A Multi-Radio Rendezvous Algorithm Based on Chinese Remainder Theorem in Heterogeneous Cognitive Radio Networks

Jang-Ping Sheu and Ji-Jhen Lin

Abstract—In cognitive radio networks (CRNs), secondary users can utilize the temporary unused spectrum opportunistically without affecting the quality of services of the licensed users, also called primary users. It is a fundamental operation for a user to rendezvous with another user on the same channel and establish a communication link. Traditional rendezvous algorithms assume homogeneous CRNs and each user equipped with a single radio. In recent years, the cost of wireless transceivers has fallen dramatically. It is more feasible for users to apply multiple radios to reduce the time to rendezvous significantly. In this paper, we propose a Chinese Remainder Theorem (CRT) Based Multi-Radio Rendezvous (CMR) algorithm for heterogeneous CRNs, where the users are unaware of the total number of channels and are allowed to have different spectrum-sensing capabilities. Our CMR scheme applies CRT to achieve fast rendezvous. Simulation results show that CMR has better performance than the previous works.

Index Terms—Cognitive radio networks, rendezvous, multiple radios, heterogeneous networks

1 Introduction

With exploding popularity of mobile devices, the unlicensed spectrum such as Industrial Scientific and Medical (ISM) band has been overcrowded and hence has become a scarce resource. On the other hand, it has been observed that only 15 % of the licensed spectrum such as the TV bands are utilized efficiently when the spectrum is allocated in a static manner [1]. For the efficient use of licensed spectrum, the concept of cognitive radio has emerged as a promising technique of dynamic spectrum access [2-6]. In cognitive radio networks (CRNs), unlicensed users, also called secondary users (SUs) can utilize the temporary unused spectrum opportunistically without affecting the quality of services of the licensed users, also called primary users (PUs).

In CRN, each SU is equipped with one or multiple cognitive radios (CRs). A CR is an intelligent radio that can be programmed and configured dynamically. With cognitive radios, each SU opportunistically identifies and accesses temporary vacant licensed channels of PUs. Two SUs, when both are on the same available channel, can establish link and exchange control information. The process of meeting and establishing a link on a common channel of two or more users is referred as rendezvous. Rendezvous, an essential operation in CRNs, is difficult to implement as SUs do not aware the presence of each other and their available channels may be different. Moreover, channels availability changes dynamically due to the activities of neighboring PUs.

Traditional rendezvous methods have either utilized centralized controller or dedicated common control channel (CCC) to accomplish rendezvous. In CCC approaches, a dedicated common control channel is used for SUs to exchange the control messages and two SUs can establish a communication link via their common available channels. However, CCC has several drawbacks. In addition to scalability problems and security concerns, it is difficult to find a CCC due to the fluctuations of channel availability.

To overcome the drawbacks of CCC, blind rendezvous is more applicable to the reality. Channel Hopping (CH) has emerged as one of the most effective techniques for blind rendezvous. Each SU hops on the available channels of a deterministic CH sequence in order to meet other SUs on the same channel. The main performance metrics in designing a CH algorithm are time to rendezvous (TTR) and rendezvous diversity. Time to rendezvous includes (i) Maximum Time to Rendezvous (MTTR) and (ii) Expected Time to Rendezvous (ETTR). There exists a tradeoff between optimizing the MTTR and the ETTR [7]. The random CH approaches have better performance in average cases but the MTTR may be infinite while the sequence-based CH approaches ensure rendezvous within finite duration at the cost of higher ETTR. Rendezvous diversity means the number of distinct rendezvous channels. Since SUs may have no knowledge of the available channels of other SUs, in this paper, we want to maximize the rendezvous diversity in order to reduce the risk of rendezvous failure due to the appearance of PU signals [8, 9]. That is, any pair of SUs are required to rendezvous on all the commonly available channels.

Moreover, SUs that would like to rendezvous with each other may suffer from the following challenges.

i) Asynchronous clock. Synchronous algorithms [6, 10] assume there exists a synchronous global clock and beacon messages to make the SUs hop on the same index of their own CH sequences in the same time slot. However,

Jang-Ping Sheu and Ji-Jhen Lin are with the Department of Computer Science, National Tsing Hua University, Hsinchu, Taiwan. E-mail: sheujp@cs.nthu.edu.tw, ikuma121@gmail.com.

it is not practical since time synchronization is hard to achieve. In asynchronous algorithms [7-9, 11-13], neither synchronous global clock nor beacon messages will be used. SUs may hop on the same index of their CH sequences in different time slots, which will make the rendezvous more difficult and longer the sequence period relatively.

ii) Asymmetric available channels. Since the available channel sets may be time varying, there are two models of channel availability often considered in literature: i) symmetric model, in which all SUs have the same available channels; and ii) asymmetric model, in which different SUs have different available channels.

iii) Heterogeneous sensing capacity. Homogeneous rendezvous algorithms [7, 8, 10-24] implicitly assume there is a universe spectrum and each CR can sense and access the whole spectrum. In the heterogeneous model, various kinds of wireless devices are equipped with heterogeneous CRs whose spectrum-sensing capabilities may be different and are unaware of the global channel number. Some of the existing works [9, 25-30] take the lead in designing their rendezvous approaches for heterogeneous CRs.

iv) Anonymous information. Some algorithms such as ID-based algorithms [8, 9, 11-14] and role-based algorithms [6, 14-16] utilize prior knowledge (SU's unique IDs and pre-assigned roles) to achieve fast rendezvous. In the ID-based algorithms, SUs generate their CH sequences based on their unique IDs and achieve fast rendezvous. In the role-based algorithms, each SU is assigned a role of sender or receiver in advance of the rendezvous process. SUs with different roles generate the CH sequences by different algorithms. The role-based algorithms only guarantee rendezvous between sender-receiver pairs. Nevertheless, SUs in distributed environments may not have any prior knowledge of each other in the most cases.

Traditional rendezvous algorithms implicitly assume that each SU is equipped with exactly one radio. The authors of [31] indicated that the popularity of wireless systems has led to the cost of RF transceivers (radios) whose prices have fallen dramatically. It is of high possibility that SUs are allowed to achieve fast rendezvous through multiple radios in the future. Only small portion of existing works [14, 22-24, 28-30] have designed their rendezvous algorithms for multiple radios.

In this paper, we proposed a multi-radio rendezvous algorithm in the heterogeneous model. In the model, we assume that users have asynchronous clocks, asymmetric available channels, heterogeneous sensing capacity, no pre-assigned roles, no individual identifiers, and no global channel number, which is a fully distributed environment. To the best of our knowledge, there is no any single-radio algorithm can guarantee bounded TTR in the heterogeneous model. The contributions of our work are summarized as follows: (1) Based on the Chinese Remainder Theorem, we propose a new rendezvous algorithm for heterogeneous multi-radio CRNs. In previous works, each SU chooses two primes to implement their multi-radio rendezvous algorithms [28-30]. Because the MTTR is related to the size of the primes. By choosing the larger primes, the longer will be the MTTR. Assuming a user has m radios. In our protocol, we can select at most m smaller primes (repeatable) to construct the CH sequence for multiple radios instead of two larger primes. To the best of our knowledge, we are the first protocol to use multiple primes on the multi-radio model. (2) To find the optimal solution of primes with the minimum average length of primes is an integer-programming problem. We proposed a heuristic algorithm to solve this problem in polynomial time. In addition, we proposed an algorithm to construct the CH sequence in multiple radios to increase the rendezvous probability. (3) Simulation results show that our scheme has smaller ETTR and MTTR than the previous multi-radio works [22, 28-30].

The rest of the paper is organized as follows. In Section 2, we discuss the performance and limitations of existing multi-radio rendezvous algorithms. In Section 3, we present our system model, define the rendezvous problem, and then propose our multi-radio rendezvous algorithm. Simulation results are presented in Section 4 for comparison with other multi-radio rendezvous algorithms. Finally, Section 5 concludes this paper.

2 Related Works

The CH approaches can be classified according to two criteria, homogeneous versus heterogeneous, and single-radio versus multi-radio. Most of the existing CH approaches are homogeneous single-radio approaches [7, 8, 10-13, 15-21], which assume that there is exactly one radio equipped at SUs and SUs have the same spectrum-sensing capability. It may be noted that an asynchronous homogeneous CH algorithm with a single radio can achieve rendezvous no less than N^2 time slots in the worst case, where N is the total number of channels [8]. In heterogeneous case, the homogeneous CH algorithms are not applicable in practice [25]. Recently, several single-radio approaches, like LS[12], HH [25], ICH [26], and MTP [27] are proposed to guarantee bounded TTR in the heterogeneous model. However, the LS assumes that each user has an ID which can be represented as a unique binary string. The assumption is not favorable for anonymous users with no explicit IDs in distributed environments. Both HH and ICH suffer the limitation that each SU has the capability of sensing a range of consecutive channels. The MTP assumes that the global channel number is known, which is not fully distributed.

