

Distributed and Centralized Schemes for Channel Sensing Order Setting in Multi-user Cognitive Radio Networks

Junyuan Huang · Huaibei Zhou · Yongqun Chen ·
Bo Chen · Ruoshan Kong

Published online: 6 October 2013

© The Author(s) 2013. This article is published with open access at Springerlink.com

Abstract Channel sensing order setting is crucial for efficient channel exploration and exploitation in cognitive radio (CR) networks. This paper investigates the sensing order setting problem in multi-channel multi-user CR networks for both distributed scenario and centralized scenario. As the optimal solution is too complicated, two suboptimal greedy search algorithms with much less computational complexities are proposed. The channel availability, channel achievable rate, multi-user diversity and collisions among CR users are considered comprehensively in our proposed methods. For the distributed scenario, a novel *potential function* is proposed to represent the relative advantage of a channel used by a user among multi channels and multi users, based on which each user can get its own sensing order. For the centralized scenario, a sensing matrix is obtained by a coordinator for all the users. It is shown that, CR users' average throughput increases and collision probability decreases with the number of channels due to increased transmission opportunities. The total network throughput increases with the number of user pairs due to multi-user diversity. Simulation results validate the efficacy of the proposed schemes in elevating CR users' throughput and decreasing the probability of collision, and show the performance improvement of the proposed schemes by comparisons with existing works.

Keywords Cognitive radio networks · Channel sensing order · Multi-user diversity · Greedy search

J. Huang (✉) · Y. Chen · B. Chen
School of Physics and Technology, Wuhan University, Wuhan 430072, China
e-mail: leon.hjy@gmail.com

Y. Chen
e-mail: yiyuxiniaoy@gmail.com

B. Chen
e-mail: innerpeacer@gmail.com

H. Zhou (✉) · R. Kong
International school of software, Wuhan University, Wuhan 430072, China
e-mail: bzhou@whu.edu.cn

R. Kong
e-mail: krs1024@126.com

1 Introduction

Conventional wireless communication systems employ static spectrum allocation strategies, which results in both spectrum scarcity and spectrum underutilization. Cognitive radio (CR) is a promising technology to solve these spectrum inefficiency problems [1]. In cognitive radio networks, the CR users (also referred to as secondary users, SUs) can select and reuse some of the spectrum bands/channels opportunistically when the licensed users (also referred to as primary users, PUs) are not active in these channels. In order to find available spectrum opportunities and guarantee the smooth communication of the primary system, SUs need to sense the spectrum channels before transmission [2]. For ease of implementation and considering hardware constraints, we assume that SUs can sense and transmit on only one channel at a time.

Since there are a number of potential channels available, SUs can choose one channel to sense and transmit on it for the rest of the time slot if it's available or wait until next time slot otherwise; or SUs can sense the channel sequentially till an available channel is found and transmit on it for the rest of the time slot. For the former case, Zhao et al. [3] and [4] proposed optimal dynamic access strategies using periodic sensing based on Markov decision process. In the latter case, SUs sense the channels sequentially, and after each sensing, a decision on whether to continue sensing or to start transmission is made according to the channel availability and channel quality. In this case, SUs can exploit multi-channel diversity by opportunistically selecting a good channel for transmission. Sequential sensing brings a new problem: channel sensing order setting, which means that the SUs need to decide which channel to sense first. Channel sensing order is important for SUs to find a good available channel as fast as possible, and better channels and shorter sensing time will lead to higher throughput. Multi-armed bandit problems are used in [5–7] to solve the channel sensing ordering problems in uncertain environments, and polynomial regrets are obtained. In this paper, we assume that the primary users' activity and channels' achievable rates are known a priori to SUs. The authors in [8] and [9] derive the optimal sensing-sequence for channels with homogeneous capacities, which means that the intrinsic features of a multi-path channel are not taken advantage of. Dynamic programming is used in [10] and [11] to obtain the optimal channel sensing order for a single secondary user pair, heterogeneous channel capacities are considered. In the above researches only one sensing order is considered. However in this paper, we try to solve the sensing-order problem in a general multi-user network where multiple autonomous CRs have to search multiple channels according to their own sensing orders. Closely related to our research work here are [12–14]. However [12] only considered the two-user case. Cheng and Zhuang [13] only considered channel achievable rate in designing sensing orders. And Khan and Lehtomaki [14] proposed to select sensing orders from a Latin Square which is not optimal. Compared to the above works, we comprehensively consider the channel availability, channel achievable rate, multi-user diversity and collisions among CR users in designing channel sensing orders for multi users.

The main contributions of our paper are as follows. In the distributed case, with limited information exchange among SUs, we derive a novel metric which exploits the multi-user diversity simply and efficiently. Using this metric we have modified the *potential function* of traditional single-user multi-channel greedy search algorithm for multi-user case. In the centralized case, we first get the expected throughput of a CR network for given sensing orders of all users. Then we propose a suboptimal greedy search algorithm which has much lower complexity than the brute-force search. The mutual collisions among SUs are formulated as a loss to the network throughput. Plenty of simulations are carried out to investigate the

impacts of number of channels, sensing errors, number of SUs, etc. The efficacies of the proposed schemes are validated by comparisons with existing works.

The remainder of this paper is organized as follows. The related works and our motivation are presented in Sect. 2. Section 3 describes the system model. Distributed and centralized channel sensing-order setting schemes are discussed in Sects. 4 and 5, respectively. The performance of the proposed methods is evaluated in Sect. 6. Finally, conclusions are presented in Sect. 7.

2 Related Works

In the literatures, the channel sensing order setting problems have been discussed for both single user and multi-user cases. The optimal sensing order setting problem is formulated under the multi-armed bandit framework in [5–7], where a balance is found between long-term channel exploration and short-term channel exploitation to optimize the total throughput throughout the entire transmission of the SUs. However in this paper we focus on exploiting the instantaneous channel condition under the assumption that the statistics of channel availability and channel achievable rate are known a priori. The authors in [8] and [9] proposed an efficient sensing order setting scheme that incurs a small opportunity–discovery delay. Channels are sensed at the descending order of channel idle probabilities, which is also referred to as the intuitive sensing order. However the multi-channel diversity is not taken advantage of. In [15], it is shown that fluctuating nature of heterogeneous spectrum bands is crucial in spectrum decision. In [10], the optimal channel sensing order and the related stopping rule is derived for a single-user case, with an assumption that recall (i.e., a previously sensed channel can be accessed) and guess (i.e., an un-sensed channel can be accessed) are allowed. Dynamic programming is used in [11] to obtain the optimal channel sensing order considering heterogeneous channel capacities and channel availabilities jointly. The optimal stopping rule is well studied in [16] and [17] under the homogeneous channel utilization scenario, however the optimal sensing order setting problem is not solved. In [18] sensing order and power allocation are jointly considered to achieve the maximum energy efficiency. Our previous study in [19] proposed sensing order setting schemes for both real-time and best-effort applications. However in the above researches the sensing ordering problem is discussed only for single-user case. In this paper we focus on the sensing ordering problem for multi-user case. In a multi-user network, two new factors will influence the channel utilization and network throughput: collision (i.e., two or more SU pairs sense and transmit in the same channel at the same time) and useless sensing (i.e., a SU senses a channel which is being used by another SU).

The authors in [12] discussed the sensing ordering problem for a two-user case, it is shown that the traditional stopping rule is not optimal anymore. Designing an optimal stopping rule for multi-user case is nearly impossible. So in our work, we focus on sensing ordering problem but ignore the stopping rule design. In [13] the channels are sensed according to the descending order of their achievable rates, where the primary free probabilities are not considered. A novel metric comprehensively incorporating the channel availability and collision among SUs is proposed in [20], based on which the sensing order is obtained for each user in a distributed manner. However the metric is designed improperly as it is mutable and uncertain in the process of designing a sensing order. In [21], a systematic neural network-based optimization approach is developed to set proper sensing order to maximize the average throughput of a CR network. However the training process may be too long when the number of SUs or channels is large, which is not suitable for adaption to changing environment.

