Maximizing Rendezvous Diversity in Rendezvous Protocols for Decentralized Cognitive Radio Networks

Kaigui Bian, *Member, IEEE,* and Jung-Min "Jerry" Park, *Senior Member, IEEE*

Abstract—In decentralized *cognitive radio* (CR) networks, establishing a link between a pair of communicating nodes requires that the radios "rendezvous" in a common channel—such a channel is called a *rendezvous channel*—to exchange control information. When unlicensed (secondary) users opportunistically share spectrum with licensed (primary or incumbent) users, a given rendezvous channel may become unavailable due to the appearance of licensed user signals. Ideally, every node pair should be able to rendezvous in every available channel (i.e., maximize the *rendezvous diversity*) so that the possibility of rendezvous failures is minimized. Channel hopping (CH) protocols have been proposed previously for establishing pairwise rendezvous. Some of them enable pairwise rendezvous over all channels but require global clock synchronization, which may be very difficult to achieve in decentralized networks. Maximizing the pairwise rendezvous diversity in decentralized CR networks is a very challenging problem. In this paper, we present a systematic approach for designing CH protocols that maximize the rendezvous diversity of any node pair in decentralized CR networks. The resulting protocols are resistant to rendezvous failures caused by the appearance of primary user (PU) signals and do not require clock synchronization. The proposed approach, called *asynchronous channel hopping (ACH)*, has two noteworthy features: (1) any pair of channel hopping nodes are able to rendezvous on *every* channel so that the rendezvous process is robust to disruptions caused by the appearance of primary user signals; and (2) an upper bounded time-to-rendezvous is guaranteed between the two nodes even if their clocks are *asynchronous*. We propose two optimal ACH designs that maximize the rendezvous diversity between any pair of nodes and show their rendezvous performance via analytical and simulation results.

Index Terms—Cognitive radio, maximum rendezvous diversity, asynchronous channel hopping.

1 Introduction

Link establishment in decentralized *cognitive radio* (CR) networks requires two communicating nodes to "rendezvous"—i.e., find each other on a common channel (referred to as a *rendezvous channel*) to exchange control information—prior to initiating data communications. In the opportunistic spectrum sharing (OSS) paradigm, unlicensed or secondary user (SU) nodes equipped with CRs are required to refrain from transmitting in the channels where licensed or primary user (PU) signals are detected. The use of a *single* common control channel simplifies the rendezvous process but it creates a single point of failure—it may become unavailable due to the appearance of licensed user signals. Ideally, we want to maximize the *rendezvous diversity* between two secondary nodes (trying to establish a link) so that the rendezvous points are spread out over *all* available channels. Maximizing rendezvous diversity minimizes the risk of rendezvous failures due to the appearance of primary user signals in the rendezvous channels.

In decentralized networks, such as ad hoc networks, precise clock synchronization among nodes may not be feasible. Asynchronous channel hopping (CH) protocols have been proposed to enable *asynchronous rendezvous* between any pair of nodes (i.e., successful rendezvous without requiring clock synchronization between nodes). These asynchronous CH schemes can be classified into two categories.

- *Sequence-based* CH: Every node generates its CH sequence according to a sequence-generation algorithm [6], [7], [10], [28]. However, sequence-based CH schemes only support a very limited number of distinct rendezvous channels between any two nodes. For instance, the asynchronous CH scheme in [6] only supports two pairwise rendezvous channels. This drawback may be very problematic for secondary nodes accessing spectrum opportunistically because the presence of primary user signals in one or more of the few rendezvous channels can severely limit the secondary nodes' ability to establish links.
- *Random* CH: Each node hops from one channel to another randomly at a certain hopping rate. All nodes may have the same hopping rate [23] (e.g., channel hopping occurs at the beginning of every timeslot), or the uncoordinated sender and receiver nodes have different hopping rates [22], [26]. In random CH, the latency before successful rendezvous (i.e., time-to-rendezvous (TTR)) has no upper bound, and, as a result, two secondary nodes may experience unreasonably long delays before achieving rendezvous.

We coin the term *asynchronous rendezvous* problem in

- *K. Bian is with Institute of Network Computing and Information Systems (NCIS), School of EECS, Peking University, Beijing, China 100871. E-mail: bkg@pku.edu.cn*
- *J. Park is with the Bradley Department of Electrical and Computer Engineering, Virginia Tech, Blacksburg, VA, USA 24061. E-mail: jungmin@vt.edu*
- *A preliminary version of portions of this material has appeared in [5].*

the context of decentralized CR networks to denote the following problem: *How can two channel hopping secondary nodes, without clock synchronization, achieve maximum "rendezvous diversity" with a guaranteed upper bounded TTR?* Here, rendezvous diversity denotes the number of distinct rendezvous channels. Note that a solution to the asynchronous rendezvous problem satisfies the following two requirements: (1) enable pairwise rendezvous between any two nodes on *every* available channel to maximize rendezvous diversity; and (2) ensure that the TTR is *upper bounded*. Satisfying the first requirement ensures that rendezvous failures due to the presence of primary user signals are minimized. The second requirement enables the latency of link establishment (between a pair of secondary user nodes) to be upper bounded.

This paper presents a systematic approach for designing asynchronous channel hopping (ACH) protocols that address the asynchronous rendezvous problem. The primary challenge in designing such ACH protocols is devising a way to generate CH sequences that maximize rendezvous diversity. In this paper, we propose two *optimal* ACH designs that achieve *maximum* rendezvous diversity for a given CH sequence period.

- We propose an *asymmetric* ACH system that uses an array-based quorum system. This approach generates a significantly greater number of CH sequences than the approach using Latin squares; the latter approach was used in [7]. In an asymmetric ACH system, every node needs to be pre-assigned a role as either a sender or a receiver prior to rendezvous—a la Bluetooth pairing. The sender and receiver nodes use *different* methods to generate their CH sequences for achieving rendezvous.
- By leveraging the bit-sequence design technique, we devised a *symmetric* ACH system that does not require the pre-assignment of the sender/receiver role for each node. In a symmetric ACH system, every node follows the *same* method for generating its CH sequence.
- We provide both analytical and simulation results to compare the performance of existing rendezvous protocols and the proposed ACH systems under various real-world conditions, such as random clock drift, dynamic primary user traffic, spectrum sensing errors, etc.

This paper focuses exclusively on the problem of enabling a pair of nodes to rendezvous in a common channel for the purpose of link establishment in decentralized networks. The follow-on tasks after initial rendezvous—such as handshake [1], channel contention procedure [14], and data packet transmission [25]—are outside the scope of this paper.

The rest of this paper is organized as follows. We discuss related work in Section 2. We provide the system model and formulate the problem in Section 3. In Sections 4 and 5, we describe asymmetric and symmetric ACH systems, respectively. We analytically compare the proposed ACH system designs with existing CH schemes in Section 6. We evaluate our proposed ACH schemes using simulation results in Section 7, and conclude the paper in Section 8.

2 Related Work

2.1 Control Channel based Rendezvous

Static control channel. The static control channel approach significantly simplifies the link establishment process. Most existing work on CR MAC protocols assumes a global control channel for all network nodes [16], [21], [24], or a local control channel that is established for a group/cluster of network nodes [8], [31]. In opportunistic spectrum sharing, the availability of any channel, including the common control channel, cannot be guaranteed due to the unpredictable appearance of primary user signals. Thus, maintaining a common control channel is very difficult, and may be impossible in some cases.

Dynamic control channel. In CR networks that employ a dynamic control channel approach, link establishment between two nodes is carried out in two phases: (1) initial rendezvous phase and (2) link establishment phase. In the initial rendezvous phase, a pair of nodes (that needs to establish a link) "rendezvous" at a common frequency channel in order to exchange *control* messages. For example, two nodes may exchange information needed to share a common channel hopping sequence (if they execute common channel hopping) or information on a dedicated out-of-band control channel (if out-of-band signaling is used). In the link establishment phase, the two nodes establish a channel for exchanging *data* packets using the control information exchanged during the initial rendezvous phase. A number of dynamic control channel schemes have been proposed in the literature [9], [11], [18], [19].

The C-MAC [9] scheme requires the synchronization of the nodes' clocks and efficient dissemination of spectrum availability information, which may be difficult to implement in decentralized networks. In [11], swarm intelligence is used as a way to determine the channels that can serve as backup control channels. A commonly shared, ordered channel list is collectively determined and maintained by the CR nodes using the sensing results of those nodes [19]. This scheme requires the exchange of a common channel list among neighbors. In [18], a cluster-based channel hopping scheme is proposed. The cluster head of each cluster dynamically determines the hopping sequences that are used by the nodes in its cluster. The hopping sequences are designed such that any two nodes in the same cluster are guaranteed to rendezvous in a channel during a designated time slot.

The dynamic control channel approach has two drawbacks when compared to the static control channel approach. First, the dynamic control channel approach is more complex to implement and incurs greater overhead (e.g., overhead required for network synchronization). Second, it may induce greater latency for link establishment in certain scenarios. When the dynamic control channel approach is employed, control information is exchanged only during rendezvous timeslots, and it may take several channel hopping periods to exchange all of the control information.

Virtual network coded control channel. A virtual control channel scheme is proposed for the efficient dissemination of control information to all network nodes [2], [4]. Nodes visit channels in a pseudo-random manner and exchange control information by means of network coding when they meet in a particular channel.

