modified value iteration algorithm for the two-channel case. The parameters of the two primary channels are set to be: for channel 1, $r_0 = 55$s and $r_1 = 25$s; for channel 2, $r_0 = 25$s and $r_1 = 55$s, which means channel 1 is statistically better than channel 2. In Fig. 6, we first show the convergence performance of the proposed algorithm, where the X-axis is the iteration times and the Y-axis is the mean-square differences of two adjacent iterations, i.e., $\text{E}(||\pi(t+1) - \pi(t)||_2)$. We can see that the average iteration times are less than 20 iterations.

In the following simulations, our proposed strategy is compared with centralized strategy, myopic strategy and random

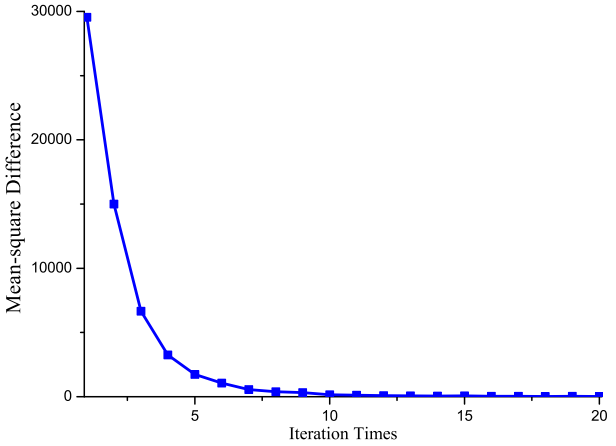


Fig. 6. Convergence performance of modified value iteration algorithm when $N = 3$, $M = 5$ and $L = 5$.

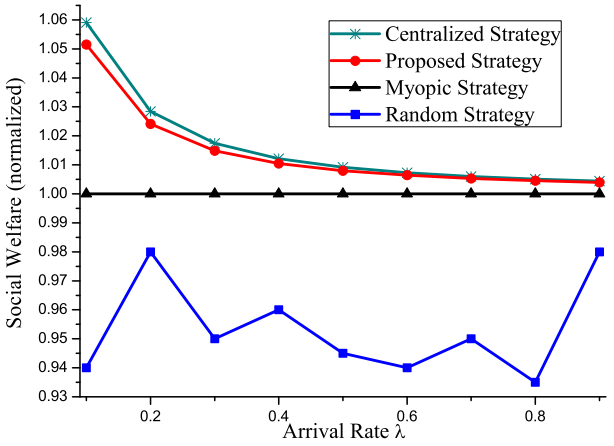


Fig. 7. Social welfare comparison under 2-channel with $M = 2$ and $L = 2$.

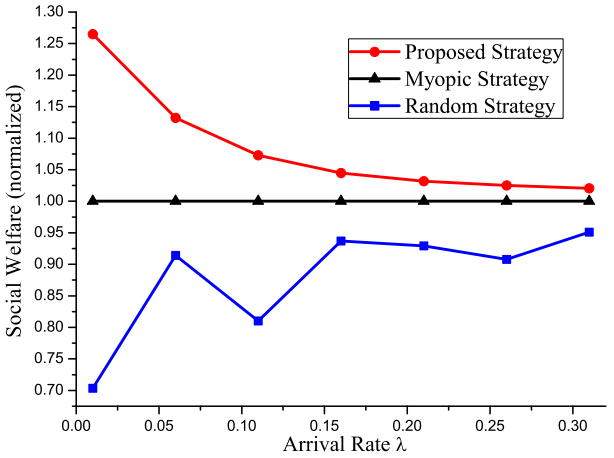


Fig. 8. Social welfare comparison under 2-channel with $M = 5$ and $L = 5$.