In this paper, we focus on the multi-radio rendezvous algorithm in distributed heterogeneous environments. The MTTR of homogeneous multi-radio algorithms increases with the number of channels N, such as RPS [22]. This implies that the MTTR of homogeneous algorithms may be large even if only a small portion of channels are available. On the other hand, the heterogeneous multi-radio algorithms generate their CH sequences only based on the available channels and thus the MTTR increases with the number of available channels but it is irrelevant to N, such as AMRR [28], MSS [29], and GCR [30]. In the following, we review the aforementioned multi-radio CH algorithms.

The authors in [22] first proposed independent sequence and parallel sequence to generalize the existing

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TMC.2018.2790408, IEEE Transactions on Mobile Computing

single-radio algorithms to use multiple radios to achieve rendezvous, and then proposed a new multi-radio algorithm known as the role-based parallel sequence (RPS). The basic idea of RPS is to divide m radios into one dedicated radio and m-1 general radios. The m-1 general radios hop on P channels in the round-robin manner and the dedicated radio stays on one channel for $\lceil P/(m-1)\rceil$ time slots and switch to next channel for the same duration, where P is the least prime not smaller than the number of global channels N. Assume that a pair of SUs which want to rendezvous with each other are equipped with m_1 and m_2 radios and have G channels in common. RPS can guarantee MTTR within $O(P(N\text{-}G)/\min(m_1, m_2))$ time slots, which is about $O(1/\min(m_1, m_2))$ of the optimal single-radio algorithm for homogeneous CRNs, where m_1 and m_2 are the numbers of radios equipped at user 1 and user 2, respectively.

In [28], the authors proposed AMRR which is the only adjustable algorithm allowing users to choose optimizing MTTR or ETTR. The design of AMRR is based on RPS, there are also dedicated and general radios equipped at users. However, there are two major differences between the AMRR and RPS. One is that AMRR can adjust the number of dedicated radios to give better performance on MTTR or ETTR. From performance analysis and simulations in [28], AMRR has better performance on MTTR when allocating half of the radios as the dedicated radios and have better ETTR when allocating one radio as the dedicated radio. The second difference is that AMRR can be used in heterogeneous CRNs. The MTTR of AMRR is $O(\max((N_1/m_1)^2, (N_2/m_2)^2))$, irrelative to N, where N_1 and N_2 are the numbers of local available channels of user 1 and user 2, respectively.

MSS [29] adopts the sunflower sets to generate the CH sequence for the first radio of user u. The remaining radios rotate the sequence of the previous radio by $2P_u$ time slots one by one, where P_u is the least prime not smaller than the number of user u's available channels. MSS has MTTR as $O(\max(P_1^2P_2,\ P_1P_2^2)/m_1m_2)$, where P_1 and P_2 are the smallest primes not smaller than the numbers of local available channels of user 1 and user 2, respectively. Though MSS is designed for heterogeneous CRN, the MTTR of MSS is proportional to the cube of a number of available channels. The MTTR increases significantly with the increasing of available channels. Moreover, MSS has a constraint that two users cannot start rendezvous process simultaneously, otherwise, they cannot guarantee rendezvous.

The authors in [30] have first derived the lower bound of MTTR as $\Omega(N_1N_2/m^2)$ when all SUs in the CRN are equipped with exactly m radios. In addition, the authors proposed a multi-radio rendezvous algorithm GCR near to the lower bound, which has the MTTR $O(N_1N_2/m_1m_2)$. First, the m_u radios of SU are divided into pairs. In the sequel, the available channel set C_u is divided into $m_u/2$ subsets and assign each pair of radios a subset. Each pair of radios selects two primes to generate their CH sequences based on channels in the subset. However, this algorithm implicitly constrains the number of radios to be even and does not utilize the unpaired radio efficiently. In this paper, we proposed a heterogeneous CRT-based multi-radio rendezvous (CMR) algorithm, which can have ETTR and MTTR better than previous works. In our algorithm, we can select multiple smaller primes (repeatable) to construct the CH sequence instead of two larger primes. For example, if the number of available channels = 16 and number of radios = 4, the GCR can select p = 11 and q = 13 to construct its CH sequence. Our algorithm can select four primes 13, 7, 7, 5 to construct the CH sequence whose average length of primes is smaller than GCR. The MTTR of the aforementioned multi-radio algorithms and our algorithm are listed in Table I.

3 CRT-BASED MULTI-RADIO RENDEZVOUS ALGORITHM

In this Section, we present the system model and define our multi-radio rendezvous problem for heterogeneous CRNs. We present a new multi-radio algorithm: CRT-based multi-radio rendezvous algorithm (CMR). The time complexity and the MTTR of our algorithm are also analyzed in this Section.

TABLE I. MTTR FOR MULTI-RADIO RENDEZVOUS ALGORITHMS

Algorithms	MTTR	Heterogeneous
RPS [22]	$O(\frac{P(N-G)}{min(m_1,m_2)})$	no
AMRR [28]	$O(\max((\frac{N_1}{m_1})^2, (\frac{N_2}{m_2})^2))$	yes
MSS [29]	$O(\frac{\max(P_1{}^2P_2, P_2{}^2P_1)}{m_1m_2})$ (a)	yes
GCR [30]	$\frac{72N_1N_2}{m_1m_2} = O(\frac{N_1N_2}{m_1m_2})$ (b)	yes
CMR	$\frac{32N_1N_2}{m_1m_2} = O(\frac{N_1N_2}{m_1m_2})$	yes

Remarks: (a) only have guaranteed rendezvous when two users do not start rendezvous process at the same time. (b) m_1 and m_2 are even.

3.1 System Model and Problem Definition

We consider a heterogeneous CRN, where SUs (hereafter referred to as "users") may not sense all channels. Let V_u be the set of all channels sensible by user u and $C_u = \{c_u(0), c_u(1), \ldots, c_u(N_u\text{-}1)\} \subseteq V_u$ be the set of channels available to user u, where $N_u = |C_u|$ is the cardinality of C_u. User u is equipped with m_u radios to sense the availability of channels ($m_u > 1$). The CH sequences of user u is denoted by $S_u = \{S_{u,0}, S_{u,1}, S_{u,2}, \ldots\}$, where $S_{u,t} = \{s_{u,t}^1, \ldots, s_{u,t}^{m_u}\}$ and $s_{u,t}^i$ represents that S_u hops on channel $s_{u,t}^i$ on radio i at time slot t and $1 \le i \le m_u$.

Consider the rendezvous problem between any two users 1 and 2. Assume that users 1 and 2 have at least one channel in common, i.e., $C_1 \cap C_2 \neq \emptyset$. Without loss of generality, suppose that user 1 starts rendezvous process earlier than user 2 by δ time slots. To cope with slot boundary misalignment, the information exchange time between two users can be set to 1/2 δ. User 1 and user 2 are said to rendezvous if there exists a finite t such that $S_{1,t+\delta} \cap S_{2,t} \neq \emptyset$. The minimal t to satisfy the above condition is called time to rendezvous (TTR).

Our multi-radio rendezvous problem is to design a CH algorithm such that $\forall\delta > 0$, user 1 and user 2 always rendezvous with each other within some finite time t. In the following, we would use the Maximum TTR (MTTR) and Expected TTR (ETTR) to measure how many time slots for two users to complete rendezvous in the worst and average cases.

3.2 Rendezvous with Chinese Remainder Theorem

In the following, we illustrate how single-radio users can use CRT to rendezvous. Then, we apply CRT to design rendezvous algorithm for multi-radio users. Suppose that user 1 and user 2 have available channel sets C_1 and C_2, respectively. User 1 has CH sequence (CHS) S_1 which contains all channels in C_1, while user 2 has another CHS S_2 which contains all channels in C_2. Theorem 1 shows that user 1 and user 2 have guaranteed rendezvous when $|S_1|$ is co-prime with $|S_2|$.

Theorem 1. *Suppose the CHSs of user 1 and user 2 are* $S_1 = \{a_0, a_1, \dots, a_{|S_1|-1}\}$ *and* $S_2 = \{b_0, b_1, \dots, b_{|S_2|-1}\}$, *respectively. If* $S_1 \cap S_2 = a_x = b_y = c$ *and* $|S_1|$ *is co-prime with* $|S_2|$, *then user 1 and user 2 will rendezvous on channel c within* $|S_1| \times |S_2|$ *time slots irrespective of the clock drift.*

Proof: This theorem can be derived from Chinese Remainder Theorem. □

However, it is impractical to assume the co-primality constraint on $|S_1|$ and $|S_2|$ since two users may have no knowledge of the length of the other user's CHSs. With the help of multiple radios, the co-primality constraint can be satisfied by selecting two primes on a pair of radios such as GCR approach. In our protocol, we can select at most m_u smaller primes (repeatable) to construct the CH sequence for user u instead of two larger primes. For a multi-radio user, each radio of the user is assigned a radio channel hopping sequence (RCHS). That is, user u equipped with m_u ($m_u \geq 2$) radios have m_u RCHSs whose lengths are not necessarily the same. To assure the co-primality of sequence lengths, we have the following two constraints on the RCHSs of each user u.

