

Optimizing Average-Maximum TTR Trade-off for Cognitive Radio Rendezvous

Lin Chen^{1,2}, Shuyu Shi^{1,3}, Kaigui Bian², and Yusheng Ji³

¹Department of Electrical Engineering, Yale University, New Haven, CT, USA

²Institute of Network Computing and Information System, School of EECS, Peking University, Beijing, China

³National Institute of Informatics, Tokyo, Japan

Abstract—In cognitive radio (CR) networks, “TTR”, a.k.a. time-to-rendezvous, is one of the most important metrics for evaluating the performance of a channel hopping (CH) rendezvous protocol, and it characterizes the rendezvous delay when two CRs perform channel hopping. There exists a trade-off of optimizing the average or maximum TTR in the CH rendezvous protocol design. On one hand, the random CH protocol leads to the best “average” TTR without ensuring a finite “maximum” TTR (two CRs may never rendezvous in the worst case), or a high rendezvous diversity (multiple rendezvous channels). On the other hand, many sequence-based CH protocols ensure a finite maximum TTR (upper bound of TTR) and a high rendezvous diversity, while they inevitably yield a larger average TTR. In this paper, we strike a balance in the average-maximum TTR trade-off for CR rendezvous by leveraging the advantages of both random and sequence-based CH protocols. Inspired by the neighbor discovery problem, we establish a design framework of creating a wake-up schedule whereby every CR follows the sequence-based (or random) CH protocol in the awake (or asleep) mode. Analytical and simulation results show that the hybrid CH protocols under this framework are able to achieve a greatly improved average TTR as well as a low upper-bound of TTR, without sacrificing the rendezvous diversity.

I. INTRODUCTION

“Rendezvous” in *cognitive radio* (CR) networks refers to the process for two secondary users (SUs) to find each other on a *rendezvous/control channel* prior to data communications [1]. A *rendezvous failure* occurs between two SUs when the rendezvous channel is unavailable due to the detection of primary users (PUs) or interference signals.

To alleviate the rendezvous failure problem, two types of *channel hopping* (CH) rendezvous protocols have been widely used to create multiple rendezvous channels between two SUs [3], [5], [10], [13]. In a random CH protocol, two SUs hop across channels at random in search of each other. In a sequence-based CH protocol, each SU starts a channel hopping process according to its own CH sequence and local clock; two SUs’ CH sequences are carefully chosen to spread out rendezvous points over multiple pairwise common channels. The *time-to-rendezvous* (TTR) or rendezvous delay, is usually used for evaluating the performance of a CH rendezvous protocol.

There exists a trade-off of optimizing the average or maximum TTR in the design of a CH rendezvous protocol. The random CH protocol leads to the best “average” TTR of N timeslots given N channels. However, two SUs may never rendezvous in the worst case, which implies that the

“maximum” TTR can be infinite. Besides, the random CH protocol cannot guarantee multiple rendezvous channels, and the number of rendezvous channels between two SUs is called the *rendezvous diversity*. In contrast, many sequence-based CH protocols ensure a finite maximum TTR (upper bound of TTR) with a high rendezvous diversity, at the expense of incurring a large average TTR.

Naturally, we are particularly interested in the following question: *Is it possible for an SU to determine when to switch to the random or sequence-based CH protocol for achieving the best performance in terms of both average and maximum TTR, while preserving a high rendezvous diversity?*

In wireless sensor networks, the wake-up schedule approach has been widely studied for addressing the neighbor discovery problem, which allows each node to switch between two modes (awake or asleep) such that two neighboring nodes can maintain the link connectivity with energy constraints [2], [6], [9]. Our research findings indicate that it is feasible to combine the CH processes of random and sequence-based protocols by enforcing each SU to mimic the behavior of switching between two modes in a wake-up schedule.

In this paper, we strike a balance in the average-maximum TTR trade-off for CR rendezvous by leveraging the advantages of both random and sequence-based CH protocols. The contributions of this work are summarized as follows.

- 1) We establish a design framework of creating a wake-up schedule whereby every SU follows the sequence-based (or random) CH protocol in the awake (or asleep) mode, such that two SUs can achieve rendezvous with significantly improved average TTR, while an upper-bounded TTR and rendezvous channel diversity are guaranteed as well.
- 2) We present a unified approach of devising a series of hybrid CH rendezvous protocols that interleave random and sequence-based CH processes, and show that such protocols can achieve a high rendezvous diversity within an average TTR of N slots, given N channels.
- 3) Analytical and simulation results confirm that the hybrid protocols under the design framework are able to preserve a small average TTR as well as a low upper-bound of TTR, without sacrificing the rendezvous diversity.

The rest of the paper is organized as follows. We provide the system model and formulate the problem in Section II. In Section III, we describe the design framework of hybrid CH rendezvous protocols based on interleaving techniques.

