Channel Selection for Rendezvous with High Link Stability in Cognitive Radio Network*

Zhenhua Han[1,2], Haisheng Tan[2**], Yongcai Wang[3], and Jipeng Zhou[2]

[1] School of Electronic Engineering, University of Electronic Science and Technology of China, Chengdu, China
[2] Department of Computer Science, Jinan University, Guangzhou, China
[3] Institute for Interdisciplinary Information Sciences, Tsinghua U., Beijing, China

Abstract. Channel selection is a fundamental problem in Cognitive Radio Networks (CRNs). One basic channel assignment problem is *rendezvous*, which investigates how secondary users can establish a connection by selecting a common available working channel in the same time slot. Rendezvous is the prerequisite for communication, which however is challenging especially in a dynamic spectrum environment. Link stability is a key factor that benefits the performance of a wireless network. In this work, we study the dynamic channel selection for rendezvous with high link stability. In the centralized case, we propose an algorithm to make as many secondary users as possible to establish stable links. In the distributed case, we derive a novel fast rendezvous strategy in the two-user scenario based on the local idle time of channels. The strategy can be utilized in asynchronous systems. Extensive simulations show that our algorithms have a higher stability in various environments compared with other rendezvous strategies.

Keywords: Cognitive Radio Networks; Channel Selection Problem; Rendezvous Problem; Link Stability

1 Introduction

Wireless spectrum is a kind of scarce resource in recent wireless communication. It is widely believed that the cognitive radio (CR) technique will play a key role in solving the spectrum scarcity problem[1]. In a cognitive radio network (CRN), the licensed entities are often called the primary users (PUs), which have absolute priority in accessing the spectrum band. PUs allow the unlicensed secondary users (SUs) to access the licensed band without causing interference to the PUs. Therefore, before communication, secondary users need to sense a wide range of band and choose a free channel as the working channel, which is called *Channel Selection*.

* This work was supported in part by the Fundamental Research Funds for the Central Universities in China, and the National Natural Science Foundation of China Grant 61373125 and 61202360.

** Contact him at thstan@jnu.edu.cn.

In CRNs, to avoid any possible interference to the PUs, we consider the case that SUs are only allowed to use the free channels which are not used by nearby PUs, i.e., in an overlay mode. Due to the existence of the PUs, the available channels for SUs are dynamic over both space and time. SUs at different locations may have different available spectrum. And the available spectrum for SUs may also change with time. One fundamental problem of SUs is how to establish a link with each other by selecting the same available channel to work on in a time slot, which is called *rendezvous*.

Channel selection for rendezvous in cognitive radio networks has attracted great attention in both academics and industry. The outputs can be divided into two groups: centralized and distributed. In a centralized system, secondary users adopt a central controller to help secondary users to build connections[2]. Although such a central controller may achieve high performance in rendezvous, it requires large information exchange overhead, which may not be practical. Some other centralized methods use a dedicated channel, called the Common Control Channel (CCC), to exchange information between users for rendezvous[3–7]. However, these methods have a key drawback that such a free channel used as CCC may be occupied by PUs, and also it may become an easy attack point.

To overcome the above drawbacks, blind rendezvous in a distributed system without any CCC or central controller is preferred. Among distributed strategies, the main group are to utilize channel-hopping (CH) technique [8–12] or a quorum system [7, 13–15]. Most of these methods can achieve low expected time to rendezvous (TTR). However, although they considered the spatial dynamics of the available channels for SUs, they did not take the channel dynamics over time into account.

