Deterministic Distributed Rendezvous Algorithms for Multi-Radio Cognitive Radio Networks

Guyue Li
Tsinghua University
Beijing, P.R. China
liguyue12@gmail.com

Zhaoquan Gu
Tsinghua University
Beijing, P.R. China
demin456@gmail.com

Xiao Lin
Tsinghua University
Beijing, P.R. China
jackielinxiao@gmail.com

Haosen Pu
Tsinghua University
Beijing, P.R. China
phs199205@gmail.com

Qiang-Sheng Hua
Tsinghua University
Beijing, P.R. China
qshua@mail.tsinghua.edu.cn

ABSTRACT

Rendezvous is a fundamental process in constructing Cognitive Radio Networks (CRNs), through which the user can communicate with its neighbors by establishing a link on some licensed frequency band (channel). Most of the existing elegant rendezvous algorithms assume each user is equipped with a single radio. Nowadays the multi-radio cognitive radio architecture, where each user can access $k \geq 2$ channels at the same time, has become a reality. In this paper, we study the rendezvous problem in multi-radio CRN to see whether and to what extent the multi-radio capability can improve the rendezvous performance. To begin with, we propose a family of deterministic distributed algorithms for two special situations when $k = 2$ and $k = O(\sqrt{n})$, where n is the number of all channels. These algorithms show that the maximum time to rendezvous ($MTTR$) can be reduced (largely) in multi-radio CRN. Then we derive a lower bound of $MTTR$ as $\Omega(\frac{|V_i||V_j|}{k^2})$ for arbitrary k (V_i, V_j represents two users' available channel sets) and present a distributed algorithm to guarantee rendezvous in $O(\frac{|V_i||V_j|}{k^2})$ time slots, which meets the lower bound. Extensive simulations are conducted to corroborate our theoretical analyses.

Categories and Subject Descriptors

C.2.1 [**Network Architecture and Design**]: Wireless communication

Keywords

Multi-Radio; Cognitive Radio Network; Rendezvous

1. INTRODUCTION

Due to the rapid growth of wireless devices and the increasing demand for wireless services, the wireless spectrum has become very scarce. The unlicensed spectrum has been overcrowded such as the Industrial Scientific and Medical (ISM) band [11], while the utilization of some licensed spectrum is pretty low. Cognitive Radio Network (CRN) is such a promising paradigm to tackle the spectrum scarcity problem [2], where primary users (PUs) who own the licensed spectrum coexist with secondary users (SUs) that can opportunistically exploit and access the unused licensed spectrum. For convenience, "user" mentioned hereafter in the paper refers to SU.

Rendezvous is a fundamental process in constructing a CRN, through which two neighboring users can establish a link on some common frequency band (channel) for communication. Being a key role, rendezvous is the cornerstone of many interesting problems, such as message broadcasting [15, 24], routing [14], and data collection [6]. Generally speaking, the licensed spectrum is assumed to be divided into n non-overlapping channels (frequency bands) and the users are equipped with cognitive radios that can sense the usage of these channels. After the spectrum sensing stage, the users can find out the unused channels by the PUs, which we called *available channels* and they can access these channels for rendezvous at any time. Once two users access the same channel at the same time, rendezvous is achieved and they can communicate with each other. *Time to rendezvous*(TTR) is used to measure the time cost before rendezvous and denote *maximum time to rendezvous*($MTTR$) as the worst situation of the rendezvous algorithm, considering two users may have different sets of available channels when the spectrum usage of the PUs varies temporally and geographically.

The state-of-the-art rendezvous algorithms can be mainly divided into two categories: global sequence (GS) based algorithms [13, 19, 23] and local sequence (LS) based ones [7, 12]. GS algorithms design channel access strategies on the basis of all licensed channels, regardless of whether they are available. The best result [13] guarantees rendezvous for any two users in $O(n^2)$ time slots if their available channels sets intersect. LS algorithms make use of each user's available channels and they can reduce the TTR largely when the portion of available channels counts for a small fraction. Specifically, for two users with available channel sets V_i, V_j, [12] guarantees rendezvous in $O(|V_i|^2)$ (suppose $|V_i| \geq |V_j|$) time slots when each user has a distinct iden-

MSWiM'14, September 21–26, 2014, Montreal, QC, Canada.

http://dx.doi.org/10.1145/2641798.2641827.

tifier and [7] guarantees rendezvous in $O(|V_i||V_j|\log\log n)$ time slots. However, all these works are proposed for single-radio CRN setting, which means each user can only access 1 channel in each time slot. In this paper, we focus on designing distributed rendezvous algorithms for multi-radio CRN, where each user can access $k \geq 2$ channels at the same time.

In recent years, multi-radio architecture has been widely used in wireless mesh networks [10,22]. Due to the hardware limitations, most works use the single-radio architecture for cognitive radio network, which implies only 1 channel can be accessed at the same time. However, [16] has implemented a multi-radio cognitive radio network at UCLA where each node is equipped with multiple radios to sense and access channels. [1] also proposed robust channel assignment for multi-radio cognitive radio network. To the best of our knowledge, only two papers considered the rendezvous process for multi-radio CRN, which should play an important role in constructing the CRN. [20] proposed a multi-interface rendezvous in self-organizing CRN. And [25] investigated multiple radios for effective rendezvous in CRN. However, these two works either did not give the theoretic $MTTR$ bound regarding the benefits of using multi-radios [20] or did not show how far their rendezvous performance is away from the optimal [25]. In our paper, we aim to answer one question: to what extent could multi-radio improve the $MTTR$ compared to single-radio CRN.

In order to solve the rendezvous problem in multi-radio CRN, we first design different algorithms for two special situations. When $k = 2$, which means the user can access two channels in the same time slot, we design both GS and LS algorithms, where the GS algorithm guarantees rendezvous in $O(n^2)$ time slots (only constant factor lower than single-radio CRN [13]) and the LS one guarantees rendezvous in $O(|V_i||V_j|)$ time slots ($\log\log n$ factor lower than [7]). For the other case $k = O(\sqrt{n})$, rendezvous is guaranteed in $O(n)$ time slots based on the quorum system method. Second, we derive the lower bound as $\Omega(\frac{|V_i||V_j|}{k^2})$ when the user can access arbitrary k channels at the same time, and we present how to meet the bound based on the LS algorithm. Finally, we conduct extensive simulations and these results corroborate our theoretical analyses.