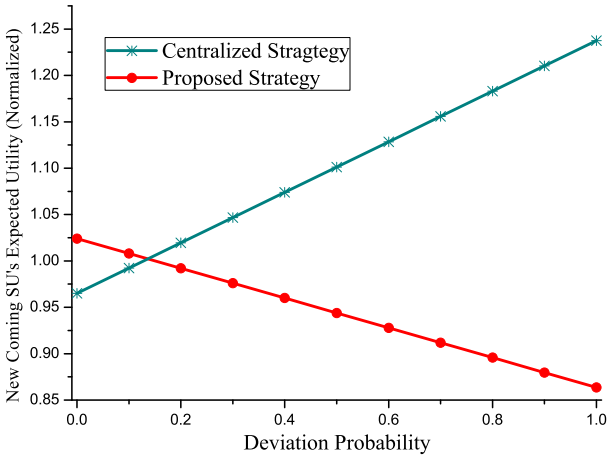


Fig. 9. NE verification under 2-channel with $M = 2$ and $L = 2$.

strategy in terms of social welfare. We first define the social welfare, $W$, when given a strategy profile $\pi = \{a_S, \forall S \in \mathcal{X}\}$ as

$$W = \sum_{S \in \mathcal{X}} \sigma^{\pi}(S)\Big(g_1 U(B_1, g_1) + g_2 U(B_2, g_2)\Big), \qquad (55)$$

where $S = (B_1, B_2, g_1, g_2)$ in the two-channel case, and $\sigma^{\pi}(S)$ is the stationary probability of state $S$. The four strategies we test are defined as follows.

- **Proposed strategy** is obtained by our proposed value iteration algorithm in Algorithm 1.
- **Centralized strategy** is obtained by exhaustively searching all possible $2^{|\mathcal{X}|}$ strategy profiles to maximize the social welfare, i.e., $\pi^c = \arg\max_{\pi} W^{\pi}$, where the superscript $c$ means centralized strategy. We can see that the complexity of finding the centralized strategy is NP-hard.
- **Myopic strategy** is to maximize the immediate utility, i.e., to choose the channel with the largest immediate reward by $\pi^m = \{a_S = \underset{i \in \{1,2\}}{\operatorname{argmax}} U(B_i, g_i), \forall S \in \mathcal{X}\}$, where the superscript $m$ means myopic strategy.
- **Random strategy** is to randomly choose one channel with equal probability 0.5, i.e., $\pi^r = \{a_S = \text{rand}(1,2), \forall S \in \mathcal{X}\}$, where the superscript $r$ means random strategy.

In the simulation, we use the myopic strategy as the comparison baseline and show the results by normalizing the performance of each strategy by that of the myopic strategy.

In Fig. 7, we evaluate the social welfare performance of different methods. Due to the extremely high complexity of the centralized strategy, we consider the case with 2 belief levels and maximally 2 SUs in each channel, i.e., $M = 2$ and $L = 2$. Note that if $M = 2$ and $L = 3$, there are totally $2^{2^{2\cdot(3+1)}} = 2^{64}$ possible strategy profiles to verify, which is computational intractable. Therefore, although slightly outperforming our proposed strategy as shown in Fig. 7, the centralized method is not applicable to the time-varying primary channels. Moreover, we also compare the proposed strategy with the myopic and random strategies under the case with $M = 5$ and $L = 5$ in Fig. 8. We can see that the proposed strategy performs the best among all the strategies.

We verify that the proposed strategy is a Nash equilibrium through simulating a new coming SU's expected utility in Fig. 9. The deviation probability in x-axis stands for the probability that a new coming SU deviates from the proposed strategy or centralized strategy. We can see that when there is no deviation, our proposed strategy performs better than the centralized strategy. Such a phenomenon is because the

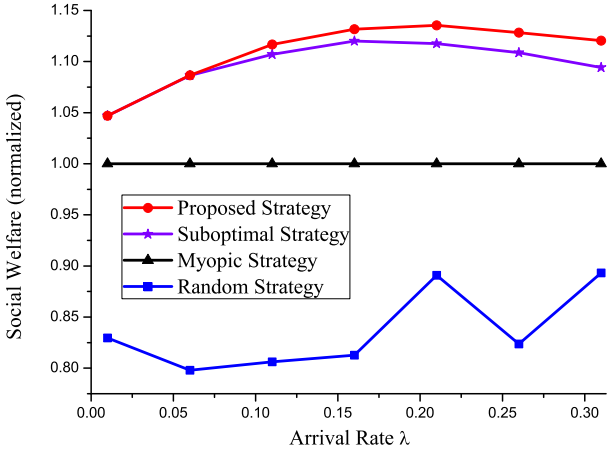


Fig. 10. Social welfare comparison under 3-channel case.

