Nearly Optimal Asynchronous Blind Rendezvous Algorithm for Cognitive Radio Networks

Zhaoquan Gu*, Qiang-Sheng Hua*, Yuexuan Wang* and Francis C.M. Lau†
*Institute for Theoretical Computer Science, Institute for Interdisciplinary Information Sciences,
Tsinghua University, Beijing, 100084, P.R. China
†Department of Computer Science, The University of Hong Kong, Hong Kong, P.R. China

***Abstract*—Rendezvous is a fundamental process in Cognitive Radio Networks, through which a user establishes a link to communicate with a neighbor on a common channel. Most previous solutions use either a central controller or a Common Control Channel (CCC) to simplify the problem, which are inflexible and vulnerable to faults and attacks. Some blind rendezvous algorithms have been proposed that rely on no centralization. Channel Hopping (CH) is a representative technique used in blind rendezvous, with which each user hops among the available channels according to a pre-defined sequence. However, no existing algorithms can work efficiently for both symmetric (both parties have the same set of channels) and asymmetric users. In this paper, we introduce a new notion called Disjoint Relaxed Difference Set (DRDS) and present a linear time constant approximation algorithm for its construction. Then based on the DRDS, we propose a distributed asynchronous algorithm that can achieve and guarantee fast rendezvous for both symmetric and asymmetric users. We also derive a lower bound for any algorithm using the CH technique. This lower bound shows that our proposed DRDS based distributed rendezvous algorithm is nearly optimal. Extensive simulation results corroborate our theoretical analysis.**

***Index Terms*—Rendezvous, Time to Rendezvous, Disjoint Relaxed Difference Set, Cognitive Radio Networks**

I. Introduction

The wireless spectrum has become a scarce resource because of the burgeoning development and deployment of wireless technologies. The scarceness however occurs mainly in the unlicensed section of the spectrum where there is often overcrowding due to the rapidly increasing demand for wireless services, whereas the utilization of the licensed spectrum is very low. Dynamic spectrum access (DSA) through the use of cognitive radios has been proposed to alleviate the spectrum scarcity problem in wireless communications. A cognitive radio network (CRN) consists of secondary users (SUs) (i.e. unlicensed users) sharing the licensed spectrum with the licensed primary users (PUs). Each SU is equipped with cognitive radios that can sense the spectrum for the opportunity to access the licensed spectrum when it is left unused by the PUs.[1]

Because there can be many wireless channels made available to the SUs by the PUs, the first act of a communication task in a CRN is to establish a common link on a common channel between two users, which is referred to as the process of *rendezvous*. Since the spectrum usage of PUs varies temporally and geographically, the available channels each SU can access may be different. If two users have exactly the same set of available channels, we have a *symmetric* situation; otherwise *asymmetric*. In real scenarios, two users who need to communicate by default cannot tell which situation they are in, and the algorithm for rendezvous must work efficiently for either situation. The time for them to establish a link is called the *time to rendezvous* (TTR). In this paper, we assume all users are asynchronous which means that they can start a rendezvous at any time, and two users may have different start times, and our goal is to minimize the *Maximum Time to Rendezvous* ($MTTR$) for two such asynchronous users.

Most previous works on rendezvous use either a central controller [14], [16] or a common control channel (CCC) [8], [15] to simplify the process. They therefore may suffer from several problems: the central controller or CCC may become too overloaded with the increase of users, thus forming a bottleneck in the CRN; a system with centralization is not flexible; and the CCC is vulnerable to adversary attacks. Therefore, *blind rendezvous algorithms* have been proposed, which depend on no central controller or dedicated CCC. The majority of these algorithms are based on Channel Hopping (CH) techniques [2], [3], [11], [18], [20], whereby each user would hop from channel to channel based on a pre-defined sequence.

Nevertheless, no existing algorithms can work efficiently for both the symmetric and asymmetric scenarios (see Table I). Jump-Stay [11] is a state-of-the-art algorithm guaranteeing efficient rendezvous for the symmetric case with two users, but their $MTTR$ value is unacceptable for asymmetric users. Channel Rendezvous Sequence (CRSEQ) [18] guarantees efficient rendezvous for two asymmetric users, but their $MTTR$ value is much larger than Jump-Stay for symmetric users. Deterministic Rendezvous Sequence (DRSEQ) [21] can guarantee efficient rendezvous for symmetric users, while it is not directly applicable to asymmetric users.

In this paper, we introduce a new notion called Disjoint Relaxed Difference Set (DRDS) and present a linear time constant approximation algorithm for its construction. Then by combining what we call a Common Channel Hopping Sequence (CCHS), we propose a distributed asynchronous algorithm based on the DRDS construction. Our algorithm can

[0]This version corrected a small error in Alg. 2 (Line 4) of the published version. Thanks to Zhenhua Han for pointing out the error!

[1]Unless specified otherwise, "users" in the paper refer to SUs.

TABLE I
MTTR COMPARISON FOR ALL ALGORITHMS

Algorithms	Symmetric	Asymmetric
Jump-Stay [11]	$3P = O(N)$	$3NP(P-G) = O(N^3)$
CRSEQ [18]	$P(3P-1) = O(N^2)$	$P(3P-1) = O(N^2)$
DRSEQ [21]	$2N+1 = O(N)$	–
GOS [5]	$N(N+1) = O(N^2)$	–
Our algorithm	$3P = O(N)$	$3P^2 + 2P = O(N^2)$

Remarks: 1) "–" means DRSEQ and GOS are inapplicable to asymmetric users; 2) P is the smallest prime number $\geq N$, $P = O(N)$; G is the number of common channels.

guarantee rendezvous in $MTTR = O(N)$ time slots for symmetric users and $MTTR = O(N^2)$ time slots for asymmetric users, where N is the number of all non-overlapping licensed channels. We also reveal the equivalence of DRDS and CCHS. Based on the equivalence, we derive a lower bound for any algorithm using the Channel Hopping technique. This lower bound shows that our DRDS based asynchronous rendezvous algorithm is nearly optimal. Extensive simulations show that our algorithm can achieve better performance than all existing algorithms.

The remainder of the paper is organized as follows. The next section highlights the related work. Preliminaries are provided in Section III. We introduce Disjoint Relaxed Difference Set and present its construction in Section IV. Section V gives an asynchronous algorithm and its performance analysis. We establish the equivalence of DRDS and CCHS and derive the lower bound in Section VI. Section VII presents our simulation experiments. We conclude the paper in Section VIII.

II. RELATED WORK

The existing rendezvous algorithms can be classified into two categories: centralized and decentralized.

A. Centralized Algorithms

Centralized Algorithms assume there exists a central controller that each user can access directly during the rendezvous process. Most centralized algorithms use a pre-selected Common Control Channel (CCC) [16] which is accessible to all users. This simplifies the rendezvous process since the communication between the users is much easier. However, there are several drawbacks: the pre-selected CCC can easily get congested with the increase of users, and it is vulnerable to adversary attacks; the cost to maintain CCC is high and it is impractical in real CRNs. The other class of centralized algorithms work with no CCC. [14] proposed an exhaustive search based protocol to achieve rendezvous.

Failure or overloading of the central controller makes it a weak spot or bottleneck of the CRN, and thus centralized algorithms generally are not practical.