Constraint 1. The lengths of all RCHSs are prime and there are at least two different prime lengths.

Constraint 2. Each channel in C_u must appear in at least two different RCHSs with different lengths. Thus the sum of lengths of the RCHSs of user u is larger than or equal to $2N_u$.

If constraint 1 and constraint 2 are satisfied, for each user, the common channel c appears in two RCHSs of different prime lengths. There must exist one RCHS of user 1 containing channel c and one RCHS of user 2 containing channel c whose lengths are co-prime with each other. By theorem 1, they can rendezvous within a finite time. To satisfy constraint 2, the lengths of all RCHSs need to satisfy equations (1)-(3). Otherwise, the constructed RCHSs may not satisfy constraint 2.

Theorem 2. *Suppose the lengths of the RCHSs of user u consist of* α *distinct primes* $p_1, p_2, \dots, p_\alpha$, *and* p_i *appears* x_i (x_i: *nonnegative integer) times, for i = 1 to* α. *The RCHSs satisfies constraint 2 if the following equations are true.*

$$\sum_{i=1}^{\alpha} x_i = m_u \tag{1}$$

$$\sum_{i=1}^{\alpha} p_i x_i \geq 2N_u \tag{2}$$

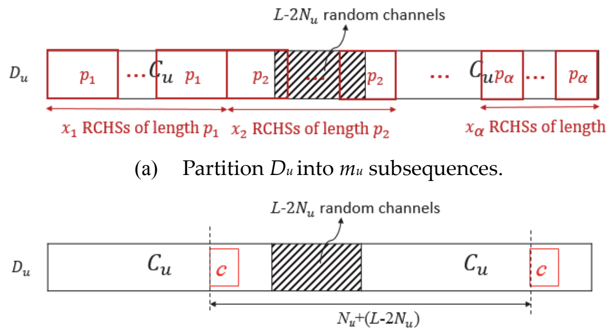
$$p_j x_j \leq \sum_{i=1}^{\alpha} p_i x_i - N_u, \text{ for } j = 1 \text{ to } \alpha \tag{3}$$

Proof. Since the number of radios is m_u, equation (1) must hold. Each available channel in C_u appears at least twice, so equation (2) must hold too. If there exists any j such that $p_j x_j > \sum_{i=1}^{\alpha} p_i x_i - N_u$, then we have $N_u > \sum_{i=1, i\neq j}^{\alpha} p_i x_i$. Thus, at least one available channel in C_u cannot satisfy constraint 2. For example, assume C_u= {0, 1, 2, 3, 4, 5, 6, 7}, N_u= 8, and m_u=3. If the lengths of three primes are 5, 5, 7, they satisfy equations (1) and (2) but not equation (3). This because prime 7 is smaller than N_u. So, equation (3) must hold too. □

Now, we can construct the RCHSs satisfying constraint 2 as follows. First, we create a sequence D_u={$d_u(0)$, $d_u(1)$, …, $d_u(L\text{-}1)$} of length $L= \sum_{i=1}^{\alpha} p_i x_i$ by concatenating N_u available channels, $L\text{-}2N_u$ random channels, and N_u available channels as shown in Fig. 1. A random channel can be any available channel in C_u. The i-th element of D_u is defined as follows.

$$d_u(i) = \begin{cases} cu(i), & \text{if } 0 \leq i < Nu, \\ \text{random channel } r, & \text{if } Nu \leq i < L - Nu, \\ cu((i-(L-Nu)) \bmod Nu, & \text{if } L - Nu \leq i < L \end{cases} \tag{4}$$

We can partition D_u into x_1 subsequences of length p_1, x_2 subsequences of length p_2, …, and x_α subsequences of length p_α as shown in Fig. 1(a). The distance between two same available channel c in D_u is exactly $L\text{-}N_u$ as shown in Fig. 1(b). By equation (3), the total length of any x_i RCHSs with length p_i is always no larger than $L\text{-}N_u$, i.e., $p_i x_i \leq L\text{-}N_u$. So, any channel can appear at most once in one of the RCHSs of same length p_i. In this way, each available channel must appear in two RCHSs of different lengths and thus constraint 2 is satisfied.



(a) Partition D_u into m_u subsequences.

(b) The distance between the same available channel c in D_u is $L\text{-}N_u$.

Fig. 1. An example of D_u.

We can express the equations (2) and (3) by vector multiplications (5) and (7), respectively. Let $x=[x_1, x_2, \dots, x_\alpha]^T$ be an unknown α-vector and $P=[p_1, p_2, \dots, p_\alpha]$ be a given α-vector. From equation (2), we have the following equations.

$$Px \geq 2N_u. \tag{5}$$

Let A be an α by α matrix as follows.

$$A = \begin{bmatrix} 0 & p_2 & p_3 & \cdots & p_\alpha \\ p_1 & 0 & p_3 & \dots & p_\alpha \\ p_1 & p_2 & 0 & \dots & p_\alpha \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ p_1 & p_2 & p_3 & \dots & 0 \end{bmatrix}, \tag{6}$$

Let $b=[N_u, \dots, N_u]^T$ be a given α-vector, equation (3) can be expressed as equation (7).

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TMC.2018.2790408, IEEE Transactions on Mobile Computing

$$Ax = \begin{bmatrix} 0 & p_2 & p_3 & \cdots & p_\alpha \\ p_1 & 0 & p_3 & \dots & p_\alpha \\ p_1 & p_2 & 0 & \dots & p_\alpha \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ p_1 & p_2 & p_3 & \dots & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ \vdots \\ x_\alpha \end{bmatrix} \geq \begin{bmatrix} N_u \\ N_u \\ \vdots \\ N_u \end{bmatrix}. \quad (7)$$

So, the equations in Theorem 2 can be rewritten as the following equivalent simultaneous equations.

$$\begin{cases} \sum_{i=1}^{\alpha} x_i = m_u \\ Px \geq 2N_u \\ Ax \geq b \\ x \geq 0 \text{ and integral} \end{cases}. \quad (8)$$

By Theorem 1, the TTR is always smaller than the product of the maximum length of the RCHSs of two users and the MTTR is proportional to the maximum length of the RCHSs of user 1 and that of user 2. Moreover, it is obvious that the ETTR could be smaller if the average length of the RCHSs $\sum_{i=1}^{\alpha} p_i x_i / m_u$ become smaller. To have better performance, our object is to minimize the average of the chosen m_u primes and the maximum prime. Since searching the optimal solution from a large number of possible combination of m_u primes satisfying equation (8) is an integer-programming problem, it is hard to obtain the optimal solution of m_u primes whose average of m_u primes and the maximum prime are minimum. Thus, we propose a heuristic method to reduce the average of m_u primes and the maximum prime.

3.3 Algorithm Description

In the following, we introduce a CRT-based multi-radio rendezvous algorithm (CMR), which solves the multi-radio rendezvous problem for heterogeneous CRN. It is worth noting that a user does not need to run any rendezvous algorithm when the user is equipped with radios more than the number of available channels, i.e., $m_u \geq N_u$. Our CMR algorithm consists of feasible primes searching algorithm and RCHSs construction algorithm. In the first algorithm, we find m_u primes satisfying equation (8) as the lengths of m_u RCHSs of user u such that both the maximum and the average of m_u primes are as small as possible. In the second algorithm, we construct the m_u RCHSs such that each available channel appears in at least two RCHSs of different prime lengths.

3.3.1 Feasible Primes Searching Algorithm

In our primes searching algorithm, we select m_u primes (some of them may be duplicate) to satisfy equation (8) and thus we can construct the RCHSs which satisfy constraint 2. Let the number of different primes in the m_u primes be α ($\alpha \geq 2$). Intuitively, if α is larger, the probability to have a smaller average of m_u primes is higher. For example, Assume N_u = 15 and m_u = 5. The possible solution of five primes for α = 2 is 11, 11, 7, 7, 7 whose average length is 8.6. The possible solution for α = 3 is 11, 7, 5, 5, 5 whose average is 6.6. However, the larger α also spends longer computation time to search for suitable primes satisfy equation (8). To have bounded time complexity, we constrain that α must be smaller than a constant $T\alpha$. From the simulation results, the performance of $T\alpha$ = 4 is comparable to that of $T\alpha$ = 5, so we set $T\alpha$ = 4 for our algorithm. In the following, we would search a feasible solution of m_u primes for α = 2. Then, we extend the feasible solution for α = 2 to a solution for $\alpha = T\alpha$.

Here, we illustrate how to find a feasible solution for α = 2. First, we find the smallest two consecutive primes p_1 and p_2 ($p_1 > p_2$) such that $\lceil N_u/p_1 \rceil + \lceil N_u/p_2 \rceil \leq m_u$. Since there are m_u radios, we let p_1 appears $x_1 = \lceil N_u/p_1 \rceil$ times and p_2 appears $x_2 = m_u - x_1$ ($x_2 \geq \lceil N_u/p_2 \rceil$) times. In this way, equation (8) is satisfied as follows.