Link stability is a key factor to determine the performance of a wireless network, particularly for the dynamic channel environment in CRNs. Unstable links mean frequent channel switches. In CRNs, the change of channel availability may cause disconnection between a pair of nodes. The channel selection strategies will also have a great effect on the link stability. In our paper, we focused on designing channel selection approaches to establish stable links for SUs. The main contribution includes:

- Centralized Environment: we proposed a centralized algorithm for multi-user networks to maximize the number of connected links, which will also have a property of improved link stability.
- Distributed Environment: we also derived a distributed rendezvous strategy for two secondary users, which has a low time to rendezvous when two users have a similar spectrum environment. A connected link will also have better stability compared with other exist rendezvous methods. The strategy does not need a global clock.
- We conducted extensive simulations to validate the performance of the above algorithms. Other than the random setting of the available channels for SUs, we set up the channel environment of the SUs based on the behaviors of the PUs. This setting is more close to the environment in practice.

The rest of the paper is organized as follows. We present the network model and the problem definitions in Section 2. We then propose a centralized algorithm to solve stable channel selection problem in Section 3. We also propose a distributed rendezvous strategy for two secondary users in Section 4. Section 5 contains the simulation results, and we conclude our work in Section 6.

2 Network Model and Problem Definitions

2.1 Network Model

We study a cognitive radio network which comprises the following elements:

- $U = \{SP_1, SP_2, ..., SP_N\}$ is the set of N pairs of secondary users. S_i and D_i, where $i = \{1, 2, ..., N\}$, are the sets of source nodes and destination nodes respectively;
- $C = \{c_1, c_2, ..., c_L\}$ is the set of L channels in the network;
- $PU = \{PU_1, PU_2, ..., PU_M\}$ is the set of M primary users.

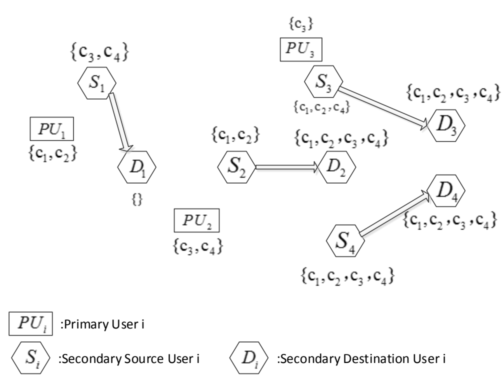

Fig. 1. A cognitive radio network with different available channels to secondary users

As stated before, we consider the network works under an overlay mode, i.e., SUs can only use channels that are free to its nearby PUs. The available channel set for a secondary user u is denoted as A_u. Figure 1 illustrates a scenario of a CRN with PUs and SUs. The whole channel set is $C = \{c_1, c_2, c_3, c_4\}$. The channel set beside a PU indicates its current working channel, while the set beside a SU indicates its available channels. For example, $A_{S_1} = \{c_3, c_4\}$, since S_1 can not use the working channels of PU_1; $A_{D_1} = \varnothing$, since all the channels are occupied by its nearby PUs. In this work, we assume that each SU has already known its local available channels through spectrum sensing or accessing to a centralized spectrum resource database. Our work focuses on the selection of an available channel as the working channel for each SU.

2.2 Stable Channel Selection Problem

Two secondary nodes can communicate if and only if they select the same available channel at the same time slot. For the pair SP_i of secondary users, we define

$$x_l^i = \begin{cases} 1, & \text{both two nodes of the pair } SP_i \text{ select channel } c_l \\ 0, & \text{otherwise.} \end{cases}$$

A node can only choose one available channel at each time slot (also called round). We adopt the global interference model, so an available channel can be chosen by only a pair of nodes. In addition, we define *the communication time* of a link is the duration from its establishment on a working channel c to the end of its connection on channel c no matter due to the disconnection of the link or the change of its working channel.

In the centralized case, our main aim is to maximize the number of connected links. The assignment may not be unique. We select the one with the highest stability. At each round, we will re-choose working channels for each SU to achieve the global optimization in maximizing the connected links and the network stability . That is to say, the working channel of a link might be switched to another one even its old working channel is still available. Therefore, in the centralized case we define stability as follow:

- Stability decreases if a secondary link disconnects;
- Stability decreases if a link is still connected but the nodes switch their communication channel to another one.