The rest of the paper is organized as follows. The next section introduces some related works. Model and problem formulation are provided in Section 3. We show the algorithms for two special situations in Section 4. The lower bound is derived in Section 5 and a general construction for rendezvous to meet this lower bound is presented in Section 6. Simulation results are depicted in Section 7 and we conclude the paper in Section 8.

2. RELATED WORK

Rendezvous algorithms can be divided into two categories: centralized and decentralized algorithm. Assuming a central controller or a dedicated Common Control Channel (CCC) exists [17,21], centralized algorithm can be realized by communicating through the central controller or the CCC, but it's vulnerable to adversary attacks and easily overcrowded when the number of users increases.

The main part of decentralized algorithms is *blind rendezvous algorithm*, where no centralization or CCC exists. The common technique used in blind rendezvous algorithms is Channel Hopping (CH), which means each user can generate a specific sequence and hop among the available channels according to it. The state-of-the-art distributed rendezvous algorithms for single-radio CRN are summarized in Table 2. Generally speaking, there are two types of sequences used in extant works: *global sequence* (GS) is constructed based on all channels' information, and *local sequence* (LS) is generated on the basis of the user's available channels. Thus different users generate the same global sequence, but they could have different local sequences.

Table 1: MTTR for Single-Radio Rendezvous

Algorithms	$MTTR$
GOS [9]	$n(n+1) = O(n^2)$
Jump-Stay [19]	$3nP^2 + 3P = O(n^3)$
CRSEQ [23]	$P(3P-1) = O(n^2)$
DRDS [13]	$3P^2 = O(n^2)$
AHW [8]	$3P^2 \log M = O(n^2 \log M)$
MLS [12]	$O(\|V_i\|^2) = o(n^2)$
Result [7]	$O(\|V_i\|\|V_j\|\log\log n) = O(n^2\log\log n)$

Remarks: 1)P is the smallest prime number $P \geq n$, $n \leq P < 2n$; 2)M is the maximum value of the users' identifiers; 3) V_i, V_j represents two users' available channels sets and supposing $|V_i| \geq |V_j|$;

Table 2: MTTR for Multi-Radio Rendezvous

Algorithms	MTTR
RPS [25]	$O(\lceil \frac{P}{max\{m,n\}} \rceil \times (Q-G))$
EAR [20]	No explicit MTTR bound but simulations

Remarks: 1)Q is the number of total channels, P is the smallest prime number larger than Q, m, n are the number of radios each user is equipped with; 2) G is the number of common channels.

Generated Orthogonal Sequence (GOS) [9] is a pioneering work which generates a GS of length $n(n+1)$ based on a random permutation of $\{1, 2, \cdots, n\}$. However, this algorithm is limited to the situation that all these channels are available. Quorum-based Channel Hopping (QCH) [4, 5] works efficiently for synchronous users (i.e. the users start the rendezvous algorithm at the same time), which generates the GS based on the quorum system. Asynchronous QCH [3] is modified for asynchronous users (i.e. the users' start time is different), but only applicable to two available channels.

Channel Rendezvous Sequence (CRSEQ) [23] is the first one guaranteeing rendezvous in bounded time. It firstly computes the smallest prime $P > n$ and constructs the GS with P periods. For each period, $3P - 1$ elements are then generated based on the triangle number and modular operation. Jump-Stay (JS) [19] generates the GS of P periods and each period contains two *jump* frames and one *stay* frame, where each frame contains P numbers. CRSEQ guarantees rendezvous in $O(n^2)$ time slots for any two users and it works badly for symmetric users (i.e. the users have the same available channels). JS guarantees rendezvous for symmetric users in $O(n)$ time slots but in $O(n^3)$ time slots for two asymmetric users (i.e. the users may have different available channels). This result is later improved in [18]. Disjoint Relaxed Difference Set (DRDS) [13] is the first algorithm guaranteeing quick rendezvous for both symmetric and asymmetric users. It reveals the equivalence between DRDS and GS. By constructing an appropriate DRDS and transforming it into a GS, rendezvous can be guaranteed in

$O(n^2)$ time slots for asymmetric users and in $O(n)$ time slots for symmetric users.

There are mainly three LS based algorithms. Alternate Hop-and-Wait (AHW) [8] generates different sequences when each user has a distinct identifier that can be represented as a unique binary string. However, it also uses the information of all channels to construct the sequence. [12] generates different sequences on the basis of each user's identifier and the available channels. For two sets V_i, V_j (suppose $|V_i| \geq |V_j|$), it guarantees rendezvous in $O(l_i|V_i|^2)$ time slots (l_i is a constant for most situations), which doesn't rely on the n channels. [7] constructs different sequences based on edge coloring without the user's identifer and it guarantees rendezvous in $O(|V_i||V_j|\log\log n)$ time slots.

For multi-radio rendezvous algorithm, [25] first studies how to generalize the existing algorithms to use multiple radios to achieve rendezvous. It used two strategies: (1)applying an existing algorithm to each radio independently; (2) first applying an existing algorithm to generate a CH sequence, then in each time slot the user accesses m consecutive channels of the sequence, where m is the number of user's radios. In this way, the $MTTR$ is reduced to $\frac{1}{m}$ of an existing single-radio algorithm. Then [25] proposed a new algorithm. The key idea is dividing the radios into 1 dedicated radio and $m-1$ general radios. Users stay in a specific channel in the dedicated radio for a duration while hop on consecutive $m-1$ channels in the general radios. From table 2 we can see the $MTTR$ is also about $\frac{1}{m}$ of an existing single-radio algorithm. Besides, [20] also proposed a multi-interface rendezvous algorithm based on Jump-and-Stay algorithm [19], where each user is equipped with 3 radios. The main idea is sorting the available channels by the channel quality. Channel sequence is composed of jump sequence and stay sequence. Compared to JS algorithm in [19], the difference is user hops on better channels more often. And the available channel sets are divided among 3 radios. This paper only gives a performance simulation that shows the TTR is reduced compared to JS algorithm. From the above, we can see both of the algorithms didn't study whether their rendezvous performance is optimal or not.