B. Decentralized Algorithms

Decentralized algorithms without a central controller have been proposed to avoid the drawbacks of centralized algorithms. Similarly, there are two classes depending on whether CCC is required.

Some decentralized algorithms assumes the existence of a global CCC that is known to all users [8], [15]. However, a global CCC might not be feasible in practice because the availability of the CCC depends on all PUs' usage. [10], [22] focus on establishing local CCCs through which each user can contact with their neighbors. These algorithms however incur substantial overhead in establishing and maintaining local CCCs.

Since CCC has its inherent limitations, some researchers turned to decentralized algorithms without CCC, which are called *blind rendezvous algorithms*. The main technique used is Channel Hopping (CH). Each user can hop among the available channels according to a pre-defined hopping sequence and the rendezvous is achieved when two users happen to hop onto the same channel at the same time.

Generated Orthogonal Sequence (GOS) [5] is a pioneering work which generates an $N(N+1)$ sequence based on a random permutation of $\{1, 2, \cdots, N\}$ where N is the number of all channels. Quorum-based Channel Hopping (QCH) [2], [3] is proposed for synchronized users while Asynchronous QCH (A-QCH) works for asynchronous scenarios, but it is only applicable to two channels. Deterministic Rendezvous Sequence (DRSEQ) of length $2N+1$ is proposed in [21] and it can be expressed as $\{1, 2, \cdots, N, null, N, N-1, \cdots, 1\}$. [20] proposes the Modular Clock (MC) algorithm and the Modified Modular Clock (MMC) algorithm to achieve rendezvous. The basic idea of MC and MMC is that each user chooses a proper prime number $P > N$ and picks a rate value $r < P$ randomly, and generates the CH sequence via some modular operations. GOS, DRSEQ and MC can work only when two users are symmetric, and MMC is suitable for the asymmetric case but it cannot guarantee rendezvous in finite time.

Channel Rendezvous Sequence (CRSEQ) [18] is the first algorithm that guarantees rendezvous when two users are asymmetric. The basic idea is that each user picks the smallest prime number $P \geq N$ and generates the CRSEQ which consists of P periods, each period containing $3P-1$ numbers based on the triangle number and modular operations.

Jump-Stay [11] is another work guaranteeing rendezvous for both symmetric users and asymmetric users. It is built on top of MC: each user finds the smallest prime $P > N$ and generates the CH sequence as the *jump* step in MC. In order to overcome the drawback of MC, the *stay* step is introduced where the user stays on a particular channel for P time slots.

However, neither CRSEQ nor Jump-Stay can work efficiently under both symmetric and asymmetric scenarios. CRSEQ guarantees rendezvous in $P(3P-1) = O(N^2)$ time slots for both scenarios, while Jump-Stay guarantees $3P = O(N)$ time slots for symmetric users and $3NP(P-G) + 3P = O(N^3)$ time slots for asymmetric users (G is the number of common channels).

There are some algorithms not based on the CH technique. [1], [14] elect a leader to discover the neighbors and to assist the users' to achieve rendezvous. Cyclostationary signatures

are utilized in [19]. A grid based method is presented in [17] to guarantee rendezvous with high probability.

III. Preliminaries

A. System Model

In this paper, we focus on the rendezvous process between two users and it can be extended to multi-users just like the method in [6], [11]. Assume the licensed spectrum is divided into $N(N \geq 1)$ non-overlapping channels that are labeled $1, 2, \cdots, N$ uniquely and the labeling is known to all users. Each user is equipped with cognitive radios to sense the spectrum for its set of available channels, where a channel is said to be *available* if the user can communicate through the channel with no interference to the PUs.

Assume time is divided into slots of equal length and the length of each time slot is sufficient for establishing a link if two users choose the same channel for communication. (In practice, rendezvous involves a more detailed process comprising beaconing, handshaking, etc. In this paper, we focus on the design of CH algorithms and assume the rendezvous is successful if they choose the same channel.) All users are physically dispersed and the wake-up time of each user may be different. Therefore, the rendezvous algorithm needs to be applicable to asynchronous users. When two users begin the rendezvous process, they will end with a common channel if their rendezvous succeeds. We refer to the time used by the second user (who starts the process later than the other one), from start time to termination, as the *Time to Rendezvous (TTR)*. In this paper, we use the *Maximum Time to Rendezvous ($MTTR$)* to evaluate the algorithms, which denotes the longest time for the second user to achieve rendezvous.

B. Problem Definition

Consider two users, Alice and Bob, each equipped with cognitive radios to sense the spectrum. Suppose the available channel sets for them are $C_A, C_B \subseteq C$ respectively, where $C = \{1, 2, \cdots, N\}$ denotes the set of all channels. Both Alice and Bob know the label of each channel, and thus they can devise a strategy based on the available channel sets.

For example, $C = \{1, 2, 3, 4, 5, 6\}, C_A = \{1, 2, 5\}, C_B = \{3, 4, 5, 6\}$, and Alice accesses the channels at different time slots as follows

$$f_{C_A}(t) = \begin{cases} 1 & \text{When } t \equiv 0 \mod 3 \\ 2 & \text{When } t \equiv 1 \mod 3 \\ 5 & \text{When } t \equiv 2 \mod 3 \end{cases}$$

Bob accesses the channels by repeating the sequence:

$$\{3, 3, 3, 3, 4, 4, 4, 4, 5, 5, 5, 5, 6, 6, 6, 6\}$$

If Alice and Bob start at the same time, $TTR = 9$ when they both access channel 5 as shown in Fig. 1. However, $MTTR = 15$ when Bob starts the process at time slot 0 and Alice starts at time slot 10, as in Fig. 2.

In the example, $C_A \neq C_B$ and this is an asymmetric case (the case would be symmetric if $C_A = C_B$). Both scenarios should be considered in order to guarantee rendezvous. Alice and Bob use different strategies in the example; however, it is impractical to design different strategies for different users, especially when there are many of them. We therefore assume each user starts the rendezvous process with the same algorithm.

Time	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14
Alice	1	2	5	1	2	5	1	2	5	1	2	5	1	2	5
Bob	3	3	3	3	4	4	4	4	5	5	5	5	6	6	6

Fig. 1. Alice and Bob start at time 0; rendezvous is achieved at time 8.

Time	...	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
Alice			1	2	5	1	2	5	1	2	5	1	2	5	1	2	5
Bob	...	5	5	5	6	6	6	6	3	3	3	3	4	4	4	4	5

Fig. 2. Bob starts at time 0 and Alice starts at time 10; rendezvous is achieved at time 24.

We formulate the *Blind Rendezvous Problem* as follows:

Problem 1: Given a channel set $C' \subseteq C$, design a channel access strategy for different time slots $f_{C'}(t) \in C'$ such that: $\forall C_A, C_B \subseteq C, \forall \delta_t$:

$$\exists T \text{ s.t. } f_{C_A}(T + \delta_t) = f_{C_B}(T)$$

The $MTTR$ value of strategy f is $MTTR_f = \max_{\forall \delta_t} T$. Our goal is to find a strategy g minimizing the $MTTR$ value among all strategies: $MTTR_g = min_{\forall f} MTTR_f$.

Remark 3.1: If Alice starts later than Bob, $\delta_t < 0$ in the description of Problem 1.

IV. Disjoint Relaxed Difference Sets

In this section, we first introduce Disjoint Relaxed Difference Set (DRDS) and show the hardness of finding the maximum DRDS. And then we present a linear time constant approximation algorithm to construct a DRDS.

