Multiple Radios for Fast Rendezvous in Cognitive Radio Networks

Lu Yu, Hai Liu, Yiu-Wing Leung, Xiaowen Chu, and Zhiyong Lin

Abstract—Rendezvous is a fundamental operation in cognitive radio networks (CRNs) for establishing a communication link on a commonly-available channel between cognitive users. The existing work on rendezvous implicitly assumes that each cognitive user is equipped with one radio (i.e., one wireless transceiver). As the cost of wireless transceivers is dropping, this feature can be exploited to significantly improve the rendezvous performance at low cost. In this study, we investigate the rendezvous problem in CRNs where cognitive users are equipped with multiple radios and different users may have different numbers of radios. We first study how the existing rendezvous algorithms can be generalized to use multiple radios for faster rendezvous. We then propose a new rendezvous algorithm, called *role-based parallel sequence* (RPS), which specifically exploits multiple radios for more efficient rendezvous. Our basic idea is to let the cognitive users stay in a specific channel in one *dedicated radio* and hop on the available channels with parallel sequences in the remaining *general radios*. We prove that our algorithm provides guaranteed rendezvous (i.e., rendezvous can be completed within a finite time) and derive the upper bounds on the maximum time-to-rendezvous (TTR) and the expected TTR. The simulation results show that i) multiple radios can cost-effectively improve the rendezvous performance, and ii) the proposed RPS algorithm performs better than the ones generalized from the existing algorithms.

Index Terms—cognitive radio, blind rendezvous, channel hopping

1 Introduction

With the traditional static spectrum management, a significant portion of the licensed spectrum is underutilized in most of time while the unlicensed spectrum is over-crowded due to the growing demand for wireless radio spectrum from exponential growth of various wireless devices [1]. Dynamic Spectrum Access utilizes the wireless spectrum in a more intelligent and flexible way. Cognitive radios are a promising enabler for Dynamic Spectrum Access because they can sense and access the idle channels. With cognitive radios, the unlicensed users (SUs) can opportunistically identify and access the vacant portions of the spectrum of the licensed users (PUs).

In cognitive radio networks (CRNs), multiple idle channels may be available to SUs. If two or more SUs want to communicate with each other, they must select a channel which is available to all of them. The process of two or more SUs to meet and establish a link on a commonly-available channel is known as *rendezvous* [1]. Rendezvous is a fundamental and essential operation for establishing communication links of SUs. Channel-hopping (CH) is one of the most representative techniques for rendezvous. With CH technique, each SU selects a set of available channels and hops among these channels. A rendezvous is said to be achieved if two SUs hop on the same channel simultaneously.

Many effective rendezvous algorithms have been proposed in the literature and they are described in Section 2. To the best of our knowledge, all the existing rendezvous algorithms implicitly assume that each user is equipped with one radio (i.e., one wireless transceiver). As the cost of wireless transceivers is dropping, this feature can be exploited to significantly improve the rendezvous performance at low cost. In particular, when a SU is equipped with multiple radios, the time-to-rendezvous (TTR, i.e., the time required by the rendezvous operation) can potentially be reduced by a large amount while the additional cost (i.e., cost of the extra radios) is low. In addition, the energy consumption can also be reduced (if the number of radios is increased from 1 to n, n radios would consume energy but the time spent on rendezvous could be reduced by more than n times, so the total energy consumption can be reduced).

In this paper, we study the rendezvous problem in CRNs where each SU is equipped with multiple radios and different SUs may have different numbers of radios. We make three contributions.

1) We investigate a new approach (i.e., exploiting multiple radios per user) to significantly improving the rendezvous performance at low cost.
2) We generalize the Random algorithm and the existing rendezvous algorithms in order to use multiple radios for faster rendezvous.
3) We propose a new rendezvous algorithm, called *role-based parallel sequence* (RPS), which specifically exploits multiple radios for more efficient rendezvous. We derive upper bounds on the maximum TTR (MTTR) and the expected TTR (E(TTR)) of this algorithm. We conduct extensive simulation to demonstrate that its MTTR and E(TTR) decrease significantly with the increase of the number of radios.

In the literature, there are two models to describe the channel availability [4, 9, 20, 21]: i) *symmetric model* in which all users have the same available channels; and ii) *asymmetric model* in which different users may have different available channels. Both the symmetric

- *L. Yu, H. Liu, Y.-W. Leung, X. Chu, and Z. Lin are with the Department of Computer Science, Hong Kong Baptist University, Kowloon Tong, Hong Kong SAR.*
  E-mail: lyu, hliu, ywleung, chxw, zylin@comp.hkbu.edu.hk.
- *A preliminary version was presented at ICC2013.*

TABLE 1
COMPARING THE UPPER BOUNDS OF MTTR

	Algorithms	Symmetric Model	Asymmetric Model
Existing algorithms generalized to multiple radios	Random	Infinity	Infinity
	JS/Independent	$3P$	$3MP(P-G)+3P$
	JS/Parallel	$\lceil\frac{3P}{m}\rceil$ when $m=n$ Infinity when $m \neq n$	$\lceil\frac{3MP(P-G)+3P}{m}\rceil$when $m=n$ Infinity when $m \neq n$
New Algorithm	RPS	$2 \times \lceil\frac{P}{max\{m,n\}-1}\rceil - 1$	$(\lceil\frac{P}{m-1}\rceil \times (Q-G+1))$ when $m=n$ $(2 \times \lceil\frac{P}{max\{m,n\}-1}\rceil - 1) + \lceil\frac{P}{min\{m,n\}-1}\rceil \times (Q-G)$ when $m \neq n$

TABLE 2
COMPARING THE UPPER BOUNDS OF E(TTR)

	Algorithms	Symmetric Model	Asymmetric Model
Existing algorithms generalized to multiple radios	Random	$\frac{A_Q^m A_Q^n}{A_Q^m A_Q^n - A_Q^{m+n}}$	$\frac{A_{\lvert C_1\rvert}^m A_{\lvert C_2\rvert}^n}{A_{\lvert C_1\rvert}^m A_{\lvert C_2\rvert}^n - A_{\lvert C_1\rvert-G}^m A_{\lvert C_2\rvert-G}^n}$
	JS/Independent	$\frac{5P-4}{3} + \frac{1}{3Q^{m+n-1}} \times (2P+1+\frac{1}{P})$	$\frac{5P-4}{3} + \frac{1}{3Q^{m+n-1}} \times (2P+1+\frac{1}{P})$ $+3QP \times (Q-G)$
	JS/Parallel	$\lceil\frac{5P/3+3}{m}\rceil$when $m=n$ Infinity when $m \neq n$	$\lceil\frac{2QP(P-G)+(Q+5-P-(2G-1)/Q)P}{m}\rceil$when $m=n$ Infinity when $m \neq n$
New Algorithm	RPS	$(\lceil\frac{P}{m-1}\rceil)$ when $m=n$ $\lceil\frac{P}{max\{m,n\}-1}\rceil + \frac{(\lceil\frac{P}{max\{m,n\}-1}\rceil-1)^2}{2\times\lceil\frac{P}{min\{m,n\}-1}\rceil}$ when $m \neq n$	$(\lceil\frac{P}{m-1}\rceil \times (Q-G+1))$ when $m=n$ $\lceil\frac{P}{max\{m,n\}-1\rceil} + \frac{(\lceil\frac{P}{max\{m,n\}-1}\rceil-1)^2}{2\times\lceil\frac{P}{min\{m,n\}-1}\rceil}$ $+\lceil\frac{P}{min\{m,n\}-1}\rceil \times (Q-G)$ when $m \neq n$

Remarks: i) m, n are the numbers of radios of two users; Q is the number of all channels; P is the smallest prime number which is not smaller than Q; G is the number of commonly-available channels of two users; $|C_1|$, $|C_2|$ are the numbers of available channels of two users, respectively; A_j^i is the number of possible permutations of i objects from a set of j. ii) We select the Jump-Stay (JS) algorithm [9] for comparison since it was recently proposed and was shown to have a very good performance [9]. The upper bounds of MTTR and E(TTR) of the existing algorithms are derived in Section 4. iii) In a CRN, different SUs may be equipped with different numbers of radios.

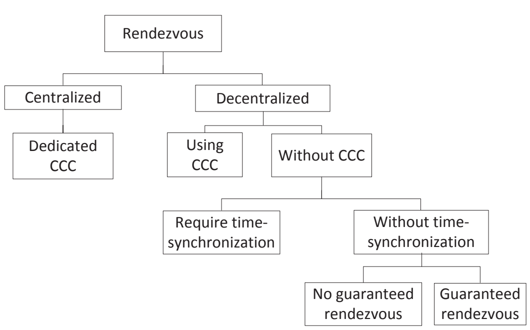


Fig. 1. A taxonomy of the existing rendezvous algorithms.

and asymmetric models are important in practice. For example, the symmetric model is suitable for SUs who are located in a relatively small area (compared with their distance to PUs) while the asymmetric model is applicable if geographical locations of SUs are far. Table 1 and Table 2 summarize the differences between: i) the proposed algorithm, and ii) the existing rendezvous algorithms after they are generalized to use multiple radios.

The rest of this paper is organized as follows. Related works are reviewed in Section 2. System model and problem formulation are presented in Section 3. In Section 4, we generalize the Random algorithm and the existing rendezvous algorithms to use multiple radios for faster rendezvous. Then we propose a new rendezvous algorithm which specifically exploits multiple radios for more efficient rendezvous. We present simulation results in Section 5 for performance evaluation and conclude our work in Section 6.

2 RELATED WORK

The existing CH algorithms can be classified into two categories based on their structures: i) centralized systems where a central server is preselected to allocate the spectrum for all SUs, and ii) decentralized systems where there is no central server. The decentralized systems can be further classified into two subcategories: i) using a common control channel (CCC), and ii) not using CCC. Fig. 1 shows a possible taxonomy of the existing rendezvous algorithms.

Centralized systems: Under centralized system, such as DSAP[3] and DIMSUMNet [4], a centralized server is operated to schedule the data exchanges among users. With a centralized server for global coordination, this approach eases the rendezvous process but it involves the overhead of maintaining the server and this server is a single point of failure [2].

Decentralized systems using CCC: In decentralized system, a channel is preselected as a CCC. In [5] and [6], a global CCC is preselected and known to all users. In [7] and [22], a cluster-based control-channel method was proposed, in which a local CCC is selected for each cluster. However, the extra costs in establishing and maintaining the global/local CCCs are considerable.

Decentralized systems without using CCC: This approach does not use CCC and hence it is known as blind rendezvous [9]. With this desirable feature, this approach has drawn significant attention in the literature and some effective algorithms for blind rendezvous have been proposed (e.g., Jump-Stay [8][9], M-/L-QCH [10] and ASYNC-ETCH [11]). Time-synchronization is a key point we should pay attention to in blind rendezvous. Based on it we can further divide decentralized systems without using CCC into two types (Fig. 1).