We evaluate our proposed framework using simulation results in Section IV, and conclude the paper in Section V.

II. PROBLEM FORMULATION

A. System Model

We assume a CR network where each secondary user/node is equipped with a CR operating over a set of orthogonal frequency channels that are licensed to primary users. We denote each node in the network by its unique identifier (ID), say $i \in \Lambda$, where Λ is the set of all possible IDs —i.e., the secondary node with its ID i is termed *node i* . The set of channels which the nodes in this network can sense and operator over is called the *sensible channel set*, denoted by $C = \{1, 2, 3, \dots, N\}$. The cardinality of the sensible channel set is known as the *sensible channel number*, or simply *channel number*, written as $N = |C|$.

CH sequence and clock drift. We consider a time-slotted communication system in which time is divided into consecutive time slots of equal length $2t_0$, where t_0 is the time necessary for link establishment. For example, as prescribed by IEEE 802.22 [12], $t_0 = 10$ ms and thus time is divided into time slots of 20 ms in the IEEE 802.22 context. We double the link establishment time to be the slot duration because two nodes can still have slot overlap no less than t_0 , which is adequate for link establishment, even if slot boundaries are misaligned between them (please see Fig. 1) [8]. Thus we can safely assume that slot boundaries are aligned between two nodes.

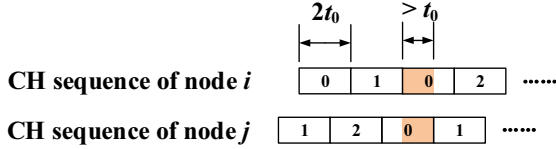


Figure 1: The CH sequences of nodes i and j are $\{0, 1, 0, 2, \dots\}$ and $\{1, 2, 0, 1, \dots\}$, respectively. They will rendezvous in the third slot on channel 0. Although the boundaries of their time slots are misaligned, since the duration of a slot is $2t_0$, the slot overlap on channel 0 is greater than t_0 .

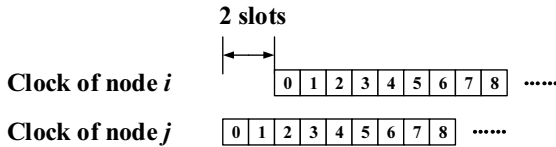


Figure 2: Nodes i and j have their respective local clocks. Time slots are divided in accordance with each node's local clock, numbered as slot 0, slot 1, slot 2, etc. Local clocks of two nodes may differ from each other by a certain amount of *clock drift*—i.e., if node i 's clock is two slots behind that of node j , then node j 's slot 2 will be its slot 0.

Each node has its local clock. Time slots are divided in accordance with each node's local clock, numbered as slot 0, slot 1, slot 2, etc. Local clocks of two nodes may differ

from each other by a certain amount of *clock drift*—i.e., if node i 's clock is two slots behind that of node j , then node j 's slot 2 will be its slot 0 (see Fig. 2). In order to achieve rendezvous, each node is capable of hopping across different channels in accordance with its CH sequence and local clock. We formulate node i 's CH sequence as a function

$$\begin{aligned} \theta_i : \mathbb{N} \cup \{0\} &\rightarrow C \\ t &\mapsto \theta_{i,t}, \end{aligned}$$

where $\mathbb{N} \cup \{0\}$ denotes the set of time slots that begin with slot 0 and θ_i maps each slot to a channel in C , which means that node i hops onto the frequency channel $\theta_{i,t}$ at slot t .

Rendezvous. As aforementioned, there may be a certain amount of clock drift between nodes i and j 's local clocks. We denote it by σ_{ij} , which means that node i 's clock is σ_{ij} slots behind that of node j . If $\sigma_{ij} < 0$, node i 's clock is ahead of node j 's in fact. If $\sigma_{ij} = 0$, they are synchronized. A rendezvous between nodes i and j is said to occur if they hop onto the same frequency channel simultaneously. Formally, a rendezvous is said to occur at node i 's slot t if $\theta_{i,t} = \theta_{j,t+\sigma_{ij}}$. In this case, node i 's slot t is called a *rendezvous slot* and the frequency channel $\theta_{i,t}$ is called a *rendezvous channel*.

Let $\mathcal{T}(\theta_i, \theta_j, \sigma_{ij})$ denote the set of rendezvous slots given that node i 's clock is σ_{ij} slots behind that of node j . A noteworthy problem is whether the serial numbers of rendezvous slots should be with regard to the local clock of node i or node j . In our definition, rendezvous slots are indexed in accordance with the local clock left behind. For example, if node i 's clock is behind that of node j ($\sigma_{ij} > 0$), then rendezvous slots in $\mathcal{T}(\theta_i, \theta_j, \sigma_{ij})$ are with regard to the clock of node i ; otherwise, the clock of node j . If $\sigma_{ij} > 0$, $\{2, 6\} \subseteq \mathcal{T}(\theta_i, \theta_j, \sigma_{ij})$ implies that nodes i and j rendezvous at node i 's 2nd and 6th slots; if $\sigma_{ij} \leq 0$, it means that they rendezvous at the 2nd and 6th slots with respect to node j 's clock. This idea is natural because the zeroth slot of the clock left behind denotes when both nodes start channel hopping.