An adaptive persistent sensing order selection strategy is proposed in [14], sensing orders are selected from a Latin Square matrix. It is assumed that none of the SUs can gain any reward for the rest of the slot if a collision happens. However, under our framework, the collision can be detected by the SUs using a probing signal, and then the SUs can keep sensing the rest spectrum opportunities. So, though the strategy in [14] enables the CRs to converge to collision-free channel sensing orders, the network throughput is not optimized. In this paper, aiming to maximize the average SU throughput and minimize collisions, two greedy search algorithms are proposed to obtain the sensing orders in both centralized and distributed manners.

3 System Model

We consider a CR network with M cognitive transmitter/receiver pairs and a set $\mathbf{N} = \{1, 2, \dots, N\}$ of channels, and we only consider the case that $M \leq N$ (for the case that $M > N$, an admission control scheme similar to [22] can be adopted). The cognitive users are synchronous with each other and they are synchronous with the primary network. Time is partitioned into slots of duration T , and each channel is either busy (i.e., with primary activities) or free (i.e., no primary activities) for an entire time slot. We assume that the channel availability probability and channel achievable rate are known to the SUs at the beginning of each time slot.

In each timeslot, the source node of a SU pair needs to scan (sense and probe) the channels according to its sensing sequence before transmission. Denote τ as the time duration needed for scanning one channel, which satisfies $N\tau < T$. If a channel is found to be free, the SU pair transmits in that channel. However, if no available channel is found after scanning all N channels, the SU pair stays silent for the remaining duration of that time slot. The slot structure is depicted in Fig. 1, where after sensing the first $(k - 1)$ channels to be busy, the SU finds the k -th channel is free and transmits on it for the remainder of that time slot. Thus the period of time spent on channel scanning and that on data transmission are $k\tau$ and $T - k\tau$, respectively. Define the *effectiveness* of a slot as the ratio of the transmission phase length to the slot length. Therefore, if a user stops at the k -th channel in its sensing order, the effectiveness is $c_k = 1 - k\tau/T$.

As depicted in Fig. 1, after each sensing, a probing phase is carried out. In specific, once a channel is sensed to be free, the SU transmitter transmits a channel probing packet (CPP) to its receiver so as to estimate the channel gain of this transmission link. Then the SU receiver feeds the estimated channel gain back to the transmitter with a probing feedback packet (PFP). It is assumed that the SU transmitter and its receiver are always synchronized through a centralized coordinator (or a dedicated control channel for the distributed case), so that they always tune to the same channel for sensing, probing and transmission. A receiver will not send back a PFP in two cases: (1) The receiver senses a channel to be busy, and (2) Probing

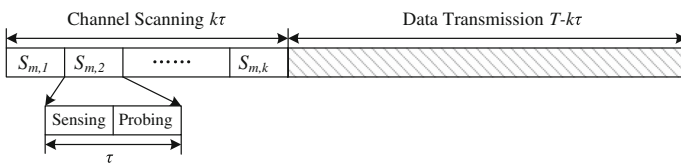


Fig. 1 An illustration of the sequential channel sensing process of user m with its sensing order given as $\{S_{m,1}, S_{m,2}, \dots, S_{m,N}\}$

collision happens among multiple transmitters. In both cases the transmitter will skip the channel and keep sensing. In the second case, multiple transmitters simultaneously sense the same channel to be free and transmit a CPP on that channel. Thus a probing collision happens and none of them can use that channel, a spectrum opportunity is wasted.

In this paper, we consider sensing order setting problem for both distributed and centralized cases. In the distributed case, each SU pair should determine their own sensing order by themselves. Before carrying out sensing and probing, a negotiation phase is needed where the SUs exchange their primary free probability and channel achievable rate information with neighbors on a common channel. In the centralized case, a coordinator exists in the cognitive radio network. According to estimated primary free probabilities and channel achievable rates, the coordinator determines the sensing orders of the SU pairs and announces the sensing orders to the pairs. Thus the coordinator needs to determine a sensing matrix \mathbf{S} with dimensions of $M \times N$, in which the m -th row $\mathbf{S}_m = \{S_{m,1}, S_{m,2}, \dots, S_{m,N}\}$ is the channel sensing order dedicated to the m -th SU pair. Each row $\mathbf{S}_m = \{S_{m,1}, S_{m,2}, \dots, S_{m,N}\}$ is a permutation of the set $(1, 2, \dots, N)$, where $S_{m,k}$ denotes the channel that user m senses at the k -th position in its sensing order. In this study, we assume that each SU pair can potentially see different primary user occupancy behavior on each channel due to their geographic dispersion. And we focus on the simplest complete-interference case when all secondary users interfere with each other and must therefore each be allocated distinct channels. The same assumption is also adopted in [7] and [20], and is discussed in [23]. We use a $M \times N$ matrix \mathbf{P} to denote the primary free probabilities where its element $P_{m,i}$ is the primary free probability of channel i for user m , and $P_{m,i}$ is uncorrelated with $P_{l,i}$ for $m \neq l$; and a $M \times N$ matrix \mathbf{R} to denote the channel achievable rates where its element $R_{m,i}$ is the achievable rate of channel i for user m , and $R_{m,i}$ is uncorrelated with $R_{l,i}$ for $m \neq l$. We assume that \mathbf{P} and \mathbf{R} are known to the users a priori, the estimation of them is not our concern. We ignore the effect of the missed detection (due to strict protection of primary network) and only consider the effect of false alarm P_f . Thus we define the unconditional probabilities matrix that the SUs determine that the channels are idle as $\boldsymbol{\theta}$ with its element $\theta_{m,i} = P_{m,i}(1 - P_f)$ representing the probability that user m determines channel i to be free.

4 Sensing Order Selection for Distributed Case

4.1 Greedy Search Algorithm for Single User Scenario

We first briefly introduce the main idea of the greedy search algorithm for single user scenario in this section. Consider a single user pair m with its aim to maximize the average throughput. Let $\mathbf{S}_m^0 = \{S_{m,1}, S_{m,2}, \dots, S_{m,N}\}$ be the optimal channel sensing order of user m . The expected throughput using this optimal sensing order can be obtained as

$$U_m^0 = \sum_{k=1}^N \left[\prod_{i=1}^k (1 - \theta_{S_{m,i-1}}) \right] \theta_{S_{m,k}} c_k R_{S_{m,k}} \tag{1}$$

where we define $\theta_{S_{m,0}} = 0$, $\theta_{S_{m,k}} \prod_{i=1}^k (1 - \theta_{S_{m,i-1}})$ is the probability that the user stops at the k -th channel, and $c_k R_{S_{m,k}}$ is the obtained normalized throughput if the user transmits in the k -th channel.

The optimal sensing sequence can also be expressed as $\{L_{k-1}, S_{m,k}, S_{m,k+1}, L_{k+2}^c\}$, where $L_{k-1} = (s_{m,1}, \dots, s_{m,k-1})$ is the former part of the sequence and $L_{k+2}^c = (s_{m,k+2}, \dots, s_{m,N})$ is the later part. Let \mathbf{S}_m^* be the counterpart of \mathbf{S}_m^0 constructed by switching the order of k -th and

$k + 1$ -th channels, $\mathbf{S}_m^* = \{L_{k-1}, S_{m,k+1}, S_{m,k}, L_{k+2}^c\}$. Since \mathbf{S}_m^0 is the optimal sensing order, the expected throughput U_m^0 should be no less than that of its counterpart U_m^* , $U_m^0 \geq U_m^*$. We have:

$$\begin{aligned}
 U_m^0 - U_m^* &= \prod_{i=1}^{k-1} (1 - \theta_{S_{m,i}}) \{ \theta_{S_{m,k}} (T - k\tau) R_{S_{m,k}} + (1 - \theta_{S_{m,k}}) \theta_{S_{m,k+1}} (T - (k+1)\tau) R_{S_{m,k+1}} \\
 &\quad - \theta_{S_{m,k+1}} (T - k\tau) R_{S_{m,k+1}} - (1 - \theta_{S_{m,k+1}}) \theta_{S_{m,k}} (T - (k+1)\tau) R_{S_{m,k}} \} \geq 0 \\
 &\Rightarrow \frac{R_{S_{m,k}}}{\tau/\theta_{S_{m,k}} + T - (k+1)\tau} \geq \frac{R_{S_{m,k+1}}}{\tau/\theta_{S_{m,k+1}} + T - (k+1)\tau} \tag{2}
 \end{aligned}$$

Thus we define the *potential function* of channel i at the k -th round in a sensing order for user m as:

$$g_m(i, k) = R_{m,i} / (\tau/\theta_{m,i} + T - (k+1)\tau) \tag{3}$$

which satisfies that:

$$g_m(i, k) \geq g_m(j, k) \Rightarrow U_m(\{L_{k-1}, i, j, L_{k+2}^c\}) \geq U_m(\{L_{k-1}, j, i, L_{k+2}^c\}) \tag{4}$$

Then a greedy search algorithm can be obtained as follows:

Let L_{k-1} be the set of already chosen channels, at the initial step of the algorithm ($k = 1$), $L_0 = \emptyset$. For finding the k -th channel in the optimal sensing sequence, the channel i with the largest potential function $g_m(i, k)$ is chosen, where $i \in \{1, 2, \dots, N\} \setminus L_{k-1}$. Till $k = N$, the whole optimal sequence is obtained.