- *Algorithms that require time-synchronization*: Many CH algorithms which require time-synchronization have been proposed. Two CH algorithms M-/L-QCH were proposed in [10] based on quorum systems which can guarantee a rendezvous of users under the symmetric model. Another CH algorithm called A-QCH was proposed in [10] for the asynchronous systems. Bahl et al. designed a link-layer protocol named SSCH [12]. Each user can select more than one pairs and generate the CH sequence based on these pairs. It was designed to increase the capacity of IEEE 802.11 networks. However, similar to M-/L-QCH, they are not applicable in the asymmetric model. Authors in [23] proposed a deterministic approach in which each user is scheduled to broadcast on every channel in an exhaustive manner.
- *Algorithms that work without time-synchronization*: There are CH algorithms which do not require time-synchronization. In [16], they presented a ring-walk (RW) algorithm which guarantees the rendezvous under both models. In RW, each channel is represented as a vertex in a ring. Users walk on the ring by visiting the vertices (channels) with different velocities and rendezvous is guaranteed since users with lower velocities will be caught by users with higher velocities. However, RW requires that each user has a unique ID and knows the upper bound of network size. Recently, a notable work by Theis et al. presented two CH algorithms: modular clock algorithm (MC) and its modified version MMC for the symmetric model and the asymmetric model, respectively [14]. The basic idea of MC and MMC is that each user picks a proper prime number and randomly selects a rate less than the prime number. Based on the two parameters, the user generates its CH sequence via predefined modulo operations. Although MC and MMC are shown to be effective, both algorithms cannot guarantee the rendezvous if the selected rates or the prime numbers of two users are identical. Yang et al. proposed two significant algorithms, namely deterministic rendezvous sequence (DRSEQ) [18] and channel rendezvous sequence (CRSEQ) [19], which provide guaranteed rendezvous for the symmetric model and the asymmetric model, respectively. In CRSEQ, the sequence is constructed based on triangle numbers and modular operations. In terms of MTTR, CRSEQ is quite good under the asymmetric model but it does not perform well under the symmetric model. Bian et al. [24] presented an asynchronous channel hopping (ACH) algorithm which aims to maximize rendezvous diversity. It assumes that each user has a unique ID and ACH sequences are designed based on the user ID. Though the length of user ID is a constant, it may result in a long TTR in practice given that a typical MAC address contains 48 bits. In [27], an efficient rendezvous algorithm based Disjoint Relaxed Difference Set (DRDS) was proposed while DRDS only costs a linear time to construct. They proposed a distributed asynchronous algorithm that can achieve and guarantee fast rendezvous under both the symmetric and the asymmetric models. They derived the lower bounds of MTTR which are $3P$ and $3P^2 + 2P$ under the symmetric model and the asymmetric model, respectively. They showed that it is nearly optimal. In [28], the authors addressed the pairwise as well as the multicast rendezvous problems under fast primary user (PU) dynamics. They considered the issue of adaptively adjusting the channel hopping (CH) sequences to account for the spectrum heterogeneity and PU dynamics, which are the two main challenges in DSA systems. Wu et al. proposed an effective rendezvous algorithm called Heterogeneous Hopping (HH) which is based on the available channel set only. There are other algorithms in this category such as synchronous QCH [10], SYNCETCH and ASYNC-ETCH [11], AMRCC [15], C-MAC [17], and MtQS-DSrdv [25]. Due to limited space, these algorithms are not reviewed and readers may refer to the survey in [2] for details. Most of the existing rendezvous algorithms use one single radio per user for rendezvous. The authors in [29] studied the multi-interface rendezvous problems for CRNs. However, they assumed that all nodes have the same number of radios (say, n), the channels are divided into n partitions, and each radio performs rendezvous over the channels in one partition.

3 SYSTEM MODEL AND PROBLEM FORMULATION

We consider a CRN consisting of K ($K \geq 2$) users. Time is divided into slots of equal duration. The licensed spectrum is divided into Q non-overlapping channels $c_1, c_2, ..., c_Q$, where c_i is called channel i. Let C be the channel set $\{c_1, c_2, ..., c_Q\}$. Let $C_i \in C$ be the set of available channels of user i ($i = 1, 2, ..., K$), where a channel is said to be available to a user if the user can communicate on this channel without causing interference to any PUs. The available channels can be identified by any spectrum sensing method (e.g., [4]). Without loss of generality, we consider the rendezvous of a pair of users, say user i and user j, $i \neq j$ and $i, j = 1, 2, ..., K$. User i is equipped with

m ($m \geq 1$) radios and user j is equipped with n ($n \geq 1$) radios. Note that m may not be equal to n in CRNs. Let G be the number of commonly-available channels of user i and user j. The CH sequence of user i is denoted by $\{\overrightarrow{S_1^i}, \overrightarrow{S_2^i}, \overrightarrow{S_3^i}, ...\}$, where vector $\overrightarrow{S_t^i} = \{S_{t1}^i, S_{t2}^i, S_{t3}^i, ..., S_{tm}^i\}$ represents that user i hops on channels S_{th}^i on radio h in time slot t. Fig. 2 shows the sequence structure where user i has 3 radios and user j has 2 radios. For any given and fixed C_i and C_j where C_i and C_j have at least one commonly available channel, if a rendezvous algorithm can ensure that rendezvous can be completed within a finite time, we say that this algorithm provides *guaranteed rendezvous*.

In this study, we do not require time-synchronization in the networks. Without time synchronization, the slot time could be doubled so as to ensure that the overlap of two time slots is long enough to complete all necessary steps of rendezvous [14] [19]. In this sense, the CH sequences of two users are slot-aligned even without time-synchronization [2]. In each time slot, user i hops on m channels and user j hops on n channels to attempt rendezvous. We say that a rendezvous is achieved if user i and user j hop on the same channel on any of the radios in the same time slot. Typically the time-to-rendezvous (TTR) is usually in the order of tens of milliseconds, which is very small compared with the PU dynamic. To illustrate, let us consider an example. Suppose it is necessary to send a handshaking packet of 100 bytes (e.g., containing information of user IDs) for rendezvous and the data rate of the wireless channel is 10 Mbps. That is, The duration of each frame is $(100 \times 8)/(10 \times 10^6) = 0.08$ms. Suppose that the SIFS (Short Inter Frame Spacing) is 10µs (e.g., in IEEE 802.11b or IEEE 802.11n [31]) and the three-way handshaking is adopted. We can estimate the time necessary for rendezvous as $3 \times (0.08 \times 10^{-3} + 10 \times 10^{-6}) = 0.27$ms. Thus, the duration of each time slot is 0.54ms around (i.e., the overlap of two time slot is no less than 0.27ms). If it takes 10 to 100 time slots to achieve rendezvous (see the numerical results in our paper), the time-to-rendezvous is only 5.4ms to 54ms. On the other hand, a common type of primary users quoted in the literature is the TV station which only uses its TV channels at certain time of each day (say, evenings and nights), and the activity of this PU changes very slowly compared with the time-to-rendezvous. Therefore, channels availabilities are assumed to be static in the process of rendezvous. We define the rendezvous problem as follows.

Rendezvous problem for two users: Suppose two users have m and n ($m, n \geq 1$) radios respectively and they may start the rendezvous process at different time. The problem is to determine a CH sequence for each radio of each user, such that these users will hop on a commonly-available channel in the same time slot.

4 SOLUTIONS

In this section, we generalize the Random algorithm and the existing algorithms to use multiple radios for faster rendezvous. Then we design a new rendezvous algorithm, called *role-based parallel sequence* (RPS), which specifically exploits multiple radios for more efficient rendezvous.

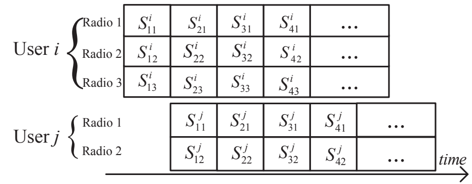


Fig. 2. Sequence structure when user i has 3 radios and user j has 2 radios.

4.1 Generalized Random Algorithm

When there is a single radio, the Random algorithm randomly selects an available channel in each time slot and attempts to achieve rendezvous on this channel in this time slot. When there are multiple radios, this Random algorithm can be generalized as follows: each radio randomly and independently selects an available channel in each time slot and attempts to achieve rendezvous on this channel in this time slot. When two or more radios happen to select the same channel, these radios will randomly select again until they select different channels (suppose that for each user the number of available channels is no less than that of radios). Obviously, the Random algorithm cannot guarantee the rendezvous within finite time and hence the MTTR is infinity. The algorithm is formally presented as follows.

Algorithm 1: Random Algorithm

Require: Q, m, C_i *//for user i*
1: $t = 1$
2: $\overrightarrow{S_t^i} = \{S_{t1}^i, S_{t2}^i, S_{t3}^i, ..., S_{tm}^i\}$
3: **while** not rendezvous **do**
4: **for** $k = 1$ to m **do**
5: $S_{tk}^i = RandomSelect(C_i)$
6: **for** $j = 1$ to $(k-1)$ **do**
7: **if** $S_{tk}^i == S_{tj}^i$ **then**
8: $S_{tk}^i = RandomSelect(C_i)$
9: $j = 1$
10: **end if**
11: **end for**
12: **end for**
13: $t = t + 1$
14: Attempt rendezvous on $\overrightarrow{S_t^i}$
15: **end while**

In line 2, $\overrightarrow{S_t^i} = \{S_{t1}^i, S_{t2}^i, S_{t3}^i, ..., S_{tm}^i\}$ denotes the set of channels that user i hops on m respective radios in time slot t. In lines 4-12, user i randomly selects an available channel for each radio. In line 14, the rendezvous is achieved when one channel in $\overrightarrow{S_t^i}$ is equal to one channel of another user. The following theorem gives the performance properties of the generalized Random algorithm.

Theorem 1. *The E(TTR) of the generalized Random algorithm is equal to* $\frac{A_Q^m A_Q^n}{A_Q^m A_Q^n - A_Q^{m+n}}$ *under the symmetric model where Q is the number of all channels and* A_j^i *is the number of possible permutations of* i *objects from a set of* j.

Proof: In time slot k, user i hops on m different channels $\overrightarrow{S_k^i}$ and user j on n different channels $\overrightarrow{S_k^j}$ (Fig. 2). If any channel in $\overrightarrow{S_k^i}$ is the same as one channel in $\overrightarrow{S_k^j}$, the rendezvous is achieved. In any time slot, each channel has Q choices. The probability that all m channels of user i are not equal to any channel of user j is $\frac{A_Q^{m+n}}{A_Q^m A_Q^n}$. So $p(TTR=t)=(\frac{A_Q^{m+n}}{A_Q^m A_Q^n})^{t-1}(1-\frac{A_Q^{m+n}}{A_Q^m A_Q^n})$. The $E(TTR)=\sum_{i=1}^{+\infty} t\times p(TTR=t)=\sum_{i=1}^{+\infty} t\times(\frac{A_Q^{m+n}}{A_Q^m A_Q^n})^{t-1}(1-\frac{A_Q^{m+n}}{A_Q^m A_Q^n})=\frac{1}{1-\frac{A_Q^{m+n}}{A_Q^m A_Q^n}}=\frac{A_Q^m A_Q^n}{A_Q^m A_Q^n-A_Q^{m+n}}$. □

Theorem 2. *The E(TTR) of the generalized Random algorithm is equal to* $\frac{A^m_{|C_1|}A^n_{|C_2|}}{A^m_{|C_1|}A^n_{|C_2|}-A^m_{|C_1|-G}A^n_{|C_2|-G}}$ *under the asymmetric model where* $|C_1|$ *and* $|C_2|$ *are the numbers of available channels of two users and* G *is the number of commonly-available channels of the two users.*