Given that node i 's clock is σ_{ij} slots behind that of node j , let $\mathcal{C}(\theta_i, \theta_j, \sigma_{ij})$ denote the set of rendezvous channels between nodes i and j .

B. CH Protocol Design Problem

CH protocol. A CH protocol is a fully distributed algorithm whereby each node autonomously generates its CH sequence only employing the information of its ID i . Formally, a CH protocol is a map

$$\begin{aligned} \theta : \Lambda &\rightarrow \Theta \\ i &\mapsto \theta_i, \end{aligned}$$

where Λ is the set of all possible IDs and $\Theta = \{f : \mathbb{N} \cup \{0\} \rightarrow \mathbb{N}\}$ is the set of all CH sequences. Note that θ only depends on the ID i (i.e., θ is independent of the clock drift between nodes, or the information of other nodes apart from i).

Many existing CH protocols are *periodic* in the sense that there exists a (minimum) positive constant integer τ (known as the period length) such that

$$\forall i \in \Lambda, t \in \mathbb{N} \cup \{0\}, \theta_{i,t+\tau} = \theta_{i,t}.$$

The period length τ is a function of the channel number N , denoted by $\tau(N)$. For example, CRSEQ [11] has a period length of $\tau(N) = N(3N - 1)$ if N is prime; for the Jump-stay (JS) algorithm in [10], the period length is $\tau(N) = 3N$ if N is prime.

Performance metrics. We introduce three performance metrics for evaluating the performance a CH rendezvous protocol.

- 1) *Maximum time-to-rendezvous (MTTR)*. The latency before nodes i and j 's first rendezvous can be characterized by $\min \mathcal{T}(\theta_i, \theta_j, \sigma_{ij})$, i.e., the minimal value in the set $\mathcal{T}(\theta_i, \theta_j, \sigma_{ij})$. However, this latency relies on their clock drift σ_{ij} . In practical scenarios, we are unaware of what the clock drift would be, which, in fact, is a random variable. Thus we care about the worst-case (maximum) latency given an arbitrary amount of clock drift, which is exactly the *maximum time-to-rendezvous (MTTR)*. Formally, the MTTR between nodes i and j is given by

$$MTTR(\theta_i, \theta_j) = \max_{\sigma_{ij} \in \mathbb{Z}} \min \mathcal{T}(\theta_i, \theta_j, \sigma_{ij}).$$

MTTR is a bound for time-to-rendezvous. In the random channel hopping protocol (RCH) [4], each node i randomly hops onto a channel at each time slot, and it is easy to show that $MTTR = +\infty$ in this case—i.e., RCH fails to have a bounded time-to-rendezvous or a guaranteed rendezvous diversity.

- 2) *Average time-to-rendezvous (ATTR)*. Apart from MTTR, we take into account the average (expected) latency before the first rendezvous, termed as *average time-to-time (ATTR)*. Formally, the ATTR between nodes i and j is given by

$$ATTR(\theta_i, \theta_j) = \mathbb{E}[\min \mathcal{T}(\theta_i, \theta_j, \sigma_{ij})].$$

- 3) *Rendezvous channel diversity rate*. In practical applications, some channels may encounter problems such as congestion, attack, eavesdropping etc., and thus become inappropriate for rendezvous and information exchange. Ideally, we want to maximize the rendezvous diversity in hopes that the two nodes can attempt to rendezvous on all channels. We introduce the metric called *rendezvous channel diversity rate*, which is defined for two nodes i and j such as

$$div(\theta_i, \theta_j) = \min_{\sigma_{ij} \in \mathbb{Z}} \frac{|\mathcal{C}(\theta_i, \theta_j, \sigma_{ij})|}{N}$$

and quantifies the minimum ratio of the number of rendezvous channels to that of all sensible channels. It follows immediately from the definition that the rendezvous channel diversity rate ranges from 0 to 1.

CH protocol design problem. The CH protocol design problem is how to devise a fully distributed CH protocol θ whereby each node i autonomously generate its CH sequence θ_i such that the protocol can achieve bounded MTTR, small ATTR and high rendezvous diversity rate in CR networks environments, in spite of a random clock drift between two nodes.