3. MODEL AND PROBLEM DEFINITION

We consider the blind rendezvous problem where the users try to discover each other without a dedicated common control channel and the knowledge of each other. Assume the licensed spectrum is divided into a set of n non-overlapping channels as U={1,2,...,n}, where the indices are known to all the users. Supposing time is divided into slots of equal length $2t$ where t is the sufficient time to establish a link if two users access the same channel at the same time. Each user i is equipped with multiple cognitive radios to sense the licensed spectrum and it results in an available channel set $V_i \subseteq U$. Two users i and j can discover each other if they have overlapping channels, i.e., $V_i \cap V_j \neq \emptyset$. Different from extant works, assuming each user is equipped with $k(k \geq 2)$ cognitive radios, which implies the user can access k channels simultaneously in a single time slot, rather than 1 channel in previous works.

We mainly discuss whether the rendezvous time can be improved under the k-radio scenario and to what extent it could help. Different users may not have the same available channel sets as they are distributed in different positions and may have different interferences, which we call it asymmetric

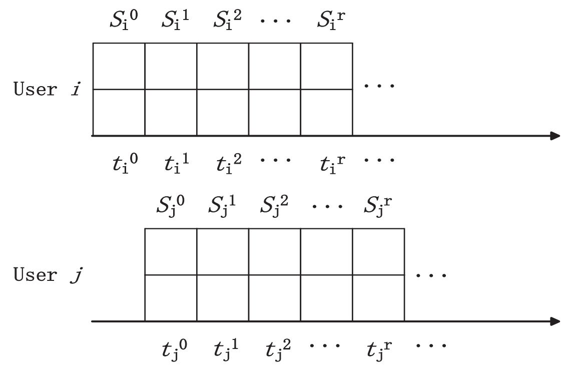

Figure 1: Channel hopping sequence and time slots with $k = 2$ in asynchronous model. User i can access 2 channels simultaneously each time slot, it hops to channel set S_i^r at t_i^r.

case. As all the users are independent, they may "wake up" at different time, then their channel-hopping sequences may have a shift. In order to be more close to reality, we focus on asymmetric and asynchronous scenarios.

As the most rendezvous algorithms do, we also adopt channel hopping sequences. In each time slot, each user i hops on k channels in V_i to attempt rendezvous with their neighbors. For example, user i obtains its own channel hopping sequence $S_i = [S_i^0, S_i^1, \cdots, S_i^x, \cdots]$, where set $S_i^x \subseteq V_i$ and $|S_i^x| = k$. When user i wakes up, it hops to channel set S_i^0 at time slot t_i^0, then hops to channel set S_i^1 at t_i^1, and so on. Now we give an example of $k = 2$, as shown in Fig. 1.

Definition 1. Given a pair of users i and j, $V_i \cap V_j \neq \emptyset$, the users i and j **rendezvous** if $S_i^x \cap S_j^y \neq \emptyset$ for some finite integers x, y, considering the different "wake-up" time.

Time to rendezvous(TTR) is an important metric to evaluate the rendezvous algorithms. In this paper, we mainly focus on asynchronous case, so our goal is to minimize the *Maximum Time to rendezvous*($MTTR$) for two users. When $MTTR$ is bounded, rendezvous is guaranteed.

4. SPECIAL CASES FOR MULTI-RADIO RENDEZVOUS

In this section, we present different distributed rendezvous algorithms for two special cases $k = 2$ and $k = O(\sqrt{n})$.

4.1 Special Case 1: $k = 2$

When each user can access 2 channels at each time slot, we extend previous single radio rendezvous algorithm to this problem and compare their performances. Firstly, we consider the global sequence (GS) based rendezvous algorithm [13]. Before presenting the algorithm, we give the definition of Disjoint Relaxed Difference Set(DRDS).

Definition 2. A set $D = \{a_1, a_2, ..., a_k\} \subseteq Z_n(Z_n = [0, n-1])$ is called a Relaxed Difference Set(RDS) if for every $d \neq 0$ (mod n), there exists at least one ordered pair (a_i, a_j) such that $a_i - a_j \equiv d$ (mod n), where a_i, $a_j \in D$.

LEMMA 1. *If D is an RDS under Z_n, then $D_k = \{(a_i+k)$ mod $n | a_i \in D\}$ is also an RDS under Z_n.*

Definition 3. A set $S = \{D_1, D_2, \cdots, D_h\}$ is called a Disjoint Relaxed Difference Set(DRDS) under Z_n if $\forall D_i \in S$, D_i is an RDS under Z_n and $\forall D_i, D_j \in S$, $i \neq j$, $D_i \cap D_j = \emptyset$.

[13] reveals the equivalence of DRDS and GS that can be used in rendezvous scheme. We present the asynchronous rendezvous algorithm based on the DRDS construction [13]. The set of all licensed channels is $U = \{1, 2, \cdots, n\}$. Find the smallest prime P such that $P \geq \frac{n}{2}$.

Algorithm 1 Global sequence based rendezvous algorithm

```
Divide the total channel set U = {1, 2, ..., n} into two parts U_1 = {1, 2, ..., n/2} and U_2 = {(n+1)/2, ..., n};
Construct DRDS of Z_m where m = 3P^2;
For each user i, apply the DRDS based Rendezvous Algorithm in [13] to channel set U_1 and U_2 simultaneously, and denote the outputs to be CH_1 = {c_1^0, c_1^1, ..., c_1^t, ...}, CH_2 = {c_2^0, c_2^2, ..., c_2^t, ...};
For any t, pick two channels from CH_1 and CH_2 respectively to form S_i^t = {c_1^t, c_2^t};
```

Theorem 1. *For two users i and j with available channel sets $V_i, V_j \subseteq U$, whenever they start Alg. 1, they can achieve rendezvous in $O(n^2)$ time slots if $V_i \bigcap V_j \neq \emptyset$.*

Proof. In Alg. 1, we divide the channels U into two parts U_1 and U_2. Then we construct a DRDS DR_1 of Z_m where $m = 3P^2, P \geq \frac{n}{2}$. Based on DR_1 we can get a global channel hopping sequence CH_1 for channel set U_1. According to Lemma 1, we can get a similar DRDS DR_2 used for channel set U_2, then get CH_2. At each time slot,the user can access two channels from CH_1 and CH_2 respectively, and from the rendezvous guarantee in [13], we can get the $MTTR = 3P^2 = O(n^2)$. $\square$

It's easy to verify that $\frac{n}{2} \leq P < n$, and thus the $MTTR \in [\frac{3n^2}{4}, 3n^2)$, which is nearly 4 times lower than single-radio CRN as Table 2. Moreover, when the number of available channels is not large, we can use the following local sequence (LS) based rendezvous algorithm to achieve a better result.