Let $\beta(i, l)$ denote the stability of channel $c_l \in C$ for secondary pair $SP_i \in U$. There might be various definitions of $\beta(i, l)$. Here, in order to improve the probability that a link remains working on the same channel as long as the channel is available, we propose a definition only based on the network status of the previous round:

$$\beta(i, j) = \begin{cases} 1, & \text{Secondary Pair } SP_i \text{ used channel } c_j \text{ as} \\ & \text{the communication channel in the previous round;} \\ 0, & \text{otherwise.} \end{cases}$$

Based on the above, we can formulate the following the stable channel selection (SCS) problem:

$$\begin{aligned}
\mathbf{max} \quad & < \sum_{l=1}^{L} \sum_{i=1}^{N} x_l^i \quad , \quad \sum_{l=1}^{L} \sum_{i=1}^{N} x_l^i \beta(i, l) > & \\
\mathbf{s.t.} \quad & x_l^i \in \{0, 1\}, & \forall\ SP_i \in U, \forall\ c_l \in C \\
& \sum_{l=1}^{L} x_l^i \leq 1, & \forall\ SP_i \in U \\
& \sum_{i=1}^{N} x_l^i \leq 1, & \forall\ c_l \in C \\
& x_l^i = 0, & \forall\ SP_i \in U, \forall\ c_l \notin A_{S_i}\ or\ c_l \notin A_{D_i}
\end{aligned}$$

variables $\boldsymbol{x} = \{x_l^i\}$.

In addition, the order of $< a, b >$ and $< c, d >$ is defined as follows: $< a, b >$ is greater than $< c, d >$ if and only if $a > c$, or $b > d$ when $a = c$.

In the distributed case, due to the dynamics of the available channels, the rendezvous of two nodes is already challenging. In this work, we investigate the stable rendezvous problem of two SUs, and leave the rendezvous for multiple users as the future work. Once the two SUs achieve rendezvous on a channel c, they will keep on working on this channel until c is occupied by a PU. Therefore, we define the stability in the distributed case as:

– Once two SUs achieve rendezvous, the longer communication time indicates the better link stability.

In every round, both SUs independently choose an available channel and attempt to build connection on this channel. If they choose the same channel, the rendezvous is achieved. Our goal is to design a strategy which can help two SUs to build a stable connection with a short time to rendezvous (TTR).

3 A Centralized Stable Channel Selection Algorithm

In the centralized condition, secondary users adopt a central controller to help them selecting channels. The controller acquires all channel state of SUs before making the assignment. At any time t, we design the Stable Channel Selection Algorithm (Algoirthm 1) to achieve the maximum number of connected links with high link stability. The algorithm contains the following steps:

1. Line 1–9: *Construct a weighted directed bipartite graph between the sets secondary user pairs U and channels C.*
   We denote the bipartite graph as $G = (V(U, C), E)$. An edge $e_{ij} \in E$ pointing from a node in U to a node in C if and only if channel c_j is available for both nodes of pair SP_i (Figure 2). Then, the channel selection for a time t, denoted as P_t, is a subset of links in E. For example, if $e_{ij} \in P_t \subseteq E$, it means at time t we assign channel c_j to the pair SP_i.
   In addition ,we set $\beta(i, j)$ as the weight (or cost) of the edge e_{ij}.

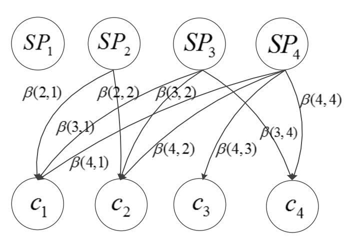

Fig. 2. the constructed bipartite graph based on the scenario in Figure 1

2. Line 10–15: *Construct a graph to compute the maximum cost maximum flow, denoted as G_1.*
   We add two virtual nodes N_{source} and N_{sink} to G. N_{source} points to each node in U with cost 0; and each node in C points to N_{sink} with cost 0. We set the capacity of all edges in G_1 to 1.