III. INTERLEAVING-BASED FRAMEWORK FOR HYBRID CH RENDEZVOUS PROTOCOLS

In this section, we propose a design framework for creating hybrid CH rendezvous protocols that minimize the ATTR of CH protocols while preserving their TTR bound. We begin with introducing the neighbor discovery wake-up schedule design problem. In accordance with a specified neighbor discovery wake-up schedule, any existing CH protocol can be easily extended to a hybrid protocol under our framework.

A. Definition of the Neighbor Discovery Wake-up Schedule

In energy-constraint wireless sensor networks (WSNs), a neighbor discovery *wake-up schedule* of node i is a binary sequence that consists of only zeros and ones,

$$\mathbf{x}^i = \{\mathbf{x}_0^i, \mathbf{x}_1^i, \dots, \mathbf{x}_{T_i-1}^i\}$$

where $\forall 0 \leq t \leq T_i - 1$, $\mathbf{x}_t^i \in \{0, 1\}$. The length of the sequence, i.e., T_i , is termed the *period length* of the wake-up schedule \mathbf{x}^i . At the t -th time slot, if $\mathbf{x}_t^i = 1$, the node will be active (awake); otherwise, it will be inactive (asleep).

The *duty cycle* of a wake-up schedule quantifies the percentage of slots in which the node is active—i.e., the duty cycle of \mathbf{x}^i is

$$\frac{\sum_{t=0}^{T_i-1} \mathbf{x}_t^i}{T_i}.$$

Obviously, the duty cycle ranges from 0 to 1. For instance, for $\mathbf{x}^i = \{0, 0, 1, 1, 0\}$, its duty cycle will be $\frac{2}{5} = 40\%$. For the purpose of energy saving, the duty cycle is supposed to be as small as possible.

We introduce the notion of *cyclic rotation* to characterize clock drift between nodes. For a wake-up schedule \mathbf{x}^i , we define $rotate(\mathbf{x}^i, k) \triangleq \{\mathbf{x}_0^i, \mathbf{x}_1^i, \dots, \mathbf{x}_{T_i-1}^i\}$ where $\forall t \in [0, T_i - 1]$, $\mathbf{x}_t^i = \mathbf{x}_{(t+k) \bmod T_i}^i$.

Given any clock drift, a *neighbor discovery* between nodes i and j successfully occurs if and only if $\forall k \in \mathbb{Z}$, $\exists t \geq 0$ such that

$$\mathbf{x}_{t \bmod T_i}^i = rotate(\mathbf{x}, k)_{t \bmod T_j}^j = 1.$$

A *neighbor discovery protocol*, ν , assigns each node a wake-up schedule in accordance with its desired duty cycle in order to guarantee successful neighbor discovery between any two neighboring nodes. If node i 's desired duty cycle is $\delta_i \in [0, 1]$, then it will be assigned the wake-up schedule $\mathbf{x}^i \triangleq \nu(\delta_i)$. Formally, a neighbor discovery protocol is a map ν from $\Delta \subseteq [0, 1]$ to the set of binary sequences, where Δ , which is a subset of $[0, 1]$, is called the set of its *supported duty cycles*, and ν is expected to satisfy that $\forall \delta_i, \delta_j \in \Delta$, $\forall k \in \mathbb{Z}$, $\exists t \geq 0$ such that

$$\nu(\delta_i)_{t \bmod T_i} = rotate(\nu(\delta_j), k)_{t \bmod T_j} = 1,$$

where T_i (or T_j) is the length of $\nu(\delta_i)$ (or $\nu(\delta_j)$). Specifically, $\forall \delta \in \Delta$, $\forall k \in \mathbb{Z}$, $\exists t \geq 0$ such that

$$\nu(\delta)_{t \bmod T} = rotate(\nu(\delta), k)_{t \bmod T} = 1,$$

where T is the length of $\nu(\delta)$.

B. Hybrid CH Protocols by Interleaving Random and Sequence-based CH Processes

Suppose that ν is an arbitrarily given neighbor discovery protocol and that θ is a periodic CH protocol with period length $\tau(N)$.

Padding scheme. According to the *padding scheme*, we increase the channel number N to some integer $N' \geq N$. We view the newly added $(N' - N)$ channels as aliases of the original N channels. For example, if the channel number is 3, we add a new channel, say, channel 4, so that the new channel number amounts to 4. Channel 4 serves as a random channel sampled from the original N channels.

The first step of our proposed algorithm is to choose a supported duty cycle, say, δ , of the neighbor discovery protocol ν . Let \mathbf{x} denote $\nu(\delta)$ and we write T for the length of \mathbf{x} . As aforementioned, \mathbf{x} satisfies that $\forall k \in \mathbb{Z}, \exists 0 \leq t \leq T-1$ such that

$$\mathbf{x}_t = \text{rotate}(\mathbf{x}, k)_t = 1.$$

Then, by the padding scheme, the algorithm will slightly increase the channel number N to $N' = \min\{N' \in \mathbb{N} : N' \geq N, \gcd(\tau(N'), \sum_{t=0}^{T-1} \mathbf{x}_t) = 1\}$. Given N' as the new channel number, with the CH protocol θ at hand, every node i has its CH sequence θ_i with period length $\tau = \tau(N')$, where $\gcd(\tau, \sum_{t=0}^{T-1} \mathbf{x}_t) = 1$.