4.2 Greedy Search Algorithm for Multi-user Scenario

When multiple autonomous SUs search multiple potentially available channels synchronously, then from an individual SU's perspective one of the following three events will happen in each sensing step [14]: (1) The SU senses a channel to be free, then it sends a probing signal on this channel to its receiver and successfully receives the response, the SU then has the channel for itself for the remainder of the time slot; (2) The SU senses a channel to be busy (occupied by primary users or another SU), then it keeps quiet for the probing phase and keeps looking in the next sensing step; (3) The SU senses a channel to be free, then it sends a probing signal, but so does at least one other SU; thus a collision occurs, none of these SUs can use this channel and they all skip into next sensing step, a spectrum opportunity is wasted. From above discussions, we can conclude that the optimal sensing order setting of an individual SU is not only affected by the channel free probabilities and channel achievable rates of its own perspective, but also affected by the sensing orders of other SUs.

For example, consider a CR network with $N = 3$ channels and $M = 2$ users, the scanning time for each channel is 0.1, slot duration is 1, and the achievable rate for each channel is 1. The channel free probabilities for user 1 are $\theta_{1,1} = 0.9$, $\theta_{1,2} = 0.5$ and $\theta_{1,3} = 0.2$; the channel free probabilities for user 2 are $\theta_{2,1} = 0.7$, $\theta_{2,2} = 0.4$ and $\theta_{2,3} = 0.6$. If the users search their sensing orders individually and don't consider the other user's sensing order, the sensing order setting of the two users is (1,2,3) for user 1 and (1,3,2) for user 2, and the total throughput can be calculated as $0.5576 + 0.5755 = 1.1331$. However By brute-force search, we can obtain the optimal sensing order setting of the two users, which is (1,2,3) for user 1 and (3,1,2) for user 2, and the total throughput can be calculated as $0.8528 + 0.6614 = 1.5142$. We can see that in the former sensing order setting, both users sense channel 1 at their first

sensing step. So a collision will happen with a high probability $\theta_{1,1}\theta_{2,1} = 0.63$ (when both users sense channel 1 to be free and send a probing signal), the spectrum opportunity of channel 1 is wasted which causes the decrease of the network throughput.

Although it is possible that a SU can obtain the optimal channel sensing order by brute-force search individually with the knowledge of the channel free probabilities and channel achievable rates of its own and all other SUs, the complexity of brute-force search is $O((N!)^M)$ if the complexity to calculate the expected network throughput with a specific sensing-order setting is $O(1)$, which is unacceptable for an individual user due to hardware constraint. Thus in this section we try to give a simple yet efficient greedy search algorithm for the distributed network scenario by modifying the *potential function* in Eq. (3) as follows:

$$g_m^{\text{mod}}(i, k) = (g_m(i, k) - \frac{1}{M-1} \sum_{l \in \mathbf{M} \setminus m} g_l(i, k)) + (g_m(i, k) - \frac{1}{N-1} \sum_{j \in \mathbf{N} \setminus i} g_m(j, k)) \tag{5}$$

where \mathbf{M} and \mathbf{N} are the sets of all SUs and potential channels, $g_m(i, k)$ is presented as in Eq. (3). $g_m(i, k) - \frac{1}{M-1} \sum_{l \in \mathbf{M} \setminus m} g_l(i, k)$ represents the relative advantage of channel i being used by user m at its k -th sensing step compared with which being used by other users; $g_m(i, k) - \frac{1}{N-1} \sum_{j \in \mathbf{N} \setminus i} g_m(j, k)$ represents the relative advantage of channel i being used by user m at its k -th sensing step compared with other channels being used by user m . $g_m(i, k) - \frac{1}{M-1} \sum_{l \in \mathbf{M} \setminus m} g_l(i, k)$ guarantees that channel i should be first allocated to user m which can get a larger reward than other users; $g_m(i, k) - \frac{1}{N-1} \sum_{j \in \mathbf{N} \setminus i} g_m(j, k)$ guarantees that user m should first choose the channel i which can bring larger reward than other channels.

Using this *modified potential function*, the greedy search algorithm for multi-user case in distributed network scenario can be stated as follows:

- Step 1: Negotiation. Users exchange their channel free probability and channel achievable rate information, $\boldsymbol{\theta}$ and \mathbf{R} , with their neighbors on a common control channel;
- Step 2: Initialization. $k = 1$, the set of already chosen channels of user m is $L_m^{k-1} = L_m^0 = \emptyset$.
- Step 3: Find the k -th channel in the optimal sensing order, $S_{m,k} = \underset{i \in \{1,2,\dots,N\} \setminus L_m^{k-1}}{\text{arg max}} g_m^{\text{mod}}(i, k)$. And $L_m^k = L_m^{k-1} \cup \{S_{m,k}\}$.
- Step 4: If $k = N$, end; else, $k = k + 1$, go to step 3.

Let's briefly explain how this *modified potential function* works in the former mentioned example. We have $g_1^{\text{mod}}(1, 1) = 0.25$, $g_1^{\text{mod}}(2, 1) = 0.1142$ and $g_1^{\text{mod}}(3, 1) = -0.5449$; $g_2^{\text{mod}}(1, 1) = 0.0301$, $g_2^{\text{mod}}(2, 1) = -0.1428$ and $g_2^{\text{mod}}(3, 1) = 0.2933$. For the first channel in the sensing order, user 1 will choose channel 1 and user 2 will choose channel 3 (though from user 2's individual perspective channel 1 is the best and should be chosen first, our method can choose another but still good one to avoid the collision), finally our greedy search method will get the same optimal sensing orders as brute-force search does.

From the procedure of the algorithm, the complexity of the greedy search algorithm is $N + (N - 1) + \dots + 1 = N(N - 1) / 2$ if the complexity of calculating one *modified potential function* is $O(1)$, which is much less than the brute force search $O((N!)^M)$.

5 Sensing Matrix Setting for Centralized Case

5.1 Problem Formulation

In our centralized scenario, a coordinator exists to help the SUs to set their channel sensing orders. The coordinator needs to decide a $M \times N$ sensing matrix \mathbf{S} to maximize the network throughput, in which the m -th row $\mathbf{S}_m = \{S_{m,1}, S_{m,2}, \dots, S_{m,N}\}$ is the channel sensing order dedicated to the m -th SU pair. Each row $\mathbf{S}_m = \{S_{m,1}, S_{m,2}, \dots, S_{m,N}\}$ is a permutation of the set $\{1, 2, \dots, N\}$, where $S_{m,k}$ denotes the channel that is at the k -th position in the sensing order of user m . For a given sensing matrix, let $\phi_{m,k}$ denotes the probability that user m successfully stops and transmits at its k -th sensing step. Let $p(i, \mathbf{S}_m)$ denotes the position of channel i in user m 's sensing order \mathbf{S}_m .