3. Line 16–17: *Compute the channel selection.*
   We compute the maximum cost maximum flow in G_1 taking N_{source} and N_{sink} as the source and sink respectively. The edges between nodes U and C whose residual capacity is 0 are the elements of P_t.

The input of Algorithm 1 are the time slot t and the channel selection of the previous slot P_{t-1}. Here, the maximum matching of the bipartite graph G are the maximum connected links at time t. The maximum cost maximum flow in G_1 guarantees that the cost is maximized with the precondition that the flow is maximized, which can be computed in polynomial time. Moreover, when achieving the maximum flow, the edges P_t, whose residual capacities are 0, form a maximum matching of the bipartite graph G. The cost of an edge e_{ij} is set as $\beta(i, j)$. Therefore, the maximum cost indicates our channel selection P_t also have the maximum stability.

Algorithm 1: Stable Channel Selection Algorithm

Input: t, P_{t-1}
Output: P_t
1 Initialize a weighted bipartite graph $G = (V(U, C), E = \varnothing)$ and $P_t = \varnothing$;
2 **for** $i \in \{1, 2, ..., |U|\}$ **do**
3   **for** $j \in \{1, 2, ..., |C|\}$ **do**
4   **if** *channel* c_j *is open to the source and destination node of* SP_i **then**
5   **if** $t = 0$ *or* $e_{ij} \in P_{t-1}$ **then**
6                 $\beta(i, j) \leftarrow 1$;
7   **else**
8                 $\beta(i, j) \leftarrow 0$;
9   Add edge e_{ij} to G with cost $\beta(i, j)$;
10 Construct a weighted graph $G_1 = (V \cup \{N_{sourse}, N_{sink}\}, E)$ based on G;
11 **for** $i \in \{1, 2, ..., |U|\}$ **do**
12   Add edge between N_{source} and SP_i in G_1 with cost 0;
13 **for** $i \in \{1, 2, ..., |C|\}$ **do**
14   Add edge between N_{sink} and c_i in G_1 with cost 0 ;
15 Set the capacity of all edges in G_1 to 1;
16 Compute the maximum cost maximum flow of G_1 with source N_{source} and sink N_{sink};
17 P_t = All edges between U and C where the residual capacity is 0 when achieving the maximum cost maximum flow;
18 **return** P_t;

4 Distributed Strategy for 2-user Rendezvous Problem

In a distributed system, a node only knows the available channels of its own. In this section, we propose a novel rendezvous strategy for two secondary users based on the local idle rate of channels.

Let T denote the timeslots counted from the moment that the SU enters the network and $Idle(c)$ denote the total available time of channel c during the T slots. In our method, a node is required to calculate and memorize $Idle(c)$ of every channel from the moment it enters the network. At time T, the largest available probability among all the channels is estimated as $p = \frac{Idle(c_m)}{T}$, where c_m is the channel with the longest available time so far. Figure 3 shows an example of the calculation.

At the first τ timeslots, our strategy just calculate the idle rates of channels and does not try to achieve rendezvous. This is to avoid rendezvous on an unstable channel with insufficient information. After τ slots, we sort all the channels based on their available time so far [4], and the SU will select the working channel based on a geometric distribution probability(Algorithm 2).

Time	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
Channel 1	Off	Off	On	On	On	Off	Off	Off	Off	On	On	On	On	Off	Off	Off	Off	Off	On	On
$Idle(c_1)$	0	0	1	2	3	3	3	3	3	4	5	6	7	7	7	7	7	7	8	9
Channel 2	On	On	On	Off	Off	Off	On	On	On	Off	Off	Off	Off	Off	Off	On	On	On	On	Off
$Idle(c_2)$	1	2	3	3	3	3	4	5	6	6	6	6	6	6	6	6	7	8	9	9
p	1	1	1	0.75	0.6	0.5	0.57	0.625	0.667	0.6	0.545	0.5	0.538	0.5	0.467	0.438	0.412	0.444	0.473	0.45

Fig. 3. An example of calculating $Idle(c)$ and p for one SU in the case of two channels (the shadowed grid is the channel c_m which has the longest available time so far.)