$x_1 + x_2 = m_u$,

$Px = p_1 x_1 + p_2 x_2 \geq p_1 \cdot \left\lceil \frac{N_u}{p_1} \right\rceil + p_2 \cdot \left\lceil \frac{N_u}{p_2} \right\rceil \geq 2N_u$, and

$Ax = \begin{bmatrix} 0 & p_2 \\ p_1 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \geq \begin{bmatrix} N_u \\ N_u \end{bmatrix}$.

Therefore, we have a feasible solution of m_u primes for α = 2. For example, N_u = 15, m_u = 5. We set the (p_1, p_2) as (2, 3), (3, 5), (5, 7) and so on to see whether $\lceil 15/p_1 \rceil + \lceil 15/p_2 \rceil \leq 5$. The smallest two consecutive primes satisfying the requirement are p_1 = 11, p_2 = 7, and the occurrences of the two primes are $x_1 = \lceil 15/11 \rceil$ =2, x_2 =5 - 2= 3, respectively. The output of five primes for α = 2 are 11, 11, 7, 7, 7.

After we have a feasible solution for α = 2, we can modify it to a solution of more than two distinct primes such that the average of the m_u primes become smaller than the feasible solution. Let β be the number of primes smaller than or equal to p_2. We have at most β-2 primes smaller than p_1 and p_2 to reduce the average of current m_u primes of the feasible solution for α = 2. So, let γ = min(β, $T\alpha$). We construct a prime list $P = [p_1, p_2, \ldots, p_\gamma]$, where $p_3, \ldots, p_\gamma$ are prior γ-2 consecutive primes before p_2 ($p_\gamma < \cdots < p_2 < p_1$). Then we construct the corresponding occurrence list $x = [x_1, x_2, \ldots, x_\gamma]$, where x_i denotes the occurrence of p_i for $1 \leq i \leq \gamma$. The above feasible solution ($p_1 = 11, p_2 = 7, x_1 = 2,$ and $x_2 = 3$)) can be represented by two γ-vectors $P = [p_1, p_2, \ldots, p_\gamma]$ and $x = [x_1, x_2, \ldots, x_\gamma]$, where $x_i = 0 \; \forall i > 2$. To verify equation (8), we extend the matrix A from the size 2 by 2 to the size γ by γ as follows.

$$A = \begin{bmatrix} 0 & p_2 & p_3 & \cdots & p_\gamma \\ p_1 & 0 & p_3 & \dots & p_\gamma \\ p_1 & p_2 & 0 & \dots & p_\gamma \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ p_1 & p_2 & p_3 & \dots & 0 \end{bmatrix}.$$

And set $b = [N_u, \ldots, N_u]^T$ as a γ-vector. It is obvious that both $Px \geq 2N_u$ and $Ax \geq b$ still hold. For example, $T\alpha = \gamma = 4$, $P = [p_1, p_2, p_3, p_4]$ = [11, 7, 5, 3], $x = [x_1, x_2, x_3, x_4]$ = [2, 3, 0, 0], b= [15, 15, 15, 15] and create A as follows.

$$A = \begin{bmatrix} 0 & 7 & 5 & 3 \\ 11 & 0 & 5 & 3 \\ 11 & 7 & 0 & 3 \\ 11 & 7 & 5 & 0 \end{bmatrix}.$$

Then we have $Px = 43 \geq 2N_u$ and $Ax = [21, 22, 43, 43] \geq b$.

Starting from the feasible solution for α = 2, we extend it to a solution for α = 3 by replacing as many p_1 and p_2 as possible with p_3 without violating equation (8). In this case, equation (8) holds if and only if both equations (5) and (7) hold. In next step, we extend the solution for α = 3 to a solution for α = 4 by replacing as many p_1, p_2, and p_3 as possible with p_4 without violating equations (5) and (7). We

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TMC.2018.2790408, IEEE Transactions on Mobile Computing

keep this procedure until the solution is extended to a solution for $\alpha = \gamma$. For example, suppose that C_u = {0, 1, 2, 4, 5, 14, 15, 17, 19, 20, 21, 23, 24, 25, 27}, N_u = 15, m_u = 5, and $T\alpha$ = 4. Fig. 2(a) shows the steps to extend a solution from α = 2 to the solution α = 3. We replace one p_1 by p_3 and x is reset from [2, 3, 0, 0] to [1, 3, 1, 0]. Since both $Px \geq 2N_u$ and $Ax \geq b$ hold, we accept this replacement. Then we replace another p_1 by p_3 and x is reset from [1, 3, 1, 0] to [0, 3, 2, 0]. It is obvious that setting x= [0, 3, 2, 0] violates equation (7). It also contradicts to the fact that p_1 and p_2 are two smallest primes satisfying $\lceil N_u/p_1 \rceil + \lceil N_u/p_2 \rceil \leq m_u$. So, we abort the replacement and reset x as [1, 3, 1, 0]. Since we cannot replace more p_1 by p_3, we stop and try to replace p_2 by p_3. As a result, we replace at most two p_2 by p_3 and x is set as [1, 1, 3, 0]. So the solution of five primes for α = 3 are 11, 7, 5, 5, 5 with P = [11, 7, 5, 3] and x = [1, 1, 3, 0].

Fig. 2(b) shows the steps to extend the solution for α = 3 to the solution for α = 4. In this example, we cannot replace any p_1 or p_2 by p_4 but can replace one p_3 by p_4. The solution of five primes for α = 4 can be represented by P = [11, 7, 5, 3] and x = [1, 1, 2, 1], and the corresponding five primes are 11, 7, 5, 5, 3.

	Before replacement	After replacement			Keep or abort the replacement
	$[x_1, x_2, x_3, x_4]$	$[x_1, x_2, x_3, x_4]$	$P \cdot x \geq 2N_u$	$Ax \geq b$	
(1)	[2, 3, 0, 0]	[1, 3, 1, 0]	True	True	Keep
(2)	[1, 3, 1, 0]	[0, 3, 2, 0]	True	False	Abort
(3)	[1, 3, 1, 0]	[1, 2, 2, 0]	True	True	Keep
(4)	[1, 2, 2, 0]	[1, 1, 3, 0]	True	True	Keep
(5)	[1, 1, 3, 0]	[1, 0, 4, 0]	True	False	Abort
Result	[1, 1, 3, 0]				

(a) Extend a solution for α = 2 to a solution for α = 3

	Before replacement	After replacement			Keep or abort the replacement
	$[x_1, x_2, x_3, x_4]$	$[x_1, x_2, x_3, x_4]$	$P \cdot x \geq 2N_u$	$Ax \geq b$	
(1)	[2, 3, 0, 0]	[1, 3, 1, 0]	True	True	Keep
(2)	[1, 3, 1, 0]	[0, 3, 2, 0]	True	False	Abort
(3)	[1, 3, 1, 0]	[1, 2, 2, 0]	True	True	Keep
(4)	[1, 2, 2, 0]	[1, 1, 3, 0]	True	True	Keep
(5)	[1, 1, 3, 0]	[1, 0, 4, 0]	True	False	Abort
Result	[1, 1, 3, 0]				

(b) Extend a solution for α = 3 to a solution for α = 4.

Fig. 2. The steps to extend the feasible solution of two distinct primes to the solution of four primes

Our feasible primes searching algorithm is summarized in Algorithm 1. In lines 1 - 5, we search m_u primes consisting of two different primes without violating equation (8). In lines 6 - 18, we extend this feasible solution to a solution of α different primes gradually. In line 8, we initialize the occurrence of prime p_i as zero. Then, we try to replace the occurrences of the primes $p_1, p_2, \ldots, p_{i-1}$ by p_i in order. In the while loop of lines 10 - 16, we check if $Px < 2N_u$ or $Ax < b$ holds. If false, we continue to replace one occurrence of p_j with one occurrence of p_i. If true, we restore the value of x_i and x_j in line 13, and exit the while loop.