Algorithm 2 Local sequence based rendezvous algorithm

```
For each user i, the available channel set V_i = {a_1, ..., a_|V_i|}, time slot t_i^0, t_i^1, ..., t_i^r, ..., channel hopping sequence S_i = [S_i^0, S_i^1, ..., S_i^r, ...]
Choose primes p, p' from [|V_i|, 3|V_i|], p ≠ p'
For each time slot t_i^r (stage r), f = r mod p, g = r mod p'
if either f or g is not in [1, |V_i|] then
    choose a_f or a_g randomly from channel set V_i
end if
S_i^r = {a_f, a_g}
```

Theorem 2. *For arbitrary two users i, j, $V_i \cap V_j \neq \emptyset$, if they execute Alg.2 asynchronously, they can achieve rendezvous in $O(|V_i| \cdot |V_j|)$ time slots.*

Proof. Consider two users i, j, available channel sets $V_i = \{a_1, ..., a_{|V_i|}\}, V_j = \{b_1, ..., b_{|V_j|}\}$. For user j, suppose the primes are q, q'.

Suppose $V_i \cap V_j = a_x = b_y$. In synchronous model, in order to guarantee rendezvous, we need to find a time slot r such that $r \equiv x \bmod p$ and $r \equiv y \bmod q$, where $p \neq q$. According to Chinese Remainder Theorem, we can find a solution of r in no more than pq steps. As for S_i^r, S_j^r, we can judge whether they rendezvous in single time slot, so we need no more than $pq = O(|V_i| \cdot |V_j|)$ time slots to guarantee rendezvous.

In asynchronous model, we can double each stage r in the above construction, that is, we should execute $S_i^r = \{a_f, a_g\}$ for 2 time slots. Assume user i and j "wake up" in t_i^0 and t_j^0 respectively, $t_i^0 < t_j^0$, then we can find r such that the r^{th} stage of user i and the $\{r - \frac{t_j^0 - t_i^0}{2}\}^{th}$ stage of user j overlap for at least 1 time slot. Like the proof above, we can guarantee rendezvous between S_i^r and $S_j^{r - \frac{t_j^0 - t_i^0}{2}}$ in pq steps. Therefore, we need no more than $2pq = O(|V_i||V_j|)$ time slots to guarantee rendezvous. $\square$

After applying both GS and LS rendezvous algorithms to 2-channel scenario, we compare their performance as follows. Given two users i, j with available channel sets V_i, V_j, if they can access 2 channels in each time slot, for GS based rendezvous Alg. 1, MTTR is $O(3P^2)$, where P is the smallest prime larger than $\frac{n}{2}$, which is improved 4 times compared to the scenario of accessing 1 channel each time slot. For LS based rendezvous Alg. 2, MTTR is $O(|V_i| \cdot |V_j|)$, while accessing 1 channel is $O(|V_i||V_j|\text{loglog}n)$ [7], which implies the performance has been improved with $\log\log n$ factor. Generally speaking, when each user can access 2 channels at each time slot, the performance is improved and the LS based algorithm improves more than GS ones when n is larger, for example $n > 16$.

4.2 Special Case 2: $k = O(\sqrt{n})$

When the number of channels the user can access at the same time is much larger, such as $k = O(\sqrt{n})$, we show that the rendezvous can be guaranteed much more quickly. The main technique used is quorum system, which has intersection property and the size of each quorum is $O(\sqrt{n})$, we can utilize this property to provide rendezvous between two channel hopping sequences. A quorum system can be defined as follows:

Definition 4. Given a set $S = \{s_1, s_2 ... s_n\} (n \geq 1)$, a set system **QS** is a quorum system over S, if and only if

$$\forall Q_1, Q_2 \in \mathbf{QS} : Q_1 \cap Q_2 \neq \emptyset.$$

Denote the total channel set $U = \{1, 2, ..., n\}$, arrange the channels in U in a square matrix $A_{\sqrt{n} \times \sqrt{n}}$, $a_{p,q}$ = channel $(p-1) * \sqrt{n} + q$ is a element in row p and column q. Now we consider the case when $k = 2\sqrt{n} - 1$.

Theorem 3. *The distributed rendezvous algorithm based on quorum system can guarantee rendezvous asynchronously in $O(n)$ time slots.*

Proof. Consider two users i and j with $V_i \cap V_j = a_{p,q}$. According to the construction above, the channel hopping sequences we obtained are periodic with period $\sqrt{n} \times \sqrt{n} = n$, and denote them as $S_i = \{Q_{p,q}\}$, $S_j = \{Q'_{p,q}\}$. Consider a period, let $Q_p = \{Q_{p,1}, \cdots, Q_{p,\sqrt{n}}\}$, the same for $Q'_p (1 \leq p \leq \sqrt{n})$. And the channel hopping sequence $S_i = \{Q_1, Q_2, \cdots, Q_{\sqrt{n}}\} = \{Q_{1,1}, \cdots, Q_{1,\sqrt{n}}, Q_{2,1}, \cdots, Q_{2,\sqrt{n}}, \cdots,$

Algorithm 3 Quorum system based rendezvous algorithm

For user i, denote time slot $t_i^0, t_i^1, \cdots, t_i^r, \cdots$, $S_i = [S_i^0, S_i^1, \cdots, t_i^r, \cdots]$, $k = 2\sqrt{n} - 1$

Construct a quorum system **QS**=$\{Q_{p,q}, 1 \le p, q \le \sqrt{n}\}$ as follows: $Q_{p,q}$ contains all elements in row p and column q of square matrix $A_{\sqrt{n}\times\sqrt{n}}$. Then **QS** contains $\sqrt{n} \times \sqrt{n}$ quorums, each quorum has $2\sqrt{n} - 1$ elements

if there exist some channels in $Q_{p,q}$ that are not in V_i **then**

  Replace them by channels randomly selected from V_i

end if

At time slot t_i^r, hopping sequence $S_i^r = Q_{y+1,z+1}$, where $r = x \cdot n + y \cdot \sqrt{n} + z$, x, y, z are integers

