Symmetric Channel Hopping algorithm for a Cognitive Radio Network in Asynchronous environment

Sangeeta Sa and Arunanshu Mahapatro, Senior Member IEEE

Abstract—Rendezvous is a critical step in Cognitive Radio Network (CRN) prior to transmission for establishing communication links between secondary users (SUs). Due to the long-term blocking, channel saturation, and scalability issues encountered by dedicated common control channels (CCC) in a distributed CRN, rendezvous is preferable on the available idle channels of the primary users (PUs). In fact, each SU is unaware of the other SUs’ available channel sets, and the blind rendezvous is performed through the channel hopping (CH) process. As a result, ensuring a rendezvous between two SUs in a finite period in an asynchronous environment remains a challenging problem. In this work, a disjoint difference set based CH (DDS-CH) method is developed that ensures the highest degree of rendezvous in comparatively less time. In an asymmetric model, where at least one channel is common to available channel sets of SUs, the proposed algorithm ensures maximum conditional time to rendezvous \( MCTTR = 2N^3 + \left( \frac{N}{2} \right) \times N - N \) in an asynchronous environment with lower channel loading (CL) value and maximum degree of rendezvous \( DoR \). Here, \( N \) is the number of licensed channel in the CRN.

Index Terms—Cognitive Radio network, blind rendezvous, channel Hopping, difference sets

I. INTRODUCTION

With the advent of smart ubiquitous technology, the usage of smart wireless gadgets and appliances has become mainstream. To deploy these overcrowded wireless devices, spectrum distribution faces a shortage in unlicensed spectrum bands such as the ISM (2.4 MHz) band. To solve the scarcity issue, unlicensed SUs can exploit the underutilized licensed band while the owner of the licensed band, i.e., PUs, is not using it. To enable this channel exploitation, cognitive radio network (CRN) has recently evolved, with each SU equipped with one or more cognitive radios to sense unoccupied spectrum \[1][2]. When the PU is not present, the SU uses the channel; when the PU appears, the SU hops to another sensed channel to avoid interference. Prior to data transmission, SUs must communicate control information to create a link in order to negotiate communication opportunities on these licensed channels. Traditionally, this information exchange is either enabled by a centralized entity or through the use of common control channels (CCC), which are accessible to all SUs [3][4]. The usage of CCC may result in saturation, congestion, and a bottleneck of CRN with a large number of SUs [5]. Hence, the licensed channels can be used as control channel, where the SUs need to rendezvous on a commonly available channel to exchange control information. It is mostly a blind rendezvous since the SUs have no prior knowledge about the other SUs, such as their presence in the network or channel sensing data. Hence, they need to hop on their available channels independently to meet on a common channel at the same time. While achieving a blind rendezvous through channel hopping (CH), the following significant challenges need to be addressed [6][7].

Though the asymmetric role based CH sequence shows better performance \[8][9][10], where sender-receiver roles are preassigned to users, assigning roles to users prior to transmission in CRN is an impractical and hard task. Hence, the CH sequence generating method should be either role independent or symmetric-role based.

In synchronous environment, the SUs start their CH at the same time. However, clock synchronization is difficult for spatially distributed SUs, each with its own local clock. Thus, the algorithm must assure rendezvous in an asynchronous environment with unpredictable drifts in the CH sequences of the SUs.

The majority of CRNs are heterogeneous in nature, with each SU having different sensing capacity, and possibly having different network information. As a result, the system may not be symmetric, with SUs having an equivalent set of available channels. The asymmetric situation requires a rendezvous between two SUs with different sets of available channels.

The performance of the CH scheme is evaluated in most of the literature mainly with the following metrics \[11][12][13].

- **Average time to rendezvous (ATTR):** The average time required for the first successful rendezvous on the available channel using a pair of CH sequences. It is also termed as Expected time to rendezvous (ETTR).
- **Maximum conditional time to rendezvous (MCTTR):** The maximum time required for a pair of SU to rendezvous on the available channels common to them when SUs might have different set of available channels. So MCTTR of value \( n \) means the pair of CH sequence definitely rendezvous within \( n \) time slots if at least one channel is common to available channel sets of any pair of SUs.
- **Degree of rendezvous (DoR):** Out of all the available channels, the minimum number of channels at which a pair of SU can rendezvous. CH algorithm with \( N \) available channels has \( DoR = 2 \) means a pair of SU would rendezvous atleast on two channels. Hence, higher
is the DoR, higher is the possibility of rendezvous for an asymmetric model.

- **Channel loading (CL):** This metric represents the maximum possible number of CH sequences ending up having rendezvous on the same channel at the same time following their CH sequence. Hence, large CL represents high possibility of channel blocking or channel congestion by SUs. CL value of $\frac{x+y}{2}$ means out of $y$ possible CH sequences $x$ distinct number of sequences are on the same channel at the same time.

It is clear from the above description that for a successful blind rendezvous, the proposed CH algorithm must have a small ETTR, CL, and MCTTR with a large DoR.

In this paper, a CH method is proposed to achieve relatively better performance, where the SUs achieve the blind rendezvous in distributed, and asynchronous environment. The contribution of this paper can be summarized as follows.

- A disjoint difference set (DS) is constructed for the generation of CH to achieve shorter TTR, which guarantees rendezvous for both symmetric and asymmetric model in an asynchronous environment.
- The performance of the rendezvous scheme has been presented through extensive simulation results under symmetric, and asymmetric model.
- The degree of rendezvous, channel loading and the upper-bound for the ETTR under the symmetric, and asymmetric model are derived.

The rest of the paper is organized as follows. Related works have been discussed in Section II and the problem formulation is done in section III. In Section IV, along with the theoretical background of difference set, a disjoint DS-based CH algorithm is proposed to achieve blind rendezvous. A brief performance evaluation and simulation work for the proposed scheme are discussed in Section V and VI respectively. Finally, the Section VI concludes this paper.

II. RELATED WORKS

As briefly noted in Table I, many channel hopping methods for a blind rendezvous have been developed by researchers focused on different CRN environments, and with varying performance. The notations used in the literature are as follows: $N$ represents the number of channels in the network, and $P$ represents the smallest prime number greater than $N$. In Jump and Stay (JS) [15], the CH is composed of jump and stay patterns. The SU hops on available channels in the jump pattern of length $2P$ time slots, and remains on a given channel in the stay pattern of length $P$ time slots. Though for the symmetric model JS works well, but it has a large rendezvous time in asymmetric model. Channel rendezvous sequence (CRSEQ) [7] creates a periodic CH sequence with $N$ subsequences, each with two jump patterns. The jump pattern is generated with the available channels taking advantages of triangular number. Three different type of CH methods named as rendezvous couple channel-hopping scheme (RCCH), Asynchronous rendezvous channel-hopping scheme (ARCH) and symmetric asynchronous rendezvous channel-hopping scheme (SARCH) are introduced in [10] targeting specific types of network environments. Despite having an optimum value of $\text{MTTR}=\frac{N}{2}$ and $\text{MCTTR}=\frac{N^2}{2}$, the scheme RCCH is a role-based CH suited to synchronous environments. ARCH is intended for an asynchronous environment, but users must still be assigned roles as sender or receiver. SARCH is a symmetric role-based CH scheme guaranteeing rendezvous in an asynchronous environment. However, this method falls short of providing full DoR. Alternative Hop-and-Wait (AHW), an ID-based CH method is proposed in AHW [16], and is further enhanced as Enhanced AHW (E-AHW) in EAHW [17], which uses the ID bits of SUs to produce the CH. However, as the length of the ID and the number of SUs increase, so does the TTR value. In Matrix-based T-Channel Hopping(T-CH) [20], a default matrix containing jump columns and stay columns is used to generate CH sequence by concatenating the rows of the matrix. Here, the jump column contains distinct available channels, while a stay column has a specific available channel.