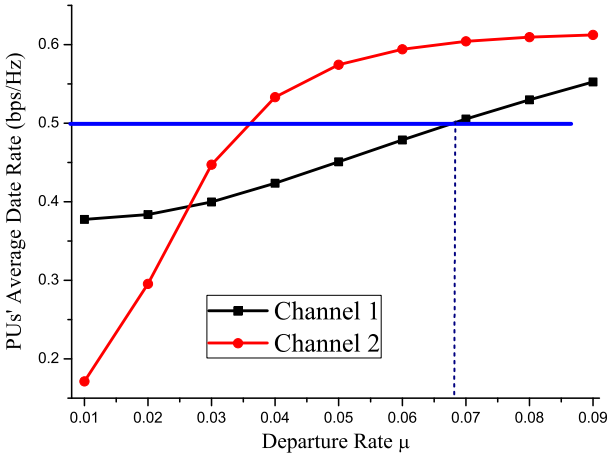


Fig. 11. PU's average data rate when $M = 5$ and $L = 5$.

centralized strategy is to maximize the social welfare and thus sacrifices the new coming SU's expected utility. Moreover, we can see that the expected utility of a new coming SU decreases as the deviation probability increases, which verifies that the proposed strategy is a Nash equilibrium. On the other hand, by deviating from the centralized strategy, a new coming SU can obtain higher utility, which means that the centralized strategy is not a Nash equilibrium and SUs have incentive to deviate.

### C. Fast Algorithm for Multiple Channel Access

In this simulation, we evaluate the performance of the proposed fast algorithm for multi-channel case, which is denoted as suboptimal strategy hereafter. In Fig. 10, the suboptimal strategy is compared with the proposed strategy, myopic strategy and random strategy in terms of social welfare under 3-channel case, where the channel parameters are set to be: for channel 1, $r_0 = 55$s and $r_1 = 25$s; for channel 2, $r_0 = 45$s and $r_1 = 40$s; for channel 3, $r_0 = 25$s and $r_1 = 55$s. We can see that the suboptimal strategy achieves the social welfare very close to that of the optimal one, i.e., the proposed strategy using modified value iteration, and is still better than the myopic and random strategies. Therefore, considering the low complexity of the suboptimal strategy, it is more practical to use the suboptimal strategy for the multi-channel case.

### D. Interference Performance

Fig. 11 shows the simulation results of the PU's average data rate in each channel $R_{pi}$ versus SUs' departure rate $\mu$ under the two-channel case, where we set $\text{SNR}_p = 5$db and $\text{INR}_p = 3$db. We can see that $R_{pi}$ is an increasing function in terms of $\mu$. Such a phenomenon is because an increase of departure rate $\mu$ means fewer SUs in the primary channels, which leads to less interference to the PU. Suppose that the PU's data rate in each channel should be at least $0.5$bps/Hz, $\mu$ should be no smaller than the value indicated by the vertical dotted line in Fig. 11, i.e., $\mu$ should be approximately larger than $0.07$. Therefore, the secondary network should appropriately control SUs' departure rate $\mu$, i.e., the average transmission time, to control the interference and ensure the PU's average data rate.

## V. Conclusion

In this paper, we extended the previous Chinese Restaurant Game work [9] into the Dynamic Chinese Restaurant Game, in which customers arrive and leave by Bernoulli process. Based on the Bayesian learning rule, we introduced a table state learning method for customers to estimate the table state. In the learning method, we assume that all the customers truthfully report their beliefs to others. How to ensure the truthful reporting is not considered, which is one of our on-going works. On one hand, truthful reporting can be achieved by effective mechanism design. On the other hand, an alternative scenario can be considered where each customer does not reveal his/her belief information and only action information can be observed. We modeled the table selection problem as an MDP problem, proposed a Multi-dimensional MDP model and a modified value iteration algorithm to find the optimal strategy. We further discussed the application of the Dynamic Chinese Restaurant Game into cognitive radio networks. The simulation results show that compared with the centralized approach that maximizes the social welfare with an intractable computational complexity, the proposed scheme achieves comparable social welfare performance with much lower complexity, while compared with random strategy and myopic strategy, the proposed scheme achieves much better social welfare performance. Moreover, the proposed scheme maximizes a new coming user's expected utility and thus achieves Nash equilibrium where no user has the incentive to deviate. Such a Dynamic Chinese Restaurant Game provides a a very general framework for analyzing the learning and strategic decision making in a dynamic social network with negative network externality.