Proof: In time slot k, the probability that all m channels of user i are not equal to any channel of user j is $\frac{A^m_{|C_1|-G}A^n_{|C_2|-G}}{A^m_{|C_1|}A^n_{|C_2|}}$. So $p(TTR=t)=(\frac{A^m_{|C_1|-G}A^n_{|C_2|-G}}{A^m_{|C_1|}A^n_{|C_2|}})^{t-1}(1-\frac{A^m_{|C_1|-G}A^n_{|C_2|-G}}{A^m_{|C_1|}A^n_{|C_2|}})$. The $E(TTR)=\sum_{i=1}^{+\infty} t\times p(TTR=t)=\sum_{i=1}^{+\infty} t\times(\frac{A^m_{|C_1|-G}A^n_{|C_2|-G}}{A^m_{|C_1|}A^n_{|C_2|}})^{t-1}(1-\frac{A^m_{|C_1|-G}A^n_{|C_2|-G}}{A^m_{|C_1|}A^n_{|C_2|}})=\frac{1}{1-\frac{A^m_{|C_1|-G}A^n_{|C_2|-G}}{A^m_{|C_1|}A^n_{|C_2|}}}=\frac{A^m_{|C_1|}A^n_{|C_2|}}{A^m_{|C_1|}A^n_{|C_2|}-A^m_{|C_1|-G}A^n_{|C_2|-G}}$. □

4.2 Generalized Existing Rendezvous Algorithms

In the literature, several rendezvous algorithms have been proposed for CRNs. They implicitly assume one radio per user. In this subsection, we generalize these existing rendezvous algorithms such that each user can use multiple radios for faster rendezvous. We consider two strategies to generalize these algorithms to use multiple radios:

1) *Independent Sequence*: Apply an existing rendezvous algorithm to independently generate a CH sequence for each radio. If this algorithm always generates the same CH sequence using a deterministic method, then the sequence is rotated by x positions where x is a randomly generated integer. For example, suppose a user is equipped with two radios and an existing rendezvous algorithm generates two CH sequences $\{s_1, s_2, ...\}$ and $\{r_1, r_2, ...\}$. In the first time slot, the two radios hop on channels s_1 and r_1 respectively. In the second time slot, the two radios hop on channels s_2 and r_2 respectively.
2) *Parallel Sequence*: Apply an existing algorithm to generate a CH sequence and apply this CH sequence on all radios in parallel. For example, suppose a user is equipped with three radios and an existing rendezvous algorithm generates the CH sequence $\{s_1, s_2, s_3, s_4, s_5, s_6, ...\}$. In the first time slot, the three radios hop on channels s_1, s_2 and s_3, respectively. In the second time slot, the three radios hop on channels s_4, s_5 and s_6, respectively.

We propose two schemes (namely, independent sequence and parallel sequence) to generalize the existing single-radio algorithms to use multiple radios. While the ideas of these schemes are simple, their theoretical analysis is not trivial. When different nodes have different number of radios, only the independent sequence scheme works. Nevertheless, the parallel sequence scheme also has its own merit: when the nodes have the same number of radios, the parallel sequence scheme works and it gives better performance than the independent sequence scheme. Therefore, we report both schemes in our paper.

When the Independent Sequence strategy is applied to generalize an existing rendezvous algorithm to use multiple radios for faster rendezvous, the steps are given in Algorithm 2.

Algorithm 2: Independent Sequence

Require: Q, m, A, C_i *//an existing algorithm denoted by A and user i*
1: $t = 1$
2: $\overrightarrow{S_t^i} = \{S_{t1}^i, S_{t2}^i, S_{t3}^i, ..., S_{tm}^i\}$
3: **for** $k = 1$ to m **do**
4:   **if** A is a deterministic method and the sequence is S_{At} **then**
5:     $S_{Akt} = S_{At}$ rotated by random positions
6:   **else**
7:     S_{Akt} is generated by algorithm A
8:   **end if**
9: **end for**
10: **while** not rendezvous **do**
11:   **for** $k = 1$ to m **do**
12:     $S_{tk}^i = S_{Akt}$
13:   **end for**
14:   $t = t + 1$
15:   Attempt rendezvous on $\overrightarrow{S_t^i}$
16: **end while**

In line 3, since the user has m radios, it generates m independent CH sequences by an existing algorithm. In lines 4-8, k-th sequences are generated by this algorithm. In line 12, the $k\text{-}th$ sequence is performed in the $k\text{-}th$ radio. Let $MTTR_0$ be the MTTR of the existing rendezvous algorithm using one radio. The following theorem gives the MTTR of Algorithm 2.

Theorem 3. *If two users perform an existing algorithm on multiple radios with Independent Sequence, the maximum time-to-rendezvous (MTTR) is equal to* $MTTR_0$ *under both the symmetric model and the asymmetric model.*

Proof: When the users run an existing algorithm independently on multiple radios, each pair of radios of the two users will achieve rendezvous on or before $MTTR_0$. As a result, the two users will achieve a guaranteed rendezvous on or before $MTTR_0$ by any pair of radios. So $MTTR \le MTTR_0$. Next we prove $MTTR \ge MTTR_0$. Consider a worst case in which all the m radios of a user use the same sequence (e.g., the existing algorithm happens to independently generate m identical sequences). In this case, the rendezvous is the same as the original algorithm. So

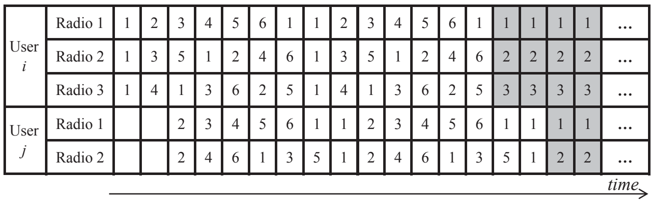
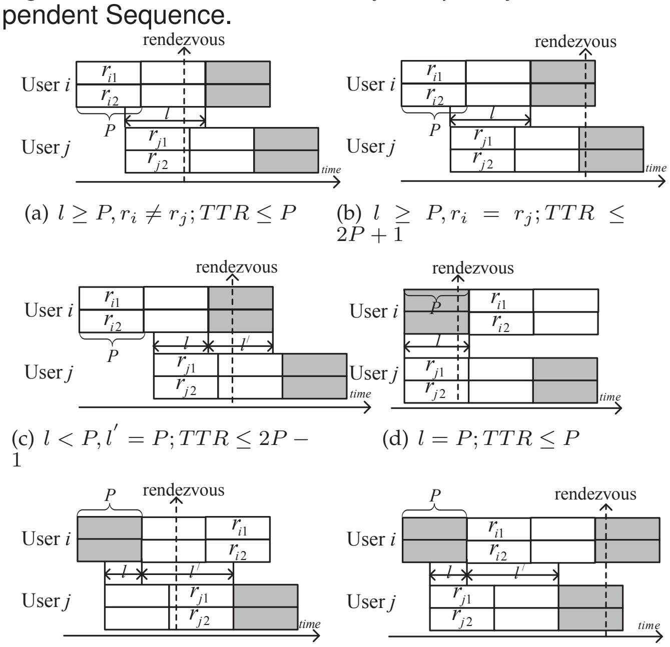


Fig. 3. Rendezvous achieved by Jump-Stay with Independent Sequence.



(a) $l \geq P, r_i \neq r_j; TTR \leq P$ (b) $l \geq P, r_i = r_j; TTR \leq 2P+1$

(c) $l < P, l' = P; TTR \leq 2P-1$ (d) $l = P; TTR \leq P$

(e) $l < P, r_i \neq r_j; TTR \leq 2P-1$ (f) $l < P, r_i = r_j; TTR \leq 3P-1$

Fig. 4. Six cases under the symmetric model. Remarks: We say $r_i = r_j$ (Fig. 4(b) and Fig. 4(f)) if all step-lengths of CH sequences with user i are identical and are equal to those with user j, and $r_i \neq r_j$ (Fig. 4(a) and Fig. 4(e)) otherwise.

$MTTR \geq MTTR_0$. Overall, the MTTR is equal to $MTTR_0$. □

Among the existing rendezvous algorithms, the Jump-Stay algorithm performs well in terms of MTTR and E(TTR) [9]. When the Jump-Stay algorithm is generalized by the Independent Sequence strategy, we have the following results.

Corollary 1. *Under the symmetric model, any two users performing the Jump-Stay algorithm on multiple radios with Independent Sequence achieve rendezvous in at most $3P$ time slots which is an upper bound of MTTR. The E(TTR) is not greater than $\frac{5P-4}{3}+\frac{1}{3Q^{m+n-1}}\times(2P+1+\frac{1}{P})$, where P is the smallest prime number which is not smaller than Q.*

Proof: Fig. 3 illustrates how the sequences are generated. Suppose that $Q = 6$ and $P = 7$. User i has 3 radios with $(i_1 = 1, r_1 = 1)$, $(i_2 = 1, r_2 = 2)$, and $(i_3 = 1, r_3 = 3)$. User j has 2 radios with $(i_1 = 2, r_1 = 1)$ and $(i_2 = 2, r_2 = 2)$. The rendezvous is achieved in time slot 2 when user i hops on channels $(4, 1, 3)$ while user j hops on channels $(3, 4)$.

Fig. 4 shows the six cases when users perform the Jump-Stay algorithm independently on multiple radios. Step-length r takes integer value in $[1, Q]$; two users select different step-lengths, $r_{k_1} \neq r_{k_2}$, with probability $(1-\frac{1}{Q^{m+n-1}})$ while select the same step-length with probability $\frac{1}{Q^{m+n-1}}$. Thus, we compute the occurrence probabilities of six cases when all users have multiple radios are $\frac{2}{3}\times\frac{P+1}{2P}\times(1-\frac{1}{Q^{m+n-1}})$, $\frac{2}{3}\times\frac{P+1}{2P}\times\frac{1}{Q^{m+n-1}}$, $\frac{2}{3}\times\frac{P-1}{2P}$, $\frac{1}{3}\times\frac{1}{P}$, $\frac{1}{3}\times\frac{P-1}{P}\times(1-\frac{1}{Q^{m+n-1}})$ and $\frac{1}{3}\times\frac{P-1}{P}\times\frac{1}{Q^{m+n-1}}$, respectively. So the upper bound of MTTR should be $3P$ and E(TTR) should be: $\frac{2}{3}\times\frac{P+1}{2P}\times(1-\frac{1}{Q^{m+n-1}})\times P+\frac{2}{3}\times\frac{P+1}{2P}\times\frac{1}{Q^{m+n-1}}\times(2P+1)+\frac{2}{3}\times\frac{P-1}{2P}\times(2P-1)+\frac{1}{3P}\times P+\frac{1}{3}\times\frac{P-1}{P}\times(1-\frac{1}{Q^{m+n-1}})\times(2P-1)+\frac{1}{3}\times\frac{P-1}{P}\times\frac{1}{Q^{m+n-1}}\times(3P-1) = \frac{5P-4}{3}+\frac{1}{3Q^{m+n-1}}(2P+1+\frac{1}{P})$.

Similar to the proof in [9], we can prove that the upper bound of E(TTR) under the asymmetric model is smaller than $\frac{5P-4}{3}+\frac{1}{3Q^{m+n-1}}(2P+1+\frac{1}{P})+3QP(P-G)$. □

When the Parallel Sequence strategy is applied to generalize an existing rendezvous algorithm (say, algorithm A) to use multiple radios for faster rendezvous, the steps are given in Algorithm 3.

In line 3, an existing rendezvous algorithm is run to generate one CH sequence S_{At}. In lines 5-7, this CH sequence S_A is applied in parallel to the m radios of the user. The following theorem gives the MTTR of Algorithm 3.