Many other CH techniques have been proposed in the literature that are based on the concepts of differences set (DS) and Quorum-system. Two such methods such as MTTR-minimization Quorum-based CH (M-QCH) and Load-minimization Quorum-based CH (L-QCH) in [13] have relatively smaller MTTR and higher DoR. They are, however, only suitable in a synchronous environment, and with a large CL and MCTTR value for M-QCH and L-QCH, respectively. In [18], a Disjoint Relaxed Difference Set (DRDS) namely $D$ is used to generate a CH sequence to ensure rendezvous in an asynchronous environment. Here, $D = \{D_1, D_2, \ldots, D_p\}$ and $D_i = i^{th}$ difference set with $3P$ elements. The DRDS-based CH scheme is being improved in interference-based DRDS (I-DRDS), where the rendezvous is performed on channels with less interference [19], but the TTR for this CH method is still too high. In Union of disjoint difference sets (UDDS) [21], two methods UDDS-SCH and UDDS-ACH are proposed for the synchronous and asynchronous environment respectively. Instead of utilizing a defined disjoint difference set like in DRDS, the author proposes a searching method to find the best disjoint set out of all possible sets to obtain the lowest TTR. However, the computational complexity is high for a large value of $N$.

According to the literature, most of the CH schemes, although having substantial performance, are relevant to a single model or environment, and the performance of CH schemes that are applicable to all types of environments still needs to be improved.

III. SYSTEM MODEL AND PROBLEM FORMULATION

A. System Model

In this article, a distributed CRN with $N$ non-overlapping channels licensed to the PUs is examined. These channels are labeled as $0, 1, 2, \ldots, N-1$, and the coexist SUs are allowed to exploit these channels opportunistically. Each SU is equipped with a half-duplex cognitive radio transceiver for channel sensing and data transmission. Prior to data transmission, a pair of SUs, $S_U$ and $S_V$, must rendezvous on a common available channel to exchange control information in
order to establish a connection. The available channel set for $SU_i$ is a subset of the $N$ licensed channels and is denoted as $C_i$. Due to heterogeneity of the CRN, the available channel set of each SU might be different. Hence, for an asymmetric model with $C_u \cap C_v \neq \emptyset$, $SU_u$ and $SU_v$ would rendezvous on a channel $c \in C_u \cap C_v$. In this system, the time is assumed to be divided into time slots of constant length. The time-slot length should be long enough for a SU to make a hop, and exchange control information with the other SU. The SUs need to hop on specific available channels in each time slot according to their respective CH sequence. To rendezvous, both users must be on the same channel at the same time-slot. In this paper, an asynchronous environment is considered, where local clock of each SU may have random number of slots drift with the global clock.

### B. Problem formulation

A blind rendezvous algorithm needs to be designed focusing all the required criteria discussed in the introduction section. The sender and receiver SUs generate their CH sequences for the rendezvous, while keeping the following factors in mind:

- In an asymmetric model with different available channel sets, the SUs, $SU_u$ and $SU_v$ should use a CH algorithm that ensures rendezvous if there exist at least one common channel $c \in C_u \cap C_v$. This CH algorithm, designed for an asymmetric model, can be extended to a symmetric model with SUs with identical available channel sets.
- It is difficult to maintain a synchronous environment for SUs with their respective local clocks. In practise, an asynchronous environment is the primary consideration for rendezvous system design. The CH sequences of these SUs required to achieve rendezvous even when there is a $\tau$ time-slots drift between their respective clocks, when $\tau \geq 0$.
- Since assigning roles to SUs as sender and receiver is difficult, an effective rendezvous algorithm should be built, in which all SUs follow a common CH procedure.
- Along with the design requirements mentioned above, the rendezvous algorithm should have the highest degree of rendezvous with the lowest values for the performance metrics ETTR, MCTTR, and channel loading.

### IV. Proposed Channel Hopping scheme

To propose a rendezvous method that meets the discussed design criteria, a CH algorithm based on disjoint relaxed difference sets is developed.

#### A. Disjoint Difference Set

**Definition 1:** A k-element set $D = \{a_0,a_1,a_2,\cdots,a_{k-1}\}$ $\subset Z_n$ is called a relaxed cyclic difference set or simply $(n,k)$-DS, if $\forall d \in (0,n)$ there exist an ordered pair $(a_i,a_j)$ such that $a_i - a_j \equiv d \mod n$. Here, $a_i,a_j \in D$ and $Z_n$ is the set of non negative integers represented as $\{0,1,2,\cdots,n-1\}$.

**Corollary 1:** If set $D = \{a_0,a_1,\cdots,a_{k-1}\} \subset Z_n$ is an $(n,k)$-DS, then the set $\text{rotate}(D,i)$ is also an $(n,k)$-DS, where $i \in (0,n)$ and $\text{rotate}(D,i)=\{a_0+i,a_1+i,\ldots,a_{k-1}+i\} \mod n$.

**Proof:** According to definition 1, for every $d \in (0,n)$ there exist an ordered pair $(a_i,a_j)$ such that $d = (a_i-a_j) \mod n$ and $a_i,a_j \in D$. Let’s consider another ordered pair $(a_i',a_j')$ such that $(a_i',a_j')=(a_i+a_j, a_j+a_j) \mod n$. Now $d'=(a_i'-a_j') \mod n = (a_i-a_j) \mod n = d$. Thus for every $d \in (0,n)$ there exist a paired element in the $\text{rotate}(D,i)$.

**Example:** For the set $D = \{1, 4, 5\} \subset Z_6$ is an DS, then $\text{rotate}(D,4) = \{5, 2, 3\}$ is also an DS.

**Definition 2:** A Cyclic Quorum System(CQS) is denoted as a set $Q = \{ A_0, A_1,\ldots,A_{n-1}\}$, where each element of this set is a subset of $Z_n$ with the following two properties for all $i,j \in [0, n-1]$:

(i) $A_i = \text{rotate}(A_0,i)$,

(ii) $A_i \cap A_j = \emptyset$.