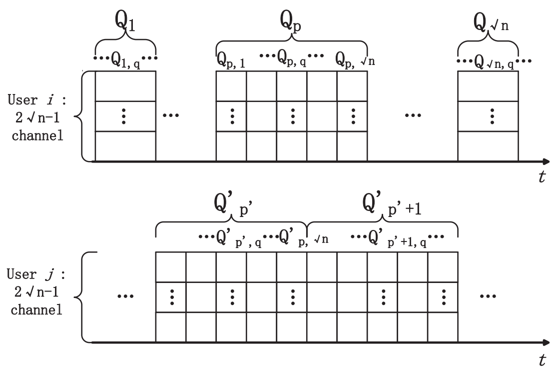

Figure 2: A period of Channel hopping sequence with $k = 2\sqrt{n} - 1$, for user i, $a_{p,q}$ appears in every element of Q_p; for user j, $a_{p,q}$ appears in $Q'_{p',q}, Q'_{p'+1,q}$.

$Q_{\sqrt{n},1}, \cdots, Q_{\sqrt{n},\sqrt{n}}\}$. In synchronous model, it is obvious that two users rendezvous in a period of $O(n)$ time slots. In asynchronous model, suppose user i wakes up earlier than user j, as shown in Fig. 2. For user i, the common channel $a_{p,q}$ will appear in consecutive $\sqrt{n}$ time slots, which correspond to channel hopping set sequence Q_p.

For user j, in the above consecutive $\sqrt{n}$ time slots, if there exist two quorum sets $Q'_{p'}$ and $Q'_{p'+1}$ that intersect with Q_p, as shown in Fig. 2. As $a_{p,q}$ appears in $Q'_{p',q} \subseteq Q'_{p'}$ and $Q'_{p'+1,q} \subseteq Q'_{p'+1}$, either $Q'_{p',q}$ or $Q'_{p'+1,q}$ intersects with Q_p, then rendezvous can be achieved. Else if there exists only one quorum set, it's obvious. So it can guarantee rendezvous asynchronously in $O(n)$ time slots. □

When $k = O(\sqrt{n})$, the $MTTR$ can be reduced largely ($O(n)$ factor compared with the best result) and we try to figure out to what extent multi-radio improves in the next section.

5. LOWER BOUND

After we consider the capability of k-radio to speed up rendezvous, we mainly focus on the limit of accelerated degree when the k value is given, actually similar to [7], we can get the following lower bound.

Theorem 4. *Any deterministic rendezvous algorithms which can access at most k channels in the single time slot require at least $\frac{|V_i||V_j|}{k^2}$ steps to guarantee rendezvous in asynchronous setting when $|V_i| + |V_j| \le n + 1$.*

Proof. Denote $a = |V_i|$ and $b = |V_j|$. Firstly select V_i uniformly at random from all subsets with a elements of $U = \{1, 2, \ldots, n\}$, then pick an element uniformly from V_i and denote it as e, and select V'_j uniformly at random from all subsets with $b - 1$ elements of $U \setminus V_i$. Let $V_j = V'_j \bigcup \{e\}$.

Let S be the deterministic rendezvous algorithm that we use, $S_i(t)$ be the channel S accesses in time slot t for user i and $S_i(x,t)$ be the number of occurrence of the element x in first t time slots, then for any t we have

$$\begin{aligned} E_x(S_i(x,t)) &= E(\sum_{y\in V_i} Pr(x = y) S_i(y,t)) \\ &= E(\frac{1}{a}\sum_{y\in V_i} S_i(y,T)) \\ &= E(\frac{1}{a}tk) = \frac{tk}{a} \end{aligned} \tag{1}$$

A similar result can be got for user j, denote $T_m = MTTR$ as the minimized rendezvous time and choose $T \gg T_m$. Because of the property of expectation, there exists an element x such that $a\frac{S_i(x,T)}{T} + b\frac{S_j(x,T_m)}{T_m} \le 2k$ which means $a\frac{S_i(x,T)}{T} \times b\frac{S_j(x,T_m)}{T_m} \le k^2$,thus

$$S_i(x,T) \times S_j(x,T_m) \le \frac{k^2}{ab} \tag{2}$$

Finally we deal with the case that user i starts at time 0,and user j starts at time $t_j \in [0, T)$. Considering the set $Q = \{(t_1, t_2) \in [0,T) \times [0, T_m) | S_i(t_1) = S_j(t_2) = x\}$. It's quite obvious that

$$\begin{aligned} |Q| &\le T \cdot T_m \cdot S_i(x,T) S_j(x,T_m) \\ &\le T \cdot T_m \frac{k^2}{ab} \end{aligned} \tag{3}$$

However since the algorithm guarantees rendezvous in time T_m, for any $t_j \in [0, T - T_m)$ there must be a corresponding point in Q which implies that

$$|Q| \ge T - T_m \tag{4}$$

Combining the two inequalities we get (using the fact that $T \gg T_m$,let T goes to infinity)

$$\begin{aligned} T \cdot T_m \frac{k^2}{ab} &\ge T - T_m \\ T_m &\ge \frac{ab}{k^2} \end{aligned} \tag{5}$$

Thus the theorem holds. □

6. GENERAL CONSTRUCTION FOR RENDEZVOUS

In order to meet the bound, we present a general construction for distributed rendezvous process.