Algorithm 1: Feasible Primes Searching Algorithm
Input: Number of available channel N_u, radio number m_u
Output: m_u primes, denoted by prime list $P = [p_1, p_2, \ldots, p_\gamma]$ and occurrence list $x=[x_1, x_2, \ldots, x_\gamma]$

/** search for a feasible solution of two primes*/
1. Select p_1 and p_2 ($p_1 > p_2$) as the smallest two consecutive primes such that $\lceil N_u/p_1 \rceil + \lceil N_u/p_2 \rceil \leq m_u$;
2. $x_1 = \lceil N_u/p_1 \rceil$, $x_2 = m_u - x_1$;
3. Let $\gamma = \min(T_\alpha, \beta)$, where β is the number of primes smaller than or equal to p_1;
4. Let $P = [p_1, p_2, p_3, \ldots, p_\gamma]$, where $p_3, \ldots, p_\gamma$ are consecutive $\gamma - 2$ primes before p_2 and $p_1 > p_2 > p_3 > \ldots > p_\gamma$;
5. Let $x = [x_1, x_2, x_3, \ldots, x_\gamma]$, where $x_i = 0\ \forall i > 2$;

/** extend to a solution of γ primes gradually*/
6. Construct an $\gamma \times \gamma$ matrix A according to (6) and γ-vector $b = [N_u, \ldots, N_u]$;
7. **for** i = 3 to γ
8. $x_i = 0$;
9. **for** j = 1 to i-1
10. **while** $x_j > 0$
11. $x_j = x_j - 1$, $x_i = x_i + 1$;
12. **if** $Px < 2N_u$ **or** $Ax < b$ **then**
13. $x_j = x_j + 1$, $x_i = x_i - 1$;
14. break;
15. **end if**
16. **end while**
17. **end for**
18. **end for**

3.3.2 RCHSs Construction Algorithm

Here, we construct the RCHSs based on Algorithm 1. The lengths of the RCHSs are m_u primes, which consists of γ distinct primes $p_1, p_2, \ldots, p_\gamma$ and each p_i has occurrence x_i. Our RCHSs construction algorithm is shown as follows. First, we construct a sequence D_u by concatenating N_u available channels in C_u, $L = \sum_{i=1}^{\gamma} p_i x_i - 2N_u$ random channels, and N_u available channels in C_u according to equation (4). Second, we partition D_u into x_1 subsequences of length p_1, x_2 subsequences of length p_2, …, and x_γ subsequences of length p_γ in order, denoted by $S_1, S_2, \ldots$, and S_{m_u}, respectively. For example, γ = 4, C_u = {0, 1, 2, 4, 5, 14, 15, 17, 19, 20, 21, 23, 24, 25, 27}, m_u = 5, P=[11, 7, 5, 3], x=[1, 1, 2, 1], the lengths of the five RCHSs are 11, 7, 5, 5, 3. We create D_u = {0, 1, 2, 4, 5, 14, 15, 17, 19, 20, 21, 23, 24, 25, 27, r, 0, 1, 2, 4, 5, 14, 15, 17, 19, 20, 21, 23, 24, 25, 27}, where r denotes a random channel in C_u. The five RCHSs according the above example are S_1 = {0, 1, 2, 4, 5, 14, 15, 17, 19, 20, 21}, S_2 = {23, 24, 25, 27, r, 0, 1}, S_3 = {2, 4, 5, 14, 15}, S_4 = {17, 19, 20, 21, 23}, and S_5 = {24, 25, 27}.

Third, for each S_i, we shuffle the elements in S_i. That is, permute the elements in S_i randomly. This shuffle procedure is used to reduce the chance that different radios hop on the same channel at the same time slot. This is because there may exist many repeat patterns between the RCHSs on different radios. For example, in the above example, there is a repeat pattern {2, 4, 5, 14, 15} in the RCHS of S_1 (radio 1) and the RCHS of S_3 (radio 3). Once radio 1 and radio 3 hopping to channel 2 in the same time slot, there are consecutive five time slots such that both radios 1 and 3 hop on the same channels. To avoid this problem, we launch the shuffle procedure. After the shuffle procedure, the five RCHSs are S_1 = {14, 2, 21, 15, 17, 5, 0, 1, 4, 19, 20}, S_2 = {r, 27, 25, 1, 24, 0, 23}, S_3 = {15, 5, 14, 4, 2}, S_4 = {19, 21, 23, 20, 17}, and S_5 = {27, 25, 24}. The five radios of user u repeatedly hop on the channels of the five RCHSs as shown in Fig. 3. Each time user u meets the random channel r, user u randomly hops on an available channel in C_u.

Radio 1	14	2	21	15	17	5	0	1	4	19	20	14	2	21	15	…
Radio 2	r	27	25	1	24	0	23	r	27	25	1	24	0	23	r	…
Radio 3	15	5	14	4	2	15	5	14	4	2	15	5	14	4	2	…
Radio 4	19	21	23	20	17	19	21	23	20	17	19	21	23	20	17	…
Radio 5	27	25	24	27	25	24	27	25	24	27	25	24	27	25	24	…

time →

Fig. 3. An example of RCHSs with five radios.

Our RCHSs construction algorithm is summarized in Algorithm 2. In line 1, we construct D_u. In lines 2 - 8, we partition D_u into m_u subsequences $S_1, S_2, \ldots, S_{m_u}$. In lines 9 - 11, we shuffle every sequence S_i, for $1 \le i \le m_u$.

Algorithm 2: Construct RCHSs for user u
Input: Available channel set $C_u = \{c_u(0), c_u(1), \ldots, c_u(N_u-1)\}$, radio number m_u, prime list $P = [p_1, p_2, .., p_\gamma]$ and the occurrences list $x = [x_1, x_2, .., x_\gamma]$
Output: RCHSs for the m_u radios S_i, for $1 \le i \le m_u$

```
1.   Construct D_u according to equation (4);
2.   idx = 0, e = 1, γ = sizeof(P);
3.   for i = 1 to γ
4.     for j = 1 to x_i
5.       S_e = D_u[idx, ..., idx + p_i − 1];
6.       idx = idx+p_i, e = e+1;
7.     end for
8.   end for
9.   for i = 1 to m_u
10.    Shuffle elements in S_i;
11.  end for
```

For example, assume that C_1 = {2, 3, 4, 10, 11, 13}, C_2 = {0, 4, 5, 7, 8, 12, 19}, $m_1 = 2$, $m_2 = 3$, and $T\alpha = 3$. Radios 1 and 2 of user 1 are assigned with RCHSs of lengths 11 and 7 respectively, while Radios 1, 2, and 3 of user 2 are assigned with RCHSs of lengths 7, 5, and 3, respectively. According to our algorithms 1 and 2, the common channel 4 appears in the RCHSs of radio 1 and radio 2 of user 1 and in the RCHSs of radio 1 and radio 2 of user 2. Since 7 is co-prime with 5, by CRT, Radio 2 of user 1 can rendezvous with Radio 2 of user 2 on channel 4 within 35 time slots as shown in Fig. 4.

3.4 Time Complexity Analysis

In Algorithm 1, p_1 and p_2 are two smallest consecutive prime numbers such that $\lceil N_u/p_1 \rceil + \lceil N_u/p_2 \rceil \le m_u$, which implies $\lceil N_u/p_2 \rceil + \lceil N_u/p_3 \rceil > m_u$. Since $p_2 > p_3$, we can derive $2N_u/p_3 > m_u$ and thus $p_3 < 2N_u/m_u$. By Bertrand-Chebyshev Theorem, there exists a prime p such that $p_3 < p < 2p_3 < 4N_u/m_u$. Note that, p_2 is the next prime larger than p_3, so $p_2 < 2p_3 < 4N_u/m_u$. Similarly, we have $p_1 < 2p_2 < 8N_u/m_u$. So, it takes $O(N_u/m_u)$ time to find a feasible solution for α = 2. To search a solution for $\alpha = \gamma$, we need to execute the replacement operation at most $(\gamma$-2$)m_u$ times, each time takes $O(\gamma^2)$ time to compute if both $Px \ge 2N_u$ and $Ax \ge b$ hold, which takes $O(\gamma^3 m_u)$ time in total. Since $\gamma \le T\alpha$ ($T\alpha$ is constant) and we execute this algorithm only when $m_u < N_u$, it takes $O(N_u)$ time to find a solution for $\alpha = \gamma$.

User 1	Radio 1	13	r	4	r	11	r	2	10	r	3	r	13	r	4	…
	Radio 2	4	2	10	13	3	r	11	4	2	10	13	3	r	11	…
User 2	Radio 1				12	7	4	8	5	r	0	12	7	4	8	…
	Radio 2				0	19	5	7	4	0	19	5	7	4	0	…
	Radio 3				12	8	19	12	8	19	12	8	19	12	8	…

time →

Fig. 4. An example of rendezvous between two multi-radio users.

In Algorithm 2, it takes $O(L)$ to construct D_u and partition it into $S_1, S_2, \ldots, S_{m_u}$, where $L = \sum_{i=1}^{\alpha} p_i x_i$. In addition, we run the modern version of Fisher-Yates Shuffle algorithm to shuffle $S_1, S_2, \ldots, S_{m_u}$, respectively. The time to complete the shuffle procedure is $O(|S_1|+|S_2|+\ldots+|S_{m_u}|)=O(L)$. Note that $L \le p_1 \sum_{i=1}^{\alpha} x_i = p_1 \times m_u \le 8N_u$. So, the time complexity of the Algorithm 2 is $O(N_u)$. Thus, the time complexity of our CMR algorithm is $O(N_u)$.

3.5 Performance Analysis

Below, Theorem 3 further shows that CMR has MTTR = $O(N_1N_2/m_1m_2)$ and maximum rendezvous diversity $|C_1 \cap C_2|$.