### Table 1: Performance comparison of Rendezvous schemes

| CH Algorithm | Degree of rendezvous | MCTTR | Channel loading | Environment | Role assignment |
|--------------|----------------------|-------|----------------|-------------|----------------|
| RCCH [10]$^*$ | $N$ | $2N^2$ | $\frac{1}{N}$ | Synchronous | Symmetric |
| FRCH [14]$^*$ | 1 | $2N^2 + N$ | $\frac{1}{2N^2 + N}$ | Asynchronous | Symmetric |
| ARCH [10]$^*$ | $N$ | $N^2$ | $\frac{1}{N^2}$ | Asynchronous | Asymmetric |
| SARCH [10]$^*$ | 1 | $8(N^2 + N)$ | $\frac{1}{8N^2 + 8N}$ | Asynchronous | Symmetric |
| JS [15] | $N$ | $3N(P - K) + 3P$ | $\frac{1}{3N(P - K) + 3P}$ | Asynchronous | Symmetric |
| AHW [16] | $K_{min}$ | $3P(1 + \lambda)(R + 1 - K)$ | $\frac{1}{3P(1 + \lambda)(R + 1 - K)}$ | Asynchronous | Symmetric |
| E-AHW [17] | $N$ | $3P^2 + 2P$ | $\frac{1}{3P^2 + 2P}$ | Asynchronous | Symmetric |
| DRDS [18] | $N$ | $3P^2$ | $\frac{1}{3P^2}$ | Asynchronous | Symmetric |
| I-DRDS [19] | $N$ | $3P^2$ | $\frac{1}{3P^2}$ | Asynchronous | Symmetric |
| CRSQ [7] | $N$ | $3P^2 - P$ | $\frac{1}{3P^2 - P}$ | Asynchronous | Symmetric |
| T-CH [20]$^*$ | $N$ | $2N^2 + \lfloor \frac{2}{2} \rfloor \times N$ | $\frac{1}{2N^2 + \lfloor \frac{2}{2} \rfloor \times N}$ | Asynchronous | Symmetric |
| DDS-CH $^*$ | $N$ | $2N^2 + \lfloor \frac{2}{2} \rfloor \times N - N$ | $\frac{1}{2N^2 + \lfloor \frac{2}{2} \rfloor \times N - N}$ | Asynchronous | Symmetric |

Algorithm supporting asynchronous also supports synchronous environment
$N$: Total number of channels in the CRN, $*: N$ is even, $\dagger$: $N$ is prime,
$K_{min}$: Minimum number of available channels, $\lambda$: ID string length of SU
$P$: Smallest prime greater than $N$.
ii) $A_i \cap A_j \neq \emptyset$

Corollary 2: If set $D = \{a_0, a_1, \ldots, a_{k-1}\}$ is an $(n, k)$-DS, then an CQS under $Z_n$ can be formed with a group of $n$ sets $A_0, A_1, \ldots, A_{n-1}$. Here, $A_i = \{(a_i + i) \mod n, (a_i + 2i) \mod n, \ldots, (a_i + (k-1)i) \mod n\} \forall i \in [0, n]$

Proof: It is suffice to prove that the set $Q = \{A_0, A_1, \ldots, A_{n-1}\}$ obtained from an $(n, k)$-DS will be a CQS if it satisfies the two properties given in definition 2. Here, each set is given by $A_i = \text{rotate}(D, i) \mod n = D$. Therefore, $A_i = \text{rotate}(A_0, i)$. For the second property we need to show, $\text{rotate}(A_0, i) \cap \text{rotate}(A_0, j) \neq \emptyset, \forall i, j \in [0, n-1]$. Say, $a_i$ and $a_m$ be the $(l+1)^{th}$ and $(m+1)^{th}$ elements in $A_0$, where $l, m \in [0, k-1]$. Then $(l+1)^{th}$ element in $\text{rotate}(A_0, i)$ is $(a_i + i) \mod n$ and $(m+1)^{th}$ element in $\text{rotate}(A_0, j)$ is $(a_m + j) \mod n$. According to definition 1, $A_0$ has number pairs for $\forall d \in [1, n-1]$. Let the number pair, $(a_i, a_m)$ has the difference, $d = (j - i) \mod n$, where $i, j \in [0, n-1]$

So, $a_i - a_m = (j - i) \mod n$

$\Rightarrow (a_i + i) \mod n - (a_m + j) \mod n \neq 0$\mod n

$\Rightarrow (l+1)^{th}$ element in $\text{rotate}(A_0, i)$ is $(m+1)^{th}$ element in $\text{rotate}(A_0, j)$

Hence, $\text{rotate}(A_0, i) \cap \text{rotate}(A_0, j) \neq \emptyset$, which makes $Q$ an CQS.

Definition 3: A group of disjoint difference sets (DDS) under $Z_n$ is denoted as $S = \{D_0, D_1, \ldots, D_{m-1}\}$. Here, $D_i$ is an DS and $D_i \cap D_j = \emptyset, \forall i, j \in [0, m-1]$

Corollary 3: If $S = \{D_0, D_1, \ldots, D_m - 1\}$ is an DDS with $D_i \subset Z_n$, then the set $\text{rotate}(S, i)$ is also an DDS, where $i \in [0, n] \mod n$ and $\text{rotate}(S, i) = \{\text{rotate}(D_0, i), \text{rotate}(D_1, i), \ldots, \text{rotate}(D_m, i)\}$.

Proof: Say, $D_1$ and $D_2$ are two disjoint DDS. For $D_1 \cap D_2 = \emptyset$, it is enough to prove that $\text{rotate}(D_1 + i) \cap \text{rotate}(D_2 + i) = \emptyset$ for $\forall i \in [0, n]$. Let $D_1 = \{a_0, a_1, \ldots, a_{k-1}\}$, $D_2 = \{b_0, b_1, \ldots, b_{k-1}\}$

As $D_1 \cap D_2 = \emptyset$

$\Rightarrow a_x - b_y \neq 0, \forall x, y \in [0, k-1]$

$\Rightarrow (a_i + i) - (b_i + i) \neq 0$

$\Rightarrow \text{rotate}(D_1, i) \cap \text{rotate}(D_2, i) = \emptyset$

Corollary 4: If $S = \{D_0, D_1, \ldots, D_{m-1}\}$ is an DDS with $D_i \subset Z_n$, then a set constitute of $m$ number of CQS can be formed with the help of $S$.

Proof: Considering the corollary 3, a total of $n$ DDS can be generated using $S$. The $i^{th}$ DDS is given by $\text{rotate}(S, i) = \{\text{rotate}(D_0, i), \text{rotate}(D_1, i), \ldots, \text{rotate}(D_m, i)\} \forall i \in [0, n] \mod n$. Now with corollary 2, it can be proved that $m$ number of CQS are formed as expressed below $\text{CQS}_j = \{\text{rotate}(D_j, 0), \text{rotate}(D_j, 1), \ldots, \text{rotate}(D_j, n-1)\}, \forall j \in [0, m-1]$. 

\[ X_0 = \{0, 1, 2, \ldots, v-1\} \]
\[ X_j = \{x, x+k, x+2k, \ldots, x+(v-1)k\} \]
\[ Z_n = \{0, 1, 2, \ldots, (Nk-1)\} \]
\[ D_1 = \{a, b, c, \ldots, (N-1)k+1, (N-1)k+2, \ldots, (N-1)k+N\} \]

2. $k = 2N + \lceil \frac{k}{N} \rceil$, $k$ is the average size of each DS in $S$

First one is applicable to all values of $n$ where as the second one is applicable for $N > 3$. It is obvious that for a given $N$, the value of $n$ in the second case is less than the value of $n$ in the first case.