CH sequence generation. We now present in Algorithm 1 how node i generates its new CH sequence in accordance with the wake-up schedule \mathbf{x} and the original CH sequence θ_i . Note that with the padding scheme, we can safely assume that the channel number N satisfies that $\gcd(\tau, \sum_{t=0}^{T-1} \mathbf{x}_t) = 1$, where $\tau = \tau(N)$ is the period length of the CH protocol θ —i.e., for convenience of notations, we simply use N to denote the resulting slightly increased channel number (namely N') after the padding scheme is conducted.

As demonstrated in Algorithm 1, at the t -th timeslot, the resulting new CH sequence generated by our proposed framework uses a slot that comes from the original CH sequence θ_i if the $(t \bmod T)$ -th bit of \mathbf{x} equals 1 (i.e., $\mathbf{x}_{t \bmod T} = 1$); otherwise, it uses a random channel. We eventually obtain an interleaved new CH sequence θ'_i .

Fig. 3 illustrates an example of the proposed framework. In the example, the original CH sequence is $\{1, 2, 3, 1, 2, 3, 1, 2, 3, \dots\}$ and the specified wake-up schedule is $\{1, 1, 1, 0, 1, 0, 0, 0\}$. We demonstrate the resulting new CH sequence generated in accordance with our proposed framework, where “ r ” represents a randomly selected channel.

The average TTR of random channel hopping is N . By interleaving the random CH process, we can improve the average performance (i.e., ATTR) of the original CH protocol θ . By leveraging the properties of wake-up schedules, the proposed framework maintains a bounded TTR and the rendezvous diversity inherited from the original CH protocol.

Theorem 1 presents the main result regarding the rendezvous performance improvement by the proposed method.

Theorem 1. *The new CH sequence generated by Algorithm 1 has the following properties:*

- 1) *It can guarantee rendezvous within τT slots (thus preserves bounded TTR).*

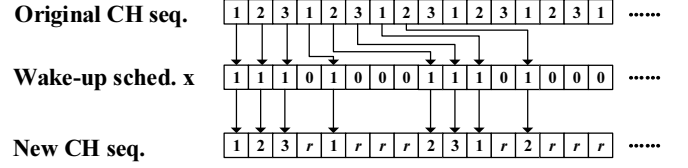


Figure 3: This figure illustrates an example of the CH sequence under the proposed framework. The original CH sequence is $\{1, 2, 3, 1, 2, 3, 1, 2, 3, \dots\}$ and the wake-up schedule is $\{1, 1, 1, 0, 1, 0, 0, 0\}$. The resulting new CH sequence is $\{1, 2, 3, r, 1, r, r, r, 2, 3, 1, r, 2, r, r, r, \dots\}$, where “ r ” represents a randomly-selected channel.

Algorithm 1 CH sequence generating algorithm

Input: Wake-up schedule, \mathbf{x} ; original CH sequence, θ_i .

Output: New CH sequence, θ'_i .

- 1: $T \leftarrow |\mathbf{x}|$;
 - 2: $\triangleright T$ denotes the length of the wake-up schedule.
 - 3: $t \leftarrow 0$;
 - 4: \triangleright In the first place, initialize t to 0, which means that we begins generating the CH sequence from timeslot 0.
 - 5: $t' \leftarrow 0$;
 - 6: $\triangleright \theta_i$'s next slot that the algorithm is going to use is $\theta_{i,t'}$.
 - 7: **while** not rendezvous **do**
 - 8: **if** $\mathbf{x}_{t \bmod T} = 0$ **then**
 - 9: $\theta'_{i,t} \leftarrow$ a random channel;
 - 10: **else**
 - 11: $\theta'_{i,t} \leftarrow \theta_{i,t'}$;
 - 12: $t' \leftarrow t' + 1$;
 - 13: \triangleright Update t' .
 - 14: **end if**
 - 15: Node i hops onto channel $\theta'_{i,t}$ at the t -th time slot;
 - 16: **end while**
-

- 2) *It achieves at least the same rendezvous diversity as the original CH protocol (thus preserves the rendezvous diversity).*
- 3) *If the clock drift is k , let B be $|\{t \in [0, T-1] \cap \mathbb{N} : \mathbf{x}_t = \text{rotate}(\mathbf{x}, k)_t = 1\}|$, then the ATTR of the new CH sequence, denoted by $\text{ATTR}_{\theta'}$, will be $\frac{B}{T} \cdot \text{ATTR}_{\theta} + (1 - \frac{B}{T}) \cdot N$ (thus when $\frac{B}{T}$ is small, $\text{ATTR}_{\theta'} \approx N$ —i.e., it improves the ATTR).*

