Neighbor Discovery for Cognitive Radio Ad Hoc Networks

Zaw Htike
Department of Computer Engineering,
Kyung Hee University,
1 Seocheon,Giheung, Yongin, Gyeonggi
449-701 Korea
+82-10-2561-2811
htike@networking.khu.ac.kr,

Choong Seon Hong
Department of Computer Engineering,
Kyung Hee University,
1 Seocheon,Giheung, Yongin, Gyeonggi
449-701 Korea
+82-31-201-2532
cshong@khu.ac.kr

ABSTRACT

Neighbor discovery is a critical task in cognitive radio ad hoc networks since the secondary users operate on available channels which are dynamically changing according to the primary users activities. In this paper, we propose a neighbors discovery mechanism that provides how a secondary user could find its neighbor in cognitive radio ad hoc networks without any collaboration. In this mechanism, no prior knowledge of neighboring users is required. In the cognitive radio networks, the available channel sets are varying and users choose neighbor discovery strategies according to the number of available channels they observed. In other words, neighbor discovery strategies are adaptively changing in a dynamic environment. The strategies are simple and totally distributed, but these provide minimum time to rendezvous (TTR).

Categories and Subject Descriptors

C.2 [Computer communication networks]: Wireless communication,

General Terms

Algorithm

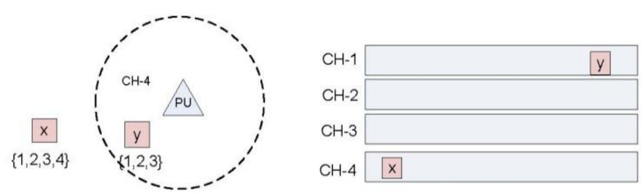
Keywords

Neighbor discovery, time to rendezvous (TTR), and cognitive radio ad hoc networks

This research was supported by Next-Generation Information Computing Development Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science and Technology (2012-0006421). Dr. CS Hong is the corresponding author.

1. INTRODUCTION

In recent years, many research works have been involved in the development of cognitive radio (CR) technology since it has been recognized as a new way to improve the spectral efficiency of wireless networks. In cognitive radio network, secondary users (SUs) are allowed to utilize free or idle portions of licensed channels (or spectrum) without causing any interference to primary users (PU) [19]. Generally, SUs detect the idle channels and access the channels [21]. The channel availability for SUs is varying according to PU activities which change dynamically in frequency, space and time [20]. Therefore, the available channels for each SU might also change dynamically. However, if a pair of SUs wishes to communicate with each other, they need to rendezvous on a channel that is commonly available to both of them and exchange necessary control information for negotiation such as request-to-send (RTS)/clear-to-send (CTS) handshaking of the 802.11 Distributed Coordination Function (DCF) [7]. This task is not trivial in CR networks since SUs may operate on different channels independently. This is generally called rendezvous or neighbor discovery problem [14].



(a) (b)

Figure 1. (a) Users have different available channel sets and (b) users are dwelling on different channels independently.

1.1. Problem Statement

The neighbor discovery is more challenging in cognitive radio ad hoc networks as there is no centralized controller [17]. Figure 1 represents the neighbor discovery problem of cognitive radio ad hoc networks. As shown in Figure 1, there are four channels for

SUs and, *user x* and *y* have different available channel sets as the user *y* is in the transmission range of a PU, and users are currently dwelling on different channels. When a user, user *x*, wants to communicate with its neighbor, user *y*, these two users need to rendezvous on a channel which is commonly available to them. In this case, the available channels are $\{CH_1, CH_2, CH_3\}$, since CH_4 is occupied by a PU and, thus, not commonly available for both SUs.

At the very initial state of the networks, SUs have no information regarding their neighbors such as the channel that the neighbor currently dwelling on. However, by performing channel sensing [16], SUs can achieve local information such as number of available channels, PU activities of each channel within its region, etc. If an SU needs to initiate any communication, it has to find its neighbor by using its own local information. Moreover, the SU needs to rendezvous with intended neighbor within a reasonable time interval which is, generally, called *time to rendezvous* (TTR).