The main idea of Alg. 4 is to generate k channels based on special case $k = 2$ in Section 4. In the first place, the available channel set V_i is divided into $\frac{k}{2}$ subsets with size $\frac{2|V_i|}{k}$ (these subsets may not be exactly the same size, we omit the details to tackle this), and then apply Alg. 2 on each subset to generate the corresponding sequences as $S_{i,j} = \{S_{i,j}^0, S_{i,j}^1, \ldots, S_{i,j}^t, \ldots\}$. It's obvious that each element $S_{i,j}^t$ has 2 channels and all these $\frac{k}{2}$ subsets can produce k channels to access, as Line 4. We can conclude that:

Algorithm 4 General Construction for Rendezvous

For user i, denote the available channel set as V_i;
Divide V_i into $k/2$ subsets as $V_{i,1}, V_{i,2}, \ldots, V_{i,\frac{k}{2}}$, where $|V_{i,j}| = \frac{2|V_i|}{k}$;
For each $V_{i,j}$, use Alg. 2 to generate sequence as $S_{i,j} = \{S_{i,j}^0, S_{i,j}^1, \ldots, S_{i,j}^t, \ldots\}$;
For any time slot t, construct $S_i^t = \bigcup_{1 \le j \le k/2} S_{i,j}^t$ and access the channels in S_i^t;

THEOREM 5. *For any two users i, j with available channel sets $V_i \bigcap V_j \neq \emptyset$, Alg. 4 guarantees rendezvous in $MTTR = O(\frac{|V_i||V_j|}{k^2})$ time slots.*

PROOF. Since $V_i \bigcap V_j \neq \emptyset$, there exist $1 \le k_1, k_2 \le \frac{k}{2}$ such that $V_{i,k_1} \bigcap V_{j,k_2} \neq \emptyset$. When they apply Alg. 2 to each subset, we can check that: for the corresponding sequences S_{i,k_1} and S_{j,k_2}, there exist corresponding x, y such that $S_{i,k_1}^x \bigcap S_{j,k_2}^y \neq \emptyset$ and they are actually in the same time slot for different wake-up time, then rendezvous can be guaranteed in $O(|V_{i,k_1}||V_{j,k_2}|) = O(\frac{|V_i||V_j|}{k^2})$ time slots according to Theorem 2. $\square$

Theorem 5 shows that we can guarantee rendezvous for any two users in $O(\frac{|V_i||V_j|}{k^2})$ time slots, which meets the lower bound in Theorem 4. Compared with the special cases in Section 4, when $k = 2$, the LS algorithm guarantees rendezvous in $O(|V_i||V_j|)$ time slots, which corroborate the analysis, and while $k = O(\sqrt{n})$, the quorum based algorithm guarantees rendezvous in $O(n)$ time slots, which meets the lower bound when both $|V_i|, |V_j| = \Omega(n)$.

7. SIMULATION

In this section, we evaluate the performance of our proposed distributed algorithms under multi-radio CRN circumstance and compare the results with the state-of-the-art single-radio rendezvous algorithms. (The algorithms we select is DRDS [13], which is a GS based rendezvous algorithm.) Since it is difficult to synchronize timers in practice, we focus on asynchronous environment.

For a multi-radio CRN, denote the total channel set $U = [1, 2, \cdots, n]$ and k to be the number of cognitive radios each user is equipped with. For two users i and j, denote the available channel sets as $V_i \subseteq U$ and $V_j \subseteq U$, where $V_i \cap V_j \neq \emptyset$. Define $\theta_i = \frac{|V_i|}{n}$, $\theta_j = \frac{|V_j|}{n}$. In each simulation, V_i and V_j are generated randomly from U satisfying some given conditions, and the wake-up time of each user is also randomly selected. MTTR is counted as the maximum time slots that it takes to achieve rendezvous since the second wake-up user begins its hopping sequence. The simulation results in the following figures are the maximal MTTR value of 10000 runs.

Since we have presented two different distributed rendezvous algorithms for the special case $k = 2$, we first evaluate the performance of GS based rendezvous algorithm and compare it with single-radio scenario (i.e. $k = 1$). Since the number of available channels for each user is an important factor, we consider the situations $|V_i|, |V_j|$ have small and large differences respectively. Fig. 3 shows the situation $\theta_i = \theta_j = 0.8$, while we set $\theta_i = 0.2$, $\theta_j = 0.8$ in Fig. 4. When n increases from 10 to 100, the $MTTR$ values both increase as shown in Fig. 3 and Fig. 4. Compared

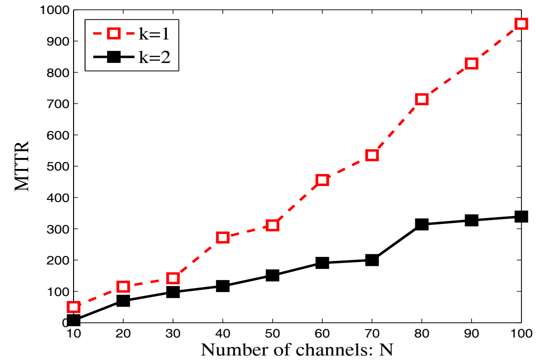

Figure 3: GS based rendezvous algorithm, $\theta_i = \theta_j = 0.8$, MTTR as n increases

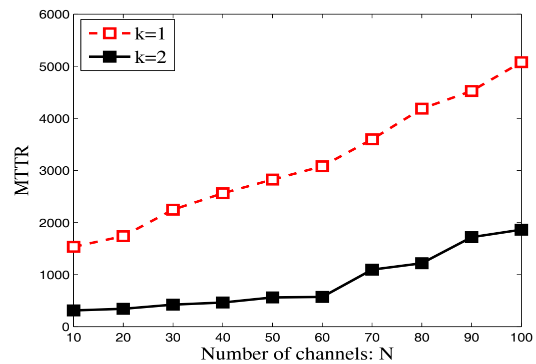

Figure 4: GS based rendezvous algorithm, $\theta_i = 0.2, \theta_j = 0.8$, MTTR as n increases

with single-radio CRN, our 2-radio CRN has better performance, which is in accordance with our theoretical analysis. As depicted, our algorithm is nearly 4 times quicker than the single-radio CRN, which verifies the analysis of Alg. 1. When θ_i and θ_j have a great difference in Fig. 4, the $MTTR$ value increase enormously compared with $\theta_i = \theta_j = 0.8$ (Fig. 3), as the number of common channels decreases.

Although the GS based rendezvous algorithm has a good performance, the LS based rendezvous algorithm (Alg. 2) achieves a better result. Like GS algorithm, we also consider the two cases of $|V_i|$ and $|V_j|$. Fig. 5 shows the situation $\theta_i = \theta_j = 0.8$, while Fig. 6 shows the result when $\theta_i = 0.2$, $\theta_j = 0.8$. Similar with Alg. 1, both Fig. 5 and Fig. 6 show that the $MTTR$ values increase as n increases from 10 to 100. However, as shown in Fig. 5, the LS based rendezvous algorithm reduces the $MTTR$ largely compared with Fig. 3 (for example, when $n = 100$, Alg. 1 shows the $MTTR$ value is about 350 time slots, while Alg. 2 has shorter $MTTR$ about 270 time slots.) This implies that Alg. 2 can improve the performance of rendezvous in 2-radio CRN when n is large. This is also verified from Fig. 6 when $|V_i|, |V_j|$ have great difference. All these results corroborate the analysis and the comparison we make in Section 4.