It should be noted that the dynamic control channel schemes proposed in this paper and those of related works do not dictate how the radios coordinate the exchange of data packets once the control messages have been exchanged (via rendezvous).

2.2 Channel Hopping based Rendezvous

There exists a body of work that proposes using channel hopping protocols to enable rendezvous (for link establishment) in multi-channel and CR networks without relying on a control channel.

Sequence-based vs. random CH schemes. As discussed before, sequence-based CH schemes (e.g., [6], [10]) are able to guarantee an upper bound for the TTR between two nodes, but enable rendezvous in only a few distinct channels per pair of nodes. In these protocols, the presence of primary user signals in one or more of the rendezvous channels can drastically lower the chance of successful rendezvous.

In [7], Bian et al. proposed using an m by m Latin square to generate the CH sequence for a sequence-based CH scheme that ensures the maximum rendezvous diversity. In this paper, we present an asymmetric ACH scheme that uses an m by n array to generate the CH sequence. Since the number of distinct m by m Latin squares for a given m is much smaller than the number of distinct m by n arrays (where $n \geq m$), our approach in this paper can create a significantly greater number of distinct CH sequences than the approach using Latin squares. Note that insufficiently large number of CH sequences can result in the *rendezvous convergence problem* [6].

The random CH scheme enables every node pair to achieve rendezvous over any channel with a positive probability. However, due to the random channel-visiting strategy, there is no guaranteed upper bound for the TTR of a node pair.

Symmetric and asymmetric CH schemes. Since the symmetric CH design is independent of the assumption that every node has a priori knowledge of its role as either a sender or a receiver, it has a wider variety of applications than the asymmetric design. Note that a CH scheme can be either a symmetric or an asymmetric design depending on whether the sender and receiver employ different CH sequence generation algorithms or hop at different rates. For example, if every node has the same hopping rate [23], [28] or the same sequence-generation algorithm [3], [6], [10], [25], then the CH scheme is a symmetric design. In contrast, if the sender and receiver have different hopping rates [26] or employ different sequence-generation algorithms [5], then the CH scheme is an asymmetric design.

3 Problem Statement

3.1 Technical Background

We provide a brief description of two properties of quorum systems [17], [20] to facilitate the readers' understanding of our approach for designing ACH systems.

3.1.1 Intersection property

Every quorum system satisfies the *intersection property* which is described below.

***Definition** 1:* Given a finite universal set $U = \mathbf{Z}_n = \{0, ..., n-1\}$ of n elements, a quorum system Q under U is a collection of non-empty subsets of U, which satisfies the following *intersection property*: $p \cap q \neq \emptyset, \forall p, q \in Q$. Each $p \in Q$ (a subset of U) is called a *quorum*. Here, $\mathbf{Z}_n$ denotes the set of non-negative integers less than n.
For example, $Q = \{\{0, 1\}, \{1, 2\}\}$ is a quorum system over $U = \{0, 1, 2\}$. In Q, two quorums, $\{0, 1\}$ and $\{1, 2\}$, "intersect" with each other—have a common element, 1.

3.1.2 Rotation closure property

Here, we introduce the concepts of *cyclic rotation* and the *rotation closure property* in the context of quorum systems.

Given a non-negative integer k and a quorum q in a quorum system Q under the universal set $U = \{0, ..., n-1\}$, we use $rotate(q, k)$ to denote a *cyclic rotation* of quorum q:

$$rotate(q, k) = \{(i + k) \bmod n | i \in q\}.$$

For example, suppose we have a quorum system $Q = \{\{0, 1\}, \{1, 2\}\}$ under $U = \mathbf{Z}_3$. For quorum $q = \{0, 1\}$ and $k = 2$, $rotate(q, k) = \{2, 0\}$. Now let us define the *rotation closure property* of quorum systems.

***Definition** 2:* A quorum system Q over $U = \mathbf{Z}_n$ is said to have the rotation closure property if the following holds:

$$\forall p, q \in Q, \forall k \in [0, n-1], rotate(p, k) \cap q \neq \emptyset.$$

3.2 System Model

We assume an opportunistic spectrum sharing environment, where secondary users equipped with CRs are operating over N orthogonal frequency channels that are licensed to the primary user. Channels are labeled as $0, 1, ..., N-1$. Each node is equipped with a single half-duplex transceiver. Hence, a node can only listen to or transmit over one channel at a time. A packet sent over a channel can be heard by any node within the transmission range of the transmitting node.

3.2.1 Time-slotted system

We consider a time-slotted communication system, where a global system clock exists. The local clock of each node may be synchronized to the global clock or may differ with the global clock by a certain amount of clock drift. Hence, there are two types of clock drift: the clock drift between global and local clocks, and the clock drift between local clocks. A network node is assumed to be capable of hopping between different channels according to a channel hopping sequence and its local clock. A packet can be exchanged between two nodes if they hop onto the same channel in the same timeslot. We assume that one timeslot is long enough to exchange multiple packets. In decentralized CR networks, it is difficult to maintain very tight synchronization among the nodes' clocks.

3.2.2 Channel hopping (CH) sequence

A CH sequence determines the order with which a node visits all available channels. We represent a CH sequence u of period T as a set of channel indexes:

$$u = \{u_0, u_1, ..., u_i, ..., u_{T-1}\},$$

where $u_i \in [0, N-1]$ represents the channel index of sequence u in the i^{th} timeslot of a CH period.

Given two CH sequences of period T, u and v, if $\exists\ i \in [0, T-1]\ s.t.\ u_i = v_i = h$, where $h \in [0, N-1]$, we say that u and v *rendezvous* in the i^{th} timeslot on channel h. The i^{th} timeslot is called the *rendezvous slot* and channel h is called the *rendezvous channel* between u and v. Given N channels, let $\mathcal{C}(u, v)$ denote the *number of rendezvous channels* between two CH sequences u and v, and $\mathcal{C}(u, v) \in [0, N]$.

3.2.3 Coexistence with primary users

Secondary user signals coexist with primary user signals in the same channels on an interference-free basis. A channel hopping secondary node employs spectrum sensing techniques to detect the primary user signals on a channel in each timeslot, and thus avoids hopping to or transmitting packets on a channel where primary user signals are detected.

3.2.4 Asynchronous channel hopping system

We extended the concept of cyclic rotation so that it is applicable to CH sequences. Given a CH sequence u, we use $rotate(u, k)$ to denote a *cyclic rotation* of CH sequence u by k timeslots, i.e.,

$$rotate(u, k) = \{v_0, ..., v_j, ..., v_{T-1}\},$$

where $v_j = u_{(j+k) \bmod T}, j \in [0, T-1]$ and k is a non-negative integer. For example, given $u = \{0, 1, 2\}$ and $T = 3$, $rotate(u, 2) = \{2, 0, 1\}$.

In [6], a channel hopping (CH) system of period T is defined as a set of CH sequences of period T. Here, we formally define an *asynchronous channel hopping* (ACH) system below.

***Definition** 3:* An asynchronous channel hopping (ACH) system of period T is a CH system of period T, which satisfies the *rotation closure property*: it consists of CH sequences such that any two distinct CH sequences u and v satisfy the following inequality:

$$\forall k, l \in [0, T-1], \mathcal{C}(rotate(u, k), rotate(v, l)) \geq m,$$

where the positive integer m is *the degree of overlapping* of the ACH system.

Using two CH sequences of an ACH system, two channel hopping nodes can rendezvous with each other on at least m distinct rendezvous channels even if their clocks are asynchronous. In addition to the degree of overlapping, we introduce two metrics for evaluating ACH systems.

Minimum Rendezvous Probability (MRP). The MRP of an ACH system is defined as the lower bound of the probability that a pair of CH sequences from an ACH system will rendezvous in a given timeslot. In an ACH system H with period T and a degree of overlapping value of m, two sequences are guaranteed to rendezvous at least m times on m distinct rendezvous channels during a sequence period. Thus, the MRP for the ACH system H is $\gamma(H) = \frac{m}{T}$.

Average Time-to-Rendezvous (ATTR). The ATTR for a given ACH system is computed by taking the mean of TTR values between two channel hopping sequences randomly selected from the ACH system when their clocks differ by a random number of timeslots. The ATTR value has a direct impact on the medium access delay of MAC protocols that utilize CH-based rendezvous in time-asynchronous environments.

3.3 The Optimal ACH System Design Problem

To minimize the chance of rendezvous failure due to the presence of primary user signals in the rendezvous channels, one needs to maximize the number of distinct rendezvous channels. We would ideally want an ACH system of period T to guarantee N distinct rendezvous channels between any two CH sequences for every T consecutive timeslots when N channels are available—i.e., the ACH system's degree of overlapping is $m = N$. The MRP of such a system is $\frac{N}{T}$, and maximizing the MRP for a fixed value of N is equivalent to minimizing the sequence period, T. We define an *optimal* ACH system as a system that has a degree of overlapping N and also the maximum possible MRP value. To devise an optimal ACH system, one has to solve the following problem:

***Problem** 1:* Given an ACH system H of period T and assuming there are N channels,

$$\begin{aligned} &\textit{minimize} && T, \\ &\textit{subject to} && \forall u, v \in H, u \neq v, \forall k, l \in [0, T-1], \\ & && \mathcal{C}(rotate(u, k), rotate(v, l)) = N. \end{aligned}$$

The lower bound for T is given by the following theorem.