We can note that given a sensing matrix, for each channel $S_{m,k}$ there are four possible cases might occur: (1) The channel is neither in the prior positions ($<k$) nor in the k -th position of any other users' (other than user m) sensing order; (2) The channel is not in the prior positions ($<k$) of any other users' sensing order but is in the k -th position of some other users' sensing order; (3) The channel is in the prior positions ($<k$) of some other users' sensing order but is not in the k -th position of any other users' sensing order; (4) The channel is in the prior positions ($<k$) and in the k -th position of some other users' sensing order. By analyzing the above four possible cases of a channel, $\phi_{m,k}$ can be obtained as follows:

For $k = 1$,

$$\phi_{m,k} = \begin{cases} \theta_{m,S_{m,k}}, & \text{for case 1;} \\ \theta_{m,S_{m,k}} \prod_l (1 - \theta_{l,S_{l,k}}), & \text{for case 2.} \end{cases} \tag{6}$$

For $k > 1$,

$$\phi_{m,k} = \begin{cases} (1 - \sum_{j=1}^{k-1} \phi_{m,j})\theta_{m,S_{m,k}}, & \text{for case 1;} \\ (1 - \sum_{j=1}^{k-1} \phi_{m,j})\theta_{m,S_{m,k}} \prod_l \left((1 - \sum_{j=1}^{k-1} \phi_{l,j})(1 - \theta_{l,S_{l,k}}) + \sum_{j=1}^{k-1} \phi_{l,j} \right), & \text{for case 2;} \\ (1 - \sum_{j=1}^{k-1} \phi_{m,j})\theta_{m,S_{m,k}} \prod_{l^*} (1 - \phi_{l^*,p(S_{m,k}, \mathbf{S}_{l^*})}), & \text{for case 3;} \\ (1 - \sum_{j=1}^{k-1} \phi_{m,j})\theta_{m,S_{m,k}} \prod_l \left((1 - \sum_{j=1}^{k-1} \phi_{l,j})(1 - \theta_{l,S_{l,k}}) + \sum_{j=1}^{k-1} \phi_{l,j} \right) \\ \times \prod_{l^*} (1 - \phi_{l^*,p(S_{m,k}, \mathbf{S}_{l^*})}), & \text{for case 4.} \end{cases} \tag{7}$$

where $(1 - \sum_{j=1}^{k-1} \phi_{m,j})\theta_{m,S_{m,k}}$ is the probability that user m reaches the k -th step and senses channel $S_{m,k}$ to be free. l satisfies $l \in \{1, 2, \dots, M\} \setminus m$ and $p(S_{m,k}, \mathbf{S}_l) = k$, is the user (other than user m) that selects channel $S_{m,k}$ at the k -th position of its sensing order; l^* satisfies $l^* \in \{1, 2, \dots, M\} \setminus m$ and $p(S_{m,k}, \mathbf{S}_{l^*}) < k$, is the user (other than user m) that selects channel $S_{m,k}$ at the prior $k - 1$ positions. $\prod_l \left((1 - \sum_{j=1}^{k-1} \phi_{l,j})(1 - \theta_{l,S_{l,k}}) + \sum_{j=1}^{k-1} \phi_{l,j} \right)$ is the probability that channel $S_{m,k}$ is not simultaneously sensed to be free (or be sensed) by other users (other than user m), $\sum_{j=1}^{k-1} \phi_{l,j}$ is the probability that user l successfully stops and transmits at the previous $k - 1$ sensing steps (so does not reach the k -th step),

$(1 - \sum_{j=1}^{k-1} \phi_{l,j})(1 - \theta_{l,S_{l,k}})$ is the probability that user l reaches the k -th step but senses channel $S_{m,k}$ to be busy. $\prod_{l^*} (1 - \phi_{l^*,p(S_{m,k},S_{l^*})})$ is the probability that channel $S_{m,k}$ has not been occupied by any other users at the previous $k - 1$ sensing steps.

Using Eqs. (6) and (7) we can iteratively get all the $\phi_{m,k}$ for each m and k . Then given a sensing matrix, the expected network throughput can be obtained:

$$U = \sum_{m=1}^M \sum_{k=1}^N \phi_{m,k} R_{m,S_{m,k}} c_k \tag{8}$$

The coordinator’s aim is to find the optimal sensing matrix S^0 to maximize the expected network throughput. This can be stated formally as:

$$S^0 = \arg \max_{S_{1,1}, S_{1,2}, \dots, S_{M,N}} U \tag{9}$$

Given this optimization problem, the coordinator can find the optimal sensing matrix by brute-force search. However the complexity of brute-force search is $O((N!)^M)$ which will result in massive computational burden and is not scalable regarding to both M and N . This paper targets at suboptimum algorithm that has much less complexity but is still efficient, as discussed in the following section.

5.2 Greedy Search Algorithm

As discussed before, for a target user to determine which channel to sense in its k -th sensing step, if a channel has already been in other users’ prior ($\leq k$) sensing positions, the probability that this channel is occupied by other users should be considered. And the throughput loss caused by collisions should also be considered when two or more users proceed to sense the same channel at the same time. Based on above observations, in this section, we will propose a greedy search algorithm which contains N rounds. Let S_m denotes the channel selection vector of user m . S_m will be updated after each rounds. So it contains k chosen channels after the k -th round. In the k -th round the coordinator needs to determine $(S_{1,k}, S_{2,k}, \dots, S_{M,k})$ for the users, where $S_{m,k} \in \overline{S_m} = N \setminus S_m$. Figure 2 depicts the channel selection process with an example.

Let $p(i, S_m)$ denotes the position of channel i in S_m if it is in S_m . For a user m selecting $S_{m,k}$ among channels from $\overline{S_m}$, the coordinator estimates the probability that each channel is free from primary users and other SUs, and accordingly assign each channel a reward. We define $G_i^{(m)}(k)$ the reward of channel i selected by user m as the k -th channel in its sensing order. It is set to the contribution of the user m to the network throughput if channel i is selected. Then the channel with the largest reward is selected as $S_{m,k}$ for user m . The selecting process is as follows:

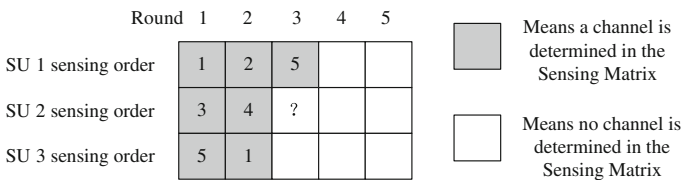


Fig. 2 An example of channel selection process. The coordinator is going to choose the third channel from {1, 2, 5} for SU 2. The probability that these channels may be already occupied by the other two users should be considered. Specifically, the possible collision with SU 1 should be considered when selecting channel 5

5.2.1 Round-1

At the beginning, the set of already selected channels for each user is empty ($\mathbf{S}_m = \emptyset$ for user m). We start with the first SU. The coordinator needs to choose $S_{1,1}$ for user 1 from $\overline{\mathbf{S}}_1 = \mathbf{N}$. For a channel $i \in \overline{\mathbf{S}}_1$, we can obtain that $G_i^{(1)}(1) = \theta_{1,i} R_{1,i} c_1$. Thus, the first channel to be sensed by user 1 is:

$$S_{1,1} = \arg \max_{i \in \overline{\mathbf{S}}_1} G_i^{(1)}(1) \tag{10}$$

After $S_{1,1}$ is determined, we have $\mathbf{S}_1 = \{S_{1,1}\}$.

While for the m -th user in the first round, a channel i may have already been in the sets of selected channels of other users ($i \in \mathbf{S}_l, l \in \mathbf{M} \setminus m$). Thus we have:

$$G_i^{(m)}(1) = R_{m,i} c_1 \theta_{m,i} \prod_{\substack{l \in \mathbf{M} \setminus m \\ i \in \mathbf{S}_l}} (1 - \theta_{l,i}) \tag{11}$$

where $\theta_{m,i} \prod_{\substack{l \in \mathbf{M} \setminus m \\ i \in \mathbf{S}_l}} (1 - \theta_{l,i})$ means the probability that channel i is free from primary users and all other SUs. If channel i has already been in the other users' selected channel set, user m selects it as the first channel will cause throughput loss to the network due to the collision with other users. Thus we can get the loss of reward as follows:

$$L_i^{(m)}(1) = \theta_{m,i} \sum_{\substack{l \in \mathbf{M} \setminus m \\ i \in \mathbf{S}_l}} \left(c_1 R_{l,i} \theta_{l,i} \prod_{\bar{l}} (1 - \theta_{\bar{l},i}) \right) \tag{12}$$

where l is the user which has already selected channel i , \bar{l} satisfies that $\bar{l} \in \mathbf{M} \setminus m, i \in \mathbf{S}_{\bar{l}}$ and $\bar{l} \neq l$. $c_1 R_{l,i} \theta_{l,i} \prod_{\bar{l}} (1 - \theta_{\bar{l},i})$ is the reward of channel i using by user l . A collision happens if user m senses channel i as free at the same time (with probability $\theta_{m,i}$), and then the reward of user l is lost. Therefore, the reward in Eq. (11) should be rewritten for user m selecting channel i as its first channel:

$$G_i^{(m)}(1) = R_{m,i} c_1 \theta_{m,i} \prod_{\substack{l \in \mathbf{M} \setminus m \\ i \in \mathbf{S}_l}} (1 - \theta_{l,i}) - L_i^{(m)}(1) \tag{13}$$