Algorithm 3: Parallel Sequence

Require: Q, m, A, C_i *//an existing algorithm denoted by A and user i*

1: $\underrightarrow{t} = 1$
2: $\overrightarrow{S_t^i} = \{S_{t1}^i, S_{t2}^i, S_{t3}^i, ..., S_{tm}^i\}$
3: Generate sequence S_{At} by algorithm A
4: **while** not rendezvous **do**
5:     **for** $k = 1$ to m **do**
6:         $S_{tk}^i = S_{A((t-1)\times m+k)}$
7:     **end for**
8:     $t = t + 1$
9:     Attempt rendezvous on $\overrightarrow{S_t^i}$
10: **end while**

Theorem 4. *Suppose two users are equipped with m radios. Under the symmetric and the asymmetric models, when the Parallel Sequence strategy is applied in conjunction with an existing rendezvous algorithm with $MTTR = MTTR_0$, rendezvous can be achieved in at most $\lceil\frac{MTTR_0}{m}\rceil$ time slots.*

Proof: Since the CH sequence is applied in parallel to the m radios of the user, the user will finish the hopping sequence in time slot $\lceil\frac{MTTR_0}{m}\rceil$. Fig. 5 shows the new rendezvous achieved when the users are equipped with the same number of radios. The channel at time slot T in the hopping sequence when the user is equipped with one radio will appear in time slot $\lceil\frac{T}{m}\rceil$ when the user is equipped with m radios. If user i and user j achieve rendezvous on channel k in time slot TTR before $MTTR_0$ in algorithm A, then the two users will hop on this channel and achieve rendezvous at $\lceil\frac{TTR}{m}\rceil$ before time slot $\lceil\frac{MTTR_0}{m}\rceil$. Therefore, rendezvous can be achieved in at most $\lceil \frac{MTTR_0}{m} \rceil$. □

Corollary 2. *Suppose all users are equipped with m radios. Under the symmetric model, when the Parallel Sequence strategy is applied in conjunction with the Jump-Stay algorithm, rendezvous can be achieved in at most $\lceil \frac{3P}{m} \rceil$ time slots. The E(TTR) is upper bounded by $(\frac{5P}{3} + \frac{11}{3} + \frac{1}{Q})/m$, where P is the smallest prime number which is not smaller than Q.*

The proof of Corollary 2 is very similar to the proof of Theorem 4 and we do not repeat the details.

Overall, the Independent Sequence strategy can guarantee rendezvous even when the users have different number of radios, while the Parallel Sequence strategy can guarantee rendezvous only when the users have the same number of radios but it can better exploit these radios to achieve smaller MTTR.

4.3 New Algorithm

In this subsection, we design and analyze a new rendezvous algorithm that specifically exploits multiple radios for more efficient rendezvous. Our basic idea is to assign one of two possible roles, called *general radio* and *dedicated radio*, to each radio. The rendezvous is expected to be achieved between the general radios of one user and the dedicated radio of the other. Our RPS algorithm generates CH sequences in rounds and the length of each round is in inversely proportional to the number of general radios. The upper-bounds of MTTR and E(TTR) of RPS are the expression of the length of the round (later shown in proof of Theorem 5&6). Therefore, large number of general radios leads to a shorter round which consequently gives smaller upper-bounds of MTTR and E(TTR). In RPS, we use only one radio as the dedicated radio and the remaining radios as the general radios to optimize the rendezvous performance. Users hop on available channels in the general radios while stay on a specific available channel in the dedicated radio. Suppose that a user is equipped with m radios. In our paper, each node is equipped with multiple radios where each radio has a role: either "dedicated" or "general". So we call the new algorithm *Role-based Parallel Sequence* (RPS). It is described as follows.

i) All radios are divided into two groups, $(m-1)$ *general radios* and one *dedicated radio*.

ii) A starting index i is randomly selected from $[1, P-1]$ and a step-length r is randomly selected from $[1, P-1]$, where P is the smallest prime number which is not smaller than Q.

iii) $(m-1)$ *general radios* hop on P channels with step-length r in the round-robin manner. For example, in Fig. 6, $P = 7$, the starting index is 1 and step-length is 2. The first channel in the channel hopping sequence is 1 and the k-th channel is $(i + r * k)\%P$ ($(1+2k)\%7$ in this example). The CH sequence is $\{1, 3, 5, 7, 2, 4, 6, 1, 3, ...\}$ and two general Radios 1 and 2 hop on this sequence in parallel as follows. Radio 1 hops on subsequence $\{1, 5, 2, 6, ...\}$ and Radio 2 hops on subsequence $\{3, 7, 4, 1, ...\}$.

iv) *Dedicated radio* stays on one channel for $\lceil \frac{P}{m-1} \rceil$ time slots and switches to next channel for the same duration, where the channel is taken from $[1, Q]$ in a round-robin manner. For example, in Fig. 6, dedicated radio 3 stays on channel 1 for 4 time slots and then switches to channel 2.

v) If the channel selected in iii) and iv) is not available to the user, randomly select an available channel.

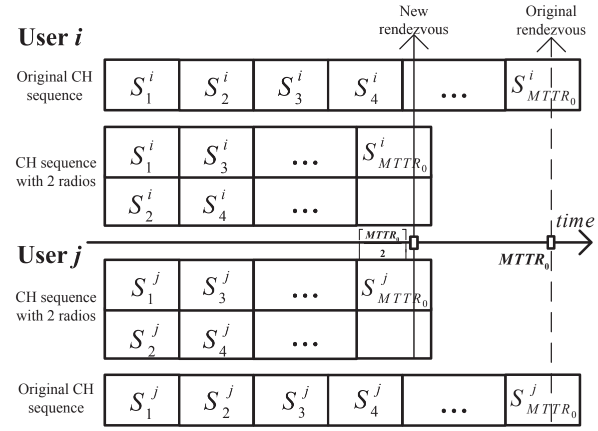


Fig. 5. Rendezvous of an existing algorithm with Parallel Sequence when $m = n = 2$.

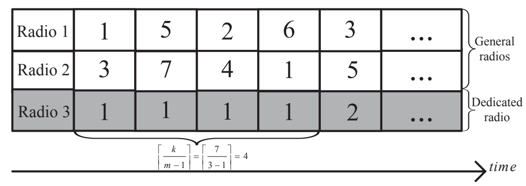


Fig. 6. CH sequence of a user with 3 radios and 7 channels.

The algorithm is formally presented as follows.

Algorithm 4: RPS Algorithm

Require: Q, m, C_i

$t = 1$
$\overrightarrow{S_t^i} = \{S_{t1}^i, S_{t2}^i, S_{t3}^i, ..., S_{tm}^i\}$
P=the smallest prime number not smaller than Q
$i = RandomSelect(1, P)$
$r = RandomSelect(1, Q)$
while not rendezvous **do**
  for $k = 1$ to $m-1$ **do**
    $S_{tk}^i = (i + ((t-1) \times (m-1) + k - 1) \times r - 1) \mod P + 1$
    if $S_{tk}^i \geq Q$ **then**
      $S_{tk}^i = S_{tk}^i \mod Q$
    end if
    if $S_{tk}^i \notin C_i$ **then**
      $S_{tk}^i = RandomSelect(C_i)$
    end if
  end for
  $S_{tm}^i = (\lceil \frac{t}{\lceil \frac{P}{m-1} \rceil} \rceil - 1) \mod Q + 1$
  if $S_{tm}^i \notin C_i$ **then**
    $S_{tm}^i = RandomSelect(C_i)$
  end if
  $t = t + 1$
  Attempt rendezvous on $\overrightarrow{S_t^i}$
end while

In line 4, starting index i and step-length r are preselected randomly. In lines 7-15, the $(m-1)$ general radios will hop on continuous $(m-1)$ channels with i and r. In line 16, the dedicated radio will switch to the next channel after $\lceil \frac{P}{m-1} \rceil$ time slots. Lines 12-14 and 17-19 ensure that the channels are available to the user.

Fig. 7 shows rendezvous of two users by performing the RPS algorithm. Suppose that $|C| = Q = 3$, $P = 3$. User i is equipped with 3 radios and the available channels are $\{1, 3\}$. It starts with channel 1. Step-length is 2. Each round consists of $\lceil \frac{P}{3-1} \rceil = 2$ time slots. User j is equipped with 2 radios and the available channels are $\{2, 3\}$. It starts with channel 2. Step-length is 2. Each round consists of $\lceil \frac{P}{2-1} \rceil = 3$ time slots. In this example, there will be a random available channel in the position with underline since the channel based on the algorithm is not available to the user. In Fig. 7(a), two users hop on channel 3 in time slot 4. Rendezvous is achieved by the dedicated radio of user j and the general radios of the user i. User j will stay on channel 3 for 3 time slots. User i has a permutation of all channels in any 3 consecutive time slots. Since there is at least one commonly-available channel between them, channel 3 in this example, there must be a rendezvous between the dedicated radio of user j and the general radios of user i. However, before this rendezvous, there is an earlier one in time slot 3. It is achieved by the general radios of user i and the general radio of the user j. In Fig. 7(b), user j starts later than user i for 1 time slot. When the dedicated radio of user i (with a shorter round) stays on the commonly-available channel (channel 3) for 2 time slots, the general radios of user j do not have a permutation of all available channels in 2 consecutive time slots. Therefore, it is possible that channel 3 is not in these 2 time slots. We cannot guarantee a rendezvous between the dedicated radio of user i and the general radios of user j. The guaranteed rendezvous realized at time slot 5.

Now we theoretically analyze the RPS algorithm. Specifically, we derive the upper bounds of MTTR of the RPS algorithm in Theorem 5 and Theorem 6 under the symmetric model and the asymmetric model, respectively. In addition, we derive the upper bounds of E(TTR) of the RPS algorithm under both the symmetric and asymmetric models.

Theorem 5. *Under the symmetric model, let m and n denote the numbers of radios of two users, respectively. If $m \neq n$, the MTTR of the RPS algorithm is upper bounded by $(2 \times \lceil \frac{P}{max\{m,n\}-1} \rceil - 1)$ and the E(TTR) of the RPS algorithm is upper bounded by $\lceil \frac{P}{max\{m,n\}-1} \rceil + \frac{(\lceil \frac{P}{max\{m,n\}-1} \rceil - 1)^2}{2 \times \lceil \frac{P}{min\{m,n\}-1} \rceil}$; if $m = n$, the MTTR of the RPS algorithm is upper bounded by $\lceil \frac{P}{max\{m,n\}-1} \rceil$ and the E(TTR) of the RPS algorithm is upper bounded by $\lceil \frac{P}{max\{m,n\}-1} \rceil$.*

Proof: Let user i be equipped with m radios while user j be equipped with n radios. Fig. 8 lists the four cases of rendezvous under symmetric model. Fig. 8(a), 8(b), and 8(c) happen when $m \neq n$. Since the results depend on which user starts hopping first, we assume

User i	Radio 1	1	2	3	1	2	3	1	2	3
	Radio 2	3	1	2	3	1	2	3	1	2
	Radio 3	1	1	2	2	3	3	1	1	2
User j	Radio 1	2	1	3	2	1	3	2	1	3
	Radio 2	2	2	2	3	3	3	1	1	1

time

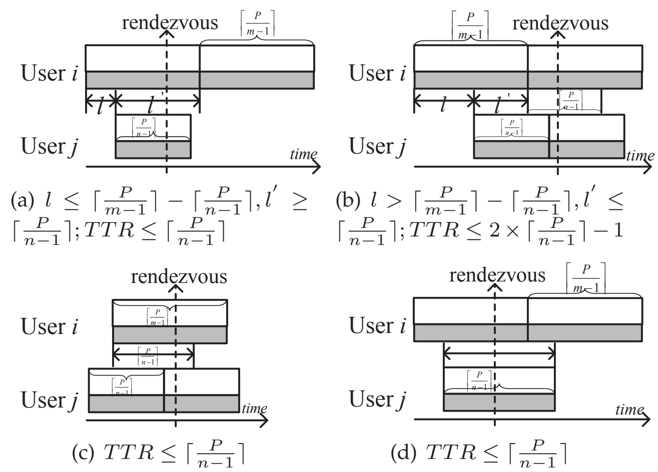
(a) offset=0

User i	Radio 1	1	2	3	1	2	3	1	2	3	
	Radio 2	3	1	2	3	1	2	3	1	2	
	Radio 3	1	1	2	2	3	3	1	1	2	
User j	Radio 1		2	1	3	2	1	3	2	1	3
	Radio 2		2	2	2	3	3	3	1	1	1

time

(b) offset=1

Fig. 7. Rendezvous of two users by performing RPS.