***Theorem** 1:* In an ACH system H whose degree of overlapping is N, the period of any CH sequence must be N^2 or greater.

Proof: Suppose we have two sequences, u and v, from an ACH system of period T. Let $k_{u,h}$ denote the number of timeslots in sequence u that are assigned with the channel index $h \in [0, N-1]$. Then, we can express the period length T as

$$T = \sum_{h=0}^{N-1} k_{u,h} = \sum_{h=0}^{N-1} k_{v,h} = \sum_{h=0}^{N-1} \left(\frac{k_{u,h} + k_{v,h}}{2} \right). \quad (1)$$

Without loss of generality, we fix u and cyclically rotate v by $l, l = 0, 1, ..., T-1$. Let us count the total accumulated number of rendezvous between u and $rotate(v,l)$ as l is incremented from 0 to $T-1$. Since $\mathcal{C}(u, rotate(v,l)) = N$, for any channel $h \in [0, N-1]$, the total number of rendezvous that involve a given timeslot, $v_i = h$, in v is $k_{u,h}$. Since there are $k_{v,h}$ timeslots in v that are assigned channel h, the total accumulated number of rendezvous between u and $rotate(v,l)$, as l is incremented from 0 to $T-1$, in which the rendezvous channel is h is $k_{u,h} \cdot k_{v,h}$.

By definition, $\mathcal{C}(u, rotate(v,l)) = N$ for any amount of cyclic rotation to v. This means that u and $rotate(v,l)$ must rendezvous in channel h at least once. Hence, the total accumulated number of rendezvous in channel h (as l is incremented from 0 to $T-1$) is at least T. Thus, we have $k_{u,h} \cdot k_{v,h} \geq T$.

Since $\left(\frac{k_{u,h}+k_{v,h}}{2}\right)^2 \geq k_{u,h} \cdot k_{v,h}$, we can readily derive

$$\frac{k_{u,h} + k_{v,h}}{2} \geq \sqrt{T}. \quad (2)$$

Combining (1) and (2), we get

$$T = \sum_{h=0}^{N-1} \left(\frac{k_{u,h} + k_{v,h}}{2} \right) \geq N\sqrt{T}.$$

Therefore, we conclude that $T \geq N^2$. □

In the next two sections, we describe two optimal ACH designs in which the period of the CH sequences is $O(N^2)$.

4 Asymmetric ACH Systems

4.1 Relationship between Quorum Systems and ACH Systems

By utilizing the intersection property of quorum systems, a set of CH sequences can be generated such that any pair selected from the set is guaranteed to rendezvous when the boundaries of the two selected CH sequences are synchronized [6]. However, merely relying on the intersection property of quorum systems is *not* sufficient to guarantee rendezvous when the two CH sequences are *asynchronous* (i.e., the boundaries of the two CH sequences are misaligned).

Our research findings indicate that quorum systems that satisfy the *rotation closure property* can guarantee pairwise rendezvous between two asynchronous CH sequences. An example is given below.

- Suppose we have a quorum system Q, under the universal set $U = \mathbf{Z}_n$, that satisfies the rotation closure property. Two quorums $p, q \in Q$ must have an intersection even if an arbitrary amount of cyclic rotation is applied to either one of them.
- Suppose we have an ACH system that includes two CH sequences of period $T = n$, u and v. Suppose we assign the channel index $h \in [0, N-1]$ to u and v in the following way: $u_i = h$ if $i \in p$ and $v_i = h$ if $i \in q$, where $i \in [0, T-1]$; other timeslots are assigned with a randomly selected channel other than h. The rotation closure property of Q guarantees that the two CH sequences rendezvous in channel h at least once even if an arbitrary amount of cyclic rotation is applied to either of the CH sequences.

The *array-based* quorum system has the rotation closure property [17], and this category of quorum systems can be constructed using a two-dimensional array. Suppose that we have an $r \times l$ array, $A[\cdot][\cdot]$. The universal set U is $\{0, ..., r{\cdot}l{-}1\}$, and there is a one-to-one mapping between the array element $A[i][j]$ and the element $(i \cdot l + j)$ in the universal set, where $i \in [0, r-1], j \in [0, l-1]$. We use the following method to construct an array-based quorum system: a quorum is constructed by picking a full column of elements plus one element from every other column in the array. We refer to the full column of elements as the *column* of the quorum and the l elements that span all the columns as the *span* of the quorum. Note that the column and the span of the same quorum have a common element. An example of an array-based quorum system is illustrated in Figure 1. In the example, an array-based quorum includes a column of elements (squares colored in grey) and a span of elements (squares labeled with a 'S'). A span and a column have one element in common which is not counted twice. One quorum's column must have an intersection with a span of another. For instance, in the figure, p's column intersects q's span at the square that is located in the first row of the third column. Similarly, p's span intersects q's column at the square located in the first row of the sixth column. Note that performing cyclic rotation by one slot to either of the array-based quorums yields a third array-based quorum (e.g., $q' = rotate(q, 1)$), and the intersection property is maintained after the cyclic rotation. The elements in common between p and q' are marked with circle markings, and those between q and q' are marked with square markings. Given the two example quorums, p and q, p's column must intersect with q's span, even if a "cyclic rotation" is applied to either p or q. From this example, we can see that the array-based quorum system satisfies the rotation closure property. The cyclic rotation of q generates another quorum, q', of the same type.

4.2 Constructing Asymmetric Designs

In an asymmetric design, the sender and the receiver use different approaches to generate their respective CH

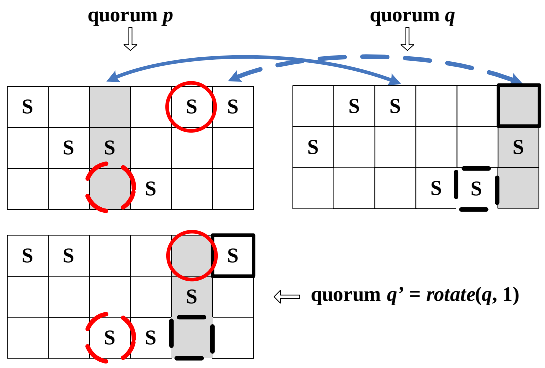

Fig. 1. Two quorums, p and q, each with 8 elements, in an array-based quorum system. Both quorums are constructed using a 3×6 array.

sequences. The sender's and receiver's procedures for constructing the sequences are described below, and the corresponding pseudo codes are given in Algorithms 1 and 2, respectively.

1) The sender constructs an $N \times N$ array, $S[\cdot][\cdot]$.
   a) The sender randomly assigns N channel indexes to the N columns of S such that each channel index is assigned to exactly one column. All array elements in the same column are assigned the same channel index.
   b) Let u denote the sender's CH sequence. The sender generates its CH sequence in the following way: $u_{i \cdot N+j} = S[i][j]$, where $i, j \in [0, N-1]$.
2) The receiver constructs an $N \times N$ array, $R[\cdot][\cdot]$. Using $R[\cdot][\cdot]$, an array-based quorum system can be constructed, from which the receiver chooses N quorums that have N *disjoint* spans. Let s_k denote the span that contains the array element $R[k][0]$ where $k \in [0, N-1]$.
   a) The receiver randomly assigns N channel indexes to the N spans such that each channel index is assigned to exactly one span, and all array elements belonging to the same span are assigned the same channel index. Let h'_k denote the channel index assigned to the span s_k.
   b) Let v denote the receiver's CH sequence. The receiver derives every channel index of its CH sequence in the following way: $v_{i \cdot N+j} = h'_k$ if $R[i][j] \in s_k$ where $i, j \in [0, N-1]$.

We refer to S and R as the *sequence array of the sender* and the *sequence array of the receiver*, respectively. The sender's CH sequence, u, is referred to as a *column-based* sequence because the channel indexes are assigned to the sender's sequence array elements in a column-wise manner. Similarly, the receiver's CH sequence, v, is referred to as a *span-based* sequence because the channel indexes are assigned to the receiver's sequence array elements in a span-wise manner. Figure 2 illustrates an example ACH system built using a 3×3 array when $N = 3$ and u^* defines the sender's CH sequence after a cyclic rotation by $(2 + \delta)$. The sender's clock is ahead of the receiver's clock by $(2 + \delta)$ timeslots where $\delta \leq 1/2$. The sequence arrays of the sender and the receiver that generated the respective CH sequences are shown. It can be easily seen that the two CH sequences have N distinct rendezvous channels despite the clock drift and that the overlap duration in each rendezvous channel is $(1 - \delta)$.

Algorithm 1 The Sender Sequence Construction.

Input: the total number of channels N, and the sender's sequence array $S[\cdot][\cdot]$
Output: the sender sequence u.

$\{h_0, h_1, ..., h_{N-1}\} \leftarrow$ a permutation of $\{0, 1, ..., N-1\}$.
Assign h_j to array elements in the j-th column of S, where $j \in [0, N-1]$.
for $j = 0$ to $N - 1$ **do**
  for $i = 0$ to $N - 1$ **do**
    $u_{i \cdot N+j} = S[i][j]$.
  end for
end for

Algorithm 2 The Receiver Sequence Construction.

Input: the total number of channels N, and the receiver's sequence array $R[\cdot][\cdot]$.
Output: the receiver sequence v.