To start the construction of disjoint sets under $Z_n$, the $Z_n$ is divided into $N$ disjoint sets of size $k$ denoted as $X_0, X_1, \ldots, X_{k-1}$ and is shown in Fig. 1. Here, $X_i = \{l, k, l+1, \ldots, (l+k-1)\}, l \in [0, N-1]$. Each $X_i$ is further divided into two subsets namely $A_i$ and $B_i$. The subset $A_i$ is having $N$ number of elements and subset $B_i$ is having $(k - N)$ number of elements. For the construction of $i^{th}$ DS of size $k$, the first $N$ elements are selected from $\{A_0, A_1, \ldots, A_{N-1}\}$ represented as $D_{Ai}$ and the rest $(k - N)$ elements are selected from $\{B_0, B_1, \ldots, B_{N-1}\}$ represented as $D_{Bi}$ by using the concept of triangular number. A triangular number $T_j$ is defined as $\frac{j(j+1)}{2}$ for any positive integer $j$. Therefore, $D_i$ is given by $D_i = D_{Ai} \cup D_{Bi}$, where

\[ D_{Ai} = \{a_{ip}\}, a_{ip} = kp + i \]
\[ D_{Bi} = \{b_{iq}\}, b_{iq} = k(T_q + i) + N + q \]

\[ T_q = \frac{q(q+1)}{2} \mod N, 0 \leq p \leq N - 1 \text{ and } 0 \leq q \leq k - N - 1 \]

Construction of $D_{Ai}$ and $D_{Bi}$ is mentioned in line 8-10 and line 12-15 in algorithm 1 respectively. For $k = 2N + \lceil \frac{k}{N} \rceil$, the disjoint set obtained in line 18 is an DDS. However, for $k = 2N + \lceil \frac{k}{N} \rceil$ all $D_i$ are DS except $D_{\frac{k}{N}}$ in $S$. To make $D_{\frac{k}{N}}$ an DS, last number from the $D_{N-1}$ set is taken and added to $D_{\frac{k}{N}}$ as shown in Fig. 3 and line 20 – 24 of Algorithm 1. In this process, $D_{N-1}$ does not loose the DS property. As an illustration, DDS for $N = 3$ and $N = 5$ is shown in the Fig. 2 and Fig. 3 respectively. Theorem 1 with the following lemmas proves that the constructed sets ($D_i$) are disjoint difference sets.

\textbf{Lemma 1:} If there exists a pair of numbers $(a_1, a_2)$ under $Z_n$ for some difference value $d$ such that $(a_1 - a_2) \mod n = d$ and $d' = d \mod k$, then for the same pair of numbers the difference value $n - d$ and $k - d'$ exist.

Proof: For the number pair $(a_1, a_2)$ if the difference $(a_1 - a_2) \mod n = d$ then $(a_2 - a_1) \mod n = (-d) \mod n$ which
Algorithm 1 Disjoint Difference Sets Construction

Require: N
Ensure: DDS S= \{D_0, D_1, D_2 D_{N-1}\}
1: \text{S} = \phi
2: \text{if } N=3 \text{ then}
3: \quad k=2N+\lceil \frac{N}{2} \rceil
4: \text{else}
5: \quad k=2N+\lceil \frac{N}{2} \rceil -1
6: \text{end if}
7: \text{for } i=0 \text{ to } (N-1) \text{ do}
8: \quad \text{for } j=0 \text{ to } (N-1) \text{ do}
9: \quad \quad a_{ij} = k^i j_1 + i
10: \quad \quad D_{A_i} = \{a_{ij}\}
11: \quad \text{end for}
12: \text{for } j=0 \text{ to } (N-1) \text{ do}
13: \quad q = \frac{j_{2} + j_{1} + 1}{2} \mod N + i
14: \quad b_{ij} = k^i q + N j_2
15: \quad D_{B_i} = \{b_{ij}\}
16: \text{end for}
17: D_i = D_{A_i} \cup D_{B_i}
18: S = S \cup \{D_i\}
19: \text{end for}
20: \text{if } N \neq 3 \text{ then}
21: \quad j_2 = N + \left\lfloor \frac{N}{2} \right\rfloor - 2
22: \quad i = N - 1
23: \quad D_{N-1} = D_{N-1} \{b_{ij}\}
24: \quad D_i = D_i \left\lfloor \frac{N}{2} \right\rfloor \cup \{b_{ij}\}
25: \text{end if}

is equivalent to \((n-d)\). Thus for a number pair with difference \(d\) if we have \(d' = d \mod k\), for the same pair with differences \((n-d)\) we can prove that \((n-d) \mod k = k - d'\).

Lemma 2: For an integer value \(m\) and a triangular number \(T(.) \mod m\), \((\Delta_0 + m \cdot l) \mod m = \Delta_0 + m \cdot l \mod m\), where \(\Delta_0\) is defined as \(T(m' + l) - T(l), 0 \leq m' \leq m - 1\) and \(l\) is an integer value.

Proof: \(\Delta_0 = T(m' + l) - T(l)\).
\[\Rightarrow \Delta_0 \mod m = (\frac{m' + l}{2} \cdot (m' + l + 1) - \frac{l(l+1)}{2}) \mod m\]
\[\Rightarrow \Delta_0 \mod m = (\frac{m_2}{2} + m' \cdot (l + 1) + m_1) \mod m\]
\[\text{Thus } \Delta_0 \mod m = \frac{(m_2}{2} + m' \cdot (l + 1) + m_1) \mod m\]

Theorem 1: The constructed set \(S\) in Algorithm 1 is a set of disjoint difference sets.

Proof: The set \(S = \{D_0, D_1, D_2, \ldots, D_{N-1}\}\) must satisfy the two properties given in definition 3. First, we need prove that each set \(D_i\) is an DS and then they are disjoint to each other.