A. Disjoint Relaxed Difference Set

Relaxed difference set is an efficient tool to construct cyclic quorum system [9], [12]. We give some definitions and examples here.

Definition 4.1: A set $D = \{a_1, a_2, \cdots, a_k\} \subseteq Z_n$ (the set of all nonnegative integers less than n) is called a Relaxed Difference Set (RDS) if for every $d \neq 0 \pmod n$, there exists at least one ordered pair (a_i, a_j) such that $a_i - a_j \equiv d \pmod n$, where $a_i, a_j \in D$.

RDS is a variation of (n, k, λ)-Difference Set [4], [12] where exactly λ ordered pairs (a_i, a_j) satisfying $a_i - a_j \equiv d \pmod n$ are required. Given any n, it is proved that any difference set D must have a cardinality $|D| \geq \sqrt{n}$ [12]. The minimal D whose size approximates the lower bound can be found when $n = k^2 + k + 1$ and k is a prime power. Such a difference set is called a *Singer* Difference Set (SDS) [4]. For

example, $D = \{1, 2, 4\}$ is both an SDS and an RDS under Z_7 while it is an RDS, but not an SDS under Z_6.

Lemma 4.1: 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.

The proof of the lemma is straightforward, which we omit.

Definition 4.2: 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 $\forall D_i, D_j \in S, i \neq j$, $D_i \bigcap D_j = \emptyset$.

For example, $S = \{\{1, 2, 4\}, \{0, 3, 5\}\}$ is a DRDS under Z_6. Such a DRDS can be used to design rendezvous algorithm and we will give the details later.

B. Hardness of Finding Maximum DRDS

For a given n, there are many DRDSs under Z_n. Define Maximum DRDS S_n to be the set with the largest cardinality. However, it is hard to find the maximum DRDS.

Lemma 4.2: Given n, the cardinality of the maximum DRDS under Z_n is bounded by $|S_n| \leq \sqrt{n}$.

This lemma is derived easily from the fact any RDS D should have cardinality $|D| \geq \sqrt{n}$ [12]. However, it is hard to find the tight bound for any give n.

For any set $\mathcal{D} = \{D_0, D_1, \cdots, D_h\}$ where D_i is an RDS under Z_n and $h \geq \sqrt{n}$, it is hard to compute the maximum DRDS from $\mathcal{D}$ because it can be reduced from the Set Packing Problem[2] which is NP-complete [13]. When each set $D_i \in \mathcal{D}$ satisfies $|D_i| \geq \sqrt{n}$, it is equivalent to Maximum $\sqrt{n}$-Set Packing which cannot be efficiently approximated within a factor of $\Omega(\frac{\sqrt{n}}{\ln \sqrt{n}})$ [7].

We compute all RDSs with cardinality in $[\sqrt{n}, \sqrt{3n}]$ and use exhaustive search to find the maximum DRDS when $n = 2, 3, \cdots, 50$. The relationship between n and $|S_n|$ is listed in Table II.

TABLE II
RELATIONSHIP BETWEEN n AND $|S_n|$ WHEN $2 \leq n \leq 50$

The number: n	**Maximum DRDS:** $\lvert S_n \rvert$
$2 \leq n \leq 5$	1
$6 \leq n \leq 14$	2
$15 \leq n \leq 23$	3
$24 \leq n \leq 30, 32 \leq n \leq 34$	4
$n = 31, 35 \leq n \leq 47$	5
$48 \leq n \leq 50$	6

C. DRDS Construction

In this section, we present a linear time algorithm to construct a DRDS under Z_n where $n = 3P^2, P \geq 3$ and P is a prime number.

Alg. 1 constructs a DRDS $S = \{D_0, D_1, \cdots, D_{P-1}\}$. We explain how each RDS D_i is constructed:

When $n = 3P^2$, divide Z_n into P disjoint subsets $Z_n = U_0 \bigcup U_1 \bigcup \cdots \bigcup U_{P-1}$, where $U_j = Z_{3P(j+1)} \setminus Z_{3P \cdot j}$. Let

[2]Given a finite set U and a list of subsets of U, the problem asks if some k subsets in the list are pairwise disjoint.

Algorithm 1 DRDS Construction of Z_n when $n = 3P^2$

```
1:  S := ∅;
2:  for i = 0 to P − 1 do
3:      D_i := (Z_(3Pi+P) \ Z_3Pi);
4:      for j = 0 to P − 1 do
5:          q_j := j^2, p_ij := ((i−q_j)(P+1))/2 mod P;
6:          t_j0 := 3Pj + P + p_ij;
7:          t_j1 := 3Pj + 2P + p_ij;
8:          D_i := D_i ⋃ {t_j0, t_j1};
9:      end for
10:     S := S ⋃ {D_i};
11: end for
```

$D_i = T_{i0} \bigcup T_{i1} \bigcup \cdots \bigcup T_{i,P-1}$ where $T_{ij} \subseteq U_j$. For each U_j, let $q_j = j^2$ and $p_{ij} = \frac{(i-q_j)(P+1)}{2} \mod P$. Choose the $(P + p_{ij})$-th and $(2P + p_{ij})$-th number of U_j to compose T_{ij}. They are t_{j0} and t_{j1} in Lines 6,7. Then T_{ij} is constructed:

$$T_{ij} = \begin{cases} \{t_{j0}, t_{j1}\} & \text{when } j \neq i \\ \{t_{j0}, t_{j1}\} \bigcup (Z_{(3Pi+P)} \setminus Z_{3Pi}) & \text{when } j = i \end{cases}$$

The intuitive idea of constructing D_i is: in order to have some ordered pairs (a_j, a_k) satisfying $a_j - a_k \equiv d \pmod n$ when d is small from 1 to P, choose the first P numbers in set U_i, i.e. $Z_{(3Pi+P)} \setminus Z_{3Pi}$. When d becomes much larger, choose two numbers from each set U_j at some appropriate positions according to the modular operations in Line 5. Below is an example of $n = 27$ and three RDSs are constructed:

$$\begin{aligned} D_0 &= \{0, 1, 2, 3, 6, 13, 16, 22, 25\}; \\ D_1 &= \{5, 8, 9, 10, 11, 12, 15, 21, 24\}; \\ D_2 &= \{4, 7, 14, 17, 18, 19, 20, 23, 26\}. \end{aligned}$$

It is easy to verify that D_0, D_1, D_2 can compose a DRDS. We prove Alg. 1 can indeed construct a DRDS formally.

Theorem 1: The set $S = \{D_0, D_1, \cdots, D_{P-1}\}$ constructed in Alg. 1 is a DRDS.

We give an important lemma before the proof.

Lemma 4.3: Each set D_i constructed in Alg. 1 is a RDS.