$\{h'_0, h'_1, ..., h'_{N-1}\} \leftarrow$ a permutation of $\{0, 1, ..., N-1\}$.
Choose N array-based quorums that have N disjoint spans, $s_0, ..., s_k, ..., s_{N-1}$, such that $R[k][0] \in s_k$.
for $i = 0$ to $N - 1$ **do**
  for $j = 0$ to $N - 1$ **do**
    if $R[i][j] \in s_k$ **then**
      $v_{i \cdot N+j} = h'_k$.
    end if
  end for
end for

Theorem 2: The respective CH sequences generated by Algorithms 1 and 2 have a period N^2, and the two CH sequences form an ACH system H that has a degree of overlapping of N.

Proof: For the sake of our discussions, suppose the length of a timeslot is 1. As previously mentioned, Algorithms 1 and 2 generate the sender sequence u and receiver sequence v, respectively, each using an $N \times N$ array. Without loss of generality, let us suppose the sender's clock is i slots ahead of the receiver's clock, where $i \in [0, T-1]$ is an arbitrary integer. With respect to the receiver's clock, sender's sequence u is equivalent to $rotate(u, i)$, and the operation $rotate(u, i)$ yields a new CH sequence u^*. In the sequence array of u^*, all the elements in each column are assigned the same channel index, and thus N columns contain N channel indexes. If the sequence array of u^* and the sequence array of the receiver (see the example in Figure 2) are superimposed on top of one another, every column of the former overlaps with one of the spans of the latter. One can readily observe that there are N overlaps of N distinct channel indexes. An overlap occurs when

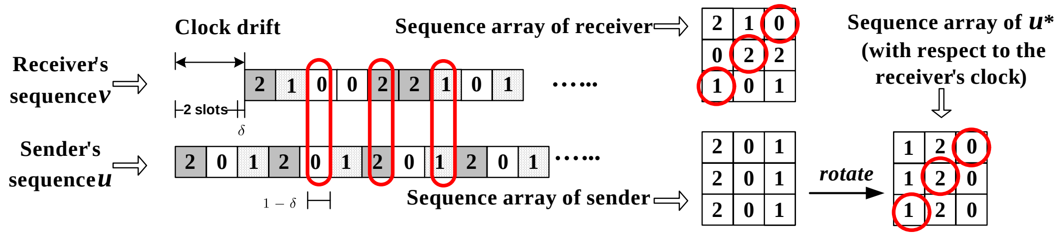

Fig. 2. An asymmetric design of an optimal ACH system when $N = 3$.

the same channel index appears in the same position (i.e., same row and column positions) in the two arrays. Thus, u^* and v have N distinct rendezvous channels within a sequence period $T = N^2$. Since $i \in [0, T-1]$ is an arbitrary integer, we can conclude that $\forall k, l \in [0, T-1], \mathcal{C}(rotate(u,k), rotate(v,l)) = N$. According to Definition 3, $H = \{u, v\}$ is an ACH system with a degree of overlapping of N. □

An ACH system generated by the asymmetric design has an MRP value of $1/N$. Given a random clock drift between two channel hopping nodes, this asymmetric design guarantees that the two nodes achieve rendezvous over N channels within N^2 timeslots, and thus the ATTR of the asymmetric design is $O(N)$. Since a sender sequence and a receiver sequence form an ACH system, they can be used by a pair of sender and receiver nodes to solve the asynchronous rendezvous problem in the context of opportunistic spectrum sharing. Note that the ability to rendezvous in N distinct channels significantly improves rendezvous robustness—it minimizes the likelihood of rendezvous failure due to the presence of primary user signals in the channels.

5 Symmetric ACH Systems

5.1 Limitations of Asymmetric ACH Systems

An ACH system created using the asymmetric design approach requires each radio to have a pre-assigned role as either a sender or a receiver. Although such a requirement may be acceptable in some scenarios (e.g., half-duplex communication systems or in Bluetooth pairing), it may not be acceptable in systems where a radio's role as sender/receiver is not pre-assigned. Designing time-asynchronous rendezvous protocols that are symmetric is a challenging problem. In the next subsection, we will explain how to devise symmetric ACH systems.

5.2 Construction of Symmetric ACH Systems

To construct a symmetric ACH system, we require every node to generate its CH sequence from a bit sequence that it possesses. Specifically, based on certain bit sequence design techniques, two channel hopping nodes are able to construct two bit sequences that have at least one distinct bit. Then, every node replaces any bit of "1" (or "0") in its bit sequence with a column-based (or span-based) sequence. The resulting CH sequences of the two nodes would achieve the maximum rendezvous diversity owing to the maximum number of rendezvous channels guaranteed between a column-based sequence and a span-based sequence. Note that every node follows the same method of constructing its CH sequence, which belongs to a symmetric ACH system.

We define a *bit sequence* as a sequence of bits where each bit takes either a value of 0 or 1. Note that a bit sequence is a special case of a CH sequence such that the channel indexes are chosen from $\{0, 1\}$. Hence, the cyclic rotation operation is also applicable to a bit sequence. Let us introduce the *bit sequence design problem* which is defined as follows: Enable two nodes to independently construct two *distinct* bit sequences. Note that the sequences resulting from cyclic rotations of a sequence are *not* considered to be distinct with respect to each other and the original sequence.

Assume that a *unique* n-bit sequence is assigned to each network node and that n is a system parameter. For example, the unique 48-bit ($n = 48$) sequence of a node can be the MAC address of the node's network interface. We refer to such a bit sequence as a node's *ID sequence*, and let α denote the n-bit ID sequence of node x. Now, let us define an n-bit sequence that contains zeros only:

$$z = \{z_0, ..., z_{n-1}\}, \forall i \in [0, n-1], z_i = 0,$$

and an n-bit sequence that contains ones only

$$o = \{o_0, ..., o_{n-1}\}, \forall i \in [0, n-1], o_i = 1.$$

Let $||$ denote a concatenation operator that concatenates two bit or CH sequences. We define the *expanded ID sequence* of node x as the concatenation of three n-bit sequences—α, z and o—as given in the following equation:

$$\mathbf{a} = \{a_0, ..., a_{3n-1}\} = \alpha || z || o.$$

According to the following lemma, two expanded ID sequences generated from two different ID sequences are guaranteed to be distinct.

***Lemma** 1:* Given any two n-bit sequences $\alpha = \{\alpha_0, \ldots, \alpha_{n-1}\}$ and $\beta = \{\beta_0, \ldots, \beta_{n-1}\}$, let $\mathbf{a} = \alpha || z || o$ and $\mathbf{b} = \beta || z || o$, where z is an n-bit sequence composed of only zeros and o is an n-bit sequence composed of only ones.

If $\alpha \neq \beta$, then

$$\mathbf{a} \neq rotate(\mathbf{b}, k), \forall k \in (0, 3n-1].$$

Proof: We prove the above statement by considering three possible scenarios, and showing, in each scenario, that a bit in $\mathbf{a}$ and another bit in $rotate(\mathbf{b}, k)$ have

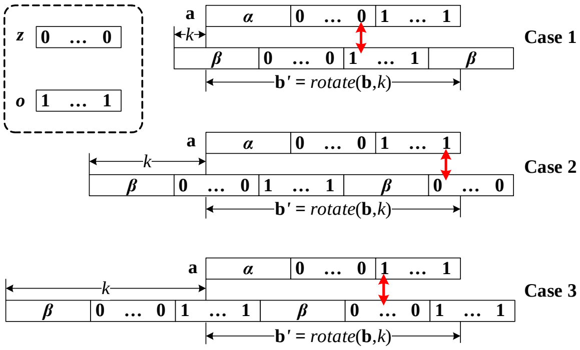

Fig. 3. Illustration of the three cases in Lemma 1's proof.

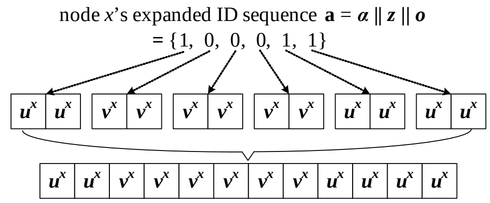

Fig. 4. Construction of a CH sequence for the symmetric ACH scheme.

different values although the two bits are in the same position within the respective expanded ID sequences. This is sufficient to prove that the two expanded ID sequences are not equal. Let $\mathbf{b}' = rotate(\mathbf{b}, k)$.

Case 1: $k \in (0, n]$. Since $\mathbf{a} = \alpha||z||o$ and $\mathbf{b} = \beta||z||o$, $a_{2n-1} = 0$ and $b'_{2n-1} = 1$ (as indicated by the arrow in the Case 1 illustration of Figure 3).

Case 2: $k \in (n, 2n]$. Since $\mathbf{a} = \alpha||z||o$ and $\mathbf{b} = \beta||z||o$, $a_{3n-1} = 1$ and $b'_{3n-1} = 0$ (as indicated by the arrow in the Case 2 illustration of Figure 3).

Case 3: $k \in (2n, 3n-1]$. Since $\mathbf{a} = \alpha||z||o$ and $\mathbf{b} = \beta||z||o$, $a_{2n} = 1$ and $b'_{2n} = 0$ (as indicated by the arrow in the Case 3 illustration of Figure 3).