## Appendix

### A. Proof of Lemma 1

According to Fig. 12, the recursive expression of $P_{11}(t)$ can be written by

$$P_{11}(t) = \begin{cases} 1 & t \le X, \\ 0 & X \le t \le X + Y, \\ P_{11}(t - X - Y) & X + Y \le t. \end{cases} \quad (56)$$

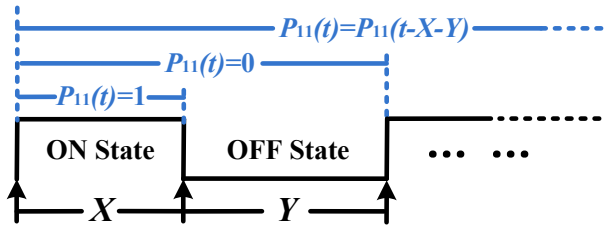


Fig. 12. Illustration of function $P_{11}(t)$.

where $X$ denotes the length of the first ON state and $Y$ denotes the length of the first OFF state. Moreover, we have $X \sim f_{\text{ON}}(x) = \frac{1}{r_1}e^{-x/r_1}$ and $Y \sim f_{\text{OFF}}(y) = \frac{1}{r_0}e^{-y/r_0}$. Since $X$ and $Y$ are independent, their joint distribution $f_{XY}(x,y) = f_{\text{ON}}(x)f_{\text{OFF}}(y)$. In such a case, we can re-write $P_{11}(t)$ as follows:

$$\begin{aligned} P_{11}(t) &= \int_{x \geq t} f_{\text{ON}}(x)dx + \iint_{x+y \leq t} f_{11}(t-x-y)f_{XY}(x,y)dxdy, \\ &= 1 - F_{\text{ON}}(t) + P_{11}(t) * f_p(t), \end{aligned} \tag{57}$$

where $F_{\text{ON}}(t) = \int_0^t f_{\text{ON}}(x)dx = 1 - e^{-t/r_1}$ is the cumulative distribution function (*c.d.f*) of the ON state's length. By taking Laplace transforms on the both sides of (57), we have

$$\begin{aligned} \mathbb{P}_{11}(s) &= \frac{1}{s} - \frac{1}{s}\mathbb{F}_{\text{ON}}(s) + \mathbb{P}_{11}(s)\mathbb{F}_p(s), \\ &= r_1\mathbb{F}_{\text{ON}}(s) + \mathbb{P}_{11}(s)\mathbb{F}_p(s), \end{aligned} \tag{58}$$

where $\mathbb{P}_{11}(s)$ is the Laplace transform of $P_{11}(t)$, $\mathbb{F}_{\text{ON}}(s) = \frac{1}{\lambda_1 s+1}$ is the Laplace transform of $f_{\text{ON}}(t)$, and $\mathbb{F}_p(s) = \frac{1}{(\lambda_1 s+1)(\lambda_0 s+1)}$ is the Laplace transform of $f_p(t)$. Then by taking the inverse Laplace transform on (58), we have

$$P_{11}(t) = r_1 f_{\text{ON}}(t) + \int_0^t P_{11}(t-w)f_p(w)dw. \tag{59}$$

### B. Proof of Lemma 2

We use induction method to prove that (48) and (49) hold for all $n \geq 0$. First, since $V_1^{(0)}(B_1,B_2,g_1,g_2)$ and $V_2^{(0)}(B_1,B_2,g_1,g_2)$ are initialized by zeros in Algorithm 1, (48) and (49) hold for $n = 0$. Second, we assume that (48) and (49) hold for some $n > 0$, and check whether (48) and (49) hold for $(n+1)$. For notation simplicity, we use $S_1 = (B_1,B_2,g_1,g_2)$ and $S_2 = (B_1,B_2,g_1+1,g_2-1)$. There are three cases for action $a_{S_1}^{(n)}$ and action $a_{S_2}^{(n)}$:

- Case 1: $V_2^{(n)}(S_1) \leq V_2^{(n)}(S_2) \leq V_1^{(n)}(S_2) \leq V_1^{(n)}(S_1)$, we have $a_{S_1}^{(n)} = a_{S_2}^{(n)} = 1$;
- Case 2: $V_1^{(n)}(S_2) \leq V_1^{(n)}(S_1) \leq V_2^{(n)}(S_1) \leq V_2^{(n)}(S_2)$, we have $a_{S_1}^{(n)} = a_{S_2}^{(n)} = 2$;
- Case 3: $V_1^{(n)}(S_1) \geq V_2^{(n)}(S_1)$ and $V_1^{(n)}(S_2) \leq V_2^{(n)}(S_2)$ we have $a_{S_1}^{(n)} = 1$ and $a_{S_2}^{(n)} = 2$.