Proof: From the definition, we need to show for any $d \neq 0 \pmod n$, there exists at least one ordered pair (a_j, a_k) such that $a_j - a_k \equiv d \pmod n$. Consider the following four cases:

- When $0 < d < P$: From Alg. 1, Line 3, P consecutive numbers are chosen, i.e. $3Pi, 3Pi+1, \cdots, 3Pi+P-1 \in D_i$; thus we can find such a pair $(3Pi + d, 3Pi)$ to meet the requirement.
- When $P \leq d < 3P^2$ and $0 \leq d \pmod{3P} < P$: Assume $d = 3Pj_1 + b_1$, $0 < j_1 < P, 0 \leq b_1 < P$; we want to find one pair (a_j, a_k) such that

$$\begin{aligned} a_j &= 3Pj_2 + b_2 \mod n \\ a_k &= 3Pj_3 + b_3 \mod n \end{aligned}$$

  where $P \leq b_3 < 2P$. $a_j - a_k \equiv d \pmod n$ implies $3Pj_2 + b_2 \equiv 3P(j_1 + j_3) + b_1 + b_3 \pmod n$, and thus

$P \le b_2 < 3P$ and both b_2, b_3 satisfy the equality from Lines 6, 7 of Alg. 1. Therefore:

$$\begin{aligned} b_2 &\equiv \frac{(i-j_2^2)(P+1)}{2} \mod P \\ b_3 &\equiv \frac{(i-j_3^2)(P+1)}{2} \mod P \end{aligned}$$

Thus we have:

$$\begin{cases} j_2 \equiv (j_1 + j_3) & \mod P \\ \frac{(i-j_2^2)(P+1)}{2} \equiv \frac{(i-j_3^2)(P+1)}{2} + b_1 & \mod P \end{cases}$$

Combining them:

$$2j_1 j_3 \equiv -(2b_1 + j_1^2) \mod P \quad (1)$$

Since P is a prime number and j_1, b_1 are constant values when d is given, j_3 has one unique solution in Z_P [4] which we write as j^*. Plugging j^* into the above equalities, we can find out the values of a_j and a_k.
For example $P = 3$, $n = 27$, when $d = 11 = 3Pj_1 + b_1$, then $j_1 = 1, b_1 = 2$. Consider set D_1 and plug j_1, b_1 into Equation (1) as:

$$2j_3 \equiv -5 \equiv 1 \mod 3$$

So $j_3 = 2$ and thus $j_2 = 0$, $b_3 \equiv 0 \pmod 3$. Since $3 \le b_3 < 6$, $b_3 = 3$ and then $b_2 = 5$. Therefore, $a_j = 3Pj_2 + b_2 = 5$ and $a_k = 3Pj_3 + b_3 = 21$. When $d = 11$, we can find such a pair $(5, 21)$ from D_1 to meet the requirement.

- When $P \le d < 3P^2$ and $P \le d \pmod{3P} < 2P$. Assume $d = 3Pj_1 + b_1, 0 \le j_1 < P, P \le b_1 < 2P$; let $c = \frac{(i-(i+j_1)^2)(P+1)}{2} \mod P$, $b = b_1 \mod P$ (both $c, b \in [0, P)$), and we find the pair (a_j, a_k) as:

$$a_j = \begin{cases} 3P(i+j_1) + P + c \mod n & \text{if } c \ge b \\ 3P(i+j_1) + 2P + c \mod n & \text{if } c < b \end{cases}$$

$$a_k = \begin{cases} 3Pi + c - b & \text{if } c \ge b \\ 3Pi + P + c - b & \text{if } c < b \end{cases}$$

It can be verified that $a_j, a_k \in D_i$ and $a_j - a_k \equiv d \pmod n$.

- When $P \le d < 3P^2$ and $2P \le d \pmod{3P} < 3P$. Assume $d = 3Pj_1 + b_1$, $0 \le j_1 < P, 2P \le b_1 < 3P$. Find (a_j, a_k) as in the second case:

$$\begin{aligned} a_j &= 3Pj_2 + b_2 \mod n \\ a_k &= 3Pj_3 + b_3 \mod n \end{aligned}$$

The difference with the second case is $2P \le b_3 < 3P$; then $P \le b_2 < 3P$ and we can find out the appropriate j_2, j_3 values. Then apply the above equalities to derive a_j and a_k.

Based on the four cases above, we claim that $\forall d \ne 0 \pmod n$, we can find at least one ordered pair (a_j, a_k) such that $a_j - a_k \equiv d \pmod n$. ∎

Now we prove Theorem 1 in two aspects:

Proof: First, each set $D_i \in S$ is an RDS from Lemma 4.3. Then, we claim that $\forall D_i, D_j \in S$, $i \ne j$, $D_i \bigcap D_j = \emptyset$.

From Alg. 1: $D_i = T_{i0} \bigcup T_{i1} \bigcup \cdots \bigcup T_{i,P-1}$ and $D_j = T_{j0} \bigcup T_{j1} \bigcup \cdots \bigcup T_{j,P-1}$, it is clear that $\forall k_1 \ne k_2$, $T_{i,k_1} \bigcap T_{j,k_2} = \emptyset$. Thus, we need to show $\forall 0 \le k < P, T_{ik} \bigcap T_{jk} = \emptyset$. If $k \ne i, k \ne j$, two numbers from U_k are chosen for T_{ik}, T_{jk} respectively according to p_{ik} and p_{jk}. From Lines 6, 7 of Alg. 1, $p_{ik} = \frac{(i-q_k)(P+1)}{2} \mod P$, $p_{jk} = \frac{(j-q_k)(P+1)}{2} \mod P$, when $0 \le i, j < P, i \ne j$; we can conclude $p_{ik} \ne p_{jk}$, and thus $T_{ik} \bigcap T_{jk} = \emptyset$. If $k = i$ or $k = j$, the first P numbers of U_k will be chosen, while the other two numbers $3Pk + P + p_{ik}$ and $3Pk + 2P + p_{ik}$ will not intersect with the first P numbers, and thus $T_{ik} \bigcap T_{jk} = \emptyset$. So $\forall k_1, k_2, T_{i,k_1} \bigcap T_{j,k_2} = \emptyset$ and it implies $D_i \bigcap D_j = \emptyset$.

Combining the two aspects, $S = \{D_0, D_1, \cdots, D_{P-1}\}$ is a DRDS. ∎

Alg. 1 constructs a DRDS with cardinality $\sqrt{\frac{n}{3}}$ in $O(n)$ time, and the approximation ratio compared to the bound in Lemma 4.2 is: $\frac{\sqrt{n}}{\sqrt{n/3}} = \sqrt{3}$ when $n = 3P^2$, $P \ge 3$ and P is a prime number.

Remark 4.1: We are unable to find a general method to construct a DRDS S under any Z_n such that $|S|$ is comparable to the bound in Lemma 4.2. However, if there exists some construction for arbitrary Z_n, we can transform it to a good rendezvous protocol as in Section VI-A.

V. DRDS Based Rendezvous

In this section, we first introduce the Common Channel Hopping Sequence (CCHS) which is based on the CH technique. And then we present the DRDS based algorithm and its performance analysis.

A. Common Channel Hopping Sequence

Channel Hopping (CH) technique is commonly used in blind rendezvous algorithms [2], [3], [11], [18], [20]. The intuition is: in order to guarantee rendezvous for asynchronous users, the rule to access channels should be periodic. Thus, we construct a fixed length sequence $HS = \{s_0, s_1, \cdots, s_{T-1}\}$ where s_i is an available channel and the user hops among the channels by repeating the sequence, i.e. they access $s_{t \bmod T}$ at time t. Since the available channel sets for asymmetric users are different, we design a common sequence for all users and each user can design its own hopping sequence based on it.