Thus, we conclude that $\mathbf{a} \neq rotate(\mathbf{b}, k), \forall k \in (0, 3n-1]$. □

We now describe how to generate the CH sequences of a *symmetric* ACH system. Suppose that the number of available channels is N.

1) Suppose node x has a unique ID sequence, α, that contains n bits, and its expanded ID sequence is $\mathbf{a} = \alpha||z||o$, which has $3n$ bits. Lemma 1, ensures that two nodes that have two distinct ID sequences must have two distinct expanded ID sequences.
2) Using the procedure outlined in Section 4 for constructing asymmetric ACH systems, node x that needs to establish rendezvous with neighboring nodes generates its own asymmetric ACH system (including a column-based sequence u^x and a span-based sequence v^x) *independently* of other nodes. Note that different nodes may generate different asymmetric ACH systems.
3) Node x constructs its CH sequence by expanding each bit in $\mathbf{a}$ to a *frame*, which is mapped to a certain sequence as described below. From $i = 0$ to $i = 3n-1$, each bit of $\mathbf{a}$ is expanded in the following way:
   a) if $a_i = 1$, then the $(i+1)^{th}$ frame of node x's CH sequence is the sequence $(u^x||u^x)$;
   b) if $a_i = 0$, then the $(i+1)^{th}$ frame of node x's CH sequence is the sequence $(v^x||v^x)$.

   Each frame is either a sequence $(u^x||u^x)$ or $(v^x||v^x)$, and thus it includes $2N^2$ slots.

A simple example of the CH sequence construction method is illustrated in Figure 4. In the figure, node x's ID sequence is $\alpha = \{1, 0\}$, $n = 2$, $z = \{0, 0\}$ and $o = \{1, 1\}$. Node x's column- and span-based sequences are u^x and v^x, respectively. Every node has a distinct expanded ID sequence. In other words, every pair of *distinct* expanded ID sequences differ by at least one bit. For the sake of this discussion, let us say that they differ by a single bit at the i-th bit position. This implies that after the two expanded ID sequences are expanded, the i-th frame in each of the two resulting CH sequences will be composed of different types of sequences—viz, one frame is composed of a column-based sequence while the other frame is composed of a span-based sequence. As stated previously, a column-based sequence and a span-based sequence always overlap in N distinct channels. Hence, any pair of nodes that construct CH sequences using the procedure described above would be able to rendezvous in the maximum number of distinct channels, which is N. To illustrate this point further, an example is given below.

Suppose node x generates a column-based sequence u^x and a span-based sequence v^x. Likewise, node y generates u^y and v^y. Moreover, suppose nodes x and y have expanded ID sequences 100011 and 000011, respectively. Following the aforementioned procedure, the two nodes generate their CH sequences respectively as

$$(u^x||u^x)||(v^x||v^x)||(v^x||v^x)||(v^x||v^x)||(u^x||u^x)||(u^x||u^x)$$

and

$$(v^y||v^y)||(v^y||v^y)||(v^y||v^y)||(v^y||v^y)||(u^y||u^y)||(u^y||u^y).$$

The two nodes' expanded ID sequences are different in the first bit position, and the first frame of each node's CH sequence is $(u^x||u^x)$ and $(v^y||v^y)$ respectively. Since the first frame of one CH sequence is composed of two concatenated column-based sequences and the other is composed of two concatenated span-based sequences, nodes x and y are able rendezvous in N distinct channels during the first half frame.

Necessity for two concatenated sequences. Note that node x uses *two concatenated* column-based sequences, $(u^x||u^x)$, to expand a bit with a value of one; or two concatenated span-based sequences, $(v^x||v^x)$, to expand a bit with a value of zero, instead of using a *single* column- or span-based sequence. This method is employed to guarantee that any pair of nodes that generate their CH sequences using the above procedure can rendezvous in N channels irrelevant of the clock drift between them. Examples are given in Figure 5 to

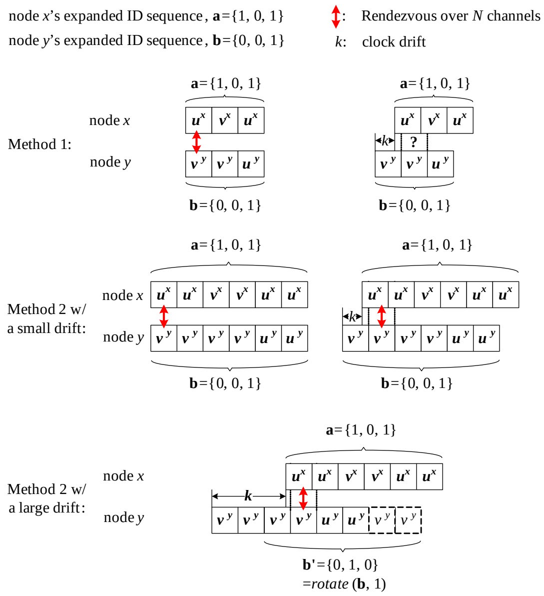

Fig. 5. Examples comparing a single column-/span-based sequence with a sequence composed of two concatenated column/span-based sequences.

show the necessity of two concatenated column- or span-based sequences. Suppose node x has an expanded ID sequence $\mathbf{a} = \{1, 0, 1\}$, and generates column- and span-based sequences u^x and v^x. Similarly, node y has an expanded ID sequence $\mathbf{b} = \{0, 0, 1\}$, and generates column- and span-based sequences u^y and v^y. The following two sequence generation methods are compared in Figure 5.

- **Method 1.** If an expanded ID sequence's bit is expanded to a single column- (or span-) based sequence, then rendezvous in N channels (i.e., the maximum rendezvous diversity) cannot be guaranteed when there is clock drift. For example, in the top figure, v^y may not have N rendezvous channels with the CH sequence composed of u^x and v^x.
- **Method 2.** When two concatenated column- or span-based sequences are used to construct a CH sequence, the maximum rendezvous diversity can be guaranteed (assuming a total of N channels and a CH period of $T = N^2$).
  - When the amount of clock drift, k, is small and less than a CH period, (i.e., $k < N^2$), $rotate(u^x, T-k)$ yields another column-based sequence for node x, which overlaps with node y's span-based sequence v^y over N channels.
  - When the amount of clock drift, k, is large and no less than a CH period, (i.e., $k \geq N^2$), node y's CH sequence is equivalent to a CH sequence that has been expanded from a rotated bit sequence $\mathbf{b}' = rotate(\mathbf{b}, \omega)$, where $\omega = \lfloor \frac{k}{2N^2} \rceil$ ($\lfloor x \rceil$ means the nearest integer to x). Note that $\mathbf{a}$ and $\mathbf{b}'$ have at least one distinct bit (according to Lemma 1). In this example, $\omega = 1$; the first bit of $\mathbf{a}$ and the first bit of $\mathbf{b}'$ are different; let $k^* = |k - \omega \cdot 2N^2|$, then $rotate(u^x, T - k^*)$ yields another column-based sequence for node x, which overlaps with node y's span-based sequence, v^y, over N channels.

Algorithm 3 CH Sequence Construction for Symmetric ACH Systems.

Input: an n-bit sequence, α, and an asymmetric ACH system $\{u, v\}$ where u is a column-based sequence and v is a span-based sequence.
Output: a CH sequence w.

$\mathbf{a} = \alpha || z || o$.
for $i = 0$ to $3n - 1$ **do**
    if $a_i == 1$ **then**
        the $(i+1)^{th}$ frame of w is $(u||u)$.
    end if
    if $a_i == 0$ **then**
        the $(i+1)^{th}$ frame of w is $(v||v)$.
    end if
end for

Algorithm 3 describes the procedure for generating CH sequences of a symmetric ACH system. The following theorem describes the properties of the CH sequences generated using Algorithm 3.

***Theorem** 3:* Algorithm 3 generates sequences of an ACH system whose period is $6nN^2$ and degree of overlapping is N. Here, N is the total number of channels, and n is the length of a radio's ID sequence.

Proof: Suppose two arbitrary channel-hopping nodes, x and y, have n-bit ID sequences α and β, respectively. Let $\mathbf{a}$ and $\mathbf{b}$ be the expanded ID sequences of nodes x and y. Each node generates a column-based sequence and a span-based sequence *independently* of the other node. Suppose node x generates a column-based sequence u^x and a span-based sequence v^x. Similarly, node y generates a column-based sequence u^y and a span-based sequence v^y. Using its n-bit ID sequence, α, and the column-based sequence u^x and the span-based sequence v^x as inputs to Algorithm 3, node x generates its CH sequence w. Similarly, using its n-bit ID sequence, β, u^y and v^y as inputs to Algorithm 3, node y generates its CH sequence w'.

Since each CH sequence generated by Algorithm 3 (either w or w') contains $3n$ frames and each frame has exactly $2N^2$ slots, the period of a CH sequence generated by the algorithm is $6nN^2$.

Without loss of generality, suppose that node x's clock is i slots ahead of node y's clock, where i is an arbitrary integer. We prove that the CH sequences $\{w, w'\}$ form an ACH system by considering the following two cases.