Algorithm 2: Rendezvous strategy for one SU in the distributed case

```
Sort all channels in descending order by their available time so far.
p = Idle(c_m)/T, where c_m is the channel with the longest available time.
if T>τ then
    Choose the i-th open channel (i ≥ 1) in the sorted order with possibility
    (p/λ)(1 − p/λ)^(i−1) (λ is a constant set by the secondary user)
else
    Do not try any channel
end if
```

In our strategy Algorithm 2, since a channel with longer available time will have a higher probability to be selected, a connected link would have a good property of stability. Meanwhile, if two SUs have similar channel environments, their sorted lists of channels will also be similar. Therefore, our strategy makes them tend to select the same available channel, which implies the time to rendezvous (TTR) is small. In practice, two SUs communicating with each other should be close, which indicates their channel environment should be similar. Our simulations in the next section will verify the performance of Algorithm 2

[4] If two channel c_i and c_j have the same available time so far, i.e., $Idle(c_i) = Idle(c_j)$, then the one with the lower channel ID comes first in the sorted list.

for both the small TTR and high stability. Moreover, our strategy dose not make use of a global clock between the two SUs, which means it can be adopted in asynchronous systems.

5 Simulation

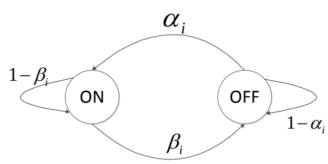

Fig. 4. Markov model of the state of PU_i's channels

To analyze the performance of the algorithms in Section 3 and 4, we conduct extensive simulations. In this work, we propose a method to set up the channel environment for SUs based on PUs. Recall that each SU can not make use of the channels occupied by nearby PUs. Therefore, for a primary user PU_i, we define a *dominating range*, denoted as r_i^{dom}. To avoid any interference to PU_i, any SU with a distance equal to or shorter than r_i^{dom} to PU_i can not use the current working channels of PU_i. Furthermore, a Markov model in Figure 4 is adopted to simulate the working channels of the PUs. At a time slot, each channel of PU_i has a probability of α_i to change its state from OFF for SUs (PU_i works on it) to ON for SUs (PU_i does not work on it); similarly, each channel has a probability of β_i to change form ON to OFF.

Based on the above method, we set up two CRN instances with different characteristics: Network 1 and Network 2 (Figure 5 and 6). The parameters are listed in Table 1. In the figures, a red dot represents the location of a PU and the red circle centered at it indicates its dominating range. A blue triangle dot represents the location of a secondary user. Two secondary users connected by a blue line means they are a secondary pair. Initially, the working channels of PUs are set randomly.

5.1 The Centralized Stable Channel Selection Algorithm

In the centralized condition, the Greedy Channel Selection Algorithm (GCS) in [16] is also implemented for comparison, which directly assigns the channels through computing the maximum bipartite matching in $G = (V(U, C), E)$. The main difference from our method is that we consider the stability of the entire network. Because both methods can achieve the maximum connected links, we only compare the property of stability through calculating switches, disconnections and average communication time of links. One disconnection means a connected link is disconnected. Switches increases by one when a disconnection happens or a link changes its working channel. We calculate the above parameters in both Network 1 and 2. We generate the two networks 200 times based on