Definition 5.1: We call $HS = \{s_0, s_1, \cdots, s_{T-1}\}$ a Common Channel Hopping Sequence (CCHS) if s_i is chosen from the whole channel set $C = \{1, 2, \cdots, N\}$.

Based on the CCHS, each user with available channel set $C' \subseteq C$ can access channel $s_{t \bmod T}$ at time t if the channel is in C'; otherwise it can choose randomly from C' or according to some fixed rule. The advantage of constructing such a CCHS is: rendezvous can be guaranteed for any two users asynchronously and asymmetrically when the CCHS is *good*; here *good* means the following requirement is satisfied:

Requirement 1: For a CCHS $HS = \{s_0, s_1, \cdots, s_{T-1}\}$, $\forall \delta_t \ge 0$ and $\forall i \in C$, there exists t such that $s_{t \bmod T} = i$ and $s_{(t+\delta_t) \bmod T} = i$.

If a CCHS is *good*, any two users repeating it will eventually achieve rendezvous on some common available channel when one user is δ_t time slots later.

B. DRDS Based Rendezvous Algorithm

We present the asynchronous rendezvous algorithm based on the DRDS construction. Suppose the whole channel set is $C = \{1, 2, \cdots, N\}$ and the available channel set for the user is $C' \subseteq C$.

Algorithm 2 DRDS Based Rendezvous Algorithm

1: Find the smallest prime P such that $P \geq N$;
2: **if** $P = 2$ **then**
3: $T := 6, t := 0$;
4: $S = \{D_0, D_1\}$, $D_0 = \{0, 1, 3\}, D_1 = \{2, 4, 5\}$;
5: **else**
6: $T := 3P^2$, $t := 0$;
7: Construct the DRDS $S = \{D_0, D_1, \cdots, D_{P-1}\}$ under Z_T as Alg. 1;
8: **end if**
9: **while** Not rendezvous **do**
10: **if** $0 \leq t < 2P$ **then**
11: Access the channel with smallest label in C';
12: **else**
13: $d := (t - 2P) \bmod T$;
14: Find $D_i \in S$ such that $d \in D_i$;
15: **if** Channel $(i + 1) \in C'$ **then**
16: Access channel $(i + 1)$;
17: **else**
18: Access an available channel in C' randomly;
19: **end if**
20: **end if**
21: $t := t + 1$;
22: **end while**

In Alg. 2, the first $2P$ time slots for the user is to access a fixed channel, which resembles the listening period when a user wakes up in many asynchronous protocols; we call this the *Listening Stage*. Afterwards, it is the *Accessing Stage* which repeats a length $T = 3P^2$ CCHS based on the DRDS construction under Z_T from Alg. 1 (if $P = 2$, the CCHS length is 6 and the DRDS is given as Line 4). Given any time t, compute $d = (t - 2P) \bmod T$ and find the RDS D_i that contains d. The user accesses channel $(i+1)$ if it is available; otherwise, it accesses an available channel randomly. Fig. 3 is an example when $N = 2$ and $C' = \{1, 2\}$. The first four time slots are the *listening stage*, and in the *accessing stage*, the user repeats the sequence with length $T = 6$.

Time	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	...
Channel	1	1	1	1	1	1	2	1	2	2	1	1	2	1	2	2	...

Fig. 3. An example of Alg. 2

C. Performance Analysis

Theorem 2: For two users Alice and Bob with available channel sets $C_A, C_B \subseteq C$, whenever they start Alg. 2, rendezvous can be guaranteed within $MTTR = 3P$ time slots if $C_A = C_B$, and $MTTR = 3P^2+2P$ time slots if $C_A \neq C_B$.

Proof: It is easy to verify when $N \leq 2$, the theorem holds. For any $N \geq 3$, without loss of generality, suppose Alice starts earlier at time 0 and Bob starts at time $\delta_t \geq 0$.

If $C_A = C_B$, the best scenario for rendezvous is $0 \leq \delta_t < 2P$ because they are both in the *listening stage* accessing the same channel. If $0 \leq (\delta_t - P) \pmod{3P} < 2P$, rendezvous occurs in the first $2P$ time slots when Bob is *listening*, while Alice is *accessing*. If $2P \leq (\delta_t - P) \pmod{3P} < 3P$, Bob can achieve rendezvous in time $[2P, 3P)$ while keeping accessing some fixed channel and the P numbers in $[\delta_t + 2P, \delta_t + 3P)$ for Alice are in P different RDSs; so they achieve rendezvous in the *accessing stage*. Thus: $MTTR \leq 3P$.

If $C_A \neq C_B$, we claim that rendezvous is guaranteed in $T+2P$ time slots. Let $d = \delta_t \pmod T$. They may not achieve rendezvous in the *listening stage* even when $\delta_t < P$. For any channel $i \in C_A \bigcap C_B$, we can find an ordered pair (a_j, a_k) from RDS D_{i-1} such that $a_j - a_k \equiv d \pmod T$ (Def. 4.1). So when Bob's time ticks $a_k + P$, they both access channel i, which implies rendezvous is guaranteed within $P + a_k \leq T + 2P$ time slots. ∎

Remark 5.1: The smallest prime $P \geq N$ has been proved[3] to be $\leq 2N$, and thus $MTTR = O(N)$ for symmetric users and $MTTR = O(N^2)$ for asymmetric users.

Remark 5.2: We assume the start time of any user is aligned to certain time slot, and doubling the length of each time slot can solve the problem when time is not aligned [11].

Remark 5.3: Alg. 2 can be extended to the multi-user multi-hop scenario through the basic idea in [6], [11]. Suppose D is the diameter of the CRN in terms of hop count, rendezvous is achieved in $MTTR = 3PD$ time slots for symmetric scenarios and $MTTR = (3P^2 + 2P)D$ time slots for asymmetric scenarios. Because of the lack of space, we omit the details.

VI. Lower Bound of Good CCHS

In this section, we first show the equivalence of DRDS and *good* CCHS, and then derive the lower bound on *good* CCHS length. This lower bound also holds for any algorithm based on the Channel Hopping technique.

A. Equivalence of DRDS and good CCHS

Lemma 6.1: Any DRDS corresponds to a *good* CCHS.

Proof: Consider a DRDS $S = \{D_0, D_1, \cdots, D_{h-1}\}$ under Z_n; we construct CCHS $HS = \{s_0, s_1, \cdots, s_{n-1}\}$ as: $s_i = j + 1$ if there exists D_j such that $i \in D_j$. If i does not belong to any set in S, assign any value in $[1, h]$ to s_i. We claim HS is *good*. Suppose Alice and Bob are repeating HS to access channels from $[1, h]$ and Bob is d time slots after Alice, if $d \equiv 0 \pmod n$, they access the same channel at

[3]Bertrand-Chebyshev Theorem: $\forall n > 1$, at least one prime p exists such that $n < p < 2n$

every time slot, and so Requirement 1 is satisfied. Let $d' = d \mod n$; thus $1 \le d' < n$, for any common available channel i, and there exists a pair (a_j, a_k) where $a_j, a_k \in D_{i-1}$ and $a_j - a_k \equiv d' \pmod n$. Therefore, HS is *good*. ∎

Lemma 6.2: Any *good* CCHS corresponds to a DRDS.

The lemma can be verified easily and we omit the proof.

The two lemmas show the equivalence between a *good* CCHS and a DRDS. We use the equivalence to derive the lower bound of any *good* CCHS length.