(a) $l \leq \lceil \frac{P}{m-1} \rceil - \lceil \frac{P}{n-1} \rceil, l' \geq \lceil \frac{P}{n-1} \rceil; TTR \leq \lceil \frac{P}{n-1} \rceil$

(b) $l > \lceil \frac{P}{m-1} \rceil - \lceil \frac{P}{n-1} \rceil, l' \leq \lceil \frac{P}{n-1} \rceil; TTR \leq 2 \times \lceil \frac{P}{n-1} \rceil - 1$

(c) $TTR \leq \lceil \frac{P}{n-1} \rceil$

(d) $TTR \leq \lceil \frac{P}{n-1} \rceil$

Fig. 8. Four cases of RPS under the symmetric model.

$m < n$, i.e., $\lceil \frac{P}{m-1} \rceil > \lceil \frac{P}{n-1} \rceil$. Fig. 8(d) happens when $m = n$. A remarkable distinguishment between them is whether the lengths of each round of the two users are the same.

1) Case 1: Fig. 8(a). $l' \leq \lceil \frac{P}{n-1} \rceil$ implies that there is a permutation of all channels before the dedicated radio of user i transfers to next channel. The rendezvous is achieved between general radios of user j and dedicated radio of user i during the first $\lceil \frac{P}{n-1} \rceil$ time slots. That is, $TTR \leq \lceil \frac{P}{n-1} \rceil$.
2) Case 2: Fig. 8(b). In this case, $l' < \lceil \frac{P}{n-1} \rceil$ implies that there is no enough time slots for user j to have permutation of all channels before the dedicated radio of user i transfers to next channel. The rendezvous can only be guaranteed between general radios of user i and dedicated radio of user j during the first $2 \times \lceil \frac{P}{n-1} \rceil - 1$ time slots. That is, $TTR \leq 2 \times \lceil \frac{P}{n-1} \rceil - 1$.
3) Case 3: Fig. 8(c). User j starts firstly. User j has a permutation of all channels in any $\lceil \frac{P}{n-1} \rceil$ consecutive time slots. When user i starts, it will stay on one channel for $\lceil \frac{P}{m-1} \rceil$ time slots. $\lceil \frac{P}{m-1} \rceil > \lceil \frac{P}{n-1} \rceil$. Thus, a rendezvous is guaranteed before $\lceil \frac{P}{n-1} \rceil$ time slots.

4) Case 4: Fig. 8(d). $m = n$. The rendezvous is achieved before the first $\lceil \frac{P}{m-1} \rceil$(or $\lceil \frac{P}{n-1} \rceil$) time slots.

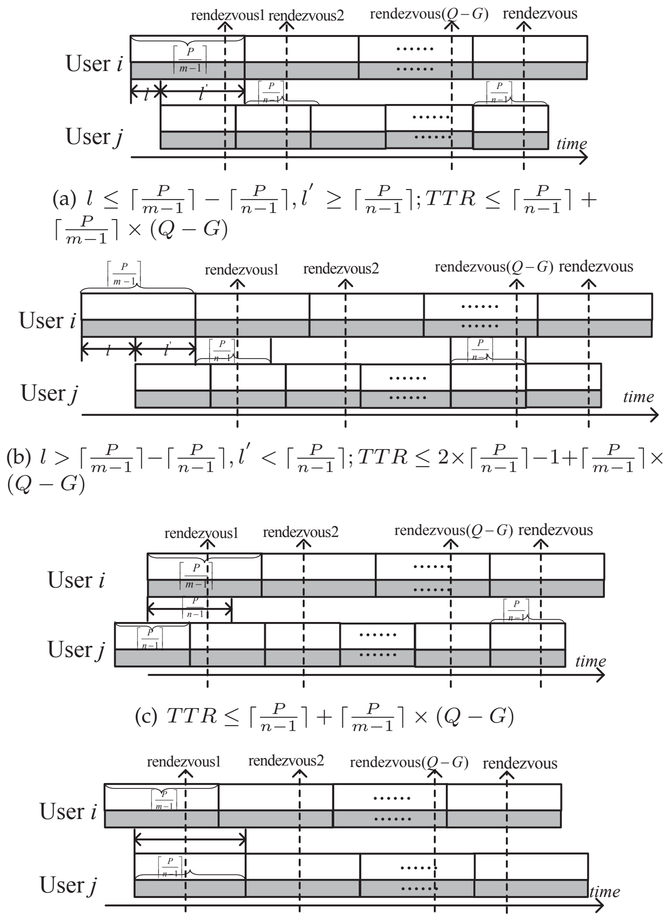
When $m \neq n$, in the above analysis, $\lceil \frac{P}{m-1} \rceil$ is replaced by $\lceil \frac{P}{min\{m,n\}-1} \rceil$ and $\lceil \frac{P}{n-1} \rceil$ by $\lceil \frac{P}{max\{m,n\}-1} \rceil$. According to the analysis of these cases, we prove that the MTTR is $(2 \times \lceil \frac{P}{max\{m,n\}-1} \rceil - 1)$. Combining with the occurrence probabilities we derive an upper bound of E(TTR) under the symmetric model. The $E(TTR) \leq \frac{1}{2} \times [\frac{\lceil \frac{P}{min\{m,n\}-1} \rceil - \lceil \frac{P}{max\{m,n\}-1} \rceil + 1}{\lceil \frac{P}{min\{m,n\}-1} \rceil} \times \lceil \frac{P}{max\{m,n\}-1} \rceil + \frac{\lceil \frac{P}{max\{m,n\}-1} \rceil - 1}{\lceil \frac{P}{min\{m,n\}-1} \rceil} \times (2 \times \lceil \frac{P}{max\{m,n\}} \rceil - 1)] + \frac{1}{2} \lceil \frac{P}{max\{m,n\}-1} \rceil \leq \lceil \frac{P}{max\{m,n\}-1} \rceil + \frac{(\lceil \frac{P}{max\{m,n\}-1} \rceil - 1)^2}{2 \times \lceil \frac{P}{min\{m,n\}-1} \rceil}$.

There is only one case when $m = n$. the MTTR and the upper bound of E(TTR) are both $\lceil \frac{P}{m-1} \rceil$ or $\lceil \frac{P}{n-1} \rceil$. $\square$

Theorem 6. *Under the asymmetric model, let m and n denote the numbers of radios of two users, respectively. If $m \neq n$, the MTTR of the RPS algorithm is upper bounded by $(2 \times \lceil \frac{P}{max\{m,n\}-1} \rceil - 1) + \lceil \frac{P}{min\{m,n\}-1} \rceil \times (Q - G)$ and the E(TTR) of the RPS algorithm is upper bounded by $\lceil \frac{P}{max\{m,n\}-1} \rceil + \frac{(\lceil \frac{P}{max\{m,n\}-1} \rceil - 1)^2}{2 \times \lceil \frac{P}{min\{m,n\}-1} \rceil} + \lceil \frac{P}{min\{m,n\}-1} \rceil \times (Q - G)$; if $m = n$, the MTTR of the RPS algorithm is upper bounded by $\lceil \frac{P}{m-1} \rceil \times (Q - G + 1)$ and the E(TTR) of the RPS algorithm is upper bounded by $\lceil \frac{P}{m-1} \rceil \times (Q - G + 1)$.*

Proof: Under the asymmetric model, since the available channel sets of two users are different from each other, the users may achieve many potential rendezvous (rendezvous 1 to $(Q - G)$ in Fig. 9). In Fig. 9(a), user j has a permutation of all channels before $\lceil \frac{P}{n-1} \rceil$ and the dedicated radio of user i stays on one channel during this period. There is a potential rendezvous before $\lceil \frac{P}{n-1} \rceil$; this channel may not be a commonly-available channel to all users. The next potential rendezvous can be guaranteed in the next round of user i ($\lceil \frac{P}{m-1} \rceil$ to $2 \times \lceil \frac{P}{m-1} \rceil$ in Fig. 9) because only after these time slots the dedicated radio of user i will transfer to the next channel. We can say, under asymmetric model, we expect a rendezvous between the dedicated radio of the user with less radios (user i) and the general radios of the user with more radios (user j). The worst case is repeating the rendezvous under symmetric model for $(Q - G)$ times. Above all, the MTTR should be equal to or smaller than $2 \times \lceil \frac{P}{max\{m,n\}-1} \rceil - 1 + \lceil \frac{P}{min\{m,n\}-1} \rceil \times (Q - G)$. We assume a uniform distribution that a commonly-available channel appears on the 1st to $(Q - G)$-th potential rendezvous. The upper bound of E(TTR) when $m \neq n$ extend for $\lceil \frac{P}{min\{m,n\}-1} \rceil \times (Q - G)$ in all cases. In this way, the $E(TTR) \leq \lceil \frac{P}{max\{m,n\}-1} \rceil + \frac{(\lceil \frac{P}{max\{m,n\}-1} \rceil)^2}{2 \times \lceil \frac{P}{min\{m,n\}-1} \rceil} + \lceil \frac{P}{min\{m,n\}-1} \rceil \times (Q - G)$.

And similarly, the MTTR and the upper bound of E(TTR) is $\lceil \frac{Q}{m-1} \rceil \times (Q - G + 1)$ when $m = n$. $\square$



(a) $l \leq \lceil \frac{P}{m-1} \rceil - \lceil \frac{P}{n-1} \rceil, l' \geq \lceil \frac{P}{n-1} \rceil; TTR \leq \lceil \frac{P}{n-1} \rceil + \lceil \frac{P}{m-1} \rceil \times (Q - G)$

(b) $l > \lceil \frac{P}{m-1} \rceil - \lceil \frac{P}{n-1} \rceil, l' < \lceil \frac{P}{n-1} \rceil; TTR \leq 2 \times \lceil \frac{P}{n-1} \rceil - 1 + \lceil \frac{P}{m-1} \rceil \times (Q - G)$

(c) $TTR \leq \lceil \frac{P}{n-1} \rceil + \lceil \frac{P}{m-1} \rceil \times (Q - G)$

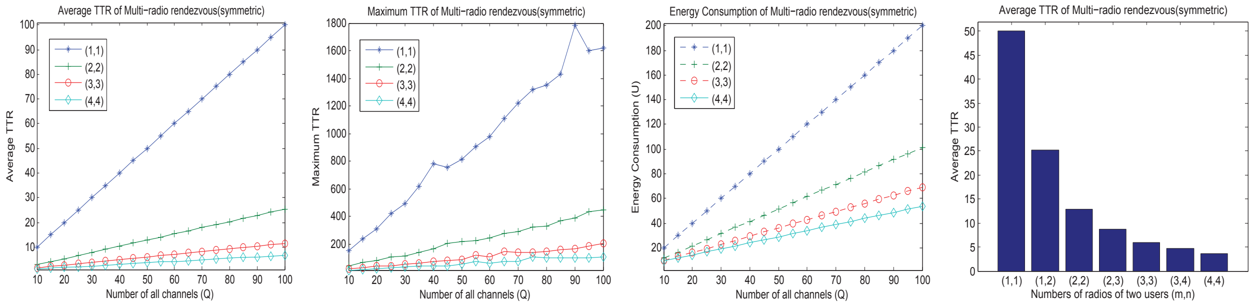
(d) $TTR \leq \lceil \frac{P}{n-1} \rceil + \lceil \frac{P}{m-1} \rceil \times (Q - G)$

Fig. 9. Four cases of RPS under the asymmetric model.