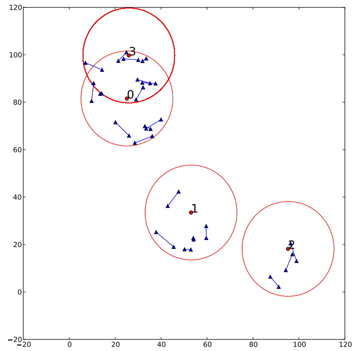

Fig. 5. Network 1

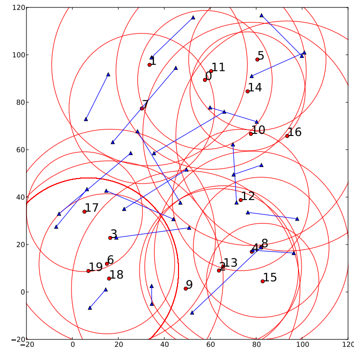

Fig. 6. Network 2

Table 1. Settings of two Networks

Settings	Network 1	Network 2	Network 3
Number of Channel	20	30	10 to 100
Number of Primary User	4	20	10
Number of Secondary User	42	40	2
Average value of α	0.3	0.9	0.3
Average value of β	0.8	0.9	0.8
Maximum dominating range of PU	20	50	80
Maximum communicating range of SU	10	40	30

their parameters in Table 1, and run the two centralized algorithms to take the average values as our simulation results. Figure 7 and 8 illustrate the numbers of disconnections at each round in Network 1 and 2 respectively. In network 1, our algorithm has a similar disconnection fluctuation with the GCS because of the simple network structure. In the complex Network 2, our method has a clearly lower numbers of disconnection. Figure 9 and 10 show that our algorithm has lower switches. Figure 11 and 12 indicate the higher average communication time per link of our algorithm. All experiments validate that our algorithm have the better stability than the GCS Algorithm.