B. Lower Bound of Good CCHS

Lemma 6.3: Suppose D is an RDS under Z_T where $T = N(N+1)$ and $|D| = N+1$, then $N \le 3$.

Proof: Consider all pairs (a_j, a_k) where $a_j, a_k \in D, j \ne k$, and define $d_{jk} = (a_j - a_k) \mod T$ which we call a *difference value*. $\forall d \in \{1, 2, \cdots, T-1\}$, there exists at least one difference value $d_{jk} = d$. Since there are $N(N+1)$ difference values, there exists two pairs (a_j, a_k) and (a'_j, a'_k) such that $d_{jk} = d_{j'k'}$ and the other difference values are all distinct. However, $d_{kj} = T - d_{jk} = T - d_{j'k'} = d_{k'j'}$ implies there exists another two pairs (a_k, a_j), (a'_k, a'_j) sharing a common difference value. The situation can happen only when $a_j = a'_k, a_k = a'_j$. Then $a_j - a_k \equiv a_k - a_j \mod T$ means $a_j - a_k \equiv \frac{T}{2} \mod T$. From Lemma 4.1, construct another RDS $D' = \{(a - a_j) \mod T | a \in D\}$, thus $0, \frac{T}{2} \in D'$.

Denote $S_1 = \{0 < a < \frac{T}{2} | a \in D'\}$, $S_2 = \{\frac{T}{2} < a < T | a \in D'$, and let $d_1 = |S_1|$ and $d_2 = |S_2|$; thus $d_1 + d_2 = |D| - 2 = N - 1$. We count the number $S_3 = \{0 < a < \frac{T}{2} | a \notin D'\}$ from two sides: Since $|S_1 \bigcup S_3| = \frac{T}{2} - 1$, thus $d_1 + |S_3| = \frac{T}{2} - 1 \Rightarrow |S_3| = \frac{T}{2} - 1 - d_1$. From the analysis above, all other pairs $(a_j, a_k) \ne (0, \frac{T}{2})$ or $(\frac{T}{2}, 0)$ should have a distinct difference value; we construct S_3 as follows:

- $\forall a \in S_1$, let $a' = \frac{T}{2} - a \in S_3$; otherwise $(a, 0)$ and $(\frac{T}{2}, a')$ share the same difference value;
- $\forall a \in S_2$, let $T - a \in S_3$ and $a - \frac{T}{2} \in S_3$;
- $\forall a_1 < a_2 \in S_1$, define $\delta = a_2 - a_1$, and let $\frac{T}{2} - \delta \in S_3$ and $\delta \in S_3$; otherwise we can find two pairs sharing a common difference value;
- $\forall a_1 < a_2 \in S_2$, define $\delta = a_2 - a_1$, $0 < \delta < \frac{T}{2}$, and then let $\frac{T}{2} - \delta$ and δ belong to S_3.
- $\forall a_1 \in S_1$, $\forall a_2 \in S_2$, define $\delta = a_2 - a_1$, if $\delta > \frac{T}{2}$; rewrite $\delta = T - \delta$, and then let $\delta \in S_3$ and $(\frac{T}{2} - \delta) \in S_3$.

It is easy to verify that when we choose one value a or two values $a, \frac{T}{2} - a$ to compose S_3, they cannot belong to S_3 before the step. (If $a = \frac{T}{2} - a$, we only add the value once and this special situation happens at most once.) Thus:

$$|S_3| \ge d_1 + 2d_2 + 2 \cdot \frac{d_1(d_1-1)}{2} + 2 \cdot \frac{d_2(d_2-1)}{2} + 2d_1d_2 - 1$$

So $\frac{T}{2} - 1 - d_1 \ge (d_1 + d_2)^2 + d_2 - 1$; plugging $d_1 + d_2 = N - 1$, we can derive $N^2 \le 3N \Rightarrow N \le 3$. ∎

Theorem 3: Any *good* CCHS $HS = \{s_0, s_1, \cdots, s_{T-1}\}$ based on N channels satisfies:

$$T \ge \begin{cases} N^2 + N & \text{If } N \le 2 \\ N^2 + N + 1 & \text{If } N \ge 3 \text{ and } N \text{ is a prime power} \\ N^2 + 2N & \text{Otherwise} \end{cases}$$

Proof: When $N = 1$, it is clear that $T \ge 2$. Suppose $N \ge 2$; from Lemma 6.2, we can construct a DRDS $S = \{D_0, D_1, \cdots, D_{N-1}\}$ under Z_T. From Lemma 4.2, $N \le \sqrt{T} \Rightarrow T \ge N^2$.

Let $h = \min_{D_i \in S} |D_i|$; if $h \le N$, the set D_i, where $|D_i| = h$, has exactly $h(h-1)$ ordered pairs (a_j, a_k) implying at most $h(h-1) \le N(N-1)$ difference values for d exist such that $a_j - a_k \equiv d \pmod T$. When $N \ge 2$, $N(N-1) < N^2 - 1 \le T - 1$, D_i cannot be a RDS. Thus $h \ge N + 1$.

Assume $h = N + 1$, from $D_0 \bigcup D_1 \bigcup \cdots \bigcup D_{N-1} \subseteq Z_T$, $T \ge \sum_{i=0}^{N-1} |D_i| \ge Nh = N(N+1)$. Three cases are analyzed.

Case 1: If $T = N(N+1)$, from Lemma 6.3, $N \le 3$. When $N = 2$, $\{\{0, 1, 3\}, \{2, 4, 5\}\}$ is a DRDS under Z_6. However, when $N = 3$, we cannot find a DRDS with three disjoint RDS through exhaustive search;

Case 2: If $T = N^2 + N + 1$, suppose D_i suits $|D_i| = h$; since $(N+1)N = T - 1$, D_i is a $(T, h, 1)$-Difference Set. In [4], this is called a *Singer* Difference Set and it can be constructed only when N is a prime power. Thus when $N \ge 3$ and N is a prime power, $T \ge N^2 + N + 1$.

Case 3: If $T \ge N^2 + N + 2$ and N is not a prime power, and suppose an RDS D_i suits $|D_i| = h$; there are at most $h(h-1)$ ordered pairs (a_j, a_k) and the difference values $a_j - a_k \equiv d \pmod n$ cannot cover $\{1, 2, \cdots, T-1\}$ since $h(h-1) = N(N+1) < N^2 + N + 1 \le T - 1$, implying D_i is not an RDS under Z_n, and so $h \ge N + 2$. From $D_0 \bigcup D_1 \bigcup \cdots \bigcup D_{N-1} \subseteq Z_T$, we conclude $T \ge \sum_{i=0}^{N-1} |D_i| \ge Nh \ge N(N+2)$. ∎

The lower bound is not always tight. Finding the minimum *good* CCHS length is equivalent of finding the maximum DRDS. As discussed in Section IV-B, it is hard to find the maximum DRDS, and thus it is also hard to find the tight lower bound for *good* CCHS. From Table II, the lower bound is tight when $N = 1, 2, 5, 6$. However, when $N = 3, 4$, the lower bound for T is $13, 21$ respectively from the theorem, but the maximum DRDS $|S_n| = 2, 3$ under Z_n, implying the lower bound is not always tight.

Corollary 1: Any blind rendezvous algorithm based on the Channel Hopping technique cannot guarantee rendezvous within T time slots, where T is the expression in Theorem 3.