1.2 Related Works

There are two famous approaches to enable neighbor discovery in cognitive radio ad hoc networks;

1. Using dedicated common control channel (CCC) and
2. Using channel hopping

Using dedicated common control channel is the simplest way to enable neighbor discovery in CR networks. Most of the proposed medium access control (MAC) protocols for cognitive radio networks were designed by assuming the existence of a CCC and further assuming that it is available for every secondary user [11]. In fact, this approach originated from the concept of MAC protocols for multi-channel wireless networks [15] [18]. In this approach, the CCC serves as a rendezvous channel and all the necessary control information is exchanged among SUs via the CCC. In the CCC approaches, time is divided into two intervals; negotiation or control interval and data interval. When an SU wants to initiate communication, it first switches to the CCC during the negotiation interval and attempts to negotiate with the intended receiver or neighbor. After negotiating on the CCC, data communication can be accomplished during the data interval via other available channels, known as data channels [10]. Figure.1 illustrates the normal operation of a network with a common control channel. As shown in Figure.1, all users attempt to negotiate on the CCC during the control interval. After the negotiation is complete on the CCC, the users move to selected channels and perform data communications simultaneously during the data interval. Obviously, using a CCC can simplify the neighbor discovery process [13]; yet, it is often not feasible or impractical due to *lack of CCC availability*.

The main drawback using the CCC approach is it is susceptible to primary user activities. When PUs appear on the CCC, all SUs must defer their transmissions on the CCC and vacate the channel immediately. Not only does PUs' presence degrade the overall throughput of a CR network, but if the transmission period of PUs is significantly long on the CCC, the presence of the PUs may also block channel access for SUs. Moreover, the available channel sets in CR networks, including the CCC, change dynamically, hindering the establishment of an ever-available control channel for all SUs. Thus, in the dynamic environment, an ever-available channel for all SUs is unlikely to exist.

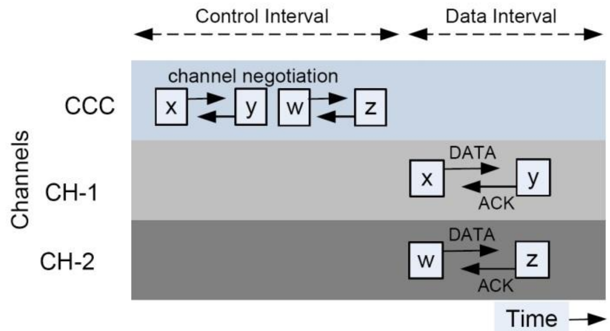


Figure 2. Operation of a protocol with CCC

Another popular solution to find the neighbors in CR ad hoc networks is using channel hopping sequences [1] [2]. The main advantage of the channel hopping sequences, compare to the CCC approach, is that SUs can rendezvous with the neighbors at any available channel. Therefore, it can overcome long-term blocking of PU and an ever-available common channel is not required. In this approach, SUs generate their own channel hopping sequences. When an SU (e.g., user *x*) needs to communicate with its neighbor (user *y*), it switches from one channel to another by following a predefined hopping sequence until it finds its neighbor. Figure 2 illustrates the operation of a channel hopping protocol. As shown in figure, user *x* and *y* attempt to find each other by following their own hopping sequences. When they rendezvous on channel 2, they can perform communication.

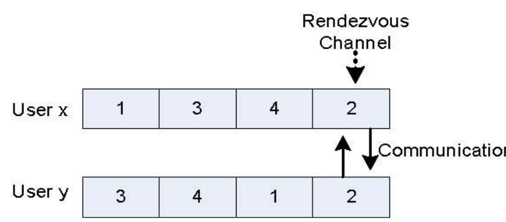


Figure 2. Operation of a sequence based protocol

However, channel hopping protocols have the following shortcomings.

a) Lack of network status information

In sequence-based approaches, SUs generate channel hopping sequences with an assumption that all SUs use the same channel labels. With this assumption, sequence-based approaches provide upper bound of time to rendezvous. However, this is a big assumption since SUs have no information regarding the neighbors at the initial state of the network.