Case 1: $2kN^2 < i \leq (2k+1)N^2$, where $k \in [0, 3n-1]$. In this case, node x's CH sequence is ahead of node y's CH sequence by k frames and $(i - 2kN^2)$ slots. With respect to node y's clock, node x's CH sequence is equivalent to a CH sequence generated from the bit sequence $rotate(\mathbf{a}, \kappa)$. Let $\mathbf{a}' = rotate(\mathbf{a}, \kappa)$, where $\kappa = i - 2kN^2$. According to Lemma 1, we have $rotate(\mathbf{a}, \kappa) \neq \mathbf{b}$. If $a'_j \neq b_j$ for some $j \in [0, 3n-1]$, there are two possible cases to consider:

1) If $b_j = 0$, then $a'_j = 1$. This implies that the $(j+1)$-th frame of w' is $(v^y||v^y)$ and the $((j+\kappa+1) \bmod 3n)$-th frame of w is $(u^x||u^x)$.
2) If $b_j = 1$, then $a'_j = 0$. This implies that the $(j+1)$-th frame of w' is $(u^y||u^y)$ and the $((j+\kappa+1) \bmod 3n)$-th frame of w is $(v^x||v^x)$.

Case 2: $(2k+1)N^2 < i \leq (2k+2)N^2$, where $k \in [0, 3n-1]$. In this case, node x's CH sequence is ahead of node y's CH sequence by an amount of $(k+1)$ frames and $(i-(2k+2)N^2)$ slots. With respect to node y's clock, node x's CH sequence is equivalent to a CH sequence generated from a bit sequence $rotate(\mathbf{a}, \kappa+1)$. Let $\mathbf{a}' = rotate(\mathbf{a}, k+1)$ and $\kappa = (2k+2)N^2 - i$. According to Lemma 1, we have $rotate(\mathbf{a}, \kappa+1) \neq \mathbf{b}$. If $a'_j \neq b_j$ for some $j \in [0, 3n-1]$, there are also two possible cases to consider:

1) If $b_j = 0$, then $a'_j = 1$. This implies that the $(j+1)$-th frame of the CH sequence w' is $(v^y||v^y)$, the $((j+\kappa+2) \bmod 3n)$-th frame of CH sequence w is $(u^x||u^x)$.
2) If $b_j = 1$, then $a'_j = 0$. This implies that the $(j+1)$-th frame of the CH sequence w' is $(u^y||u^y)$, the $((j+k+2) \bmod 3n)$-th frame of CH sequence w is $(v^x||v^x)$.

In both cases, according to Theorem 2, a pair of sequences $(v^y||v^y)$ and $(u^x||u^x)$ (or $(u^y||u^y)$ and $(v^x||v^x)$) form an ACH system that has a degree of overlapping of N. This means that the pair of sequences have N distinct rendezvous channels irrelevant of the κ-slot clock drift when $0 \leq \kappa \leq N^2$. Thus, w and w' have N distinct rendezvous channels.

Since i is an arbitrary value, we can conclude that $\forall k, l \in [0, T-1]$, $\mathcal{C}(rotate(w,k), rotate(w',l)) = N$, and $\{w, w'\}$ is an ACH system whose degree of overlapping is N. □

According to Theorem 3, each node can rendezvous with another node without clock synchronization by independently generating ACH sequences using Algorithm 3. These ACH sequences enable maximum rendezvous diversity (i.e., rendezvous in N distinct channels within a period of $6nN^2$). Hence, the resulting ACH system's MRP value is $1/6nN$, and its ATTR is $O(N)$.

Note that no two distinct nodes will generate the same CH sequence since their expanded ID sequences are different. Using Algorithm 3, any two nodes can achieve asynchronous rendezvous without being pre-assigned a sender/receiver role.

6 Discussions

In this section, we discuss the possible causes of rendezvous failures. In addition, we compare the proposed ACH designs with existing CH schemes that also support asynchronous operation.

6.1 Factors that Affect Rendezvous Success Rate

Several factors can affect the success rate of rendezvous, some of which were discussed in the previous sections.

TABLE 1
A comparison of asynchronous CH schemes.

	Degree of overlapping	MRP	ATTR
RCH	0	n/a	N
SR	1	$\frac{1}{N(N+1)}$	$O(N^2)$
Asym. ACH	N	$\frac{1}{N}$	$O(N)$
Sym. ACH	N	$\frac{1}{6nN}$	$O(N)$

Besides those factors, there are others, including location, sensing errors, packet collisions, node mobility, etc.

Two nodes at different locations may have different perceptions of channel availability, which can lead to rendezvous failures. For example, suppose node A is close to a PU transmitter that operates on channel i, while node B is out of the transmission range of the PU transmitter. In this scenario, node A will determine that channel i is occupied, while node B will determine that the channel is available. The rendezvous between the two nodes in channel i should be prohibited even if it is listed in their CH sequences. This issue is investigated in Section 7.4 using simulation results.

Spectrum sensing errors can also cause the differences in the perception of channel availability among nodes, and impact the rendezvous success rate. A local perception of channel availability will likely be different for different nodes in a network. The differences in the perception of channel availability implies that a node, say node A, hops to a channel, say channel i, because it believes that the channel is clear, whereas another node, say node B, may not hop to or transmit over channel i because it believes the channel is occupied by incumbent signals (although the channel is in node B's hopping sequence). If channel i is a rendezvous channel between CH sequences of nodes A and B, then the above scenario results in a rendezvous failure.

Packet collisions may also affect rendezvous success rates. Packet collisions will occur when multiple pairs of nodes rendezvous in the same time slot over the same channel.

Node mobility can degrade rendezvous performance when the nodes move out of each other's transmission range during the rendezvous phase.

6.2 Comparison with Other CH Schemes

We assume that N is the total number of available channels.

Random channel hopping (RCH) [23]. In this scheme, each node hops from one channel to another randomly at the beginning of every timeslot. The RCH scheme has an ATTR value of N. However, the RCH scheme does not guarantee a bounded TTR between any two sequences, and its MRP value cannot be calculated either.

Sequence-based rendezvous (SR) [10]. Every node using SR follows the same pre-determined CH sequence. The sequence period is $N(N+1)$, the degree of overlapping is 1, and thus the MRP is $\frac{1}{N(N+1)}$.

A comparison of all the aforementioned schemes is summarized in Table 1. From the table, we can see that the proposed asymmetric and symmetric ACH schemes are superior to the other schemes in terms of degree of overlapping. Both schemes' degree of overlapping is the maximum possible value of N, and therefore the two schemes are optimal in terms of rendezvous diversity. On the other hand, the two schemes' ATTR is $O(N)$, which indicates that the ATTR scales linearly with the number of channels. The proposed ACH systems and the SR scheme guarantee a bounded TTR between any two channel hopping nodes, while the RCH scheme cannot give an upper bounded TTR.

7 Performance Evaluation

7.1 Simulation Setup

In this section, we compare the performance of the proposed ACH schemes, two existing symmetric CH schemes, and two existing synchronous CH schemes, via simulation results. We simulate ten pairs of secondary nodes using ns-2 [27] in a $1000\,\text{m}\times1000\,\text{m}$ square area. Every secondary node has a single half-duplex radio, and the transmission range of every secondary radio is $250\,\text{m}$. In the simulations, the radios of all the secondary users and the primary users can access 11 channels (i.e., the number of channels available to the CR network is $N = 11$). The duration of a timeslot is $10\,\text{ms}$. Each secondary node independently generates its CH sequence using the agreed CH scheme and performs channel hopping in accordance with the sequence. Once two nodes rendezvous on a channel, the link between them is established. Each simulation point represents the average value of a number of independent simulation runs. The required number of simulation runs was calculated using a method known as "independent replications" [15], [30].

Primary user traffic generation. We simulated X primary transmitters operating on X channels, and those channels were randomly chosen in each simulation run. In most existing work, it is assumed that a primary transmitter follows a "busy/idle" transmission pattern on a licensed channel [12], [13], and we make the same assumption. The busy period has a fixed length of b timeslots, and the idle period follows an exponential distribution with a mean of l timeslots. All of the secondary nodes are within the transmission range of any primary transmitter. A channel is considered "unavailable" when primary user signals are present in it. All secondary nodes should refrain from transmitting in unavailable channels.

Random clock drift. The simulations were performed under the assumption that the nodes' clocks are not synchronized. In each simulation run, each node determines its clock time independently of other nodes. Thus, there is a random clock drift between any two nodes.

7.2 Impact of Degree of Overlapping

To show the impact of degree of overlapping on the rendezvous process, we first simulate a scenario in which

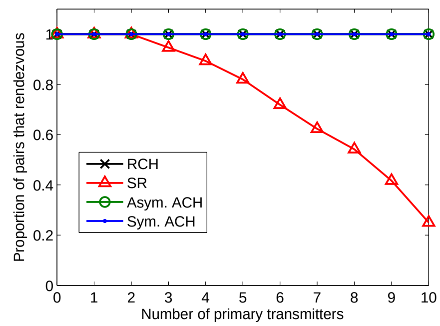

Fig. 6. The proportion of rendezvous pairs vs. number of primary transmitters.

primary transmitters continuously transmit on X channels, by setting b equal to the simulation time (or $b = \infty$) without any idle period.