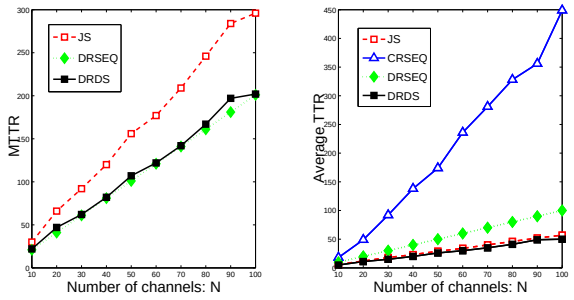
Consider any blind algorithm based on the CH technique, if two asymmetric users can rendezvous asynchronously, Requirement 1 should be satisfied because the user does not know on which channel they will achieve rendezvous. Thus the sequence can be thought of as a variation of *good* CCHS, which implies the lower bound should be T as in Theorem 3.

Corollary 2: Our DRDS based algorithm can achieve constant approximation as compared to the lower bound $\Omega(N^2)$ of any algorithm based on the CH technique. Thus, it is a nearly optimal asynchronous rendezvous algorithm.

VII. Simulation

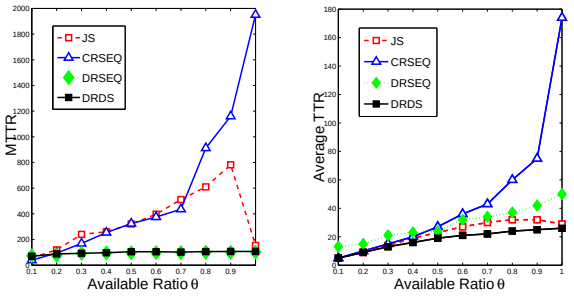
A. Symmetric Scenarios

For the symmetric scenario, we select three representative algorithms: Jump-Stay (JS) [11], DRSEQ [21] and CRSEQ [18] for comparison. We evaluate all these algorithms using



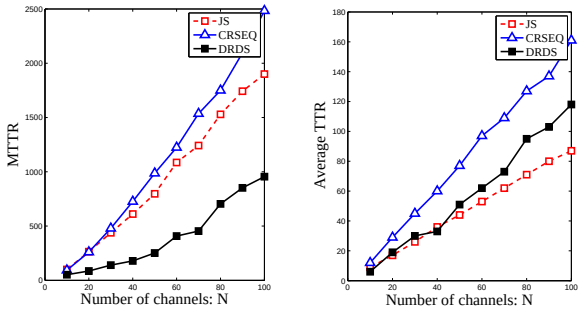
(a) $MTTR$ vs. N (b) Average TTR vs. N

Fig. 4. $MTTR$ and Average TTR as N increases



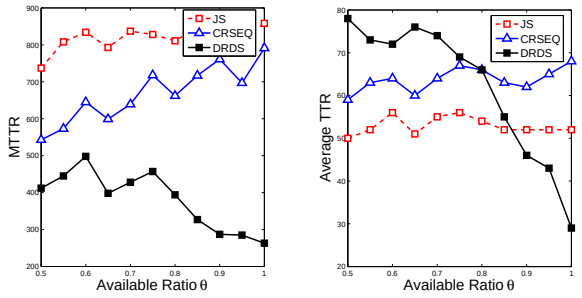
(a) $MTTR$ vs. θ (b) Average TTR vs. θ

Fig. 5. $N = 50$, $MTTR$ and Average TTR as θ increases



(a) $MTTR$ vs. N (b) Average TTR vs. N

Fig. 6. $\theta_A = 0.8$, $\theta_B = 0.8$, $MTTR$, and Average TTR as N increases



(a) $MTTR$ vs. θ_B (b) Average TTR vs. θ_B

Fig. 7. $N = 50$, $\theta_A = 0.5$, $MTTR$, and Average TTR as θ_B increases

two metrics: $MTTR$ and Average TTR. For a sequence with length T, we compute the TTR values T_d when one user is d time slots later than the other ($d = 0, 1, \cdots, T-1$); then Average TTR is defined to be $\frac{\sum_{i=0}^{T-1} T_d}{T}$. Both metrics are meaningful because small $MTTR$ value means two users can achieve rendezvous quickly even in the worst case, while small Average TTR means two users can achieve rendezvous quickly on average.

Suppose all N channels are available to the users; when N goes from 10 to 100, Fig. 4 shows both $MTTR$ and Average TTR values increase for all the algorithms because the probability to rendezvous decreases as more channels can be accessed. (Fig. 4(a) does not plot CRSEQ because its $MTTR$ value is very large. When $N = 20$, $MTTR = 1381$; when $N = 100$, $MTTR = 26070$). Fig. 4(a) shows that both DRSEQ and our DRDS based algorithm have much smaller $MTTR$ values and Fig. 4(b) shows our DRDS based algorithm has the smallest Average TTR value.

Fix $N = 50$ and suppose there are K available channels; we define available ratio as $\theta = \frac{K}{N}$, and when θ ranges from $[0.1, 1]$, Fig. 5(a) shows both DRSEQ and our algorithm work better than the others with $MTTR \simeq 100$ (the plot of DRSEQ and that of our algorithm are almost the same); Fig. 5(b) shows our algorithm works with the smallest Average TTR value.

From the two aspects, DRSEQ and our algorithm perform better than CRSEQ and JS in guaranteeing rendezvous for symmetric users.

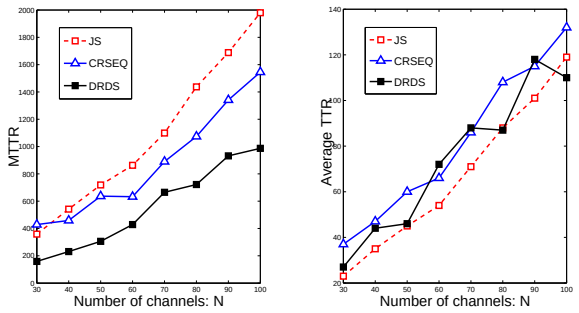
Remark 7.1: In Fig. 5(a), JS cannot guarantee rendezvous in $3P$ time slots (when $\theta = 0.9$ as in the plot, $MTTR = 782$). Because when $\theta < 1$, two users may achieve rendezvous on some unavailable channel, while JS accesses available channels randomly for the situation, implying $3P$ is not enough to guarantee rendezvous. When $\theta = 1$, all channels are available and $3P$ time slots can guarantee rendezvous, as proved in [11]. Therefore, there is a sudden drop from $\theta = 0.9$ to $\theta = 1$ in Fig. 5(a).

B. Asymmetric Scenarios

As DRSEQ is inapplicable to asymmetric users, we select CRSEQ and JS for comparison. We also evaluate these algorithms by $MTTR$ and Average TTR. Suppose the available channel sets for two users are C_A, C_B respectively. Define $\theta_A = \frac{|C_A|}{N}$, $\theta_B = \frac{|C_B|}{N}$.