b) Require Synchronization

Most of the sequence-based approaches need synchronization among users [3][4] which is hard to achieve in ad hoc networks. For example, in [3], time is divided into fixed-time intervals in which each represents one of the available channels. At the beginning of the time slot, every node in the network must switch to the corresponding channel for negotiation.

c) Complexity

The next difficulty in sequence-based protocols is overcoming the complexity of generating channel hopping sequences. Designing the channel hopping algorithm is a great challenge because when users generate channel hopping sequences, any pair of these sequences should overlap at least once within a sequence period, so that any pair of users which needs to communicate can rendezvous [6]. Moreover, the TTR values between any pair of sequences should be reasonable and, obviously, it is determined by channel hopping algorithm.

In [3], channel hopping sequences were created in a round robin fashion and, as mentioned above, this protocol requires tight synchronization among SUs which is difficult to achieve in ad hoc environment. In [8], the authors proposed biased pseudo-random sequences. These sequences do not need tight synchronization, but the average TTR may not be bounded. The authors of [2] proposed permutation-based channel hopping sequences. In their proposal, the expected time to rendezvous was bounded by a quadratic function of the number of available channels. A quorum-based scheme was proposed in [9], and the authors claimed that rendezvous between any pair of users can occur at least once within n^2 time slots, where n is the number of available channels.

In this paper, we propose an alternative way of enabling neighbor rendezvous in cognitive radio ad hoc networks. Our proposed mechanism uses neither dedicated common control channel nor channel hopping. So, the proposed mechanism does not require generating the predefined hopping sequences. In our proposed neighbor discovery mechanism, (1) SUs need only local information, such as number of available channels, to perform neighbors discovery. Therefore, it can be applied in a distributed manner. (2) The algorithms are simple and (3) these can provide less expected time to rendezvous than sequence-based approaches. Moreover, this mechanism is immune from some irrational assumptions such as (1) all channels are indexed with the same labels by secondary users and (2) tight synchronization among users.

We present our proposed mechanism in section.2. In section 2, we describe the neighbor discovery strategies and discuss how to update the strategies according to the number of available channels that SUs observed. Then, the comparisons of the numerical results are described in section 3. Section 4 concludes the paper.

2. NEIGHBORS DISCOVERY MECHANISM

We assume that neighbor discovery process can be accomplished between two SUs if they rendezvous on the same channel and exchange necessary control information (i.e. neighbors discovery message (DOV) and acknowledgment (ACK)). DOV is just a probe message and any SU, which receives DOV, can simply reply the ACK. The goal of every SU is to find its neighbors with minimum delay or TTR. We also assume that the PU activities on the channels are dynamically changing. Thus, SUs updates the neighbor discovery strategies every round according to the number of available channels they observed.

2.1 Neighbor Discovery Strategies

When an SU needs to perform neighbor discovery, it performs channel sensing first and create an available channel set. All free channels (the channels that are not currently used by PUs) will be included in the available channel set. Then SU chooses one of the following two strategies;

• ***Strategy one***: Switch one available channel after another without repeating and find a neighbor.

• ***Strategy two***: Randomly select one of the available channels and wait for a neighbor on selected channel.

SUs perform neighbor discovery until they rendezvous with their neighbors. First, the SU (user x) that chooses *strategy one* selects an available channel randomly. Then it switches to the selected channel and senses for the presence of PU. If the SU senses the channel is free, it will broadcast *DOV* and waits for the *ACK*. All message transmissions follow the principle of Distributed Coordination Function (DCF) of IEEE 802.11 [6] and the basic procedure of packets transmission can be seen in Figure.3.