For Case 1, we have the difference of $V_1$ and $V_2$ in (60). With the hypothesis that $V_1^{(n)}(S_1) - V_1^{(n)}(S_2) \geq 0$, we can see that $V_1^{(n+1)}(S_1) - V_1^{(n+1)}(S_2) \geq 0$ holds according to (60). For Case 2 and 3, same conclusions can be obtained by analyzing the difference of $V_1^{(n+1)}(S_1)$ and $V_1^{(n+1)}(S_2)$. Thus, we conclude that $V_1(S_1) \geq V_1(S_2)$. Similarly, $V_2(S_1) \leq V_2(S_2)$ can be proved by induction. Here, due to page limitation, we skip the detailed proof.

## REFERENCES


[1] H. V. Zhao, W. S. Lin, and K. J. R. Liu, *Behavior Dynamics in Media-Sharing Social Networks*. Cambridge University Press, 2011.

[2] Y. Chen and K. J. R. Liu, "Understanding microeconomic behaviors in social networking: an engineering view," *IEEE Signal Process. Mag.*, vol. 29, no. 2, pp. 53–64, 2012.

[3] W. H. Sandholm, "Negative externalities and evolutionary implementation," *Rev. Economic Studies*, vol. 72, no. 3, pp. 885–915, 2005.

[4] V. Bala and S. Goyal, "Learning from neighbours," *Rev. Economic Studies*, vol. 65, no. 3, pp. 595–621, 1998.

[5] D. Gale and S. Kariv, "Bayesian learning in social networks," *Games Economic Behavior*, vol. 45, no. 11, pp. 329–346, 2003.

[6] D. Acemoglu and A. Ozdaglar, "Opinion dynamics and learning in social networks," *Dynamic Games Applications*, vol. 1, no. 1, pp. 3–49, 2008.

[7] L. G. Epstein, J. Noor, and A. Sandroni, "Non-bayesian learning," *B.E. J. Theoretical Economics*, vol. 10, no. 1, pp. 1–16, 2010.

[8] D. Fudenberg and J. Tirole, *Game Theroy*. MIT Press, 1991.

[9] C.-Y. Wang, Y. Chen, and K. J. R. Liu., "Chinese restaurant game," *IEEE Signal Process. Lett.*, vol. 19, no. 12, pp. 898–901, 2012.

[10] D. Aldous, I. Ibragimov, J. Jacod, and D. Aldous., "Exchangeability and related topics," *Lecture Notes Mathematics*, vol. 1117, no. 12, pp. 1–198, 1985.

[11] C.-Y. Wang, Y. Chen, and K. J. R. Liu., "Sequential chinese restaurant game," *IEEE Trans. Signal Process.*, vol. 61, no. 3, pp. 571–584, 2013.

[12] K. J. R. Liu and B. Wang, *Cognitive Radio Networking and Security: A Game Theoretical View*. Cambridge University Press, 2010.

[13] B. Wang and K. J. R. Liu, "Advances in cognitive radios: a survey," *IEEE J. Sel. Topics Signal Process.*, vol. 5, no. 1, pp. 5–23, 2011.

[14] R. G. Gallager, *Draft Of Discrete Stochastic Processes*. MIT Press, 2013.

[15] M. L. Puterman, *Markov Decision Processes: Discrete Stochastic Dynamic Programming*. John Wiley & Sons, Inc, 1994.