Then the coordinator selects the channel that has the largest reward as the first channel for user m :

$$S_{m,1} = \arg \max_{i \in \mathbf{S}_m} G_i^{(m)}(1) \tag{14}$$

5.2.2 Round- k

After the $k - 1$ -th round, the coordinator has selected $k - 1$ channels for each user, which means we have got a matrix of selected channels with a dimension of $M \times (k - 1)$. Thus using this matrix and Eqs. (6) and (7) we can get the $\phi_{m,j}$ for each user m and $j \in [1, k - 1]$. At the k -th round, we define $P_i^{free}(k)$ as the probability that channel i is not occupied by any SU in the previous $k - 1$ sensing steps. We have:

$$P_i^{free}(k) = \prod_{\substack{l^* \in \mathbf{M} \\ p(i, \mathbf{S}_l^*) < k}} (1 - \phi_{l^*, p(i, \mathbf{S}_l^*)}) \tag{15}$$

where l^* represents the user that has already selected channel i at the previous $k - 1$ rounds.

At the k -th round, the coordinator should first determine with which SU it should start. Similar to [12], to achieve a certain level of fairness, we start with the user that has a less cumulative reward in the previous $k - 1$ rounds. The cumulative reward gained during the previous $k - 1$ rounds for user m is calculated as,

$$\phi_{m,1} c_1 R_{m, S_{m,1}} + \phi_{m,2} c_2 R_{m, S_{m,2}} + \dots + \phi_{m,k-1} c_{k-1} R_{m, S_{m,k-1}} \tag{16}$$

When user m selects a channel $i \in \overline{\mathbf{S}}_m$ as its k -th channel, this channel may have already been selected by other users at the same round. So the collision free probability should be considered when selecting a channel. The reward gained by user m selecting channel i at the k -th round can be obtained:

$$G_i^m(k) = c_k R_{m,i} \theta_{m,i} P_i^{free}(k) \prod_{\substack{l \in \mathbf{M} \setminus m \\ p(i, \mathbf{S}_l) = k}} \left(\sum_{j=1}^{k-1} \phi_{l,j} + (1 - \theta_{l,i}) \left(1 - \sum_{j=1}^{k-1} \phi_{l,j} \right) \right) \tag{17}$$

where $P_i^{free}(k)$ is defined in Eq. (15), l is the user that has already selected channel i at the k -th round. The formula in \prod represents the collision free probability for user m selecting channel i . A collision happens if user m and l sense channel i to be free at the same time. So the reward of user l may be lost because of user m selecting channel i . The loss of reward caused by user m selecting channel i at the k -th round is calculated as:

$$L_i^{(m)}(k) = c_k \theta_{m,i} P_i^{free}(k) \sum_{\substack{l \in \mathbf{M} \setminus m \\ p(i, \mathbf{S}_l) = k}} \underbrace{\left(\theta_{l,i} \left(1 - \sum_{j=1}^{k-1} \phi_{l,j} \right) \prod_{\bar{l}} \left(1 - \theta_{\bar{l},i} \left(1 - \sum_{j=1}^{k-1} \phi_{\bar{l},j} \right) \right) \right)}_F \tag{18}$$

where \bar{l} satisfies that $\bar{l} \in \mathbf{M} \setminus m$, $p(i, \mathbf{S}_{\bar{l}}) = k$ and $\bar{l} \neq l$. F represents the probability that user l is the only one that senses channel i as free regardless of the effect of user m . So the considered loss of reward is only because of user m selecting channel i at the k -th round. Therefore, the reward in Eq. (17) should be rewritten as follows:

$$G_i^m(k) = c_k R_{m,i} \theta_{m,i} P_i^{free}(k) \prod_{\substack{l \in \mathbf{M} \setminus m \\ p(i, \mathbf{S}_l) = k}} \left(\sum_{j=1}^{k-1} \phi_{l,j} + (1 - \theta_{l,i}) \left(1 - \sum_{j=1}^{k-1} \phi_{l,j} \right) \right) - L_i^m(k) \tag{19}$$

Then the coordinator selects the channel from $\overline{\mathbf{S}}_m$ that has the largest reward as the k -th channel for user m :

$$S_{m,k} = \arg \max_{i \in \overline{\mathbf{S}}_m} G_i^{(m)}(k) \tag{20}$$

After $S_{m,k}$ is selected, \mathbf{S}_m is updated, $\mathbf{S}_m = \{S_{m,1}, S_{m,2}, \dots, S_{m,k}\}$.

In the last round, the only channel in $\overline{\mathbf{S}}_m$ is selected as $S_{m,N}$ for user m . Till the whole sensing matrix is got, the algorithm is done.

5.2.3 Modification of Reward

Let's rewrite the reward Eq. (19) as follows:

$$G_i^m(k) = c_k R_{m,i} P_{m,i}^{free} - L_i^m(k) \tag{21}$$

Where $P_{m,i}^{free} = \theta_{m,i} P_i^{free}(k) \prod_{\substack{l \in \mathbf{M}_m \\ p(i, \mathbf{S}_l) = k}} \left(\sum_{j=1}^{k-1} \phi_{l,j} + (1 - \theta_{l,i})(1 - \sum_{j=1}^{k-1} \phi_{l,j}) \right)$ is the probability that channel i is free for user m from primary users and all other SUs. Thus from an individual user's perspective, the preceding greedy search algorithm is just like the greedy algorithm discussed in Sect. 4.1, just changing the *potential function* from Eq. (3) to $g_m(i, k) = c_k R_{m,i} \theta_{m,i}$. From the simulation result in [24] and our numerical analysis, the performance of greedy algorithm using the former *potential function* is better than that of using the later one. So it's reasonable to change our reward in Eq. (21) according to the structure of Eq. (3), $g_m(i, k) = R_{m,i} / (\tau / \theta_{m,i} + T - (k + 1)\tau)$. We modify Eq. (21) as follows:

$$G_i^m(k) = R_{m,i} / (\tau / P_{m,i}^{free} + T - (k + 1)\tau) - \frac{R_{m,i} / (\tau / P_{m,i}^{free} + T - (k + 1)\tau)}{c_k R_{m,i} P_{m,i}^{free}} L_i^{(m)}(k) \tag{22}$$

Where $P_{m,i}^{free}$ is the same of that in Eq. (21).

From the preceding procedure, we can see that the complexity of this greedy search algorithm is $(N + N - 1 + \dots + 1)M = \frac{N(N+1)}{2}M$ if the complexity of calculating one reward is $O(1)$, which is much less than the complexity of brute-force search, $O((N!)^M)$. In order to further have a mid-term fairness, we suggest that at the beginning of *Round 1*, the coordinator chooses the user to start with in a *round robin* way.

6 Performance Evaluation

6.1 Simulation Settings

In our simulations, we consider a network with M SUs and N potential channels. The primary activity is modeled as i.i.d. in both frequency channel and timeslot dimensions. We assign the values of primary free probability \mathbf{P}_1 and channel achievable rate \mathbf{R} randomly for each user and each channel at the beginning of each slot. We set the channel sensing\probing time $\tau = 0.02s$ and slot duration $T = 1s$. The achievable rates of the channels are uniformly distributed on $(0,10)kbps$. The performance metrics considered in this work are: Average SU throughput—the average number of successful bits transmitted per second per SU; Throughput difference—the average ratio of the standard deviation of the SUs' throughputs to the average SU throughput in a slot, which represents the fairness between the SUs in a slot; Collision—the average times that two or more SUs sense the same channel as free (thus the probing signals collide and none of the SUs can use the channel) in a slot. We perform the simulations 10,000 time slots for each run. We then average the performance measurements over the timeslots.

6.2 Simulation Results

We evaluate the performance of the proposed greedy search algorithms versus \mathbf{P}_1 , M , N and P_f . We compare the performance of our distributed greedy search algorithm with reference [20], and compare the sensing matrix proposed in our centralized greedy algorithm with the Latin Square matrix proposed in [14].