Considering another special case $k = O(\sqrt{n})$. Since Alg. 3 works efficiently when $k = 2\sqrt{n} - 1$, we verify the performance of the result when $\sqrt{n}$ increases from 10 to 20 as

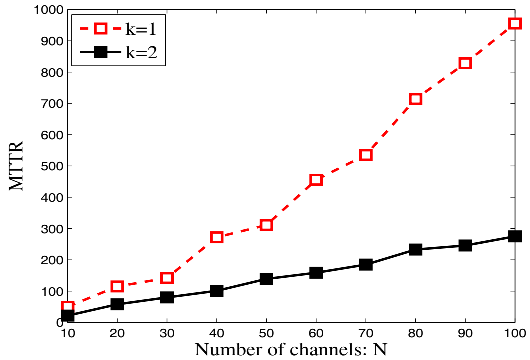

Figure 5: LS based rendezvous algorithm, $\theta_i = \theta_j = 0.8$, MTTR as n increases

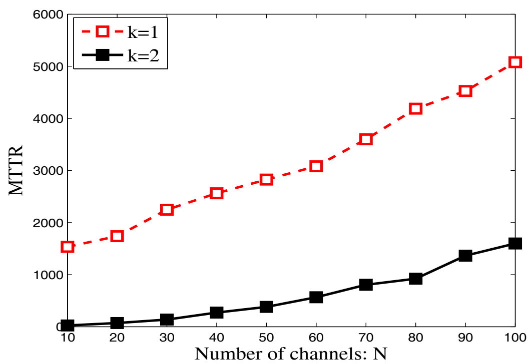

Figure 6: LS based rendezvous algorithm, $\theta_i = 0.2, \theta_j = 0.8$, MTTR as n increases

Fig. 7. For this situation, we don't compare this algorithm with single-radio ones, since the $MTTR$ value is about $O(n^2)$, which is very large. (For example, when $n = 100$, the $MTTR$ value for the multi-radio CRN can achieve rendezvous in 100 time slots, while the previous result for single-radio CRN has the smallest $MTTR$ value about 1000 time slots.) As shown in Fig. 7, the $MTTR$ values increases when $\sqrt{n}$ increases and it's almost bounded as $O(n)$ time slots as analyzed in Theorem 3.

Moreover, we evaluate the general construction in Alg. 4 for any arbitrary k. For this simulation, we fix $n = 100$ and $\theta_i = \theta_j = 0.8$. As shown in Fig. 8, the $MTTR$ value decreases when k increases from 1 to 20. When $k = 1$, which is the previous single-radio CRN, the result shows that the $MTTR$ value is very large and it decreases largely when k is large. This result corroborates the analysis of Theorem 5 where two users can rendezvous in $O(\frac{|V_i||V_j|}{k^2})$ time slots.

In a word, our simulation results show that rendezvous can be improved in multi-radio CRN. For $k = 2$, both GS and LS based algorithms can improve the state-of-the-art result. For $k = O(\sqrt{n})$, the quorum system method (Alg. 3) reduces the $MTTR$ value largely. For more general cases, our proposed algorithm guarantees rendezvous in bounded time with good performance and the $MTTR$ value decreases when the number of cognitive radio k increases.

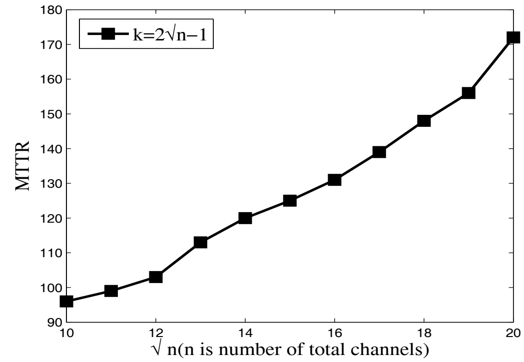

Figure 7: quorum based rendezvous algorithm, MTTR as n increases

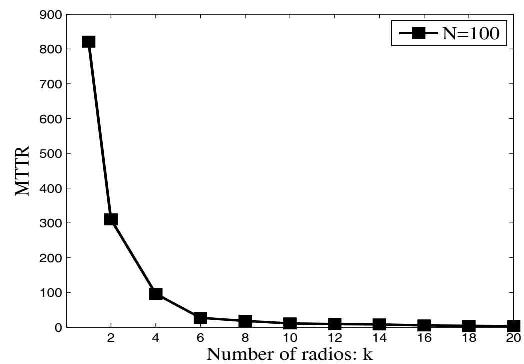

Figure 8: $n = 100$, $\theta_i = \theta_j = 0.8$, MTTR as k increases

8. CONCLUSIONS

In this paper, we study the rendezvous problem in multi-radio Cognitive Radio Networks (CRNs), which is a fundamental process in constructing a CRN. First of all, we show the improvement of $MTTR$ in k-radio scenario by considering two special cases, where $k \geq 2$ and $k = O(\sqrt{n})$. When $k = 2$, we design both global sequence (GS) and local sequence (LS) based distributed rendezvous algorithms, where GS algorithm improves the time to rendezvous only by a constant factor, whereas the LS algorithm improves by $\log\log n$ factor, where n is the number of all channels. For another special case $k = O(\sqrt{n})$, the $MTTR$ value can be reduced largely when the quorum system method is used to guarantee rendezvous. In order to figure out the limit of the improvement, we show a lower bound of $MTTR$ as $\Omega(\frac{|V_i||V_j|}{k^2})$, where V_i, V_j represents two users' available channel sets. Moreover, we present the method for general construction to rendezvous based on the LS algorithm, which meets the lower bound. From these aspects, the rendezvous time for CRN could be improved by using multiple radios for each user, and the improvement can also be bounded. Besides, compared to the existing multi-radio rendezvous algorithm, our rendezvous algorithm improves time to rendezvous by $O(k)$ factor and is optimal. Finally, we conduct extensive simulations to compare these algorithms and the results corroborate our theoretical analyses.