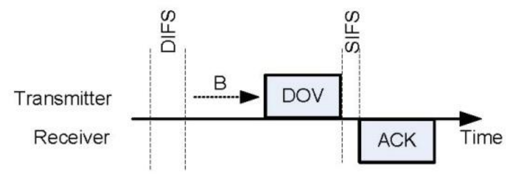


Figure 3. Procedure of packets transmission

The time interval of the whole process is defined as one time slot, T_{slot} , and it can be estimated as,

$$T_{slot} = \frac{ACK + DOV}{T_{rate}} + DIFS + SIFS + B \,, \qquad (1)$$

where, T_{rate} is transmission rate and B represents the random back-off. While the SU is waiting the ACK, it may receive DOV from other neighbors. If it receives DOV instead of ACK, it will simply reply ACK and neighbor discovery has completed for these two SUs. If *user x* does not receive any ACK or DOV, it shall switch to another available channel and broadcast the DOV again. This process is repeated and one round of neighbor discovery for *user x* is over when it has received the ACK from one of its neighbors or after it has visited all available channels. The necessary time interval for one round of neighbor discovery can be expressed as

$$T_{round} = n.T_{slot} \,, \qquad (2)$$

where, n is the total number of available channels.

The SU that chooses *strategy two*, *user y*, just selects a random channel from the available channel list and waits its neighbor for one T_{round} . If it receives *DOV* from its neighbor, let say from *user x*, it will reply *ACK* and the neighbor discovery has been successfully accomplished between these two SUs, x and y. One round of *user y* is over after it has received *DOV* and replied *ACK* or one T_{round} has expired. If user y does not rendezvous with any of its neighbor within a T_{round} , it will perform channel sensing again and update its strategies for next round. Choosing strategies for next round is independent of previous round but on the available channels it senses (the reason of why decision making depends on the available channels can be found in section 2.3).

2.2 Expected Payoffs

The expected payoffs for different events are represented with *TTR*. According to neighbor discovery strategies, any pair of SUs can be in one of the following three events.

- *Event A*: Both SUs choose the same strategy, *strategy one*.
- *Event B*: SUs choose *different strategies*.
- *Event C*: Both SUs choose the same strategy, *strategy two*.

If *event A* occurs, both SUs try to find each other by switching from one channel to another without revisiting the channels. In this event, the probability of meeting these two users on CH_i is

$$P(meeting) = \frac{1}{n}. \quad (3)$$

Then, the probability of not meeting at all within a round becomes

$$P(no\ meeting) = (1-\frac{1}{n})^n. \quad (4)$$

The probability of at least one rendezvous occurs within a round is

$$P_x(x \geq one\ meeting) = 1-(1-\frac{1}{n})^n. \quad (5)$$

Then, the expected time slots for this event can be estimated as:

$$E_A[TTR] = n(1-(1-\frac{1}{n})^n). \quad (6)$$

If *event B* occurs, SU that chooses strategy one (user *x*) switches from one channel to another without repeating and tries to find its neighbors. Neighbor SU (user *y*) that chooses strategy two selects a channel randomly and waits its neighbor for one T_{round}. The probability of meeting these two SUs on CH_i is $\frac{1}{n}$. If *user x* does not rendezvous with its neighbor, it will switch to another channel while *user y* is waiting on the selected channel. Therefore, The probability of meeting on next channel (CH_{i+1}) is $\frac{2}{n}$. Similarly, the probability of meeting on CH_{i+2} becomes $\frac{3}{n}$. Then, the expected time to rendezvous for this event is

$$E_B[TTR] = \sum_{i=1}^{n}\frac{i}{n} = \frac{1+2+...+n}{n} = \frac{n+1}{2}. \quad (7)$$

If *event C* occurs, obviously these two SUs will not meet each other within a round. One T_{round} is wasted in this situation and the expected time slots for this event can be expressed as

$$E_C[TTR] = T_{round} = n. \quad (8)$$

Suppose *p* and *q* (= *1 - p*) are probabilities of choosing strategy one and two respectively, then we can describe the overall expected time slots for one round with the following expression.

$$E[TTR] = p(1-p)(n+1) + np^2(1-(\frac{n-1}{n})^n) + n(1-p)^2. \quad (9)$$

2.3 Optimal Strategies

Obviously the aim of SUs is to minimize the expected payoff. We take derivative to (6) and we get optimal value of *p* as

$$p = \frac{n-1}{2n(1-(\frac{n-1}{n})^n)-2}. \quad (10)$$

It is clear that the optimal strategy profiles *(p, q)* are the function of number of available channels, *n*. Therefore (as mentioned in previous section), each SU updates its strategies every round according to the number of available channels it senses.