Proof: Suppose that two arbitrarily given nodes, say, nodes i and j , have their original CH sequence θ_i and θ_j with period length τ . Without loss of generality, we assume that the local clock of node i is k slots behind that of node j —i.e., the 0-th slot of node i is the k -th slot of node j . Hereafter, we number time slots in accordance with node i 's clock. Since \mathbf{x} is a wake-up schedule, we have $\exists 0 \leq t_0 \leq T-1$ such that

$$\mathbf{x}_{t_0} = \text{rotate}(\mathbf{x}, k)_{t_0} = 1.$$

We focus on the $(aT + t_0)$ -th slots, $a = 0, 1, 2, \dots$. Suppose that in the t_0 -th slot, node i hops onto channel θ_{i,c_1} while node j hops onto channel θ_{j,c_2} . Let A be $\sum_{t=0}^{T-1} \mathbf{x}_t$. In view of Algorithm 1, node i hops onto channel θ_{i,c_1+aA} while node j hops onto channel θ_{j,c_2+aA} in the $(aT + t_0)$ -th time slot.

Since $\gcd(\tau, A) = 1$, we have

$$\{((c_1 + aA) \bmod \tau, (c_2 + aA) \bmod \tau)\}_{a=0,1,2,\dots,\tau-1}$$

is a permutation of

$$\{(a, (c_2 - c_1) + a)\}_{a=0,1,2,\dots,\tau-1}.$$

Consider

$$\{(\theta_{i,a}, \theta_{j,(c_2-c_1)+a})\}_{a=0,1,2,\dots,\tau-1}.$$

Since θ is a CH protocol, given a clock drift of $(c_2 - c_1)$ slots, two nodes that use θ_i and θ_j as their CH sequences respectively will rendezvous, say, on \tilde{C} different channels (\tilde{C} is in fact the rendezvous diversity of the original CH protocol θ) and \tilde{T} different time slots in τ consecutive slots. Therefore there are \tilde{T} solutions to the equation

$$\theta_{i,a} = \theta_{j,(c_2-c_1)+a}$$

w.r.t. $a = 0, 1, 2, \dots, \tau - 1$. For $a = 0, 1, 2, \dots, \tau - 1$, when $\theta_{i,a} = \theta_{j,(c_2-c_1)+a}$, $\theta_{i,a}$ can take \tilde{C} different values, which implies that the new hybrid CH protocol θ' under our proposed framework achieves at least the same rendezvous diversity as the original CH protocol.

Hence, we conclude that the new CH sequence can guarantee rendezvous within τT slots and that it preserves the rendezvous channel diversity of the original CH sequence.

Now we calculate the ATTR of the new CH sequence. Let B be

$$|\{t \in [0, T - 1] \cap \mathbb{N} : \mathbf{x}_t = \text{rotate}(\mathbf{x}, k)_t = 1\}|.$$

Suppose X and Y are uniformly random in $[1, N] \cap \mathbb{N}$ and independent, we have $\Pr[X = c] = \Pr[X = Y] = \frac{1}{N}$ for any fixed $c \in [1, N] \cap \mathbb{N}$. The ATTR of the new CH sequence will be

$$\text{ATTR}_{\theta'} = \frac{B}{T} \cdot \text{ATTR}_{\theta} + (1 - \frac{B}{T}) \cdot N.$$

If $\frac{B}{T}$ is small, we have $\text{ATTR}_{\theta'} \approx N$. ■

IV. PERFORMANCE EVALUATION

A. Simulation Setup

In this section, we evaluate and compare the performance of existing protocols (e.g., CRSEQ [11] and Jump-Stay (JS) [10]) and those hybrid protocols extended from our proposed framework (i.e., after the interleaving operation) via simulation results.

In our simulations, there are a total number of $N = 11$ frequency channels and 20 pairs of nodes that need to rendezvous via channel hopping. Meanwhile, each node determines its clock time randomly and independently, yielding a stochastic clock drift between each pair of nodes. We simulated $X < N$ primary transmitters operating on X randomly chosen channels. A timeslot has a length of 10 ms. All secondary nodes are within the transmission range of any primary transmitter. In most existing work, it is assumed that a primary transmitter follows a “busy/idle” transmission pattern on a licensed channel [7], and we assume the same traffic pattern here—i.e., the busy period has a fixed length of b timeslots, and the idle period follows an exponential distribution with a mean

of l timeslots. A channel is viewed as “unavailable” when PU signals are present in it. The PU traffic is the probability that PU signals are active in a channel under the specified traffic pattern. Once two nodes hop onto a primary-user free channel in the same timeslot, the rendezvous between them is established.