Numbers from \(Z_n\) taken by using triangular numbers and the modular concept to construct a set that meets the DS properties. According to Definition 1, the required \(d \in [1, n - 1]\) is a range of difference values to be verified, which can be minimized to a smaller range with modular concept such that \(d' = d \mod k \in [0, k - 1]\). It is to be verified that \(\forall d' \in [0, k - 1]\), each set of \(S\) has a pair of numbers \((a_i, a_j)\) such that \((a_i - a_j) \mod n = d' \mod r, \forall r \in [0, N - 1]\).
1. For \(k = 2N + \left\lceil \frac{N}{2} \right\rceil\):

Case 1.1: \(d' = 0\). All \(D_i\) has a number pair \((a_{ij}, a_{ij})\) such that \(a_{ij} \in D_{A_i}, 0 \leq j_1, j_2 \leq N - 1\) and the difference \(d = (a_{ij} - a_{ij}) \mod n\)
\[d = (k(j_1 - j_2) + 0) \mod Nk\]
\[d = (k(j_1 - j_2) + 0), \text{ for } 0 \leq j_1 - j_2 \leq N - 1\]

Case 1.2: \(1 \leq d' \leq \left\lceil \frac{N}{2} \right\rceil\) and \(2N \leq d' \leq 2N + \left\lceil \frac{N}{2} \right\rceil - 1\). All \(D_i\) has a number pair \((b_{ij}, b_{ij})\) such that \(b_{ij}, b_{ij} \in D_{B_i}\) and \(0 \leq j_1, j_2 \leq k - N - 1\) and the difference of the numbers \(b_{ij} \text{ and } b_{ij} = d = b_{ij} - b_{ij} \mod n\)
\[d = (k(T_{j_1} - j_1) + (j_1 - j_2) \mod Nk)\]
\[d = (k(j_1 - j_2) \mod N) + (j_1 - j_2) \mod Nk\]

Case 1.3: \(\left\lceil \frac{N}{2} \right\rceil < d' \leq 2N - 1\). All \(D_i\) has a number pair \((a_{ij}, a_{ij})\) such that \(a_{ij} \in D_{A_i}, a_{ij} \in D_{B_i}\) and \(0 \leq j_1, j_2 \leq k - N - 1\). The difference is given by \(d = (b_{ij} - a_{ij})\mod n\)
\[d = (k(T_{j_2} + j_1 + i) + N + j_2 - k(j_1 + i) \mod N\]
\[d = (j_2 \mod N) \mod Nk\]
Here, \(r = (T_{j_2} + j_1 + i) \mod N\)
\[\text{Thus, } d' \text{ and } N \text{ are co-prime because } N \text{ is a prime number and } d' \text{ has a pair of numbers } (\frac{N}{2})\]

Further, using Lemma 1 the range of \(d'\) can be written as \(2N \leq d' \leq 2N + \left\lceil \frac{N}{2} \right\rceil - 1\).

Case 1.3: \(\left\lceil \frac{N}{2} \right\rceil < d' \leq 2N - 1\). All \(D_i\) has a number pair \((a_{ij}, a_{ij})\) such that \(a_{ij} \in D_{A_i}, a_{ij} \in D_{B_i}\) and \(0 \leq j_1, j_2 \leq k - N - 1\). The difference is given by \(d = (b_{ij} - a_{ij})\mod n\)
\[d = (k(T_{j_2} + j_1 + i) + N + j_2 - k(j_1 + i) \mod N\]
\[d = (j_2 \mod N) \mod Nk\]
Here, \(r = (T_{j_2} + j_1 + i) \mod N\)
\[\text{Thus, } d' \text{ and } N \text{ are co-prime because } N \text{ is a prime number and } d' \text{ has a pair of numbers } (\frac{N}{2})\]

Further, using Lemma 1 the range of \(d'\) can be written as \(2N \leq d' \leq 2N + \left\lceil \frac{N}{2} \right\rceil - 1\).

2. For \(k = 2N + \left\lceil \frac{N}{2} \right\rceil - 1\):
In this case, $|DB_{Bi}|$ is not same $\forall D_1$.

Case 2.1: $d' = 0, \forall D_1$. The proof is similar to Case 1.1.

Case 2.2: $1 \leq d' \leq \left\lfloor \frac{N}{2} \right\rfloor - 1$ and $2N \leq d' \leq 2N + \left\lfloor \frac{N}{2} \right\rfloor - 2, \forall D_1 \backslash \{D_{N-1}\}$. Similar to Case 1.2, for a given $d'$ if $0 \leq j_2 \leq N - 1 \Rightarrow d' \leq j_1 \leq (N - 1) + d'$ and as the maximum value of $j_1$ is $N - \left\lceil \frac{N}{2} \right\rceil - 2$, the range of $d'$ can be derived as $1 \leq d' \leq \left\lfloor \frac{N}{2} \right\rfloor - 1$. Then using Lemma 1, we also have $2N \leq d' \leq 2N + \left\lfloor \frac{N}{2} \right\rfloor - 2$.

Case 2.3: $\left\lceil \frac{N}{2} \right\rceil \leq d' \leq 2N - 1, \forall D_1 \backslash \{D_{N-1}\}, D_{N/2}$. Following the steps of Case 1.3, the range of $d'$ is derived for different values of $i$ and a defined range of $j_2$ in set $D_{Bi}$. For a given $d'$, if $0 \leq i \leq \left\lceil \frac{N}{2} \right\rceil - 1$ then $d' - N \leq j_2 \leq d' - N + \left\lceil \frac{N}{2} \right\rceil$, which implies $N \leq d' \leq 2N - 1$ and also using Lemma 1 the range of $d'$ is $\left\lfloor \frac{N}{2} \right\rceil \leq d' \leq N + \left\lfloor \frac{N}{2} \right\rceil - 1$. Similarly for a given $d'$ if $\left\lfloor \frac{N}{2} \right\rceil \leq i \leq N - 2$ then $d' - \left\lfloor \frac{N}{2} \right\rceil \leq j_2 \leq d' - 1$, which implies $\left\lceil \frac{N}{2} \right\rceil \leq d' \leq N - \left\lfloor \frac{N}{2} \right\rceil - 1$ and by using Lemma 1 we also have $N \leq d' \leq 2N - 1$. So the final range of $d'$ for $\forall D_1 \backslash \{D_{N-1}\}, D_{N/2}$ is $\left\lceil \frac{N}{2} \right\rceil \leq d' \leq 2N - 1$.

Case 2.4: $1 \leq d' \leq 2N + \left\lfloor \frac{N}{2} \right\rceil - 2, \forall D_{N-1}$. Following the steps of Case 2.3, the range of $d'$ can be derived with $i = N - 1$. Here, $0 \leq j_2 \leq N + \left\lfloor \frac{N}{2} \right\rceil - 3$ implies $1 \leq d' \leq 2N + \left\lceil \frac{N}{2} \right\rceil - 2$ and by using Lemma 1 we also get $N + 1 \leq d' \leq 2N + \left\lceil \frac{N}{2} \right\rceil - 2$. Therefore the final range of $d'$ is $1 \leq d' \leq 2N + \left\lceil \frac{N}{2} \right\rceil - 2$.