3. NUMERCIAL RESULTS

It is obvious that SUs do not want event C. The authors of [7] claimed that optimal values of *p* and *q* are 0.75 and 0.25 respectively as $n \to \infty$. In proposed neighbor discovery mechanism, SUs are more aggressive to choose strategy one, as $n \to \infty, p \to 0.8$. When we compare the results, proposed mechanism provides less occurrences of event C and it can achieve minimum E[TTR] as shown in Table 1.

Table 1. The E[TTR] against different optimal values.

	Optimal q		*E[TTR]*	
n	Ref [7]	Proposed ND	Ref [7]	Proposed ND
2	0.5000	0.0000	2.0000	1.5000
3	0.3333	0.1000	2.6667	2.1000
4	0.3220	0.1351	3.5683	2.7027
5	0.3012	0.1531	4.3793	3.3062
6	0.2914	0.1641	5.2133	3.9101
7	0.2842	0.1714	6.0421	4.5142
8	0.2791	0.1767	6.8716	5.1184
9	0.2753	0.1807	4.3793	5.7227
10	0.2722	0.1838	8.5300	6.3270
50	0.2521	0.2043	8.5300	30.5060
$n \to \infty$	0.2475	$q \to 0.2000$		

We also compare the results of proposed neighbor discovery (ND) mechanism and that of random algorithm *(RA)* and orthogonal sequence-based algorithm *(OSA)* from [14]. Random Algorithm is similar to event A. Nodes (SUs) try to find each other by switching from one channel to another until they rendezvous one of their neighbors on a common channel. The expected time to rendezvous of the random algorithm is

$$E_{RA}[TTR] = \frac{n^2}{P_h.n}, \quad (11)$$

where, n is the number of available channels and P_h is the probability of a successful handshake. OSA algorithm is an alternative of random algorithm. The difference is, in OSA, nodes switch channel by following a predefined hopping sequence like sequence-based approaches and the expected TTR value OSA algorithm is

$$E_{OSA}[TTR] = \frac{n^4 + 2n^2 + 6n - 3}{3n(n+1)}. \quad (12)$$

Figure.4 shows comparison of the results in terms of E[TTR]. As shown in the figure, the proposed neighbor discovery mechanism provides less TTR than that of OSA and random algorithms.

4. CONCLUSION

We have presented a neighbors discovery mechanism for cognitive radio ad hoc networks. In this mechanism, the neighbor discovery strategies are simple and totally distributed. It is flexible with dynamic nature of CR networks as SUs update their strategies according to the number of available channels. Moreover, the numerical results confirm that it provides less expected time to rendezvous than previous works.

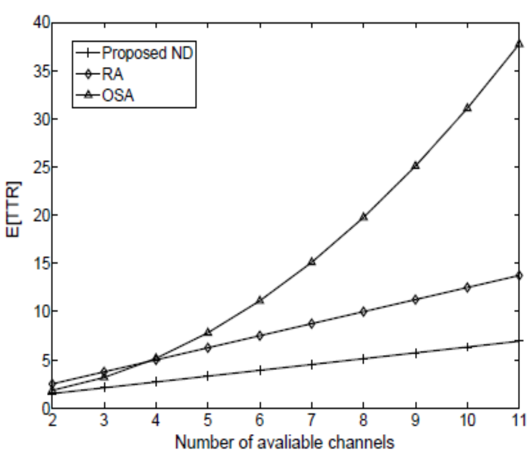


Figure 4. E [TTR] of OSA, RA and proposed neighbor discovery mechanism

REFERENCES

[1] J. Shin, D. Yang, and C. Kim, "A channel rendezvous scheme for cognitive radio networks," *IEEE COMMUNICATIONS LETTERS*, vol. 14, no. 10, pp. 954–956, November 2010.

[2] L. DaSilva and I. Guerreiro, "Sequence-based rendezvous for dynamic spectrum access." IEEE DySpan, 2008, pp. 1 – 7.