Proportion of rendezvous pairs. We use the term *rendezvous pair* to denote a pair of nodes that are able to rendezvous successfully. We also define the term *proportion of rendezvous pairs* as the ratio between the number of rendezvous pairs and the total number of node pairs that attempt rendezvous expressed as a percentage. We measured the proportion of rendezvous pairs while varying the number of primary transmitters, and the results are shown in Figure 6. As expected, SR showed inferior performance compared to the other two schemes due to the fact that they have small degree-of-overlapping values (see Table 1). When the degree of overlapping of a CH scheme is small, the scheme is more vulnerable against rendezvous failures caused by the presence of PU signals. In contrast, the proposed ACH schemes (either asymmetric and symmetric ACH) have the maximum possible value for the degree of overlapping, viz N, which explains its robustness against the aforementioned vulnerability.

Note that the RCH scheme enables 100% of node pairs to rendezvous in Figure 6. The RCH scheme provides an opportunity for any pair of nodes to achieve rendezvous over *every* channel, and it has an expected TTR value of N between any two RCH sequences. This implies that two RCH sequences will rendezvous with a probability close to one, given a sufficiently long time (that is much greater than N). In our simulations, the simulation time is sufficiently long to enable all pairs of nodes using the RCH scheme to achieve successful rendezvous.

ATTR. In Figure 7, we compare the proposed ACH schemes and RCH in terms of ATTR. Although RCH has a high proportion of rendezvous pairs (see Figure 6), it requires a significantly greater TTR, compared to the proposed ACH schemes, to establish rendezvous. From Figure 7, we can see that the symmetric ACH scheme incurs greater average TTR compared to the asymmetric ACH scheme. This phenomenon can be attributed to the symmetric ACH scheme's longer CH period.

Rendezvous rate. We define the *rendezvous rate* as the number of successful rendezvous per slot. The rendezvous rate is another measure for quantifying a CH

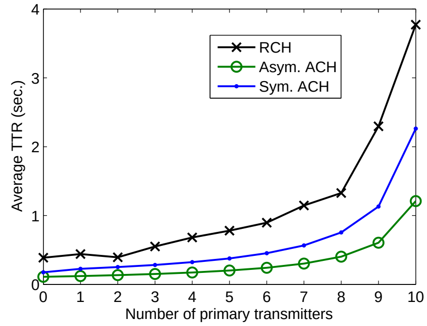

Fig. 7. Average TTR vs. number of primary transmitters.

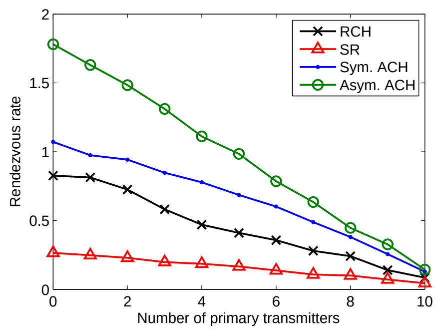

Fig. 8. Rendezvous rate.

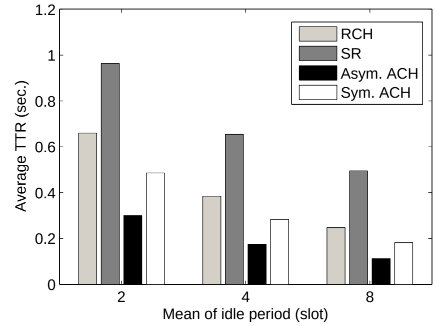

Fig. 9. Average TTR vs. mean of PU's idle period when $b = 10$.

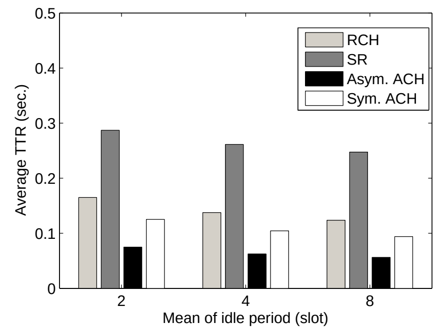

Fig. 10. Average TTR vs. mean of PU's idle period when $b = 1$.

scheme's ability to establish rendezvous in the presence of PU signals. The measured rendezvous rates for the CH schemes are given in Figure 8. Note that MRP is a theoretical estimate of the rendezvous rate when all of the N channels are free of primary user signals. The rendezvous rate is an empirically-obtained measure of the number of rendezvous per slot in the presence of PU transmissions. Figure 8 shows that the proposed asymmetric ACH scheme has the highest rendezvous rate among the simulated CH schemes.

7.3 Impact of Dynamic PU Traffic

In this set of simulations, we compare the four asynchronous CH schemes when the primary user transmission parameters vary dynamically. Specifically, two parameters are varied: the length of the busy period, b, and the mean of the idle period, l.

In Figures 9 and 10, we show the average TTR of the CH schemes as l is varied while fixing parameters X and b. From both of the figures, we can observe that as the spectrum availability of each channel ($\frac{l}{l+b}$) increases, the measured ATTR values of all CH schemes decrease. Note that the performance of SR is inferior to that of RCH and the proposed ACH schemes due to its long period length ($N(N+1)$) and low degree of overlapping value. From Figure 10, we can observe that the increase in l, from $l = 2$ to $l = 8$, while fixing $b = 1$ has little effect on decreasing the average TTR of the CH schemes. This phenomenon can be attributed to the fact that the availability of the channels is already sufficiently high when $l = 2$, which enables most node pairs to readily rendezvous without encountering primary user-occupied channels.

7.4 Discrepancy in the Perception of Channel Availability between the Sender and Receiver

In a realistic environment, the sender and receiver may have a discrepancy in their perception of channel availability, which can be caused by two factors—difference in location and sensing accuracy of the sender and receiver. We conjecture that these factors degrade the performance of rendezvous protocols.

7.4.1 Impact of location

In this set of simulations, we reduced the transmission range of PU transmitters such that some of the secondary sender and receiver nodes at different locations may have different perceptions of channel availability. We simulated ten "low-power" PU transmitters (that may represent Part 74 devices) that are operating over ten randomly-chosen channels, and fixed the transmission range of a PU transmitter to 150 m. In each secondary sender-receiver pair, we fixed the location of the sender and changed the location of the receiver by increasing the distance between the sender and receiver. From Figure 11, we can observe that the rendezvous rate drops for all four protocols as the distance between the sender and receiver increases. The results confirm our conjecture that the discrepancy in the perception of channel availability between the sender and receiver, caused by difference in location, degrades rendezvous

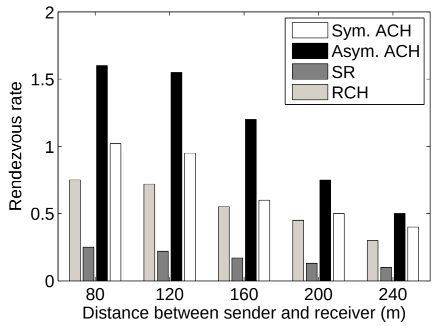

Fig. 11. The rendezvous rate vs. distance between sender and receiver.

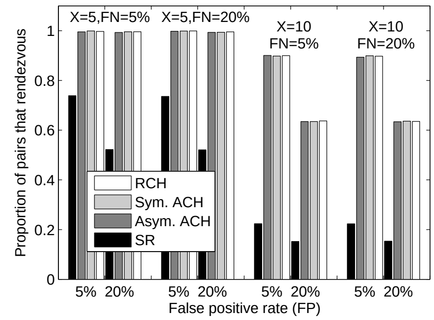

Fig. 12. The proportion of rendezvous pairs vs. false positive rates.

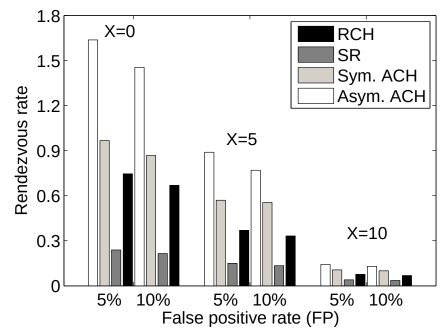

Fig. 13. The rendezvous rate vs. false positive rates (false negative rate=5%).

performance. It is worth noting that the two ACH protocols outperform the other two protocols even in such situations because of their greater rendezvous diversity compared to the other two protocols.

7.4.2 *Impact of sensing errors*

In this set of simulations, we used the simulation settings in Section 7.1, and varied the values of two types of spectrum sensing errors—i.e., false positives and false negatives. From Figures 12 and 13, we can observe that the increase of false positives causes a degradation in performance (in terms of the proportion of rendezvous pairs and rendezvous rate). The values X and FN above each group of bars denote the number of PU transmitters and the false negatives rate, respectively.

- When a false positive error occurs, a secondary user mistakenly recognizes a PU-free channel as being unavailable, and thus will not rendezvous in that channel. If there is an adequate number of PU-free channels (the case when $X = 5$ in Figure 12), then RCH and ACH provide rendezvous opportunities in every PU-free channel, and thus the false positives have little impact on the proportion of rendezvous pairs (this can be seen in Figure 12). However, if there is only one PU-free channel (the case when $X = 10$), then false positives can make a much greater impact on rendezvous performance.
- When a false negative error occurs over a channel h that is currently occupied by PU signals, two SUs will mistakenly recognize channel h as a PU-free channel, and try to rendezvous in that channel. However, this attempt to rendezvous in the mistakenly-recognized PU-free channel h will not lead to an increase in the proportion of rendezvous pairs (as can be seen from Figure 12); because in such a case, the PU signals over channel h can still cause collisions to SUs' control packets, which will prevent the two SUs from successfully exchanging control packets for completing a rendezvous process over that channel.