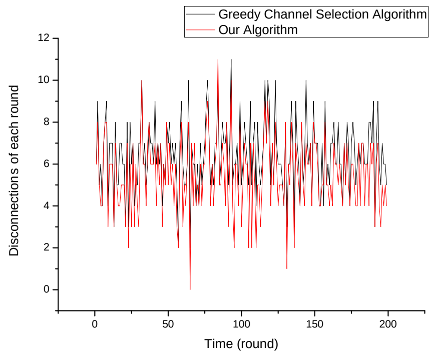

Fig. 7. Disconnections on network 1 at each round

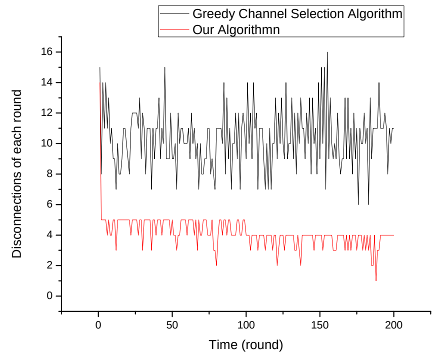

Fig. 8. Disconnections on network 2 at each round

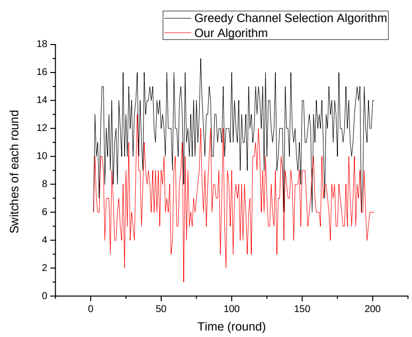

Fig. 9. Switches on network 1 at each round

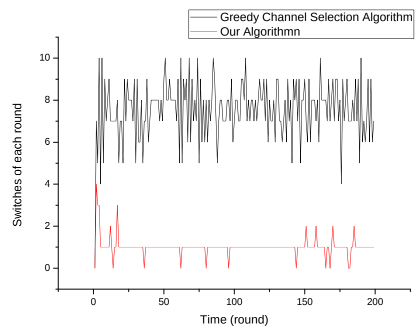

Fig. 10. Switches on network 2 at each round

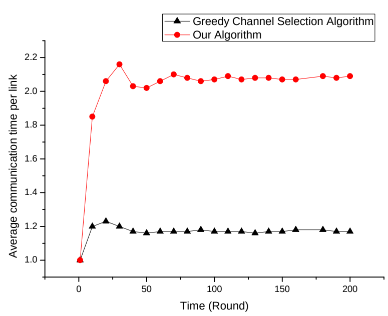

Fig. 11. Average communication time per link on network 1

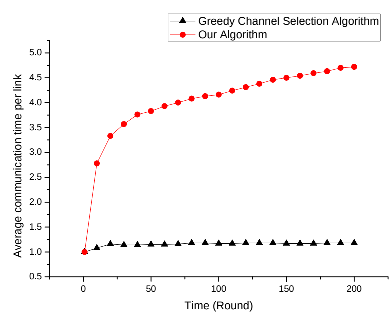

Fig. 12. Average communication time per link on network 2

5.2 The Distributed Strategy

For the rendezvous strategies for two SUs, we will compare our strategy Algorithm 2 with the well-known Jump-stay algorithm in [9] and the random selection. We calculate the link commutation time and expected time to rendezvous (ETTR) to evaluate the link stability and rendezvous speeds of the three methods.

We generate the network based on the parameters of Network 3 in Table 1. The simulation program will randomly put two SUs on 200 different sets of positions. In one set of position, 400 timeslots are observed, and each algorithm will be repeated for 100 times. We take the average communication time as our results, and the average TTR as the expected TTR (ETTR). It will be regarded as a failure once two nodes can not reach rendezvous within 400 timeslots.

We set that our algorithm will only collect the channel information for the first 30 rounds (i.e., we set $\tau = 30$). Figure 13 shows that our strategy has a higher cumulative communication time after about 60 time-slots than the other methods. Here, cumulative communication time means during the 400 timeslots, once the link is disconnected we establish the link again using the same strategy and sum up the communication times. Figure 14 shows the fact that our strategy has the lowest ETTR for all different cases of the the total number of channels if we neglect the time for collecting the channel information.

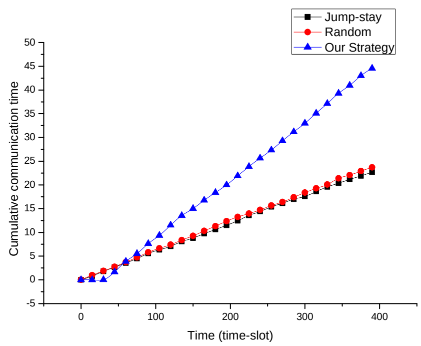

Fig. 13. Cumulative Communication time (number of channels=40)

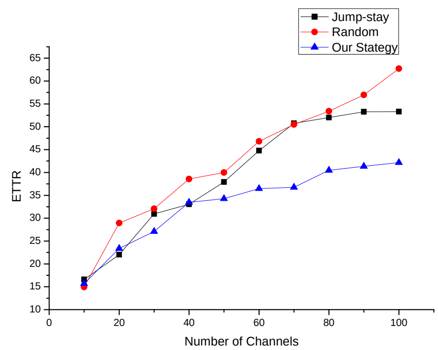

Fig. 14. Expected time to rendezvous

6 Conclusion

In this paper, we investigate the channel selection problem in cognitive radio network with high link stability. We formulate the stable channel selection (SCS) problem under centralized and distributed conditions. For the centralized case, we solve the SCS problem and achieve the maximum number of connected links based on computing the maximum cost maximum flow. The links established also have a good property of stability. For the distributed case, we consider the 2-user rendezvous problem and propose a novel algorithm which utilizes the idle rates of each channel to achieve high network stability. Our rendezvous strategy have a small time to rendezvous and can be adopted in asynchronous environments. Extensive simulations validate our algorithms have higher stability than other methods in most cases. The study of the distributed rendezvous strategies with high link stability for multiple users will be an interesting extension of this work. Future work can also include studying the channel assignment under a more realistic interference model, such as the physical interference models.