Case 2.5: $\left\lceil \frac{N}{2} \right\rceil \leq d' \leq 2N - 1, \forall D_1$. For $i = \left\lfloor \frac{N}{2} \right\rceil$ and $0 \leq j_2 \leq N + \left\lfloor \frac{N}{2} \right\rceil - 2$, following the steps of Case 2.4 we have $N - \left\lfloor \frac{N}{2} \right\rceil \leq d' \leq 2N - 1$ which implies $\left\lfloor \frac{N}{2} \right\rceil + 1 \leq d' \leq 2N - 1$. However, the considered range is $\left\lfloor \frac{N}{2} \right\rceil \leq d' \leq 2N - 1$ and $d' = \left\lceil \frac{N}{2} \right\rceil$ still need to be achieved. According to Case 2.2, to achieve $d' = \left\lceil \frac{N}{2} \right\rceil$ the range of $j_1$ and $j_2$ should be $0 \leq j_2 \leq N - 1$ and $\left\lfloor \frac{N}{2} \right\rceil \leq j_1 \leq (N - 1) + \left\lfloor \frac{N}{2} \right\rceil$. However, the maximum value of $j_1$ is $N + \left\lceil \frac{N}{2} \right\rceil - 2$. According to Case 1.2 for $d' = \left\lceil \frac{N}{2} \right\rceil$, the required $r$ should be $(T_{N + \left\lceil \frac{N}{2} \right\rceil} - T_{N - 1})$ mod $N$ and according to Lemma 2 it is equivalent to $r = (T_{N + \left\lceil \frac{N}{2} \right\rceil} - T_{N - 1} + \left\lfloor \frac{N}{2} \right\rceil)$ mod $N$. As shown in Fig. 3a, to achieve $r = (T_{N + \left\lceil \frac{N}{2} \right\rceil} - T_{N - 1} + \left\lfloor \frac{N}{2} \right\rceil)$ mod $N$, the last element of $D_{N-1}$ is added to $\{D_{N/2}\}$.

According to the above five cases, each $D_i$ in $S$ for $k = 2N + \left\lceil \frac{N}{2} \right\rceil - 1$ is a difference set since each $D_i$ has a pair of numbers with difference $d = d' + rk, \forall d' \in [0, k - 1]$ and $\forall r \in [0, N - 1]$. Therefore, it is proved that each $D_i$ in $S$ is an DS. Now their disjoint property can be proved by claiming $D_i \cap D_j = \phi, \forall D_i, D_j \in S, i \neq j$ and $i, j \in [0, N - 1]$.

According to the Algorithm 1, $D_i = D_{A_i} \cup D_{B_i}$, and $D_j = D_{A_j} \cup D_{B_j}$. The following three cases to be analyzed to prove $D_i \cap D_j = \phi$.

Case 1: The elements of the subset $D_{A_i}$ and $D_{B_j}$ are chosen from $A$ and $B$ subsets respectively (Fig.1) and $A \cap B = \phi$, therefore, $A_{D_1} \cap D_{B_1} = \phi$ and $D_{A_1} \cap D_{B_1} = \phi$.

Case 2: $D_{A_i} = (kp_1 + i)$ and $D_{A_j} = (kp_2 + j)$, where $p_1, p_2 \in [0, N - 1]$. So any element in $D_{A_i}$ and $D_{A_j}$ are the $i^{th}$ and $j^{th}$ element of $X_{p_1}$ and $X_{p_2}$ respectively (Fig.1). Since $i \neq j$, therefore, $D_{A_i} \cap D_{A_j} = \phi$.

Case 3: $D_{B_i} = (k(T_{q_1} + i) + N + q_1)$ and in $D_{B_j} = (k(T_{q_2} + j) + N + q_2)$, where $q_1, q_2 \in [0, k - 1]$. So any element in $D_{B_i}$ and $D_{B_j}$ are $(q_1 + N)^{th}$ element of the set $X_{T_{q_1}+1}$ and $(q_2 + N)^{th}$ element of the set $X_{T_{q_2}+1}$ respectively. To prove by contradiction, assume that there is a common element in $D_{B_i}$ and $D_{B_j}$, therefore, $(q_1 + N)^{th}$ number of $X_{T_{q_1}+1}$ is equal to $(q_2 + N)^{th}$ element of $X_{T_{q_2}+1}$ and since all $X$ are disjoint, we have $T_{q_1} + i = T_{q_2} + j$ and $q_1 = q_2$. Therefore, $i = j$ which is a contradiction. So for $i \neq j$ we always have $D_{A_i} \cap D_{B_j} = \phi$.

Hence, from above three cases it is proved that all $D_i$ are disjoint to each other.

C. CH generation based on DDS for the rendezvous

In blind rendezvous algorithm, CH techniques are used by the SU. Two SU hop on their sensed vacant channels in order to rendezvous on the common vacant channels available to them. In this paper, a role-independent common CH sequence generation algorithm as described in Algorithm 2 is adapted by all the SU for both synchronous as well as asynchronous CRN environment.

- **Generation of CH sequence for a SU:** As described in Algorithm 2, CH sequence is generated for $SU_u$ with $C_u \subseteq C$ by using the DDS. Here, $C = \{0, 1, 2, \ldots, N - 1\}$ is the set of licensed channels and $C_u$ is the available channel set of $SU_u$. Here, the DDS $S$ has $N$ number of
Algorithm 2 Generation of CH sequence

Require: L, C_v /L= Total licensed channels
\langle / C_v \rangle is available channel set of SU_v

Ensure: CH_v /CH sequence of user v
1: N= smallest prime number ≥ L
2: Construction of DDS S with N using Alg. 1
3: n= Total elements in S
4: x is a random number in [0, n-1]
5: S= rotate(S, x)
6: for t= 0 to n-1 do
7: find D_i ∈ S such that t ∈ D_i
8: if i ∈ C_v, then
9: CH_v[t]= i // i is accessed in slot t
10: else
11: CH_v[t]= Random channel ∈ C_v
12: end if
13: end for

disjoint sets D_i, i ∈ C. Based on these DDSs, CH sequence
CH_u with time period of n time slots is generated as follows:
CH_u = \{u_0, u_1, … , u_{n-1}\}, where n = \sum_{i=0}^{N-1} k_i and
u_i= channel i such that t ∈ D_i and i ∈ C_u. When i \notin C_u
and t ∈ D_i, a random channel from C_u is chosen.

Example: Consider an CRN network with licensed channel
set C = \{0, 1, 2\} and N = 3. Using Algorithm 1
and 2, SUs namely SU_u and SU_v with C generate their
CH sequences. The corresponding CH sequences and the
rendezvous in an asynchronous environment is shown in
Fig. 2. Similarly for N = 5 i.e. C = \{0, 1, 2, 3, 4\}, the
corresponding DDS and the rendezvous between two CH
sequences are illustrated in Fig. 3.

V. EVALUATION OF METRICS FOR THE PROPOSED METHOD

The performance metrics evaluation for the proposed
scheme is done for DoR, MCTTR and CL by considering
the number of licensed channels as N, and N is a prime number.

Theorem 2: The DDS-CH has the optimal DoR= N.