Based on our analytical and empirical results, we can make the following conclusions about the relative performance of the asynchronous CH schemes under consideration: the CH schemes that have maximum rendezvous diversity (e.g., the proposed ACH schemes) or provide rendezvous opportunities on every channel (e.g., RCH) would outperform the CH scheme that has a limited degree of overlapping value but a long period (e.g., SR), in terms of the average TTR. In addition, we can also conclude that a number of other factors, including spectrum sensing errors, can also impact rendezvous performance significantly.

8 Conclusion

In this paper, we have presented a systematic approach for designing channel hopping (CH) protocols for CR networks that enable rendezvous between any two nodes even if their clocks are asynchronous. Our approach is novel in that it focuses on maximizing rendezvous diversity between two channel hopping nodes while their clocks are asynchronous. The proposed asynchronous CH schemes addressed the asynchronous rendezvous problem by satisfying the following important requirements: (1) enable pairwise rendezvous between any pair of CH sequences (one used by the sender and other used by the receiver) on *every* available channel; and (2) ensure that any pair of CH sequences achieve rendezvous with an *upper bounded* TTR.

Our results have shown that the proposed asymmetric ACH approach incurs a smaller average TTR than the symmetric ACH approach. However, the former approach requires each node to be pre-assigned a role as either a sender or a receiver prior to rendezvous, a la Bluetooth pairing. The latter approach does not have such a requirement which makes it suitable for a wider range of network scenarios.

In addition, our results have shed light on a number of important facts about rendezvous protocol performance, including: rendezvous diversity is a critical factor in rendezvous performance; length of the CH period affects TTR; and the impact of spectrum sensing errors on rendezvous performance can vary depending on other factors such as the number of PU-free channels.

REFERENCES

[1] C. Arachchige, S. Venkatesan, and N. Mittal. An Asynchronous Neighbor Discovery Algorithm for Cognitive Radio Networks. Proc. of *IEEE DySPAN*, pp. 1–5, October 2008.

[2] A. Asterjadhi, N. Baldo, and M. Zorzi. A Distributed Network Coded Control Channel for Multihop Cognitive Radio Networks. *IEEE Network*, 23(4):26–32, July-August 2009.

[3] P. Bahl, R. Chandra, and J. Dunagan. SSCH: Slotted Seeded Channel Hopping for Capacity Improvement in IEEE 802.11 Ad Hoc Wireless Networks. Proc. of *ACM MobiCom*, pp. 216–230, September 2004.

[4] N. Baldo, A. Asterjadhi, and M. Zorzi. Multi-channel Medium access using a Virtual Network Coded Control Channel. Proc. of *ACM IWCMC*, June 2009.

[5] K. Bian and J. Park. Asynchronous Channel Hopping for Establishing Rendezvous in Cognitive Radio Networks. Proc. of *IEEE INFOCOM*, Mini-conference, pp. 236–240, April 2011.

[6] K. Bian, J. Park, and R. Chen. A Quorum-based Framework for Establishing Control Channels in Dynamic Spectrum Access Networks. Proc. of *ACM MobiCom*, pp. 25–36, September 2009.

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

[8] T. Chen, H. Zhang, G. M. Maggio, and I. Chlamtac. CogMesh: A Cluster-based Cognitive Radio Network. Proc. of *IEEE DySpan*, pp. 168–178, April 2007.

[9] C. Cordeiro and K. Challapali. C-MAC: A Cognitive MAC Protocol for Multi-Channel Wireless Networks. Proc. of *IEEE DySpan*, pp. 147–157, April 2007.

[10] L.A. DaSilva and I. Guerreiro. Sequence-based Rendezvous for Dynamic Spectrum Access. Proc. of *IEEE DySpan*, pp. 1–7, October 2008.

[11] C. Doerr, D. Sicker, and D. Grunwald. Dynamic Control Channel Assignment in Cognitive Radio Networks using Swarm Intelligence. Proc. of *IEEE Globecom* pp. 1–6, November 2008.

[12] S. Geirhofer, L. Tong, and B. Sadler. Cognitive Medium Access: Constraining Interference Based on Experimental Models. *IEEE Journal on Selected Areas of Communications*, 26(1):95–105, January 2008.

[13] S. Huang, X. Liu, and Z. Ding. Optimal Transmission Strategies for Dynamic Spectrum Access in Cognitive Radio Networks. *IEEE Transactions on Mobile Computing*, 8(12):1636–1648, December 2009.

[14] IEEE Standards Board. IEEE Standard 802.11—Wireless Lan Medium Access Control and Physical Layer Specifications, June 1997.

[15] R. Jain. The Art of Computer Systems Performance Analysis. Wiley-Interscience, New York, NY, April 1991.

[16] J. Jia, Q. Zhang, and X. S. Shen. HC-MAC: A Hardware-constrained Cognitive MAC for Efficient Spectrum Management. *IEEE Journal on Selected Areas in Communications*, 26(1):106–117, January 2008.

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

[18] L. Lazos, S. Liu, and M. Krunz. Mitigating Control-channel Jamming Attacks in Multi-channel Ad Hoc Networks. Proc. of *ACM WiSec*, pp. 169–180, March 2009.

[19] B.F. Lo, I.F. Akyildiz, and A.M. Al-Dhelaan. Efficient Recovery Control Channel Design in Cognitive Radio Ad Hoc Networks. *IEEE Transactions on Vehicular Technology*, 59(9): 4513–4526, November 2010.

[20] W.-S. Luk and T.-T. Wong. Two New Quorum Based Algorithms for Distributed Mutual Exclusion. Proc. of *IEEE ICDCS*, May 1997.

[21] H. Nan, T.-I. Hyon, and S.-J. Yoo. Distributed Coordinated Spectrum Sharing MAC Protocol for Cognitive Radio. Proc. of *IEEE DySpan*, pp. 240–249, April 2007.

[22] C. Popper, M. Strasser and S. Capkun. Anti-jamming Broadcast Communication using Uncoordinated Spread Spectrum Techniques. *IEEE Journal on Selected Areas in Communications*, 28(5):703–715, June 2010.

[23] M. D. Silvius, F. Ge, A. Young, A. B. MacKenzie, and C. W. Bostian. Smart Radio: Spectrum Access for First Responders. Proc. of *SPIE*, Vol. 6980, Wireless Sensing and Processing III, April 2008.

[24] J. So and N. Vaidya. Multi-Channel MAC for Ad Hoc Networks: Handling Multi-Channel Hidden Terminals Using a Single Transceiver. Proc. of *ACM MobiHoc*, pp. 222–233, May 2004.

[25] H. W. So, J. Walrand and J. Mo. McMAC: A Multi-Channel MAC Proposal for Ad Hoc Wireless Networks. Proc. of *IEEE WCNC*, pp. 334–339, March 2007.

[26] M. Strasser, C. Popper, S. Capkun, and M. Cagalj. Jamming-resistant Key Establishment using Uncoordinated Frequency Hopping. Proc. of *IEEE Symposium on Security and Privacy*, pp. 64–78, May 2008.

[27] The Network Simulator (ns-2). http://www.isi.edu/nsnam/ns/.

[28] N. Theis, R. Thomas and L. DaSilva. Rendezvous for Cognitive Radios. *IEEE Transactions on Mobile Computing*, 10(2):216–227, February 2011.

[29] A. Tzamaloukas and J. J. Garcia-Luna-Aceves. Channel-Hopping Multiple Access. *Proc. of IEEE ICC*, pp. 415–419, June 2000.

[30] W. Whitt. The Efficiency of One Long Run versus Independent Replications in Steady-state Simulation. *Management Science*, 37(6):645–666, 1991.

[31] J. Zhao, H. Zheng, and G.-H. Yang. Distributed Coordination in Dynamic Spectrum Allocation Networks. Proc. of *IEEE DySpan*, pp. 259–268, November 2005.

Kaigui Bian received his Ph.D. degree in computer engineering from Virginia Tech, Blacksburg, USA in 2011. He is currently an Assistant Professor in the Institute of Network Computing and Information Systems, School of Electronics Engineering and Computer Science (EECS) at Peking University, Beijing, China. His research interests include cognitive radio networks, wireless network security, and privacy enhancing technologies. He is a member of the IEEE.

Jung-Min "Jerry" Park received his Ph.D. degree in electrical and computer engineering from Purdue University, West Lafayette, USA in 2003. He is currently an Associate Professor in the Department of Electrical and Computer Engineering at Virginia Polytechnic Institute and State University (Virginia Tech). His research interests are in network security, applied cryptography, cognitive radio networks, and dynamic spectrum access technologies. Dr. Park has published his work in leading journals and conference proceedings. He is leading or participating in several sponsored research projects. Current or recent research sponsors include the National Science Foundation (NSF), SANS Institute (SysAdmin, Audit, Network Security), Samsung Electronics, and SCA Techniques Inc. More details about his research interests can be found at http://www.arias.ece.vt.edu/. Dr. Park was a recipient of a 1998 AT&T Leadership Award, an NSF Faculty Early Career Development (CAREER) Award, and a 2008 Hoeber Excellence in Research Award (equivalent to a best paper award). He is a senior member of the IEEE, a senior member of the Association for Computing Machinery (ACM), and a member of the Korean-American Scientists and Engineers Association (KSEA). He is currently an editor of the Journal of Communications and Networks.