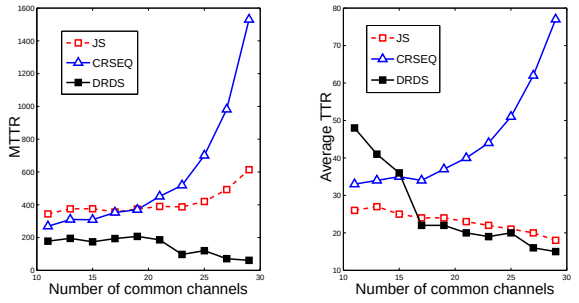
Fix $\theta_A = 0.8, \theta_B = 0.8$; when N increases from 10 to 100, Fig. 6 shows both $MTTR$ and Average TTR values increase for all three algorithms. Fig. 6(a) shows our algorithm achieves the smallest $MTTR$ value among them, and Fig. 6(b) shows our algorithm is slightly worse than JS in average TTR comparison.

Fix $N = 50$, $\theta_A = 0.5$; when θ_B ranges from 0.5 to 1, Fig. 7(a) shows our algorithm has the smallest $MTTR$ value and fluctuation exists since the channels are randomly generated. Fig. 7(b) shows that the average TTR for both JS and CRSEQ did not change much, while our algorithm improves when $\theta_B \geq 0.8$ because large θ_B increases the probability to rendezvous in the *listening stage*.



(a) $MTTR$ vs. N (b) Average TTR vs. N

Fig. 8. $|C_A \bigcap C_B| = 20$, $MTTR$ and Average TTR as N increases from 30 to 100



(a) $MTTR$ vs. N (b) Average TTR vs. N

Fig. 9. $N = 30$, $MTTR$ and Average TTR when $|C_A \bigcap C_B|$ ranges from 10 to 30

Fix $|C_A \bigcap C_B| = 20$; when N ranges from 30 to 100, Fig. 8(a) shows our algorithm has the smallest $MTTR$ value as N increases, while the average TTR values for the three algorithms do not differ much from Fig. 8(b).

Fix $N = 30$, and define $K = |C_A \bigcap C_B|$; when K ranges from 10 to 30, Fig. 9(a) shows our algorithm has the smallest $MTTR$ value among the three algorithms. Fig. 9(b) shows that the average TTR of both JS and our algorithm will decrease as K increases. When $K \simeq 16$, our algorithm achieves a smaller average TTR value than JS.

From the simulation results, our algorithm performs the best with the smallest $MTTR$ value among the three algorithms. The average TTR value changes according to θ_A, θ_B, K, N and our algorithm can also achieve good performance.

VIII. Conclusion

In this paper, we present a new method for designing blind rendezvous algorithms. We construct a DRDS consisting of P disjoint sets under Z_n where $n = 3P^2$ and P is a prime number. Then we extend the construction to design an efficient blind rendezvous algorithm, which can guarantee rendezvous within $MTTR = O(N)$ time slots for symmetric users and $MTTR = O(N^2)$ time slots for asymmetric users. Common Channel Hopping Sequence (CCHS) is introduced in the algorithm analysis, and subsequently we show the equivalence of DRDS and a *good* CCHS. Based on the equivalence, for the first time in the literature, we derive a lower bound for any blind rendezvous algorithm using the Channel Hopping technique. Extensive simulations have been conducted and the results validate our analytical results.

IX. Acknowledgment

This work was supported in part by the National Basic Research Program of China Grant 2011CBA00300, 2011CBA00302, the National Natural Science Foundation of China Grant 61073174, 61103186, 61033001, 61061130540, and Hong Kong RGC GRF grant 714311.

References

[1] C.J.L. Arachchige, S. Venkatesan, and N. Mittal. An Asynchronous Neighbor Discovery Algorithm for Cognitive Radio Networks. In *DySPAN*, 2008.

[2] K. Bian, J.-M. Park, and R. Chen. A Quorum-Based Framework for Establishing Control Channels in Dynamic Spectrum Access Networks. In *Mobicom*, 2009.

[3] K. Bian, J.-M. Park, R. Chen. Control Channel Establishment in Cognitive Radio Networks using Channel Hopping. *IEEE Journal on Selected Areas in Communications*, 29(4):689-703, 2011.

[4] C.J Colbourn, and J.H. Dintiz. Handbook of Combinatorial Designs. *CRC Presee*, 2006.

[5] L. DaSilva, and I. Guerreiro. Sequence-Based Rendezvous for Dynamic Spectrum Access. In *DySPAN*, 2008.

[6] R. Gandhi, C.-C. Wang, adn Y.C. Hu. Fast Rendezvous for Multiple Clients for Cognitive Radiso Using Coordinated Channel Hopping. In *SECON*, 2012.

[7] E. Hazan, S. Safra, and O. Schwartz. On the Complexity of Approximating k-Set Packing. *Computational Complexity*, 15(1): 20-39, 2006.

[8] 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*, 26(1):106-117, 2008.

[9] J.R. Jiang, Y.C. Tseng, and T. Lai. Quorum-Based Asynchronous Power-Saving Protocols for IEEE 802.11 Ad Hoc Network. *ACM Journal on Mobile Networks and Applications*, 10(1-2):169-181, 2005.

[10] L. Lazos, S. Liu, and M. Krunz. Spectrum Opportunity-Based Control Channel Assignment in Cognitive Radio Networks. In *SECON*, 2009.

[11] H. Liu, Z. Lin, X. Chu, and Y.-W. Leung. Jump-Stay Rendezvous Algorithm for Cognitive Radio Networks. *IEEE Transactions on Parallel and Distributed Systems*, 23(10):1867-1881, 2012.

[12] W.S. Luk, and T.T. Wong. Two New Quorum Based Algorithms for Distributed Mutual Exclusion. In *ICDCS*, 1997.

[13] R.M. Karp. Reducibility Among Combinatorial Problems. *Complexity of Computer Computations*, pp. 85-103, 1972.

[14] Y. Kondareddy, P. Agrawal, and K. Sivalingam. Cognitive Radio Network Setup without a Common Control Channel. In *MILCOM*, 2008.

[15] L. Ma, X. Han, and C.-C. Shen. Dynamic Open Spectrum Sharing MAC Protocol for Wireless Ad Hoc Networks. In *DySPAN*, 2005.

[16] J. Perez-Romero, O. Salient, R. Agusti, and L. Giupponi. A Novel On-Demand Cognitive Pilot Channel enabling Dynamic Spectrum Allocation. In *DySPAN*, 2007.

[17] S. Romaszko, and P. Mähönen. Quorum-Based Channel Allocation with Asymmetric Channel View in Cognitive Radio Network. In *MSWiM Poster and 6th ACM PM2HW2N Workshop*, 2011.

[18] J. Shin, D. Yang, and C. Kim. A Channel Rendezvous Scheme for Cognitive Radio Networks. *IEEE Communications Letters*, 14(10):954-956, 2010.

[19] P.D. Sutton, K.E. Nolan, and L.E. Doyle. Cyclostationary Signatures in Practical Cognitive Radio Applications. *IEEE Journal on Selected Areas in Communications*, 26(1):13-24, 2008.

[20] N.C. Theis, R.W. Thomas, and L.A. DaSilva. Rendezvous for Cognitive Radios. *IEEE Transactions on Mobile Computing*, 10(2):216-227, 2011.

[21] D. Yang, J. Shin, and C. Kim. Deterministic Rendezvous Scheme in Multichannel Access Networks. *Electronics Letters*, 46(20):1402-1404, 2010.

[22] J. Zhao, H. Zheng, and G.-H. Yang. Distributed Coordination in Dynamic Spectrum Allocation Networks. In *DySPAN*, 2005.