(1) Effect of primary free probability, \mathbf{P}_1

We study the impact of the primary free probability on the performance of proposed algorithms, and keep the other parameters to be $M = 5$, $N = 7$ and $P_f = 0$. We keep the standard deviation of the primary free probability to be 0.25, and change the average value between [0.25, 0.75]. The simulation results are depicted in Figs. 3, 4 and 5.

Figure 3 shows the impact of primary free probability on the average SU throughput. We can see that the values of average SU throughput of all algorithms increase as the primary free probability increases. This is easy to understand that larger primary free probability means more transmission opportunity for the SUs. It is shown that both of our distributed algorithm and centralized algorithm gain more throughput than reference [20] and Latin Square method in [14]. To be specific, the proposed distributed greedy search algorithm outperforms reference [20] by 14.78–8.56% when \mathbf{P}_1 increases from 0.25 to 0.75; the proposed centralized greedy search algorithm outperforms Latin Square method by 28.89–54.33% when \mathbf{P}_1 increases from 0.25 to 0.75. And the centralized greedy search algorithm outperforms the distributed algorithm by 1.82–4.66%. We can see that the average SU throughput of distributed greedy search algorithm increases more slowly than that of the centralized algorithm as \mathbf{P}_1 increases. This is because that when \mathbf{P}_1 increases the relative difference between free probabilities of different channels decreases, so the advantage of the distributed algorithm (which takes advantage of the relative difference of different channels, see Eq. (5)) decreases.

The throughput difference versus primary free probability is depicted in Fig. 4. It is shown that the throughput difference of each algorithm decreases as the primary free probability increases. This is because that when primary free probability is smaller, it's more

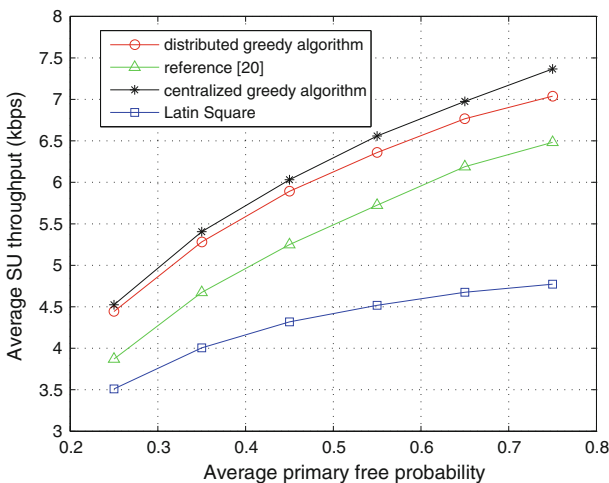


Fig. 3 Average SU throughput of different channel sensing algorithms versus the average value of primary free probability ($M = 5$, $N = 7$ and $P_f = 0$)

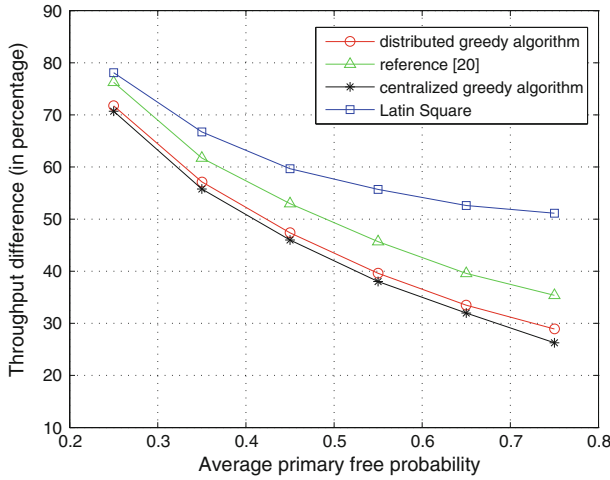


Fig. 4 Throughput difference of different channel sensing algorithms versus the average value of primary free probability ($M = 5$, $N = 7$ and $P_f = 0$)

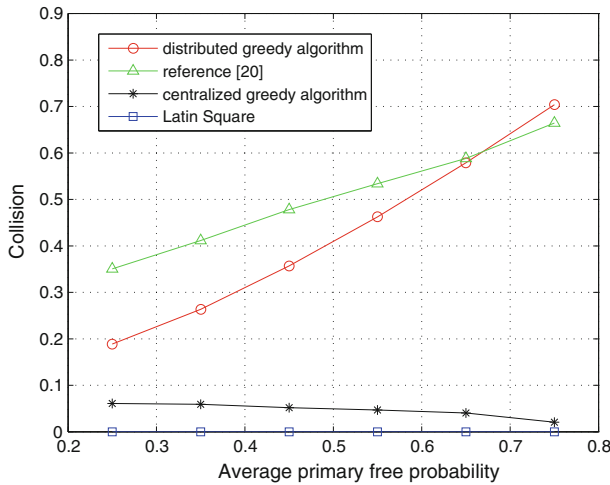


Fig. 5 Collision of different channel sensing algorithms versus the average value of primary free probability (where $M = 5$, $N = 7$ and $P_f = 0$)

likely that some users can't find an available channel which makes the difference between SUs throughputs larger. We can see that the throughput difference of our distributed greedy algorithm and centralized greedy algorithm is smaller than that of reference [20] and Latin Square method, which means that our methods can guarantee better fairness among the SUs than reference [20] and Latin Square method do. Specifically, when P_1 increases from 0.25 to 0.75, the throughput difference of the distributed greedy algorithm is 71.73–28.9%, and 70.69–26.27% for the centralized greedy algorithm, 76.26–35.38% for reference [20], 78.09–51.14% for Latin Square method. The centralized greedy algorithm is the best because that it adopts a measure to guarantee the fairness among SUs, see Eq. (16) and the related analysis.

Figure 5 shows the collision of different methods. The collisions of the distributed greedy method and reference [20] increase with \mathbf{P}_1 . This is because the probability that two or more SUs sense the same channel to be free increases with \mathbf{P}_1 . We can see that the collision of the distributed greedy algorithm is smaller than that of reference [20] in most cases. The collision of the Latin Square method is zero, and the collision of our centralized greedy algorithm is far smaller (from 0.06 to 0.02) than those of distributed greedy algorithm and reference [20]. Though Latin Square method is collision-free, it can't get the optimal SUs throughput. This is because that when a collision happens, the SU can continue to sense the rest channels in its sensing order in the slot. The collision is not a dominant factor that influences the SUs throughput. However the collisions can waste available channel opportunities, thus cause throughput loss in some degree [see Eq. (18)]. So our centralized greedy algorithm which has considered this impact can get the largest SUs throughput and very low collision.

(2) Effect of the number of channels, N

We study the impact of the number of channels on the performance of proposed algorithms, and keep the other parameters to be $M = 5$, $P_f = 0$ and \mathbf{P}_1 is randomly distributed on $(0, 1)$. Figure 6a depicts the effect of number of channels on the average SU throughput. It is shown that the average SU throughput of each method increases with the increase of the number of channels. This is because that more potential channels means more transmission opportunity for the SUs. We can see that the distributed greedy algorithm outperforms reference [20] by 21.94–8.39% when the number of channels increases from 5 to 10; the centralized greedy algorithm outperforms Latin Square method by 32.54–57.76%; and the centralized greedy algorithm outperforms distributed greedy algorithm by 8.46–1.86%. Figure 6b depicts the effect of the number of channels on the throughput difference. The throughput difference of each method decreases with the increase of the number of channels. This is because that with more potential channels it is more likely that every SU can get an available channel with good quality. It is shown that both of our distributed greedy algorithm and centralized greedy algorithm have smaller throughput difference than reference [20] and Latin Square method which confirms the fairness among the SUs when using the proposed methods. We can see that when the number of channels gets larger, the performance (both average SU throughput and throughput difference) of the distributed greedy algorithm gets closer to that