5 SIMULATION

We built a simulator in Visual Studio 2010 to evaluate the effectiveness of the proposed approach (i.e., using multiple radios for rendezvous) and the proposed rendezvous algorithm (i.e., the Role-based Parallel Sequence (RPS) algorithm). When each user has a single radio, we consider the following algorithms for comparison: i) the Random algorithm [4] (it is the most simple rendezvous algorithm), and ii) the Jump-Stay algorithm [8, 9], MMC algorithm [14], HH [30] (They are recently proposed and they have good performance). When each user has multiple radios, we consider the following algorithms for comparison: i) the generalized Random algorithm(Section 4.1), ii) the generalized Jump-Stay (MMC, HH) algorithm with the Independent Sequence strategy (Section 4.2), iii) the generalized Jump-Stay (MMC, HH) algorithm with the Parallel Sequence strategy (Section 4.2), and iv) the RPS algorithm (Section 4.3). The performance is measured in terms of the average TTR and the maximum TTR, where TTR is counted as the number of time slots required to achieve rendezvous. We consider both the symmetric model and the asymmetric model.

We use the notation (m, n) to denote the case that two users are equipped with m and n radios respectively. We consider the following key parameters: the number of channels Q in the whole channel set is varied from 10 to 100. Under the symmetric model, all channels are available to all users. Under the asym-



(a) Average TTR VS. Q (b) Maximum TTR VS. Q (c) Energy Consumption VS. Q (d) Average TTR VS. (m, n)

Fig. 10. Comparison of multi-radio and single-radio under the symmetric model (Random).

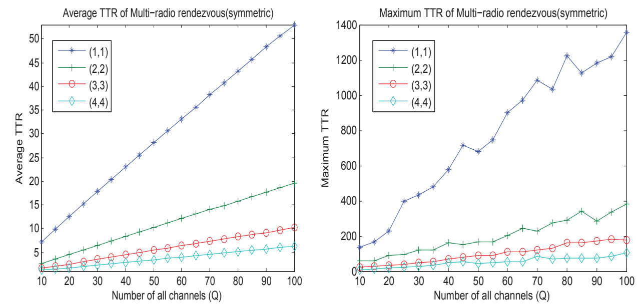
metric model, we introduce a parameter θ $(0 < \theta \leq 1)$ and randomly select channels from the channel set, such that the average number of channels available to a user is equal to θQ. If $\theta \times Q < 1$, we reset θQ to 1. For each set of parameter values, we perform 100,000 independent runs and then compute the average TTR and the maximum TTR.

5.1 Effectiveness of Proposed Approach

In this subsection, we demonstrate that multiple radios can effectively improve the rendezvous performance.

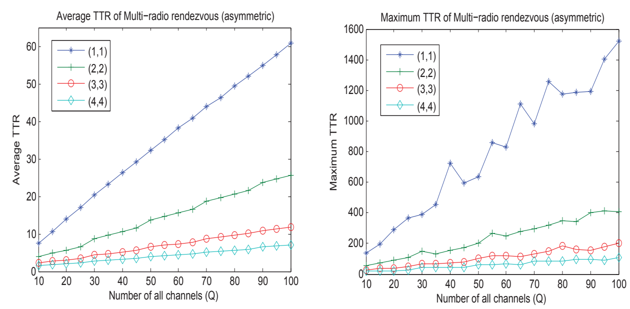
5.1.1 Under the Symmetric Model

Fig. 10 shows the performance of the Random algorithm under the symmetric model. It can be seen that: i) multiple radios can significantly reduce both the average TTR and the maximum TTR, and ii) the performance improvement is especially significant when the number of radios per user is increased from a small value. For example, we suppose that each time slot has duration of 20ms [9]. When there are 50 channels and the number of radios per user is increased from 1 to 2, the average TTR is decreased from 50.02 (1.00s) to 11.63 (0.23s) (i.e., 76.75% reduction) while the maximum TTR is decreased from 813 (16.26s) to 217 (4.34s) (i.e., 73.31% reduction). When the number of radios per user is increased from 2 to 3, the average TTR is decreased from 12.88 (0.26s) to 6.01 (0.12s) (i.e., 53.34% reduction) while the maximum TTR is decreased from 217 (4.34s) to 88 (1.76s) (i.e., 59.45% reduction). Therefore, a cost-effective tradeoff between cost of the radios and performance is to select 2 to 4 radios per user. In addition, we find that adoption of multiple radios can reduce the overall energy cost on rendezvous. For example, when the number of radios is increased from 1 to 3, the average time spent on rendezvous (E(TTR)) is reduced by 87.98% (more than 3 times). We assume that one radio performing rendezvous in one time slot costs one unit of energy, denoted by U. Then the expected energy consumption is $E(TTR) \times U \times (m + n)$ when the two users are equipped with m and n radios, respectively. Fig. 10(c) shows the expected consumption when each user is equipped with single radio and multiple radios. For example, when there are 50 channels, the energy consumption of users with 1, 2, 3 and 4 radios are $50.01U$, $25.77U$,$18.26U$ and $14.50U$, respectively. Fig. 10(d) shows the average TTR under different (m, n) when $M = 50$. We find that the average TTR drops significantly with the increase of number of radios.



(a) Average TTR VS. Q (b) Maximum TTR VS. Q

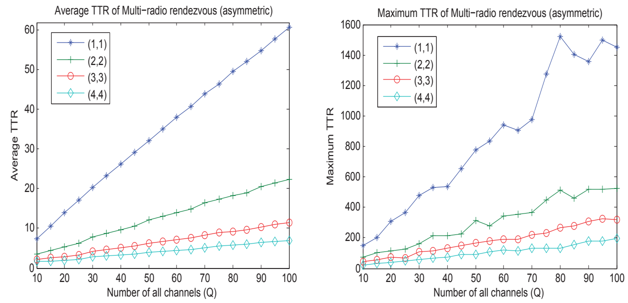
Fig. 11. Comparison of multi-radio and single-radio under the symmetric model (Jump-Stay).



(a) Average TTR VS. Q (b) Maximum TTR VS. Q

Fig. 12. Comparison of multi-radio and single-radio under the asymmetric model (Random).

Fig. 12 shows the performance of the Jump-Stay algorithm under the symmetric model. We observe similar properties as those in Fig. 10. For example, when the number of channels is 50 and the number of radios is increased from 1 to 2, the average TTR is reduced from 28.99 to 10.22 (i.e, 64.75% reduction) while the maximum TTR is decreased from 156 to 81 (i.e, 48.08% reduction). According to Theorem 1 and Corollary 1, however, the upper bound of E(TTR) is decreased from 91.33 to 85.01 while the MTTR remains 265. The theoretical results are much bigger than experimental results. It is because that Jump-Stay algorithm randomly selects available channels to replace the unavailable channels on the generated CH sequence, which practically lowers TTR in simulation.



(a) Average TTR VS. Q (b) Maximum TTR VS. Q

Fig. 13. Comparison of multi-radio and single-radio under the asymmetric model (Jump-Stay).

5.1.2 Under the Asymmetric Model

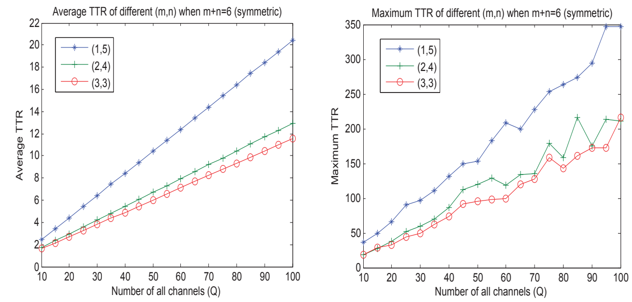
Fig. 12 shows the performance of Random algorithm under the asymmetric model. We let $\theta = 0.5$ and $G = 0.3Q$ (i.e., each user has $0.5Q$ channels and each pair of users have $0.3Q$ commonly-available channels). It can be seen that multiple radios can reduce both the average TTR and the maximum TTR significantly under the asymmetric model. For example, when there are 50 channels and the number of radios per user is increased from 1 to 2, the average TTR is decreased from 41.66 to 13.67 (i.e., 61.19% reduction) while the maximum TTR is decreased from 635 to 204 (i.e., 67.87% reduction). When the number of radios per user is increased from 2 to 3, the average TTR is decreased from 13.67 to 6.50 (i.e., 52.45% reduction) while the maximum TTR is decreased from 204 to 102 (i.e., 50% reduction). Fig. 13 shows the performance of the Jump-Stay algorithm with Independent Sequence under the asymmetric model. For example, when there are 50 channels and the number of radios per user is increased from 1 to 2, the average TTR is decreased from 32.21 to 12.11 (i.e., 62.40% reduction) while the maximum TTR is decreased from 777 to 311 (i.e., 59.97% reduction).

5.2 Property of Proposed Approach: Influence of Radio Allocation

In this subsection, we study a basic property of the proposed approach: when the total number of radios for two users is fixed (i.e., $m+n$ is a constant), how does the allocation (m, n) for different m and n affect the rendezvous performance? For example, if there are four radios, does (2, 2) give better rendezvous performance than (1, 3) or (3, 1)? In theory, the MTTR of the RPS algorithm is decided by the minimum value of m and n. The MTTR of Random algorithm [14] and other existing algorithms is almost independent of m and n. All results show that even allocation has the best performance.

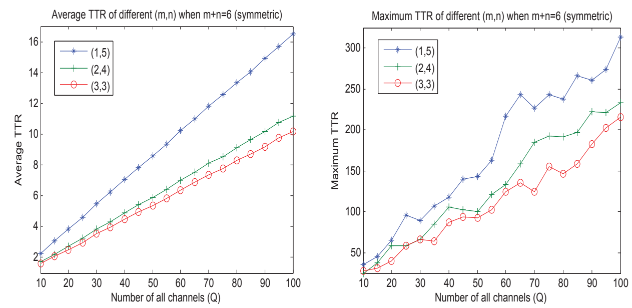
Under the symmetric model, we let $m+n=6$ and $G = Q$. Fig. 14 shows the comparison of the average TTR and the maximum TTR of Random algorithm between the three combinations of (m, n) which are (1, 5), (2, 4) and (3, 3). We can see that the combination (3, 3) has the best performance on both the average TTR and the maximum TTR. For example, when there are 50 channels, the average TTR of (1, 5), (2, 4) and (3, 3) are 10.41, 6.71 and 6.02 while the maximum TTR are 153, 120 and 95, respectively. It is consistent with the theoretical results. We also do the same comparison when we apply other existing algorithms on multiple radios. Fig. 15 shows the comparison of the average TTR and the maximum TTR of the Jump-Stay algorithm with Independent Sequence between the three combinations of (m, n) which are (1, 5), (2, 4) and (3, 3). Fig. 16 shows the comparison of the average TTR and the maximum TTR of the RPS algorithm between the two combinations of (m, n) which are (2, 4) and (3, 3).