[3] Y. Kondareddy and P. Agrawal, "Synchronized MAC protocol for multihop cognitive radio networks," IEEE ICC, 2008.

[4] C.-F. Shih, T. Y. Wu, and W. Liao, "DH-MAC: A dynamic channel hopping MAC protocol for cognitive radio networks." IEEE ICC, 2010, pp. 1–5.

[5] 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.

[6] G. Bianchi, "Performance analysis of the IEEE 802.11 distributed coordination function," *IEEE Journal on Selected Areas in Communications*, vol. 18, no. 3, pp. 535 – 547, March 2000.

[7] E. Anderson and R. Weber, "The rendezvous problem on discrete locations," *Journal of Applied Probability*, vol. 27, no. 4, pp. 839–851, 1990.

[8] Cormio and K. Chowdhury, "Common control channel design for cognitive radio wireless ad hoc networks using adaptive frequency hopping," *Ad Hoc Networks*, vol. 8, no. 4, pp. 430–438, June 2010.

[9] K. Bian, J.-M. Park, and R. Chen, "A quorum-based framework for establishing control channels in dynamic spectrum access networks,." MobiCom 2009, 2009, pp. 25–36.

[10] X. Zhang and H. Su, "CREAM-MAC: cognitive radio-enabled multichannel MAC protocol over dynamic spectrum access networks," *IEEE Journal of Selected Topics in Signal Processing*, vol. 5, no. 1, pp. 110 – 123, 2011.

[11] C. Cordeiro and K. Challapali, "C-MAC: A cognitive MAC protocol for multi-channel wireless networks." IEEE DySpan, 2008, pp. 147–157.

[12] M. Timmers, S. Pollin, A. Dejonghe, L. Van der Perre, and F. Catthoor, "A distributed multichannel mac protocol for multihop cognitive radio networks," *IEEE Transactions on Vehicular Technology,*, vol. 59, no. 1, pp. 446 – 459, 2010.

[13] B. F. Lo, "A survey of common control channel design in cognitive radio networks," *Physical Communication*, vol. 4, no. 1, pp. 26–39, March 2011.

[14] 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, February 2011.

[15] Duc Ngoc Minh Dang, Mui Van Nguyen, Choong Seon Hong, Sungwon Lee and Kwangsue Chung, "An Energy Efficient Multi-channel MAC Protocol for Wireless Ad hoc Networks", IEEE Globecom 2012, Dec 3-7, 2012, Anaheim, California, USA

[16] T. Yucek and H. Arslan, "A survey of spectrum sensing algorithms for cognitive radio applications," *IEEE Communications Surveys & Tutorials*, vol. 11, no. 1, pp. 116 – 130, First Quarter 2009.

[17] I. F. Akyildiz, W.-Y. Lee, and K. R. Chowdhury, "CRAHNs: Cognitive radio ad hoc networks," *Ad Hoc Networks*, vol. 7, no. 5, pp. 810–836, July 2009.

[18] Jungmin So, Nitin H. Vaidya. "*Multi-Channel MAC for Ad hoc Networks: Handling Multi-channel Hidden Terminals Using A Single Transceiver"*, Proceedings of the 5th ACM international symposium on Mobile ad hoc networking and computing *(Mobihoc'04)*, Pages 222-233, 2004.

[19] I.F. Akyildiz, W. Y. Lee, M. C. Vuran, and S. Mohanty, "NeXt generation / dynamic spectrum access / cognitive radio wireless networks: A survey," *Computer Networks Journal (Elsevier)*, Volume 50, Issue 13, 15 September 2006 Pages 2127–2159

[20] I. Akyildiz, W. Lee, M. Vuran, and S. Mohanty, "A survey on spectrum management in cognitive radio networks," *IEEE Communications Magazine,* April 2008, pp. 40-48.

[21] A. De Domenico, E. C. Strinati, and M.-G. Di Benedetto, "A Survey on MAC Strategies for Cognitive Radio Networks" IEEE Communications Surveys & Tutorials, First Quarter 2012, Page(s): 21 - 44