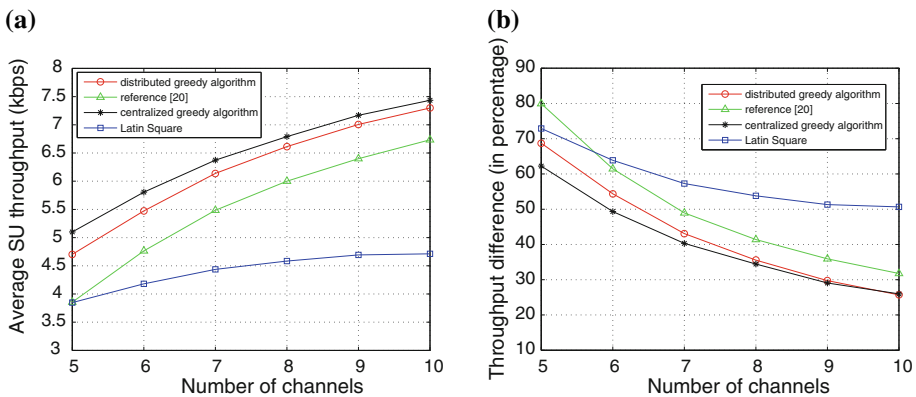


Fig. 6 Effect of number of channels on the performance (where $M = 5$, $P_f = 0$ and $\mathbf{P}_1 = \text{rand}(0, 1)$). a Average SU throughput of different methods; b Throughput difference of different methods

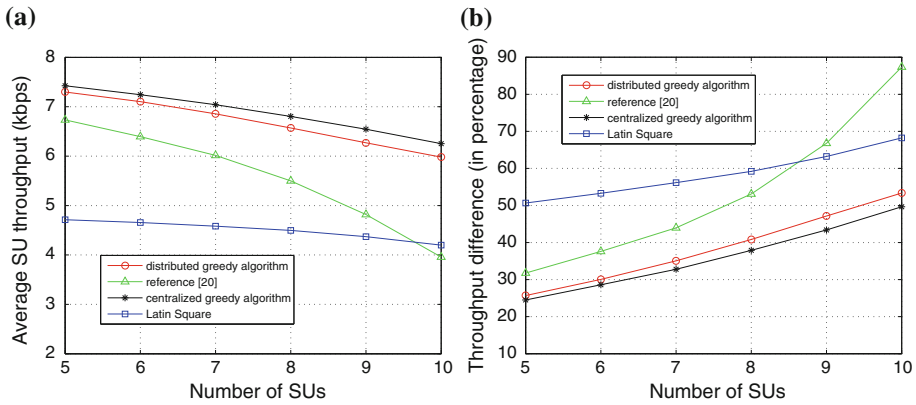


Fig. 7 Effect of number of SUs on the performance (where $N = 10$, $P_f = 0$ and $\mathbf{P}_I = \text{rand}(0, 1)$). **a** Average SU throughput of different methods. **b** Throughput difference of different methods

of the centralized greedy algorithm. This is because that the second part of Eq. (5) will give a bigger effect with a larger number of channels.

(3) *Effect of the number of SUs, M*

Figure 7 depicts the effect of the number of SUs on the performance of the algorithms, where the other parameters are kept as $N = 10$, $P_f = 0$ and \mathbf{P}_I is randomly distributed on $(0, 1)$. It is shown in Fig. 7a that the average SU throughput of each algorithm decreases with the increase of the number of SUs, since more SUs means less available channel opportunity for each SU. However the total network throughput increases with the increase of the number of SUs due to the multi-user diversity. We can see that both of our distributed greedy algorithm and centralized greedy algorithm gain more throughput than the traditional methods. In specific, the distributed greedy algorithm outperforms reference [20] by 8.39–51.11% when the number of SUs increases from 5 to 10; the centralized greedy algorithm outperforms Latin Square method by 57.54–48.95%; and the centralized greedy algorithm outperforms distributed greedy algorithm by 1.73–4.55%. It is shown in Fig. 7b that the throughput difference of each algorithm increases as the number of SUs increases. This is because that when the number of SUs increases, it’s harder to satisfy that each SU can get an available channel with good quality and it’s likely that some SUs even can’t get an available channel in a slot, so the throughput difference among the SUs increases. We can see that our distributed greedy algorithm and centralized greedy algorithm have smaller throughput difference than reference [20] and Latin Square method. Specially, the performance of reference [20] decreases very fast with the increase of the number of SUs. In contrast, both of our proposed methods can get far better performance (up to about 50%) than the traditional ones.

(4) *Effect of false-alarm probability, P_f*

We study the impact of false alarm probability on the performance of the proposed algorithms, and keep the other parameters to be $M = 5$, $N = 7$ and \mathbf{P}_I is randomly distributed on $(0, 1)$. The simulation results are shown in Fig. 8. From Fig. 8a we can see that the average SU throughput of each method decreases with the increase of false alarm probability. This is because that with larger false alarm probability, it’s more possible that the SUs miss an actual available channel opportunity. It is shown that our proposed methods gain more SU throughput: the distributed greedy algorithm outperforms reference [20] by 11.93–13.44%

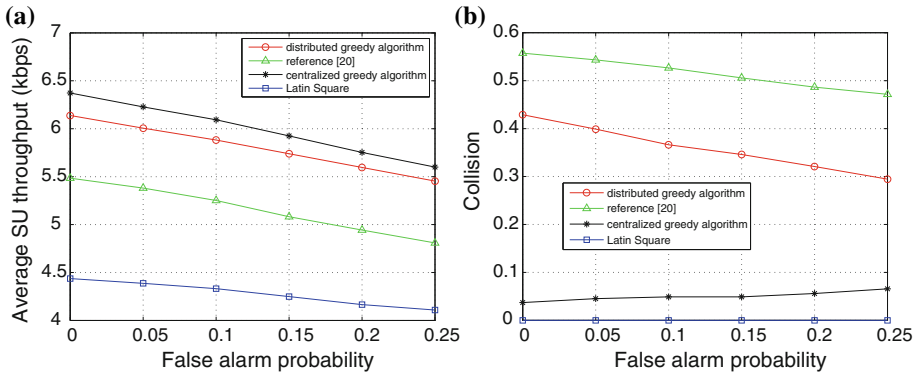


Fig. 8 Effect of false alarm probability on the performance (where $M = 5$, $N = 7$ and $\mathbf{P}_1 = \text{rand}(0, 1)$). **a** Average SU throughput of different methods. **b** Collision of different methods

when false alarm probability increases from 0 to 0.25; the centralized greedy algorithm outperforms Latin Square method by 43.63–36.29%; and the centralized greedy algorithm outperforms distributed greedy algorithm by 3.83–2.68%. Figure 8b depicts the collisions of the methods versus false alarm probability. We can see that Latin Square method is collision-free, the collisions of distributed greedy algorithm and reference [20] decrease with the increase of false alarm probability, while the collision of centralized greedy algorithm increases with the increase of false alarm probability. As we mentioned before, our centralized greedy algorithm has considered the throughput loss caused by collisions [Eq. (18)], so it can get a very low collision compared to the distributed greedy algorithm and reference [20]. However with the increase of false alarm probability, the value of Eq. (18) (thus its impact) decreases, which causes the increase of collision. In contrast, both the distributed greedy algorithm and reference [20] do not consider the impact of collisions on throughput loss, so their collisions decrease with the increase of false alarm probability because of the decrease of the probability that a channel is sensed to be free by two or more SUs simultaneously.

7 Conclusions

In this paper, we study the channel sensing order setting problem for a multi-channel multi-user cognitive radio network. Two suboptimal greedy search algorithms are proposed for both distributed scenario and centralized scenario. In the distributed greedy search algorithm, a novel *potential function* is proposed to represent the relative advantage of a channel used by a user among multi channels and multi users. In the centralized greedy search algorithm, a sensing matrix is obtained by a coordinator for all the users. The loss caused by collisions is considered and the fairness between SUs is guaranteed in this centralized greedy search algorithm. Simulation results show that our distributed greedy algorithm gets more average SU throughput, better fairness and lower collision than the distributed method in the reference. And the sensing matrix obtained by our centralized greedy algorithm outperforms the Latin Square in the reference. The distributed greedy algorithm has a very low computation complexity, and its performance gets very close to the centralized greedy algorithm (with a difference about 1%) when the primary free probability is small, the number of channels is large and the false alarm probability is large.

The stopping rules are not adopted in this paper because it should be jointly designed with the sensing order setting from a systematic point of view, this will be our future concern.

Acknowledgments The work reported in this paper was supported by the National 973 Project of China under Grant No. 2009CB320401.