Under the asymmetric model, we let $m+n=6$, $\theta = 0.75$ and $G = 0.5Q$. Fig. 17 shows the results of Random algorithm between the three combinations of (m, n). The combination (3, 3) has the best performance on both the average TTR and the maximum TTR. For example, when there are 50 channels, the average TTR of (1, 5), (2, 4) and (3, 3) are 11.56,



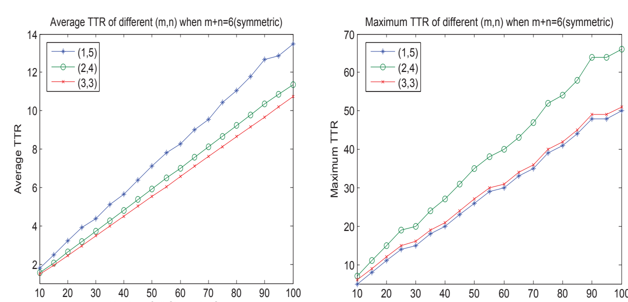
(a) Average TTR VS. Q (b) Maximum TTR VS. Q

Fig. 14. Comparison of different allocation of (m, n) when $m+n=6$ under the symmetric model (Random).



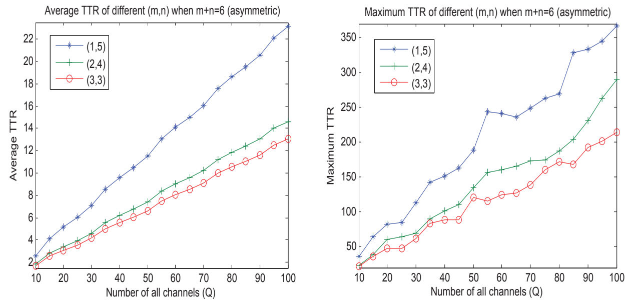
(a) Average TTR VS. Q (b) Maximum TTR VS. Q

Fig. 15. Comparison of different allocation of (m, n) when $m+n=6$ under the symmetric model (Jump-Stay).



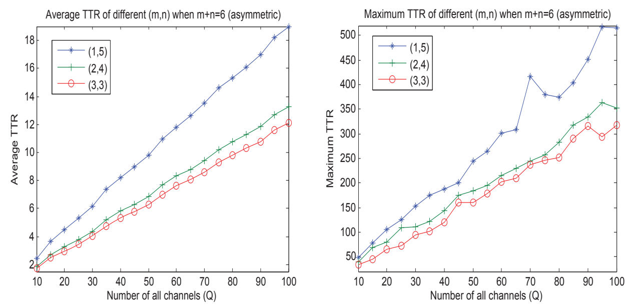
(a) Average TTR VS. Q (b) Maximum TTR VS. Q

Fig. 16. Comparison of different allocation of (m, n) when $m+n=6$ under the symmetric model (RPS).



(a) Average TTR VS. Q (b) Maximum TTR VS. Q

Fig. 17. Comparison of different allocation of (m, n) when $m + n = 6$ under the asymmetric model (Random).



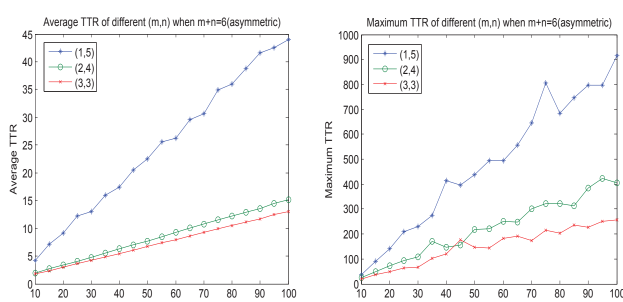
(a) Average TTR VS. Q (b) Maximum TTR VS. Q

Fig. 18. Comparison of different allocation of (m, n) when $m + n = 6$ under the asymmetric model (Jump-Stay).

7.42 and 6.08 while the maximum TTR are 188, 135 and 121, respectively. Fig. 18 shows the results of the average TTR and the maximum TTR of the Jump-Stay algorithm with Independent Sequence between the three combinations of (m, n). Fig. 19 shows the comparison of the average TTR and the maximum TTR of the RPS algorithm between the three combinations.

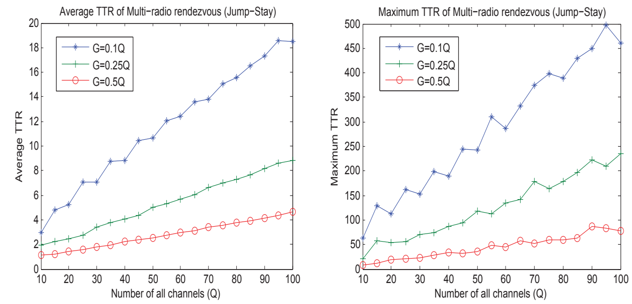
5.3 Property of Proposed Approach: Influence of G for Fixed θ

In this subsection, we study another basic property of the proposed approach: how does the number of commonly-available channels between two users affect the rendezvous performance? We let $\theta = 0.5$, $G = 0.1Q$, $G = 0.25Q$ and $G = 0.5Q$, respectively. $m = 3$ and $n = 4$. Fig. 20 shows the results when



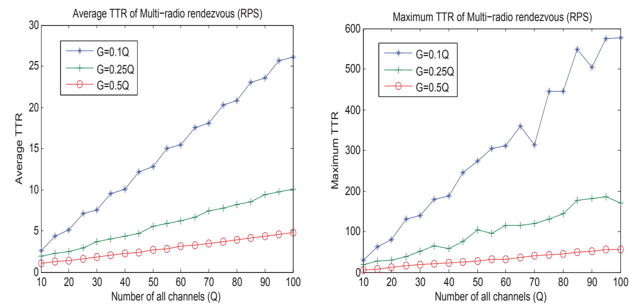
(a) Average TTR VS. Q (b) Maximum TTR VS. Q

Fig. 19. Comparison of different allocation of (m, n) when $m + n = 6$ under the asymmetric model (RPS).



(a) Average TTR VS. Q (b) Maximum TTR VS. Q

Fig. 20. Comparison of different G when $\theta = 0.5$ (Jump-Stay).



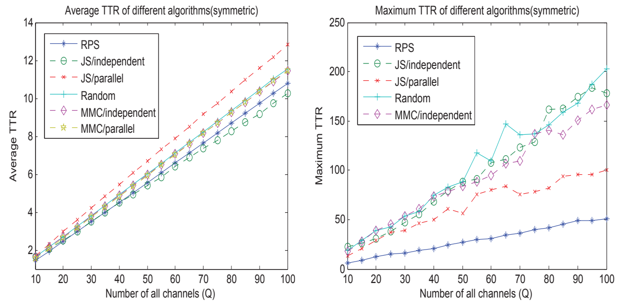
(a) Average TTR VS. Q (b) Maximum TTR VS. Q

Fig. 21. Comparison of different G when $\theta = 0.5$ (RPS).

we apply the Jump-Stay algorithm to multiple radios with Independent Sequence. When there are 50 channels, the average TTR of three scenarios are 10.69, 5.03 and 2.56 and the maximum TTR are 242, 119 and 35. Fig. 21 shows the results of the RPS algorithm. When there are 50 channels, the average TTR of three scenarios are 12.83, 5.54 (i.e. 56.82% reduction) and 2.65 and the maximum TTR are 274, 104 (i.e. 62.04% reduction) and 28. According to Theorem 6, however, when G is increased from $0.1Q$ to $0.25Q$, the upper bound of E(TTR) is decreased from 1215.72 to 1030.22 (i.e., 15.25% reduction) and the MTTR is decreased from 1200.83 to 1041.33 (i.e., 15.45% reduction). It reveals that both the average TTR and the maximum TTR drop faster than theoretical results when G increases.

5.4 Performance of Generalized Algorithms and Proposed Algorithm

We now study the performance of the proposed rendezvous algorithm and the generalized versions of the existing algorithms. We want to emphasize that all following simulations are based on multiple radios. All of them have significant improvement compared with existing algorithms. In this part, we compare the average TTR and the maximum TTR under different values of m, n, θ and G. We compare Generalized Random Algorithm (described in Section 4,1), Generalized Existing Algorithms (Section 4.2) and RPS Algorithm(Section 4.3) under the scenario when all users are equipped with multiple radios. For the existing algorithms, we apply the Jump-Stay [8], MMC [14] and HH [30] algorithm with both Independent Sequence and Parallel Sequence. HH is



(a) Average TTR VS. Q (b) Maximum TTR VS. Q

Fig. 22. Comparison of different algorithms under the symmetric model when $(m, n) = (3, 3)$.

for heterogeneous system. We only apply it under the asymmetric model and we expand it with random replacement. When the channel in the sequence is not available, we randomly select an available one to replace it.

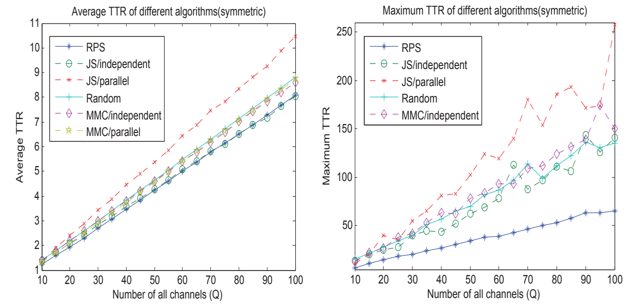
5.4.1 Under the Symmetric Model

$m = n$: Under the symmetric model, we let $\theta = 1$ and $G = Q$. Fig. 22 shows the comparison of the average TTR and the maximum TTR between the six different algorithms when $m = n = 3$. As shown in Fig. 22(a), when there are 50 channels, the average TTR of RPS, JS/Independent, JS/Parallel, Random, MMC/Independent and MMC/Parallel are 5.55, 5.40, 6.68, 6.02, 5.96 and 5.93 respectively. There is no large gap between different algorithms. However, the maximum TTR are 27, 88, 56, 88, 84 and 559 respectively. Since the Maximum TTR of MMC/Parallel is too large, we show the difference between other five algorithms in Fig. 22(b). We find that the RPS algorithm has no advantage on the average TTR but has significant improvement on the maximum TTR. Its maximum TTR is almost a third of others.

$m \neq n$: In this case, users are equipped with different numbers of radios. We let $m = 3$ and $n = 4$. Fig. 23 shows the comparison of the average TTR and the maximum TTR between the six different algorithms. For example, when there are 50 channels, the average TTR of RPS, JS/Independent, JS/Parallel, Random, MMC/Independent and MMC/Parallel are 4.28, 5.38, 4.64, 4.23, 4.59 and 4.98 respectively. However, the maximum TTR are 62, 103, 70, 34, 78 and 263 respectively. Same as the scenario $m = n$, Fig. 22(b) shows the difference between five of the algorithms. The RPS algorithm has no advantage on the average TTR but has significant improvement on the maximum TTR. Another important point is that JS/Parallel has the worst performance when $m \neq n$, which is consistent with the theoretical result.