[16] D. Bertsekas, *Dynamic Programming and Optimal Control*, 3rd ed. Athena Scientific, 2007.


$$\begin{aligned} V_1^{(n+1)}(S_1) - V_1^{(n+1)}(S_2) = B_1\Big(R(g_1) - R(g_1+1)\Big) + (1-\mu)\Bigg[ \\ \lambda\sum\nolimits_{B_1'}\sum\nolimits_{B_2'} P\Big((B_1',B_2')|(B_1,B_2)\Big)\Big(V_1^{(n)}(B_1',B_2',g_1+1,g_2) - V_1^{(n)}(B_1',B_2',g_1+2,g_2-1)\Big) + \\ \mu(g_2-1)\sum\nolimits_{B_1'}\sum\nolimits_{B_2'} P\Big((B_1',B_2')|(B_1,B_2)\Big)\Big(V_1^{(n)}(B_1',B_2',g_1,g_2-1) - V_1^{(n)}(B_1',B_2',g_1+1,g_2-2)\Big) + \\ \mu(g_1-1)\sum\nolimits_{B_1'}\sum\nolimits_{B_2'} P\Big((B_1',B_2')|(B_1,B_2)\Big)\Big(V_1^{(n)}(B_1',B_2',g_1-1,g_2) - V_1^{(n)}(B_1',B_2',g_1,g_2-1)\Big) + \\ \Big(1-\lambda-\mu(g_1+g_2-1)\Big)\sum\nolimits_{B_1'}\sum\nolimits_{B_2'} P\Big((B_1',B_2')|(B_1,B_2)\Big)\Big(V_1^{(n)}(B_1',B_2',g_1,g_2) - V_1^{(n)}(B_1',B_2',g_1+1,g_2-1)\Big)\Bigg]. \end{aligned} \tag{60}$$

[17] B. Wang, Y. Wu, and K. J. R. Liu, "Game theory for cognitive radio networks: an overview," *Comput. Netw.*, vol. 54, no. 14, pp. 2537–2561, 2010.
[18] B. Wang, K. J. R. Liu, and T. C. Clancy, "Evolutionary cooperative spectrum sensing game: how to collaborate?" *IEEE Trans. Commun.*, vol. 58, no. 3, pp. 890–900, 2010.
[19] I. F. Akyildiz, B. F. Lo, and R. Balakrishnan, "Cooperative spectrum sensing in cognitive radio networks: a survey," *Physical Commun.*, vol. 4, no. 3, pp. 40–62, 2011.
[20] I. F. Akyildiz, W.-Y. Lee, M. C. Vuran, and S. Mohanty, "Next generation/dynamic spectrum access/cognitive radio wireless networks: a survey," *Comput. Netw.*, vol. 50, no. 9, pp. 2127–2159, 2006.
[21] Z. Guan, T. Melodia, and G. Scutari, "Distributed queuing games in interference-limited wireless networks," in *Proc. 2013 IEEE ICC*, pp. 1–6.
[22] Z. Guan, T. Melodia, D. Yuan, and D. A. Pados, "Distributed spectrum management and relay selection in interference-limited cooperative wireless networks," in *Proc. 2011 ACM MobiCom*, pp. 229–240.
[23] G. S. Kasbekar and S. Sarkar, "Spectrum pricing games with bandwidth uncertainty and spatial reuse in cognitive radio networks," in *Proc. 2010 ACM MobiHoc*, pp. 251–260.
[24] M. G. Khoshkholgh, K. Navaie, and H. Yanikomeroglu, "Access strategies for spectrum sharing in fading environment: overlay, underlay, and mixed," *IEEE Trans. Mobile Comput.*, vol. 9, no. 12, pp. 1780–1793, 2010.
[25] C. Jiang, Y. Chen, K. J. R. Liu, and Y. Ren, "Analysis of interference in cognitive radio networks with unknown primary behavior," in *Proc. 2012 IEEE ICC*, pp. 1746–1750.
[26] T. Dang, B. Sonkoly, and S. Molnar, "Fractal analysis and modeling of VoIP traffic," in *Proc. 2004 International Telecommun. Netw. Strategy Planning Symp.*, pp. 123–130.
[27] H. Kim and K. G. Shin, "Efficient discovery of spectrum opportunities with MAC-layer sensing in cognitive radio networks," *IEEE Trans. Mobile Comput.*, vol. 7, no. 5, pp. 533–545, 2008.
[28] D. R. Cox, *Renewal Theory*. Butler and Tanner, 1967.
[29] C. Jiang, Y. Chen, and K. J. R. Liu, "A renewal-theoretical framework for dynamic spectrum access with unknown primary behavior," in *Proc. 2012 IEEE Globecom*, pp. 1–6.
[30] C. Jiang, Y. Chen, K. J. R. Liu, and Y. Ren, "Renewal-theoretical dynamic spectrum access in cognitive radio networks with unknown primary behavior," *IEEE J. Sel. Areas Commun.*, vol. 31, no. 3, pp. 1–11, 2013.