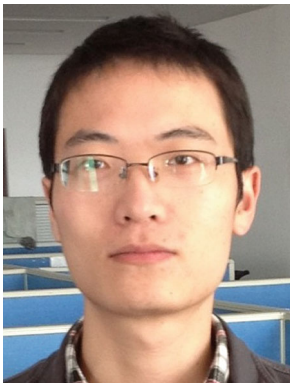
Open Access This article is distributed under the terms of the Creative Commons Attribution License which permits any use, distribution, and reproduction in any medium, provided the original author(s) and the source are credited.

References

1. Akyildiz, I., Lee, W., Vuran, M., & Mohanty, S. (2006). Next generation/dynamic spectrum access/cognitive radio wireless networks: A survey. *Computer Networks*, *50*(13), 2127–2159.
2. Liang, Y., & Zeng, Y. (2008). Sensing-throughput tradeoff for cognitive radio networks. *IEEE Transactions on Wireless Communications*, *7*(4), 1326–1337.
3. Zhao, Q., Tong, L., Swami, A., & Chen, Y. (2007). Decentralized cognitive MAC for opportunistic spectrum access in ad hoc networks: A POMDP framework. *IEEE Journal on Selected Areas in Communications*, *25*(3), 589–600.
4. Zhao, Q., Geirhofer, S., Tong, L., & Sadler, B. M. (2008). Opportunistic spectrum access via periodic channel sensing. *IEEE Transactions on Signal Processing*, *56*(2), 785–796.
5. Lai, L., El Gamal, H., Jiang, H., & Poor, H. V. (2011). Cognitive medium access: Exploration, exploitation, and competition. *IEEE Transactions on Mobile Computing*, *10*(2), 239–253.
6. Li, X.-Y., & Tang, S., (2012). Almost optimal accessing of nonstochastic channels in cognitive radio networks. In *Proceedings IEEE INFOCOM* (pp. 2291–2299).
7. Gai, Y., Krishnamachari, B., & Jain, R. (2010). Learning multiuser channel allocations in cognitive radio networks: A combinatorial multi-armed bandit formulation. In *IEEE symposium on new frontiers in dynamic spectrum (DySPAN)* (pp. 1–9).
8. Shin, K. G. (2008). Efficient discovery of spectrum opportunities with MAC-layer sensing in cognitive radio networks. *IEEE Transactions on Mobile Computing*, *7*(5) 533–545.
9. Kim, H., & Shin, K. G. (2008). Fast discovery of spectrum opportunities in cognitive radio networks. In *3rd IEEE symposium on new frontiers in dynamic spectrum access networks* (pp. 1–12).
10. Chang, N. B., & Liu, M. (2007). Optimal channel probing and transmission scheduling for opportunistic spectrum access. In *Proceedings of the 13th annual ACM international conference on Mobile computing and networking - MobiCom '07* (p. 27).
11. Jiang, H. (2009). Optimal selection of channel sensing order in cognitive radio. *IEEE Transactions on Wireless Communications*, *8*(1), 297–307.
12. Fan, R., & Jiang, H. (2009). Channel sensing-order setting in cognitive radio networks: A two-user case. *IEEE Transactions on Vehicular Technology*, *58*(9), 4997–5008.
13. Cheng, H., & Zhuang, W. (2011). Simple channel sensing order in cognitive radio networks. *IEEE Journal on Selected Areas in Communications*, *29*(4), 676–688.
14. Khan, Z., & Lehtomaki, J. (2013). Autonomous sensing order selection strategies exploiting channel access information. *IEEE Transactions on Mobile Computing*, *12*(2), 274–288.
15. Lee, W.-Y., & Akyildiz, I. F. (2011). A spectrum decision framework for cognitive radio networks. *IEEE Transactions on Mobile Computing*, *10*(2), 161–174.
16. Shu, T., & Krunz, M. (2009). Throughput-efficient sequential channel sensing and probing in cognitive radio networks under sensing errors. In *MobiCom '09: Proceedings of the 15th annual international conference on Mobile computing and networking* (pp. 37–48).
17. Shu, T., & Krunz, M. (2012). Sequential opportunistic spectrum access with imperfect channel sensing. *Ad Hoc Networks*, *11*(3), 778–797.
18. Pei, K. H. L. Y., Liang, Y. C., & Teh, K. C. (2011). Energy-efficient design of sequential channel sensing in cognitive radio networks? Optimal sensing strategy, power allocation, and sensing order. *IEEE Journal on Selected Areas in Communications*, *29*(8), 1648–1659.
19. Huang, J., Zhou, H., Chen, Y., Chen, B., Zhu, X., & Kong, R. (2012). Optimal channel sensing order for various applications in cognitive radio networks. *Wireless Personal Communications*, doi:[10.1007/s11277-012-0906-x](https://doi.org/10.1007/s11277-012-0906-x) (online first).

20. Zhao, J., & Wang, X. (2012). Channel sensing order in multi-user cognitive radio networks. In *IEEE international symposium on dynamic spectrum access, networks* (pp. 374–384).
21. Shokri-Ghadikolaei, H., & Fallahi, R. (2012). Intelligent sensing matrix setting in cognitive radio networks. *IEEE Communications Letters*, 16(11), 1824–1827.
22. Bizhani, H., & Ghasemi, A. (2012). Joint admission control and channel selection based on multi response learning automata (MRLA) in cognitive radio networks. *Wireless Personal Communications*. doi:[10.1007/s11277-012-0834-9](https://doi.org/10.1007/s11277-012-0834-9) (online first).
23. Jia, J., Zhang, Q., & Shen, X. (2008). HC-MAC: A hardware-constrained cognitive MAC for efficient spectrum management. *IEEE Journal on Selected Areas in Communications*, 26(1), 106–117.
24. Han, H., Wang, J., & Wu, Q. (2010). Optimal wideband spectrum sensing order based on decision-making tree in cognitive radio. In *Wireless communications and signal processing (WCSP), 2009* (pp. 1–5).

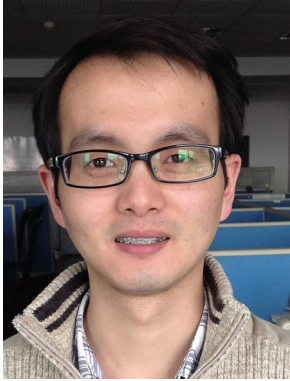
Author Biographies



Junyuan Huang received the B.S. degree in physics from Wuhan University, Wuhan, China, in 2009. Now he is working on his M.S. and Ph.D. degrees in electronics in Wuhan University. His research interests include cognitive radio and WLAN based indoor positioning.



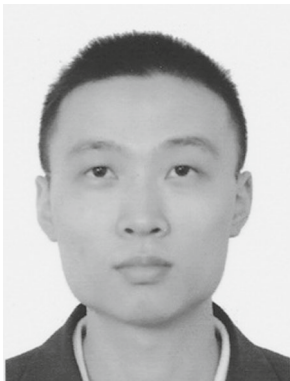
Huaibei Zhou received the B.S. and M.S. degrees in physics from Wuhan University, Wuhan, China, in 1984 and 1987, respectively, and the Ph.D. degree in physics from the University of Maryland, College Park, in 1994. From 1994 to 1996, he was a Postdoctoral Fellow with the National Institute of Standard and Technology, Gaithersburg, MD. From 1996 to 1999, he was a Senior Engineer with General Electric. From 1999 to 2002, he was a Senior Technical Manager with NEXTEL. He is currently the Dean of the International School of Software Engineering, Wuhan University. His research interests include cognitive radio and radio wave propagation.



Yongqun Chen received the B.S. degree from School of Physics and Technology, Wuhan University, Wuhan, China, in 2009. He is now working on his M.S. and Ph.D. degrees in Electrical and Information Engineering in Wuhan University. His Current research interests include machine learning, cognitive radio and WLAN based indoor positioning.



Bo Chen received the B.S. degree in physics from Wuhan University, Wuhan, China, in 2009. Now he is working on his M.S. and Ph.D. degrees in electronics in Wuhan University. His research interests include power control in cognitive radio based on Game theory, more specifically potential games.



Ruoshan Kong received the B.S. and Ph.D. degrees in computer science from Wuhan University, China in 2002 and 2007, respectively. He is currently an associate professor in the International School of Software, Wuhan University, China. His research interests include mobile communication protocols, IPv6 networking protocols, and cognitive network protocols.