We conduct simulations while varying the duty cycle of wake-up schedules and the PU traffic. Specifically, we choose wake-up schedules with duty cycles being 5/14, 7/14, 9/14, 13/14 and 1, under PU traffic of 25% and 50%. Note that if the duty cycle of the wake-up schedule happens to be 1, the resulting CH protocol is exactly the *original CH protocol*—no random CH process interleaved. And we employ the three metrics defined in Sec. II-B to evaluate the performance of rendezvous protocols.

B. Maximum TTR

The results for MTTR of original CRSEQ/Jump-stay and those resulting from our framework are shown in Fig. 4. Note that if the duty cycle is 1, the resulting CH protocol generated by our framework is exactly the original CH protocol (the original CRSEQ or Jump-stay).

We can observe that despite different PU traffic intensity and for both CRSEQ and Jump-stay, a larger duty cycle leads to an increased MTTR (i.e., worse performance). The original CRSEQ/Jump-stay has the longest MTTR compared with those generated by our framework.

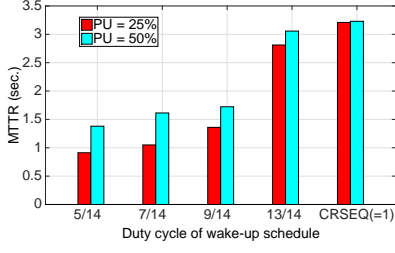
This confirms that the proposed framework can preserve bounded TTR (shown in Theorem 1) and improve MTTR in addition. With the import of random channel hopping slots, our framework achieves better worst-case performance.

C. Average TTR

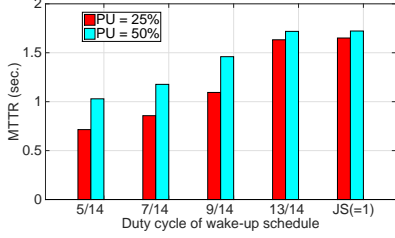
We illustrate the results for ATTR of original CRSEQ/Jump-stay and those resulting from our framework in Fig. 5. It is noteworthy that a larger duty cycle positively correlates with ATTR for both protocols and under different PU intensity. Specifically, the original CRSEQ/Jump-stay (duty cycle equals 1) has the worst performance in ATTR. This validates the theoretical analysis in Theorem 1. A smaller duty cycle implies more stochastic ingredient (an increased number of random slots are interleaved into the resulting CH sequence), which, in turn, significantly improves the average performance. To summarize, the CH protocols generated by our framework outperform the original protocols (original CRSEQ/Jump-stay).

D. Rendezvous Channel Diversity Rate

Rendezvous channel diversity rate is a metric for measuring a CH protocol’s ability to establish rendezvous in varied channels. The results for diversity rates of original CRSEQ/Jump-stay and those resulting from our framework are presented in Fig. 6(a) and 6(b), respectively. It can be observed that diversity rate declines as the duty cycle increases and that the original CRSEQ/Jump-stay has the worst/smallest diversity rate. This observation supports the conclusion of Theorem 1 that our proposed framework can preserve diversity rate. In addition, it validates that by introducing stochastic CH slots, we can obtain an improved diversity rate.

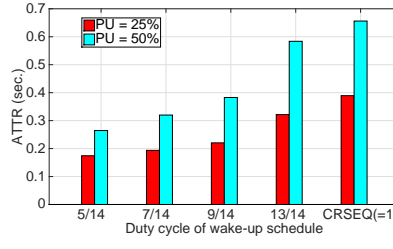


(a) Interleaving CRSEQ and random CH

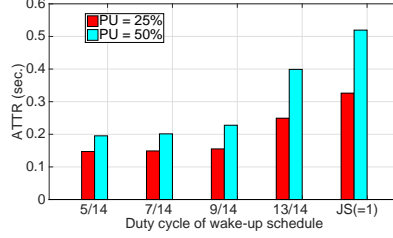


(b) Interleaving Jump-stay and random CH

Figure 4: Maximum TTR of hybrid protocols (duty cycle smaller than 1) and the original CRSEQ/Jump-stay protocol (duty cycle equals 1). We interleave CRSEQ (or Jump-stay) with random channel hopping in Fig. 4(a) (or Fig. 4(b)).