**Chunxiao Jiang** (S'09-M'13) received his B.S. degree in information engineering from Beijing University of Aeronautics and Astronautics (Beihang University) in 2008 and the Ph.D. degree from Tsinghua University (THU), Beijing in 2013, both with the highest honors. During 2011-2012, he visited the Signals and Information Group (SIG) at Department of Electrical & Computer Engineering (ECE) of University of Maryland (UMD), supported by China Scholarship Council (CSC) for one year.

Dr. Jiang is currently a research associate in ECE department of UMD with Prof. K. J. Ray Liu, and also a post-doctor in EE department of THU with Prof. Yong Ren. His research interests include the applications of game theory and queuing theory in wireless communication and networking and social networks.

Dr. Jiang received Best Paper Award from IEEE GLOBECOM in 2013, the Beijing Distinguished Graduated Student Award, Chinese National Fellowship and Tsinghua Outstanding Distinguished Doctoral Dissertation in 2013.

**Yan Chen** (S'06-M'11) received the Bachelors degree from University of Science and Technology of China in 2004, the M. Phil degree from Hong Kong University of Science and Technology (HKUST) in 2007, and the Ph.D. degree from University of Maryland College Park in 2011. His current research interests are in data science, network science, game theory, social learning and networking, as well as signal processing and wireless communications.

Dr. Chen is the recipient of multiple honors and awards including best paper award from IEEE GLOBECOM in 2013, Future Faculty Fellowship and Distinguished Dissertation Fellowship Honorable Mention from Department of Electrical and Computer Engineering in 2010 and 2011, respectively, Finalist of Deans Doctoral Research Award from A. James Clark School of Engineering at the University of Maryland in 2011, and Chinese Government Award for outstanding students abroad in 2011.

**Yu-Han Yang** (S'06) received his B.S. in electrical engineering in 2004, M.S. degrees in computer science and communication engineering in 2007, from National Taiwan University, Taipei, Taiwan, and Ph.D. degree in electrical and computer engineering in 2013 from University of Maryland, College Park, USA. His research interests include wireless communication and signal processing. He received Class A Scholarship from the ECE department, National Taiwan University in Fall 2005 and Spring 2006. He is a recipient of Study Abroad Scholarship from Taiwan (R.O.C.) government in 2009-2010. He received the University of Maryland Innovation Award in 2013.

**Chih-Yu Wang** (S'07-M'13) received the B.S. degree in Electrical Engineering from National Taiwan University, Taipei, Taiwan. in 2007. He received the Ph.D. degree from Graduate Institute of Communication Engineering, National Taiwan University. He has been a visiting student in University of Maryland, College Park in 2011. His research interests mainly are applications of game theory in wireless networking and social networking.

**K. J. Ray Liu** (F'03) was named a Distinguished Scholar-Teacher of University of Maryland, College Park, in 2007, where he is Christine Kim Eminent Professor of Information Technology. He leads the Maryland Signals and Information Group conducting research encompassing broad areas of signal processing and communications with recent focus on cooperative and cognitive communications, social learning and network science, information forensics and security, and green information and communications technology.

Dr. Liu is the recipient of numerous honors and awards including IEEE Signal Processing Society Technical Achievement Award and Distinguished Lecturer. He also received various teaching and research recognitions from University of Maryland including university-level Invention of the Year Award; and Poole and Kent Senior Faculty Teaching Award, Outstanding Faculty Research Award, and Outstanding Faculty Service Award, all from A. James Clark School of Engineering. An ISI Highly Cited Author, Dr. Liu is a Fellow of IEEE and AAAS.

Dr. Liu is Past President of IEEE Signal Processing Society where he has served as Vice President Publications and Board of Governor. He was the Editor-in-Chief of *IEEE Signal Processing Magazine* and the founding Editor-in-Chief of *EURASIP Journal on Advances in Signal Processing*.