References

1. P. Bahl, R. Chandra, T. Moscibroda, R. Murty, and M. Welsh, "White space networking with wi-fi like connectivity," *ACM SIGCOMM Computer Communication Review*, vol. 39, no. 4, pp. 27–38, 2009.
2. H. Liang, T. Lou, H. Tan, Y. Wang, and D. Yu, "On the complexity of connectivity in cognitive radio networks through spectrum assignment," *Journal of Combinatorial Optimization*, 2013.
3. C. Cordeiro, K. Challapali, D. Birru, and N. Sai Shankar, "IEEE 802.22: the first worldwide wireless standard based on cognitive radios," in *IEEE International Symposium on New Frontiers in Dynamic Spectrum Access Networks (DySPAN)*, Nov 2005, pp. 328–337.
4. J. Jia, Q. Zhang, and X. Shen, "HC-MAC: A hardware-constrained cognitive mac for efficient spectrum management," *IEEE Journal on Selected Areas in Communications*, vol. 26, no. 1, pp. 106–117, 2008.
5. J. Pérez-Romero, O. Salient, R. Agustí, and L. Giupponi, "A novel on-demand cognitive pilot channel enabling dynamic spectrum allocation," in *IEEE International Symposium on New Frontiers in Dynamic Spectrum Access Networks (DySPAN)*, 2007, pp. 46–54.
6. J. Zhao, H. Zheng, and G.-H. Yang, "Distributed coordination in dynamic spectrum allocation networks," in *IEEE International Symposium on New Frontiers in Dynamic Spectrum Access Networks (DySPAN)*, 2005, pp. 259–268.
7. K. Bian, J.-M. Park, and R. Chen, "Control channel establishment in cognitive radio networks using channel hopping," *IEEE Journal on Selected Areas in Communications*, vol. 29, no. 4, pp. 689–703, 2011.
8. R. Gandhi, C.-C. Wang, and Y. C. Hu, "Fast rendezvous for multiple clients for cognitive radios using coordinated channel hopping," in *Annual IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks (SECON)*, 2012, pp. 434–442.
9. Z. Lin, H. Liu, X. Chu, and Y.-W. Leung, "Jump-stay based channel-hopping algorithm with guaranteed rendezvous for cognitive radio networks," in *IEEE INFOCOM*, 2011, pp. 2444–2452.
10. N. C. Theis, R. W. Thomas, and L. A. DaSilva, "Rendezvous for cognitive radios," *IEEE Transactions on Mobile Computing*, vol. 10, no. 2, pp. 216–227, 2011.
11. Y. Azar, O. Gurel-Gurevich, E. Lubetzky, and T. Moscibroda, "Optimal discovery strategies in white space networks," in *ESA*, 2011, pp. 713–722.
12. E. J. Anderson and R. Weber, "The rendezvous problem on discrete locations," *Journal of Applied Probability*, pp. 839–851, 1990.
13. S. Romaszko and P. Mahonen, "Grid-based channel mapping in cognitive radio ad hoc networks," in *IEEE International Symposium on Personal Indoor and Mobile Radio Communications (PIMRC)*, 2011, pp. 438–444.
14. S. Romaszko, D. Denkovski, V. Pavlovska, and L. Gavrilovska, "Asynchronous rendezvous protocol for cognitive radio ad hoc networks," in *Ad Hoc Networks*. Springer, 2013, pp. 135–148.
15. K. Bian and J.-M. Park, "Asynchronous channel hopping for establishing rendezvous in cognitive radio networks," in *IEEE INFOCOM*, 2011, pp. 236–240.
16. F. Hou and J. Huang, "Dynamic channel selection in cognitive radio network with channel heterogeneity," in *Global Telecommunications Conference (GLOBECOM)*, Dec 2010, pp. 1–6.