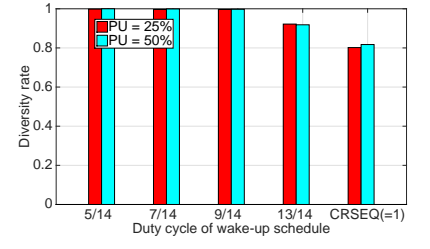


(a) Interleaving CRSEQ and random CH

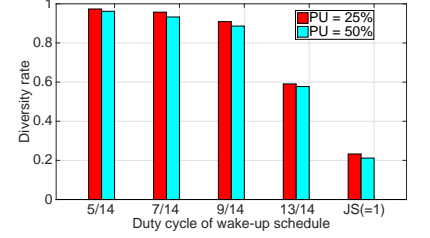


(b) Interleaving Jump-stay and random CH

Figure 5: Average TTR of hybrid protocols (duty cycle smaller than 1) and the original CRSEQ/Jump-stay protocol (duty cycle equals 1). We interleave CRSEQ (or Jump-stay) with random channel hopping in Fig. 5(a) (or Fig. 5(b)).



(a) Interleaving CRSEQ and random CH



(b) Interleaving Jump-stay and random CH

Figure 6: Diversity rate of hybrid protocols (duty cycle smaller than 1) and the original CRSEQ/Jump-stay protocol (duty cycle equals 1). We interleave CRSEQ (or Jump-stay) with random channel hopping in Fig. 6(a) (or Fig. 6(b)).

V. CONCLUSION

In this paper, by leveraging the properties of neighbor discovery wake-up schedules, we establish a design framework for creating a series of hybrid CH protocols whereby every SU employs the specified wake-up schedule to interleave the sequence-based (or random) CH protocol in the awake (or asleep) mode. Analytical and simulation results show that the hybrid CH rendezvous protocols under the proposed framework can significantly improve average time-to-rendezvous and preserve a low upper-bound of TTR and the rendezvous channel diversity simultaneously. It is also validated by extensive simulation results that our method remarkably outperforms existing CH protocols, CRSEQ and Jump-stay.

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*, 50(13):2127–2159, 2006.
- [2] M. Bakht, M. Trower, and R. H. Kravets. Searchlight: won't you be my neighbor? In *Proceedings of the 18th annual international conference on Mobile computing and networking*, pages 185–196. ACM, 2012.
- [3] K. Bian and J.-M. Park. Asynchronous channel hopping for establishing rendezvous in cognitive radio networks. In *INFOCOM, 2011 Proceedings IEEE*, pages 236–240. IEEE, 2011.
- [4] K. Bian and J.-M. Park. Maximizing rendezvous diversity in rendezvous protocols for decentralized cognitive radio networks. *Mobile Computing, IEEE Transactions on*, 12(7):1294–1307, 2013.
- [5] I. Chuang, H.-Y. Wu, K.-R. Lee, and Y.-H. Kuo. Alternate hop-and-wait channel rendezvous method for cognitive radio networks. In *INFOCOM, 2013 Proceedings IEEE*, pages 746–754. IEEE, 2013.
- [6] P. Dutta and D. Culler. Practical asynchronous neighbor discovery and rendezvous for mobile sensing applications. In *Proceedings of the 6th ACM conference on Embedded network sensor systems*, pages 71–84. ACM, 2008.
- [7] S. Geirhofer, L. Tong, and B. M. Sadler. Cognitive medium access: constraining interference based on experimental models. *Selected Areas in Communications, IEEE Journal on*, 26(1):95–105, 2008.
- [8] Z. Gu, Q.-S. Hua, and W. Dai. Fully distributed algorithms for blind rendezvous in cognitive radio networks. In *Proceedings of the 15th ACM international symposium on Mobile ad hoc networking and computing*, pages 155–164. ACM, 2014.
- [9] A. Kandhalu, K. Lakshmanan, and R. R. Rajkumar. U-connect: a low-latency energy-efficient asynchronous neighbor discovery protocol. In *Proceedings of the 9th ACM/IEEE International Conference on Information Processing in Sensor Networks*, pages 350–361. ACM, 2010.
- [10] Z. Lin, H. Liu, X. Chu, and Y.-W. Leung. Jump-stay based channel-hopping algorithm with guaranteed rendezvous for cognitive radio networks. In *INFOCOM, 2011 Proceedings IEEE*, pages 2444–2452. IEEE, 2011.
- [11] J. Shin, D. Yang, and C. Kim. A channel rendezvous scheme for cognitive radio networks. *Communications Letters, IEEE*, 14(10):954–956, 2010.
- [12] C. R. Stevenson, G. Chouinard, Z. Lei, W. Hu, S. J. Shellhammer, and W. Caldwell. IEEE 802.22: The first cognitive radio wireless regional area network standard. *IEEE Communications Magazine*, 47(1):130–138, 2009.
- [13] Y. Zhang, Q. Li, G. Yu, and B. Wang. Etch: Efficient channel hopping for communication rendezvous in dynamic spectrum access networks. In *INFOCOM, 2011 Proceedings IEEE*, pages 2471–2479. IEEE, 2011.