9. ACKNOWLEDGMENTS

This work was supported in part by the National Basic Research Program of China Grant 2011CBA00300, 2011CBA00301, the National Natural Science Foundation of China Grant 61103186, 61033001, 61361136003.

10. REFERENCES

[1] M. Ahmadi, Y. Zhuang, and J. Pan. Distributed robust channel assignment for multi-radio cognitive radio networks. In *Vehicular Technology Conference (VTC Fall), 2012 IEEE*, pages 1–5. IEEE, 2012.

[2] 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*, 50(13):2127–2159, 2006.

[3] K. Bian and J.-M. Park. Maximizing rendezvous diversity in rendezvous protocols for decentralized cognitive radio networks. *Mobile Computing, IEEE Transactions on*, 12(7):1294–1307, 2013.

[4] K. Bian, J.-M. Park, and R. Chen. A quorum-based framework for establishing control channels in dynamic spectrum access networks. In *Proceedings of the 15th annual international conference on Mobile computing and networking*, pages 25–36. ACM, 2009.

[5] K. Bian, J.-M. Park, and R. Chen. Control channel establishment in cognitive radio networks using channel hopping. *Selected Areas in Communications, IEEE Journal on*, 29(4):689–703, 2011.

[6] Z. Cai, S. Ji, J. He, and A. G. Bourgeois. Optimal distributed data collection for asynchronous cognitive radio networks. In *Distributed Computing Systems (ICDCS)*, pages 245–254. IEEE, 2012.

[7] S. Chen, A. Russell, A. Samanta, and R. Sundaram. Deterministic blind rendezvous in cognitive radio networks. *arXiv preprint arXiv:1401.7313*, 2014.

[8] I. Chuang, H.-Y. Wu, K.-R. Lee, and Y.-H. Kuo. Alternate hop-and-wait channel rendezvous method for cognitive radio networks. In *INFOCOM, 2013 Proceedings IEEE*, pages 746–754. IEEE, 2013.

[9] L. A. DaSilva and I. Guerreiro. Sequence-based rendezvous for dynamic spectrum access. In *New Frontiers in Dynamic Spectrum Access Networks. 3rd IEEE Symposium on*, pages 1–7. IEEE, 2008.

[10] R. Draves, J. Padhye, and B. Zill. Routing in multi-radio, multi-hop wireless mesh networks. In *Proceedings of the 10th annual international conference on Mobile computing and networking*, pages 114–128. ACM, 2004.

[11] A. G. Fragkiadakis, E. Z. Tragos, and I. G. Askoxylakis. A survey on security threats and detection techniques in cognitive radio networks. *Communications Surveys & Tutorials, IEEE*, 15(1):428–445, 2013.

[12] Z. Gu, Q.-S. Hua, and W. Dai. Local sequence based rendezvous algorithms for cognitive radio networks. In *Sensor, Mesh and Ad Hoc Communications and Networks (SECON)*, pages 371–379. IEEE, 2014.

[13] Z. Gu, Q.-S. Hua, Y. Wang, and F. Lau. Nearly optimal asynchronous blind rendezvous algorithm for cognitive radio networks. In *Sensor, Mesh and Ad Hoc Communications and Networks (SECON)*, pages 371–379. IEEE, 2013.

[14] X. Huang, D. Lu, P. Li, and Y. Fang. Coolest path: spectrum mobility aware routing metrics in cognitive ad hoc networks. In *Distributed Computing Systems (ICDCS), 2011 31st International Conference on*, pages 182–191. IEEE, 2011.

[15] S. Ji, R. Beyah, and Z. Cai. Minimum-latency broadcast scheduling for cognitive radio networks. In *Sensor, Mesh and Ad Hoc Communications and Networks (SECON)*, pages 389–397. IEEE, 2013.

[16] W. Kim, A. J. Kassler, M. Di Felice, and M. Gerla. Urban-x: towards distributed channel assignment in cognitive multi-radio mesh networks. In *Wireless Days (WD), 2010 IFIP*, pages 1–5. IEEE, 2010.

[17] Y. R. Kondareddy, P. Agrawal, and K. Sivalingam. Cognitive radio network setup without a common control channel. In *Military Communications Conference, 2008. MILCOM 2008. IEEE*, pages 1–6. IEEE, 2008.

[18] Z. Lin, H. Liu, X. Chu, and Y. Leung. Enhanced jump-stay rendezvous algorithm for cognitive radio networks. 2013.

[19] H. Liu, Z. Lin, X. Chu, and Y.-W. Leung. Jump-stay rendezvous algorithm for cognitive radio networks. *Parallel and Distributed Systems, IEEE Transactions on*, 23(10):1867–1881, 2012.

[20] R. Paul, Y. Z. Jembre, and Y.-J. Choi. Multi-interface rendezvous in self-organizing cognitive radio networks. In *Dynamic Spectrum Access Networks (DYSPAN), 2014 IEEE International Symposium on*, pages 531–540. IEEE, 2014.

[21] J. Pérez-Romero, O. Salient, R. Agustí, and L. Giupponi. A novel on-demand cognitive pilot channel enabling dynamic spectrum allocation. In *New Frontiers in Dynamic Spectrum Access Networks. 2nd IEEE International Symposium on*, pages 46–54. IEEE, 2007.

[22] K. Ramachandran, I. Sheriff, E. M. Belding, and K. C. Almeroth. A multi-radio 802.11 mesh network architecture. *Mobile Networks and Applications*, 13(1-2):132–146, 2008.

[23] J. Shin, D. Yang, and C. Kim. A channel rendezvous scheme for cognitive radio networks. *Communications Letters, IEEE*, 14(10):954–956, 2010.

[24] Y. Song, J. Xie, and X. Wang. A novel unified analytical model for broadcast protocols in multi-hop cognitive radio ad hoc networks. 2013.

[25] L. Yu, H. Liu, Y.-W. Leung, X. Chu, and Z. Lin. Multiple radios for effective rendezvous in cognitive radio networks. In *Communications (ICC), 2013 IEEE International Conference on*, pages 2857–2862. IEEE, 2013.