Theorem 3. *CMR has MTTR = $O(N_1N_2/m_1m_2)$ and maximum rendezvous diversity $|C_1 \cap C_2|$.*

Proof: Recall that each available channel will appear in at least two RCHSs of different prime lengths. So, we assume the common channel of the two users $c \in C_1 \cap C_2$ appears in two RCHSs of user 1 whose lengths are $l_{1,1}$ and $l_{1,2}$ ($l_{1,1} > l_{1,2}$). Similarly, suppose c appears in two RCHSs of user 2 whose lengths are $l_{2,1}$ and $l_{2,2}$ ($l_{2,1} > l_{2,2}$). We consider the following cases.

Case 1: $l_{1,2} \ne l_{2,2}$. By Theorem 1, user 1 and user 2 rendezvous within $l_{1,2} \times l_{2,2}$ time slots. By the analysis of time complexity in Section 3.4, $l_{1,2} < 4N_1/m_1$ and $l_{2,2} < 4N_2/m_2$. So, the MTTR $\le 16N_1N_2/m_1m_2$.

Case 2: $l_{1,2} = l_{2,2}$. In this case, it implies $l_{1,1} \ne l_{2,2}$. By Theorem 1, user 1 and user 2 can rendezvous within $l_{1,1} \times l_{2,2}$ time slots. It is obvious that $l_{1,1} < 8N_1/m_1$ and $l_{2,2} < 4N_1/m_1$. So, the MTTR $\le 32N_1N_2/m_1m_2$.

Combining the two cases, for each channel $c \in C_1 \cap C_2$, user 1 and user 2 rendezvous on channel c within 32 N_1N_2/m_1m_2 time slots. Therefore, CMR has maximum rendezvous diversity within $O(N_1N_2/m_1m_2)$ time. □

4 Performance Evaluation

In this section, we evaluate the proposed algorithm under multi-radio CRN circumstance and compare it with the state-of-the-arts multi-radio algorithms under various environments. Since MSS [29] may not have guaranteed rendezvous if they start the rendezvous process at the same time, we only choose the other heterogeneous multi-radio

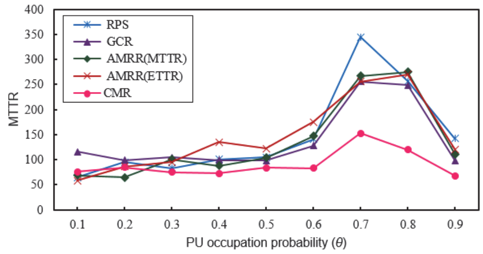
This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TMC.2018.2790408, IEEE Transactions on Mobile Computing

algorithms AMRR [28] and GCR [30] and one of the representative homogeneous algorithm RPS [22] for MTTR and ETTR comparisons. In addition, AMRR allows users to choose to optimize MTTR or ETTR, so we simulate two versions of AMRR, one is AMRR with optimized MTTR, which assigns half of the radios as stay radios, and the other is AMRR with optimized ETTR, which assigns one radio as stay radio. We implement RPS, AMRR, GCR and CMR algorithms in the asynchronous and asymmetric environment. Recall that, GCR only uses pairs of radios for CH. In the following simulations, when a user is equipped with an odd number of radios, we let the unpaired radio of GCR user execute the random CH algorithm for fairness. In each time slot, the unpaired radio randomly hops on an available channel. Consider the following parameters, the number of channels in the whole channel set C is N. It is noted that the parameter N is not necessary for the heterogeneous algorithms but is essential for homogeneous algorithms such as RPS. To evaluate the performance of our algorithm, we consider the rendezvous between two users. The numbers of available channels of two users are $N_1 = |C_1|$ and $N_2 = |C_2|$, respectively. The number of common channels between two users is denoted as G. User 1 and user 2 are equipped with m_1 and m_2 radios, respectively. For each set of parameter values, we perform independent runs and take the maximum/average time as MTTR/ETTR. The simulation data is in 95% confidence interval. Since the confidence intervals of ETTR are very small in our simulations, the confidence intervals are not drawn in figures. Because the MTTR is the worst case of TTR, we only select the largest TTR as the results.

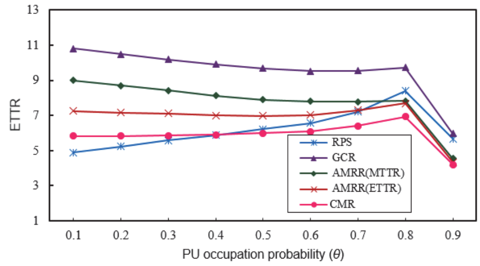
In this simulation, we introduce a new parameter θ, that defines PU occupation probability. Each channel in a channel set C has a probability of θ that occupied by PUs. That is, for each user, each channel in C has a probability of (1-θ) to be sensed as idle. So, N_1 and N_2 are normally distributed random variables. According to our simulations, the performance of $T\alpha = 4$ is similar to $T\alpha = 5$ but has lower computation cost than $T\alpha = 5$. Therefore, we set $T\alpha = 4$ in the following simulations.

4.1 Impact of the PU Occupation Probability

In this simulation, we vary the PU occupation probability from 10% ~ 90% ($\theta = 0.1 \sim \theta = 0.9$). The number of global channels $N = 80$, $m_1 = 3$, and $m_2 = 5$. In Fig. 5(a), we can see that CMR has the best MTTR when $\theta > 0.2$ compared to the previous works. The MTTR of AMRR with optimized MTTR is better than CMR on $\theta = 0.2$. However, its performance of ETTR is worse than CMR. In terms of ETTR, CMR has the best ETTR when $\theta > 0.3$ as shown in Fig. 5(b). Although RPS has better ETTR than CMR when $\theta \leq 0.3$, the MTTR of RPS is worse than CMR. In overall, CMR has the better performance of ETTR and MTTR than the previous works.

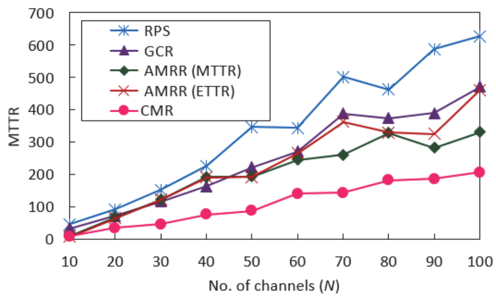


(a) MTTR vs. the PU occupation probability θ

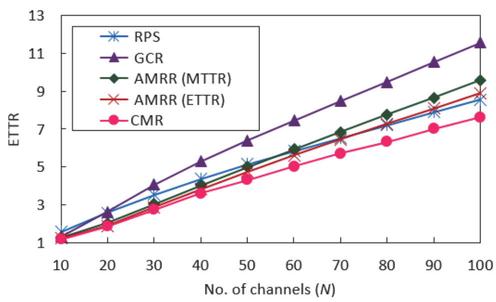


(b) ETTR vs. the PU occupation probability θ

Fig. 5. Performance of MTTR and ETTR with different protocols when $N = 80$, $m_1 = 3$ and $m_2 = 5$.



(a) MTTR vs. the number of total channels N



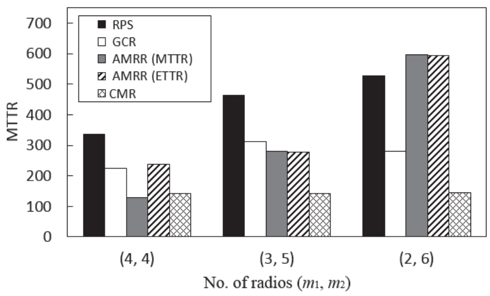
(b) ETTR vs. the number of total channels N

Fig. 6. Performance of MTTR and ETTR with different protocols when $\theta = 0.7$, $m_1 = 3$ and $m_2 = 5$.