Proof: Algorithm 1 is used to construct the DDS S
and with the given S, n distinct CH sequence can be
generated using rotate(S, x), \forall x ∈ [0, n-1] as mentioned
in algorithm 2. Let two SUs generate their CH sequences
using rotate(S, y) and rotate(S, w) respectively, where
y, w ∈ [0, n-1]. For an asynchronous environment, let
the CH sequence of the second SU is d time slots delayed
as compared to CH sequence of the first SU. These
two non-synchronized CH sequences are equivalent to the
two synchronized CH sequences generated with rotate(S, y)
and rotate(S, w + dmodn), which can be easily analyzed with
the example illustrated in Fig 4. The n number of \( (n, k)-\)
DDSs rotate(D_i, 0), rotate(D_i, 1), … , rotate(D_i, n − 1),
dedicated to channel i in algorithm 2, would form an CSQ
under Z_n according to corollary 4, \forall i ∈ [0, N − 1]. Now
the two CH sequences constructed with rotate(S, y) and
rotate(S, w) will rendezvous on the channel i at the timeslot
t ∈ rotate(D_i, y) ∩ rotate(D_i, w + dmodn), \forall i ∈ [0, N − 1]
within the time period of CH sequence. As within the
sequence period both SUs would rendezvous on all the \( N \) channels, the proposed method achieves \( N \) DoR.

**Theorem 3:** The DDS-CH has MCTTR \( = 2N^2 + \left\lfloor \frac{N}{2} \right\rfloor \times N - N \).

**Proof:** According to Theorem 2, the proposed rendezvous algorithm guarantees rendezvous at all the \( N \) licensed channels within the CH time period of \( n \) timeslots. In this paper, \( n \) is the total number of elements in all the DDSs of DDS S, which is \( 2N^2 + \left\lfloor \frac{N}{2} \right\rfloor \times N - N \). As a result, in the worst-case scenario, if only one channel is common between the available channel sets of two SUs, the rendezvous occurs on that common channel within a CH period. Hence \( \text{MCTTR} = n = 2N^2 + \left\lfloor \frac{N}{2} \right\rfloor \times N - N \).

**Theorem 4:** The DDS-CH has Channel loading \( \text{CL} = \frac{k}{n} \) or \( 1 \), where \( k \) is the average number of elements present in each DS of the DDS S and \( n = 2N^2 + \left\lfloor \frac{N}{2} \right\rfloor \times N - N \).

**Proof:** Given an DDS S, \( n \) distinct CH sequence can be generated for an SU using \( \text{rotate}(S, x), \forall x \in [0, n - 1] \). In an S each \((n,k)\)-DS is dedicated to a distinct channel. Let an \((n,k)\)-DS, \( D_j \) in S is dedicated to channel \( j \) and \( D_j = \{d_0, d_1, \ldots, d_{k-1}\} \). Hence the \( n \) set of DSs dedicated to same channel are \( \text{rotate}(D_j, 0), \text{rotate}(D_j, 1), \text{rotate}(D_j, 2), \ldots, \text{rotate}(D_j, n - 1) \), out of which exactly \( k \) number of DSs i.e., \( \text{rotate}(D_j, z - d_0), \text{rotate}(D_j, z - d_1), \ldots, \text{rotate}(D_j, z - d_{k-1}) \) would share a common element \( z \), \( \forall z \in D_j \). This proves that among \( n \) CH sequences \( k \) number of sequences are on same channel at the same time slots, which can be seen in the example given in Fig. 4. Hence, the channel loading \( \text{CL} = \frac{k}{n} \).

VI. SIMULATION RESULTS

The proposed DDS-CH is simulated in Matlab, and the performance has been compared with JS [15], CRSEQ [7], T-CH [20] and FRCH [14]. All these methods used for comparison ensure rendezvous in the asynchronous environment within the \( O(N^2) \) time slots. The detail description and comparison of them can be seen in the related work section and in the Table-1 respectively. A cognitive radio network with coexisting PUs and SUs is considered for the simulation environment. Different density of the PUs represented as \( N \) and SUs represented as \( M \) are considered in simulations and it is assumed that all the users in the network are in the communication range of each other. Both the symmetric and asymmetric models are considered, where every SUs act both as transmitter and receiver. \( C \) is considered as the licensed channel set. \( C_u \) and \( C_v \) are the available channel sets of the SU pair where the SU will rendezvous on commonly available channel. The network time environment is divided into time slots each of time duration 200ms [7], [10]. All simulations were conducted in an asynchronous environment, with each SU having a random number of time slots drift in their CH sequence with respect to the other. Hence, each SU with different available channel sets use CH sequence of length \( n \) time slots with random number of slot drift \( d \in \{0, 1, 2, \ldots, n - 1\} \). For the evaluation of each performance metric, 1000 set of simulations have been performed and their average value is considered.

A. Performance metric vs number of licensed channels: The network with \( N \) number of PUs has \( N \) the number of licensed channels, and in this paper, \( N \) is considered a prime number.

**Symmetric Case:** For a homogeneous environment, which is a special case of heterogeneous environment, the network has \( C_u = C_v \). In this case, for better performance, \( N \) is considered as \( |C_u| \) instead of \( N = |C| \).

Fig. 5(a) and (b) demonstrate the performance of ATTR and MCTTR for various \( N \) values ranging from 10 to 50. Here, ATTR represents the average time slots required for the first rendezvous, and MCTTR is the maximum time slots required to achieve rendezvous on all the available channels. Here the value of MCTTR gives the idea about the TTR when only one channel is common between \( C_u \) and \( C_v \) in the asymmetric model. Both the metrics values increase with \( N \) as the length of CH sequence including all \( N \) channels increases, and it takes more time slots for a pair of SUs to be on the same channel. Though the FRCH algorithm has a lower ATTR, it only ensures rendezvous on a channel with high probability. Hence, FRCH scheme fails when that particular channel is unavailable to the SU in the asymmetric case. The JS gives low \( \text{ATTR} > 3N \) for symmetric model, but sometimes the value increases significantly in asymmetric model because of its very large MCTTR value. The T-CH and DDS-CH (with \( k = 2N + \left\lfloor \frac{N}{2} \right\rfloor \)) will show nearly same performance as both have same theoretical MCTTR, where as the proposed DDS-
CH algorithm with \( k = 2N + \left\lceil \frac{N}{2} \right\rceil - 1 \) has better performance with lower MCTTR. In Fig. 5(b), logarithm scale has been used in y axis. Hence, though the MCTTR values of the schemes in the graph seem closer to each other, the difference is quite large.

Asymmetric Case: For an asymmetric model, CRN with \( N=|C|=30 \) is considered. In this case each SU has different available channel set \( (C_u \neq C_v) \), and \( C_r=C_u \cap C_v \) is the common channel set between two SUs.

The Fig. 6(a) shows ATTR performance for various values of \( |C_r| \), varying from 5 to 30. In the asymmetric model, the ATTR of a rendezvous method can be reduced by lowering its MCTTR. The JS method with the rapid rendezvous attribute has a low ATTR, but an irregular fluctuation in ATTR values in the asymmetric model due to its very large MCTTR. Because of the lower MCTTR, DDS-CH outperforms the other schemes in asymmetric model.