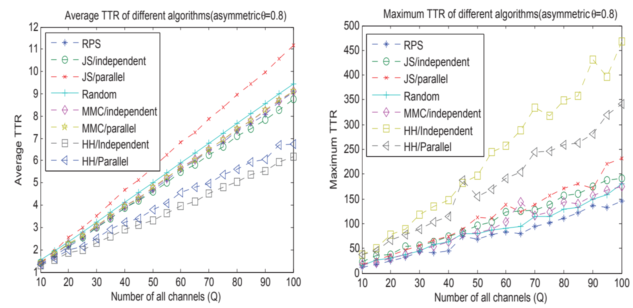
5.4.2 Under the Asymmetric Model

Large θ: Firstly we study the scenario when θ is large, that is, most channels are available to users. We let $\theta = 0.8$, $G = 0.6Q$, $m = 3$ and $n = 4$. Fig. 24 shows the comparison of the average TTR between eight different algorithms. For example, when there are 50 channels, the average TTR of RPS, JS/Independent, JS/Parallel, Random, MMC/Independent, MMC/Parallel, HH/Independent and HH/Parallel are 4.68, 4.61, 5.69,



(a) Average TTR VS. Q (b) Maximum TTR VS. Q

Fig. 23. Comparison of different algorithms under the symmetric model when $(m, n) = (3, 4)$.



(a) Average TTR VS. Q (b) Maximum TTR VS. Q

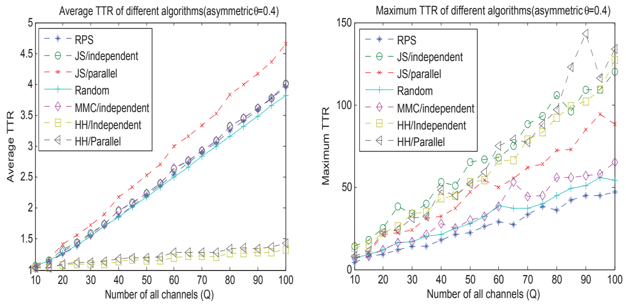
Fig. 24. Comparison of different algorithms under the asymmetric model when $\theta = 0.8$.

5.00, 4.77, 4.80, 3.33 and 3.80 respectively. However, the maximum TTR are 69, 96, 112, 80, 81, 1698, 197 and 155 respectively. HH algorithm with multiple radios has distinct advantage on the Average TTR. However, both MMC and HH has very large Maximum TTR. The reason is that their guaranteed rendezvous is based on some special conditions. Therefore, in terms of the average TTR, the generalized HH algorithm is the best; in term of the maximum TTR, the RPS algorithm is the best when θ is large.

Small θ: Then we study the scenario when θ is small, that is, only a small part of channels are available to users. We let $\theta = 0.4$, $G = 0.4Q$, $m = 3$ and $n = 4$. For example, when there are 50 channels, the average TTR of RPS, JS/Independent, JS/Parallel, Random, MMC/Independent, MMC/Parallel, HH/Independent and HH/Parallel are 2.21, 2.24, 2.53, 2.18, 2.24, 4.57, 1.16 and 1.20 while the maximum TTR are 22, 65, 47, 28, 30, 338, 53 and 52 respectively. Fig. 25 shows the comparison between the seven different algorithms (Except MMC/Parallel). Therefore, in terms of the average TTR, the generalized HH algorithm is the best; in term of the maximum TTR, the RPS algorithm is the best when θ is small.

6 Conclusions and Future Work

We investigated a new approach (using multiple radios per user) to significantly speeding up the rendezvous process in cognitive radio networks, generalized the Random algorithm and the existing algorithms in order to use multiple radios for faster rendezvous, and designed a new rendezvous algorithm (called *role-based parallel sequence* (RPS)) to specifically



(a) Average TTR VS. Q (b) Maximum TTR VS. Q

Fig. 25. Comparison of different algorithms under the asymmetric model when $\theta = 0.4$.

exploit multiple radios for fast rendezvous. We theoretically derived the upper bounds of E(TTR) and MTTR of these algorithms, and conducted extensive simulation studies to evaluate their performance. We observed the following properties:

- Multiple radios can significantly speed up rendezvous, especially when the number of radios per user is increased from a small value. For example, when there are 50 channels and the number of radios per user is increased from 1 to 2, the average TTR of the generalized Jump-Stay algorithm with Independent Sequence is reduced by 63.7% while its maximum TTR is reduced by 75.6% under the symmetric model.
- Given a fixed number of radios for two users, the rendezvous performance would be better if the radios are evenly allocated on the two users. For example, if there are 6 radios, the allocation (3, 3) achieves rendezvous faster than the allocation (4, 2) or (5, 1).
- In terms of the average TTR, the generalized HH algorithm perform better than other algorithms in our simulation. In terms of the maximum TTR, the RPS algorithm is the best.

In future work, we will analyze the tight lower-bound of MTTR in using multiple radios, and improve the MTTR of the RPS algorithm to achieve or be close to this tight lower-bound.

References

[1] I. Akyildiz, W. Lee, M. Vuran, and S. Mohanty, "NeXt Generation Dynamic Spectrum Access Cognitive Radio Wireless Networks: A Survey," *Computer Networks*, vol. 50, no. 13, pp. 2127-2159, 2006.

[2] H. Liu, Z. Lin, X. Chu, and Y. W. Leung, "Taxonomy and Challenges of Rendezvous Algorithms in Cognitive Radio Networks," invited position paper, *Proc. of IEEE ICNC*, pp. 645-649, 2012.

[3] V. Brik, E. Rozner, S. Banerjee, and P. Bahl, "DSAP: A Protocol for Coordinated Spectrum Access," *Proc. of IEEE DySPAN*, pp. 611-614, Nov. 2005.

[4] M. M. Buddhikot, P. Kolodzy, S. Miller, K. Ryan, and J. Evans, "DIMSUMnet: New Directions in Wireless Networking Using Coordinated Dynamic Spectrum," *Proc. of IEEE WoWMoM*, pp. 78-85, Jun. 2005.

[5] J. Jia, Q. Zhang, and X. Shen, "HC-MAC: A Hardware-constrained Cognitive MAC for Efficient Spectrum Management," *IEEE Journal on Selected Areas in Communications*, vol. 26, no. 1, pp. 106-117, 2008.

[6] L. Ma, X. Han, and C.-C. Shen, "Dynamic Open Spectrum Sharing for Wireless Ad Hoc Networks," *Proc. of IEEE DySPAN*, pp. 203-213, Nov. 2005.

[7] J. Zhao, H. Zheng and G.-H. Yang, "Distributed Coordination in Dynamic Spectrum Allocation Networks," *Proc. of IEEE DySPAN*, pp. 259-268, Nov. 2005.

[8] Z. Lin, H. Liu, X. Chu, Y.W. Leung, "Jump-Stay Based Channel-Hopping Algorithm with Guaranteed Rendezvous for Cognitive Radio Networks," *Proc. of IEEE INFOCOM*, pp. 2444-2452, Apr. 2011.

[9] H. Liu, Z. Lin, X. Chu, and Y. W. Leung, "Jump-Stay Rendezvous Algorithm for Cognitive Radio Networks," *IEEE Transactions on Parallel and Distributed Systems*, vol. 23, no. 10, pp. 1867-1881, 2012.

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

[11] Y. Zhang et al., "ETCH: Efficient Channel Hopping for Communication Rendezvous in Dynamic Spectrum Access Networks," *Proc. of IEEE INFOCOM*, pp. 2471-2479, Apr. 2011.

[12] 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, Sep. 2004.

[13] L. DaSilva and I. Guerreiro, "Sequence-based Rendezvous for Dynamic Spectrum Access," *Proc. of IEEE DySPAN*, pp. 1-7, Oct. 2008.

[14] N. C. Theis, R. W. Tomas, L. A. DaSilva, "Rendezvous for Cognitive Radios," *IEEE Transactions on Mobile Computing*, vol. 10, no. 2, pp. 216-227, 2011.

[15] C. Cormio and K. R. Chowdhury, "Common Control Channel Design for Cognitive Radio Wireless Ad Hoc Networks Using Adaptive Frequency Hopping," *Ad Hoc Networks*, Vol. 8, pp. 430-438, 2010.

[16] H. Liu, Z. Lin, X. Chu, and Y. W. Leung, "Ring-Walk Based Channel-Hopping Algorithm with Guaranteed Rendezvous for Cognitive Radio Networks," *Proc. of IEEE/ACM International Conference on Green Computing and Communications*, pp. 755-760, Dec. 2010.

[17] C. Cordeiro and K. Challapali, "C-MAC: A Cognitive MAC Protocol for Multi-Channel Wireless Networks," *Proc. of IEEE DySPAN*, pp. 147-157, Apr. 2007.

[18] D. Yang, J. Shin, and C. Kim, "Deterministic Rendezvous Scheme in Multichannel Access Networks," *Electronics Letters*, vol. 46, no. 20, pp. 1402-1404, 2010.

[19] J. Shin, D. Yang, and C. Kim, "A Channel Rendezvous Scheme for Cognitive Radio Networks," *IEEE Communications Letters*, vol. 14, no. 10, pp. 954-956, 2010.

[20] B. Horine and D. Turgut, "Performance Analysis of Link Rendezvous Protocol for Cognitive Radio Networks," *Proc. of CROWNCOM*, pp. 503-507, Aug. 2007.

[21] P. Sutton, K. Nolan, and L. Doyle, "Cyclostationary Signatures in Practical Cognitive Radio Applications," *IEEE Journal on Selected Areas in Communications*, vol. 26, no. 1, pp. 13-24, Jan. 2008.

[22] L. Lazos, S. Liu and M. Krunz, "Spectrum Opportunity-based Control Channel Assignment in Cognitive Radio Networks," *Proc. of IEEE SECON*, pp. 1-9, Jun. 2009.

[23] S. Krishnamurthy, M. Thoppian, S. Kuppa, R. Chandrasekaran, N. Mittal, S. Venkatesan, and R. Prakash, "Time-efficient Distributed Layer-2 Auto-configuration for Cognitive Radio Networks," *Computer Networks*, vol. 52, pp. 831-849, 2008.

[24] K. Bian and J.-M. Park, "Maximizing Rendezvous Diversity in Rendezvous Protocols for Decentralized Cognitive Radio Networks," *IEEE Transactions on Mobile Computing*, vol. 12, no. 7, pp. 1294-1307, 2013.

[25] S. Romaszko, "A Rendezvous Protocol with the Heterogeneous Spectrum Availability Analysis for Cognitive Radio Ad Hoc Networks," *Hindawi Publishing Corporation Journal of Electrical and Computer Engineering*, Article ID 715816. 2013.

[26] Z. Lin, H. Liu, X. Chu, and Y. W. Leung, "Enhanced Jump-Stay Rendezvous Algorithm for Cognitive Radio Networks," *IEEE Communications Letters*, vol. 17, pp. 1742 - 1745, 2013.

[27] Z. Gu, Q.-S. Hua, Y. Wang, F. Lau, "Nearly Optimal Asynchronous Blind Rendezvous Algorithm for Cognitive Radio Networks," *Proc. of IEEE SECON*, pp. 371-379, Jun. 2013.

[28] J. Abdel-Rahman, H. Rahbari and M. Krunz, "Adaptive Frequency Hopping Algorithms for Multicast Rendezvous in DSA Networks ," *Proc. of IEEE DySPAN*, pp. 517-528, Oct. 2012.

[29] R. Paul, Z. Yalew, Y.-J. Choi,"Multi-interface rendezvous in self-organizing cognitive radio networks." *Proc. of IEEE DySPAN*, pp. 531-540, Apr. 2014.

[30] S. H. Wu, C. C. Wu, W. K. Hon and K. G. Shin, "Rendezvous for Heterogeneous Spectrum-Agile Devices ," *Proc. of IEEE INFOCOM*, Apr. 2014.

[31] IEEE 802.11 Standard, http://standards.ieee.org/getieee802/download/802.11-2007.pdf.