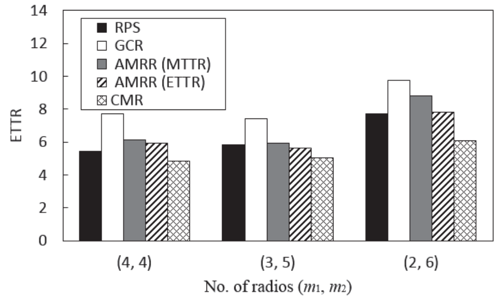
4.2 Impact of the Number of Licensed Channels

In this simulation, we vary the number of licensed channels N from 10 to 100 with fixed $\theta = 0.7$, $m_1 = 3$, $m_2 = 5$. In Fig. 6(a), we can see that CMR has significant improvement in MTTR. In this simulation, CMR has at least 45% improvement in MTTR compared to the other algorithms. As expected, the MTTR of AMRR with optimized MTTR is smaller than that of AMRR with optimized ETTR. In terms of ETTR, CMR still has the smaller ETTR than the previous works as shown in Fig. 6(b). Compared to the other algorithms, CMR has at least 10% improvement in ETTR. In addition, AMRR with optimized ETTR has better ETTR than AMRR with optimized MTTR.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TMC.2018.2790408, IEEE Transactions on Mobile Computing

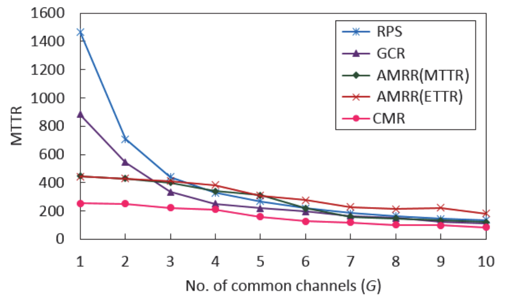


(a) MTTR vs. the number of radios m_1 and m_2

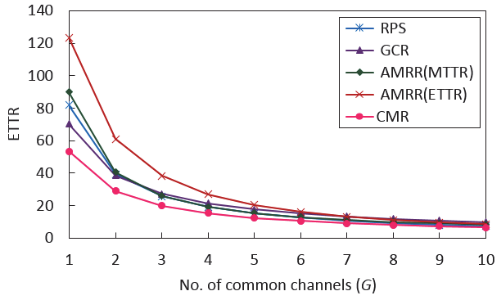


(b) ETTR vs. the number of radios m_1 and m_2

Fig. 7. Performance of MTTR and ETTR with different number of radios m_1 and m_2.

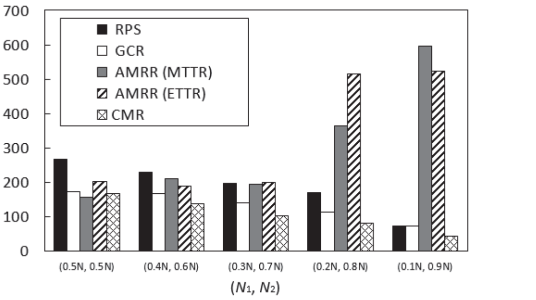


(a) MTTR vs. the number of common channels

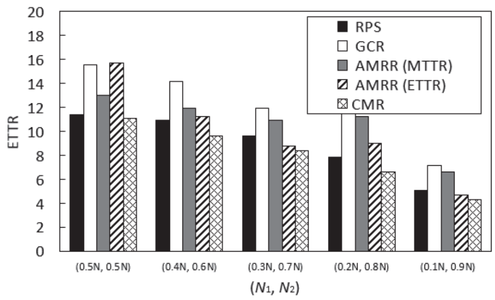


(b) ETTR vs. the number of common channels

Fig. 8. Performance of MTTR and ETTR for various common channels G with m_1 = 3, m_2 = 5.



(a) MTTR vs. the number of available channels N_1 and N_2



(b) ETTR vs. the number of available channels N_1 and N_2

Fig. 9. Performance of MTTR and ETTR for various N_1 and N_2 with m_1 = m_2 = 4.

4.3 Impact of Different Radios m_1 and m_2

In this simulation, we fix the number of all licensed channels N = 60 and PU occupation rate θ = 0.7. We study that when the total number of radios for two users is fixed (i.e., $m_1 + m_2$ is constant), how does the allocation of (m_1 and m_2) affect the rendezvous performance? We let $m_1 + m_2 = 8$ and simulate three cases: 4 radios vs. 4 radios, 3 radios vs. 5 radios, and 2 radios vs. 6 radios. We denote the three cases by (4, 4), (3, 5), and (2, 6). Fig. 7(a) shows the MTTR of the four algorithms. We can find that the MTTR of CMR is stable under the three cases. The MTTR of all algorithms except CMR and GCR increases as the difference between m_1 and m_2 increases. In theory, the MTTR of GCR and CMR should be inversely proportional to m_1m_2. However, GCR has its largest MTTR under the case of (3, 5). This is because GCR only utilize pairs of radios even with the help of unpaired radio executing random CH algorithm. AMRR has the smallest MTTR at (4, 4) and largest MTTR at (2, 6), which also verifies that the MTTR of AMRR is inversely proportional to max(m_1^2, m_2^2). Fig. 7(b) shows that CMR has the smallest ETTR compared to other algorithms under the three cases. Since CMR has stable MTTR and ETTR under different allocations of m_1 and m_2, CMR is strongly recommended in the scenario that users are unaware of the number of radios equipped by each other.

4.4 Impact of the Number of Common Channels

In this simulation, we let N = 60, $N_1 = N_2 = N/2$, m_1 = 3, m_2 = 5 and vary the number of common channels G between each pair of users. Fig. 8(a) shows the MTTR of the four algorithms decrease with the increase of the common channels. Even when G is small, AMRR and CMR can achieve fast rendezvous. Fig. 8(b) shows the ETTR of RPS, GCR, AMRR, and CMR. The CMR still has the smaller ETTR than the other three algorithms. It is obvious that the MTTR and ETTR decrease as the number of common channels G increases for all algorithms. Compared to the other algorithms, CMR has at least 50% and 25% improvement in MTTR and ETTR, respectively when the number of G is small than or equal to three.

4.5 Impact of the Different Number of Available Channels N_1 and N_2

In the following simulation, we want to know how the MTTR and ETTR is affected by the difference of available channels between two users. Let $N = 60$, $m_1 = m_2 = 4$, and $G = 3$, vary the number of available channels N_1 and N_2 with $N_1 + N_2 = N$. We randomly choose N_1 and N_2 available channels for user 1 and user 2, respectively. Fig. 9(a) shows that the MTTR of RPS, GCR, and CMR decreases with the increasing of $|N_1\text{-}N_2|$. However, the MTTR of AMRR increases with the increasing of $|N_1\text{-}N_2|$. This is because the MTTR of AMRR is proportional to $\max(N_1^2, N_2^2)$. The MTTR of CMR is the smallest compared to other algorithms. Fig. 9(b) shows that the ETTR of all algorithms decreases with the increasing of $|N_1\text{-}N_2|$ and CMR has the shortest ETTR among all algorithms.

5 Conclusion

In this paper, we have proposed an efficient multi-radio rendezvous algorithm CMR, for heterogeneous CRNs. Unlike the algorithms for homogeneous CRN, CMR allows SUs to have different spectrum-sensing capabilities and be unaware of the number of total licensed channels N. To the best of our knowledge, we are the first protocol to use multiple primes on multi-radio in the distributed heterogeneous model. To find the optimal solution of primes to minimize MTTR is an integer-programming problem, we proposed a heuristics algorithm to solve this problem in polynomial time. The CMR algorithm applies Chinese Remainder Theorem to guarantee rendezvous and can achieve maximum rendezvous diversity within $O(N_1N_2/m_1m_2)$ time. In addition, the MTTR of CMR is more stable than previous multi-radio rendezvous algorithms under various environments. The simulation results show that CMR has the shorter MTTR and ETTR than the other rendezvous schemes in most cases.

References

[1] FCC Spectrum Policy Task Force Report, ET Docket No. 02-135, Nov. 2002.

[2] A. Ghasemi and E.S. Sousa, "Spectrum Sensing in Cognitive Radio Networks: Requirements, Challenges and Design Trade-offs," *IEEE Communication Magazine*, Vol. 46, No. 4, pp. 32 – 39, Apr. 2008.

[3] C. Cordeiro, K. Challapali, D. Birru, and N. Sai Shankar, "IEEE 802.22: The First Worldwide Wireless Standard based on Cognitive Radios," *Proceedings of the IEEE International Symposium on New Frontiers in Dynamic Spectrum Access Networks* (DySPAN), pp. 328 – 337, Baltimore, MD, USA, Nov. 2005.

[4] S. Krishnamurthy, M. Thoppian, S. Venkatesan, and R. Prakash, "Control Channel Based MAC-Layer Configuration, Routing and Situation Awareness for Cognitive Radio Networks," *Proceedings of the IEEE Military Communications Conference* (MILCOM), pp. 455 – 460, Atlantic City, NJ, USA, Oct. 2005.

[5] J. Pérez-Romero, O. Salient, R. Agustí, and L. Giupponi, "A Novel On-Demand Cognitive Pilot Channel Enabling Dynamic Spectrum Allocation," *Proceedings of the IEEE International Symposium on New Frontiers in Dynamic Spectrum Access Networks* (DySPAN), pp. 46 – 54, Dublin, Ireland, Apr. 2007.

[6] C.-F. Shih, T. Y. Wu, and W. Liao, "DH-MAC: A Dynamic Channel Hopping MAC Protocol for Cognitive Radio Networks," *Proceedings of the IEEE International Conference on Communications* (ICC), pp. 1 – 5, Cape Town, South Africa, May 2010.

[7] L. Chen, S. Shi, K. Bian, and Y. Ji, "Optimizing Average-Maximum TTR Trade-off for Cognitive Radio Rendezvous," *Proceedings of the IEEE International Conference on Communications* (ICC), pp. 7707 – 7712, London, UK, June 2015.