In Fig. 6(b), MTTR represents the upper bound value of ATTR for various possible set of \( C_r \), considering all possible slot drift \( d \) between CH sequences of the SU pair. In other words, MTTR graph shows the maximum time required for the first rendezvous for a given \( C_r \). In the worst case, where \( |C_r|=1 \) it takes a longer time to rendezvous in an asynchronous environment termed as MCTTR. The MTTR performance of T-CH and DDS-CH are close to each other since they have nearly the same MCTTR as shown in Table.1, but the difference margin grows as the value of \( N \) increases.

B.Performance metric vs. density of SUs: The channel loading may be validated by taking into account the network’s SU density, and the average collision encountered by a pair of SUs before they successfully exchange control information. A collision is said to be occurred in a time slot when the SUs pair fails to exchange their control information due to the interference from other SUs. Such situation occurs when other SUs hop on the channel, on which the sender-receiver pair are supposed to rendezvous.

For the simulation, a network with 11 licensed channels \((N=11)\) is used, and the average number of collisions is observed for various SUs (M) densities ranging from 2 to 30. The Fig. 7(a) shows the average possible number of collisions a pair of SUs can face before achieving a successful rendezvous. Larger is the channel loading value more is the chances of collision at a particular time slot, and also it is obvious that the collision rate will increase with the increase of SUs density. The CRSEQ, T-CH, and the proposed DDS-CH have the same or nearly same CL value of \( \frac{1}{N} \). Thus these schemes show similar results, whereas JS with CL value \( \frac{1}{3N} \) has slightly larger number of average collisions. For the lower density environment all the schemes show nearly similar performance. However, the difference in their performance is visible in high density of SUs.

The average time required for a successful rendezvous due to the occurrence of collisions is shown in Fig. 7(b). The ATTR of each scheme is affected by the number of collisions and their rendezvous rate. Compared to CRSEQ, the JS has lower ATTR due to its quick rendezvous property. However, DDS-CH outperforms JS because it experiences fewer collisions due to its lower CL in a high-density SU environment. These systems may not be greatly affected by collisions in low SU density settings, but they have a large ATTR with a large \( N \). As a result, with an asymmetric model, we may reduce ATTR by reducing MCTTR using the suggested DDS-CH technique.
VII. CONCLUSION

In this paper, a symmetric channel hopping scheme is suggested to achieve blind rendezvous in an asynchronous environment. In this proposed DDS-CH scheme, the pre-assignment of roles for the SUs as sender and receiver is not required before the rendezvous. The rendezvous technique based on disjoint difference sets ensures a maximum degree of rendezvous($N$), with a lower MCTTR of $O(N^2)$ and an optimum CL value of $\frac{1}{2}$. The rendezvous scheme performs better in both the symmetric and asymmetric models as compared to the recent CH algorithms.

REFERENCES

[1] I. F. Akyildiz, W.-Y. Lee, M. C. Vuran, and S. Mohanty, “Next generation/dynamic spectrum access/cognitive radio wireless networks: A survey,” COMPUTER NETWORKS JOURNAL (ELSEVIER, vol. 50, pp. 2127–2159, 2006.

[2] I. F. Akyildiz, W.-Y. Lee, and K. R. Chowdhury, “Crahns: Cognitive radio ad hoc networks,” Ad Hoc Networks, vol. 7, no. 5, pp. 810 – 836, 2009.

[3] J. Perez-Romero, O. Sallent, R. Agusti, and L. Giupponi, “A novel on-demand cognitive pilot channel enabling dynamic spectrum allocation,” in 2007 2nd IEEE International Symposium on New Frontiers in Dynamic Spectrum Access Networks, 2007, pp. 46–54.

[4] V. Brik, E. Rozner, S. Banerjee, and P. Bahl, “Dsap: a protocol for coordinated spectrum access,” in First IEEE International Symposium on New Frontiers in Dynamic Spectrum Access Networks, 2005, DySPAN 2005., 2005, pp. 611–614.

[5] A. Sampath, H. Dai, H. Zheng, and B. Y. Zhao, “Multi-channel jamming attacks using cognitive radios,” in 2007 16th International Conference on Computer Communications and Networks, 2007, pp. 352–357.

[6] K. Bian and J. Park, “Asynchronous channel hopping for establishing rendezvous in cognitive radio networks,” in 2011 Proceedings IEEE INFOCOM, 2011, pp. 236–240.

[7] J. Shin, D. Yang, and C. Kim, “A channel rendezvous scheme for cognitive radio networks,” Comm. Letters., vol. 14, no. 10, p. 954956, Oct. 2010.

[8] C. J. Liyana Arachchige, S. Venkatesan, and N. Mittal, “An asynchronous neighbor discovery algorithm for cognitive radio networks,” in 2008 3rd IEEE Symposium on New Frontiers in Dynamic Spectrum Access Networks, 2008, pp. 1–5.

[9] K. Bian and J. Park, “Maximizing rendezvous diversity in rendezvous protocols for decentralized cognitive radio networks,” IEEE Transactions on Mobile Computing, vol. 12, no. 7, pp. 1294–1307, 2013.

[10] G. Chang, W. Teng, H. Chen, and J. Sheu, “Novel channel-hopping schemes for cognitive radio networks,” IEEE Transactions on Mobile Computing, vol. 13, no. 2, pp. 407-421, 2014.

[11] N. C. Theis, R. W. Thomas, and L. A. DaSilva, “Rendezvous for cognitive radios,” IEEE Transactions on Mobile Computing, vol. 10, no. 2, pp. 216–227, 2011.

[12] Y. Zhang, Q. Li, G. Yu, and B. Wang, “Etch: Efficient channel hopping for communication rendezvous in dynamic spectrum access networks,” 04 2011, pp. 2471–2479.

[13] K. Bian, J. Park, and R. Chen, “Control channel establishment in cognitive radio networks using channel hopping,” IEEE Journal on Selected Areas in Communications, vol. 29, no. 4, pp. 689–703, 2011.

[14] G. Chang and J. Huang, “A fast rendezvous channel-hopping algorithm for cognitive radio networks,” IEEE Communications Letters, vol. 17, no. 7, pp. 1475–1478, 2013.

[15] Z. Lin, H. Liu, X. Chu, and Y. Leung, “Jump-stay based channel-hopping algorithm with guaranteed rendezvous for cognitive radio networks,” in 2011 Proceedings IEEE INFOCOM, 2011, pp. 2444–2452.

[16] I. Chuang, H. Wu, K. Lee, and Y. Kuo, “Alternate hop-and-wait channel rendezvous method for cognitive radio networks,” in 2013 Proceedings IEEE INFOCOM, 2013, pp. 746–754.

[17] I. Chuang, H. Wu, and Y. Kuo, “A fast blind rendezvous method by alternate hop-and-wait channel hopping in cognitive radio networks,” IEEE Transactions on Mobile Computing, vol. 13, no. 10, pp. 2171–2184, 2014.