[8] 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, July 2013.

[9] L. Chen, K. Bian, L. Chen, C. Liu, J.-M. J. Park, and X. Li, "A Group-Theoretic Framework for Rendezvous in Heterogeneous Cognitive Radio Networks," *Proceedings of the fifteenth ACM international symposium on Mobile ad hoc networking and computing* (MobiHoc), pp. 165 – 174, Philadelphia, PA, USA, Aug. 2014.

[10] P. Bahl, R. Chandra and J. Dunagan, "SSCH: Slotted Seeded Channel Hopping for Capacity Improvement in IEEE 802.11 Ad Hoc Wireless Networks," *Proceedings of the Annual International Conference on Mobile Computing and Networking* (MobiCom), pp. 216 – 230, NY, USA, Oct. 2004.

[11] I. Chuang, H.-Y. Wu, K.-R. Lee, and Y.-H. Kuo, "A Fast Blind Rendezvous Method by Alternate Hop-and-Wait Channel Hopping in Cognitive Radio Networks," *IEEE Transactions on Mobile Computing*, Vol. 13, No. 99, pp. 2171 – 2184, Jan. 2014.

[12] Z. Gu, Q.-S. Hua, and W. Dai, "Local Sequence Based Rendezvous Algorithms for Cognitive Radio Networks," *Proceedings of the IEEE 11th International Conference on Sensing, Communication, and Networking* (SECON), pp. 194 – 202, Singapore, July 2014.

[13] G.-Y. Chang, J.-F. Huang, and Y.-S. Wang, "Matrix-Based Channel Hopping Algorithms for Cognitive Radio Networks," *IEEE Transactions on Wireless Communications*, Vol. 14, No. 5, pp. 2755 – 2768, Jan. 2015.

[14] J. Li and Jiang Xie, "Practical Fast Multiple Radio Blind Rendezvous Schemes in Ad-Hoc Cognitive Radio Networks," *Proceedings of Resilience Week* (RWS), pp. 1 – 6, Philadelphia, PA, USA, Aug. 2015.

[15] R. N. Yadav, R. Misra, "Periodic Channel-Hopping Sequence for Rendezvous in Cognitive Radio Networks," *Proceedings of International Conference on Advances in Computing, Communications and Informatics* (ICACCI), pp. 1787 – 1792, Kochi, India, Aug. 2015.

[16] L. Yu, H. Liu, Y.-W. Leung, X. Chu, and Z. Lin, "Channel-Hopping Based on Available Channel Set for Rendezvous of Cognitive Radios," *Proceedings of the IEEE International Conference on Communications* (ICC), pp. 1573 – 1579, Sydney, NSW, June 2014.

[17] 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, Oct. 2012.

[18] Z. Gu, Q.S. Hua, Y. Wang, and F. C. M. Lau, "Nearly Optimal Asynchronous Blind Rendezvous Algorithm for Cognitive Radio Networks," *Proceedings of the IEEE 10th International Conference on Sensing, Communication, and Networking* (SECON), pp. 371 – 379, New Orleans, LA, USA, June 2013.

[19] Z. Lin, H. Liu, X. Chu, and Y.-W. Leung, "Enhanced Jump-Stay Rendezvous Algorithm for Cognitive Radio Networks," *IEEE Communications Letters*, Vol. 17, No. 9, pp. 1742 – 1745, Sept. 2013.

[20] 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, Oct. 2010.

[21] G.-Y. Chang, W.-H. Teng, H.-Y. Chen, and J.-P. Sheu, "Novel Channel-Hopping Schemes for Cognitive Radio Networks," *IEEE Transactions on Mobile Computing*, Vol. 13, No. 2, pp. 407 – 421, Feb. 2014.

[22] L. Yu, H. Liu, Y. W. Leung, X. Chu, and Z. Lin, "Multiple Radios for Fast Rendezvous in Cognitive Radio Networks," *IEEE Transactions on Mobile Computing*, Vol. 14, No. 9, pp. 1917-1931, Sep., 2015.

[23] K.-Y. Cheon, C.-J. Kim, and J. K. Choi, "Rendezvous for Self-Organizing MANET with Multiple Radio," *Proceedings of International Conference on Communications Technology Convergence* (ICTC), pp. 906 – 911, Jeju, Korea, Oct. 2015.

[24] R. Paul, Y. Z. Jembre, and Y.-J. Choi. "Multi-Interface Rendezvous in Self-Organizing Cognitive Radio Networks," *Proceedings of the IEEE International Symposium on New Frontiers in Dynamic Spectrum Access Networks* (DySPAN), pp. 531 – 540, McLean, VA, USA, Apr. 2014.

[25] S.H. Wu, C.C. Wu, W.K. Hon, and K.G. Shin, "Rendezvous for Heterogeneous Spectrum-Agile Devices," *Proceedings of the IEEE Conference on Computer Communications* (INFOCOM), pp. 2247-2255, Toronto, ON, USA, Apr. 2014.

[26] C.-C. Wu and S.-H. Wu, "On Bridging the Gap between Homogeneous and Heterogeneous Rendezvous Schemes for Cognitive Radios," *Proceedings of the fourteenth ACM international symposium on Mobile ad hoc networking and computing* (MobiHoc), pp. 207 – 216, bangalore, India, July 2013.

[27] Z. Gu, H. Pu, Q.-S. Hua, and F. C. M. Lau, " Improved Rendezvous Algorithms for Heterogeneous Cognitive Radio Networks," *Proceedings of the IEEE Conference on Computer Communications* (INFOCOM), pp. 154 – 162, Kowloon, Hong Kong, May 2015.

[28] L. Yu, H. Liu, Y. W. Leung, X. Chu, and Z. Lin, "Adjustable Rendezvous in Multi-Radio Cognitive Radio Networks," *Proceedings of the IEEE Global Communications Conference* (GLOBECOM), pp. 1 – 7, San Diego, CA, Dec 2015.

[29] B. Yang, W. Liang, M. Zheng, and Y. C. Liang, "Fully Distributed Channel-Hopping Algorithm for Rendezvous Setup in Cognitive Multiradio Networks," *IEEE Transactions on Vehicular Technology,* Vol. 65, Isssue 10, pp. 8629-8643, Oct. 2016.

[30] G. Li, Z. Gu, X. Lin, H. Pu, and Q.-S. Hua, "Deterministic Distributed Rendezvous Algorithms for Multi-Radio Cognitive Radio Networks," *Proceedings of the 17th ACM international conference on Modeling, analysis and simulation of wireless and mobile systems* (MSWiM), pp. 313 – 320, Montreal, Canada, Sep. 2014.

[31] P. Bahl, A. Adya, J. Padhye, and A. Walman, "Reconsidering Wireless Systems with Multiple Radios," *ACM SIGCOMM Computer Communication Review*, Vol. 34, No. 5, pp. 39-46, Oct. 2004.

Jang-Ping Sheu received the B.S. degree in computer science from Tamkang University, Taiwan, Republic of China, in 1981, and the M.S. and Ph.D. degrees in computer science from National Tsing Hua University, Taiwan, Republic of China, in 1983 and 1987, respectively. He is currently a Chair Professor of the Department of Computer Science, National Tsing Hua University. He was a Chair of Department of Computer Science and Information Engineering, National Central University from 1997 to 1999. He was a Director of Computer Center, National Central University from 2003 to 2006. He was a Director of Computer and Communication Research Center from 2009 to 2015, National Tsing Hua University. He was an Associate Dean of the College of Electrical and Computer Science from 2016 to 2017, National Tsing Hua University. His current research interests include wireless communications, mobile computing, and software-defined networks. He was an associate editor of the IEEE Transactions on Parallel and Distributed Systems and International Journal of Sensor Networks. He is an Advisory Board Member of the International Journal of Ad Hoc and Ubiquitous Computing and International Journal of Vehicle Information and Communication Systems.

He received the Distinguished Research Awards of the National Science Council of the Republic of China in 1993-1994, 1995-1996, and 1997-1998. He received the Distinguished Engineering Professor Award of the Chinese Institute of Engineers in 2003. He received the K. -T. Li Research Breakthrough Award of the Institute of Information and Computing Machinery in 2007. He received the Y. Z. Hsu Scientific Chair Professor Award and Pan Wen Yuan Outstanding Research Award in 2009 and 2014, respectively. He received the Academic Award in Engineering from Ministry of Education and Medal of Honor in Information Sciences from Institute of Information and Computing Machinery in 2016 and 2017, respectively. Dr. Sheu is an IEEE Fellow and a member of Phi Tau Phi Society.

Ji-Jhen Lin received the B.S. degree in interdisciplinary program of science in College of Science and M.S. degree in computer science from National Tsing-Hua University, Hsinchu, Taiwan, in 2014 and 2016, respectively. Her research interests include cognitive radio networks and wireless networks. She is currently an engineer in MediaTek Inc.