Hindawi Publishing Corporation
Journal of Electrical and Computer Engineering
Volume 2013, Article ID 715816, 18 pages
http://dx.doi.org/10.1155/2013/715816

Research Article

A Rendezvous Protocol with the Heterogeneous Spectrum Availability Analysis for Cognitive Radio Ad Hoc Networks

Sylwia Romaszko

Institute for Networked Systems, RWTH Aachen University, Kackertstraße 9, 52072 Aachen, Germany

Correspondence should be addressed to Sylwia Romaszko; sar@inets.rwth-aachen.de

Received 4 May 2012; Revised 11 August 2012; Accepted 20 August 2012

Academic Editor: Fangwen Fu

In cognitive radio ad hoc networks, a frequently changing environment, varying channel occupancy statistics, and heterogeneous spectrum availability result in a need to meet on a common channel and to initiate a communication. This process of two or more CRs, meeting each other in the same channel, is called a rendezvous (RDV). RDV is essential for establishment of a communication link. Hence, methods guaranteeing that all nodes meet periodically in reasonable periods of time should be developed. In this study, we evaluate a torus Quorum System (QS) and Difference Set (DS) based rendezvous protocol in an asymmetric channel view case (heterogeneous channel availability). Regardless of the diversity of channels of CRs the protocol guarantees RDV on either all channels or almost all channels. Furthermore, the nodes meet multiple times on different channels in a period, which increases the chance of successful establishment of a communication link.

1. Introduction

Cognitive Radio Networks (CRNs) can operate in licensed and unlicensed bands. A spectrum hole is a frequency band that has been assigned to a primary user (PU) but is not utilized by this PU at a particular time and specific geographic location [1]. Secondary Users (SUs), also called Cognitive Radio (CR) users, have only an opportunistic spectrum access to these bands, that is, the licensed spectrum; that is, temporarily vacant may be used by a CR user [2]. The appearance of a PU means that CRs must vacate immediately the occupied frequency band. Hence, link recovery information (and a new determined channel) cannot be circulated over the previously used spectrum band. The dissemination of control traffic signals (on a common control channel, CCC) must be fast, since the SU might have a limited duration of time in which a spectrum hole is likely to be available. Furthermore, such dissemination should be robust and minimize the use of energy and computing resources [1]. Hence, the classical CCC of multichannel networks is not an attractive solution for CRNs. Moreover, unlike in the case of classical ad hoc networks, a CR has the heterogeneous spectrum availability which is varying over time and space due to the licensed holders' activities. In other words, the available radio resources (channel set) can be different for different CRs (also known as asymmetric channel occupancy knowledge) in the same network due to different location or PUs activities. From all these unique CRN characteristics arises a big research challenge; namely, how to achieve a rendezvous between nodes. Haykin's question, *how can we establish the dissemination of control traffic signals between neighboring SUs in cognitive radio ad hoc networks, which is rapid, robust, and efficient* [1], is still not fully answered and is a challenging problem.

In this study, we investigate a distributed rendezvous protocol for cognitive radio ad hoc networks employing frequency hopping (FH) techniques in the case of heterogeneous channel availability. FH is known for decreasing the probability of interference to PUs thanks to the frequent switching of the occupied channels. However, some undesirable assumptions of typical FH (e.g., the need of synchronization and exchange of hopping patterns, the same length of hopping sequence patterns) must be overcome (e.g., incumbents presence and as consequence the need to vacate the channel immediately by a SU; heterogeneous channel availability, etc.). Before going into details of the protocol, we formulate the rendezvous (RDV) problem with regard to channel switching and an asymmetric channel occupancy

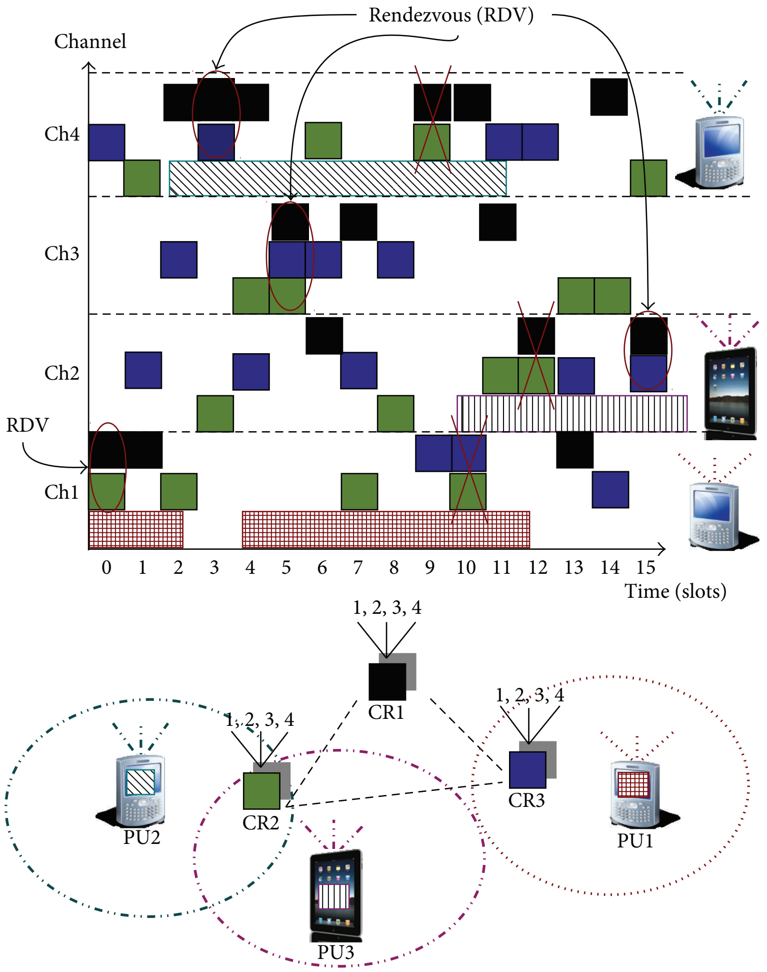


Figure 1: Rendezvous in CRNs: if there is no hatched rectangle of a PU on a particular channel, it means that this PU is not active on this channel; for example, there is no PU's activity on Channel 3 (Ch3). CR1, not being in the range of any of PUs, can use all four channels. CR3 can sporadically use Channel 1 due to the PU1 activity there. CR2 has two available channels (1, 3), while the availability in Channels 2 and 4 is limited to spectrum holes of PU2 and PU3. Hence, CRs have an *asymmetric channel occupancy knowledge*. In a 16 slots of long time period, CR1 user can meet CR2 user on Channel 1 in slot 0 and on Channel 3 in slot 5. However, they cannot communicate on Channel 4 in slot 9, nor on Channel 2 in slot 12, due to the PU2 and PU3 activities there (CR2 cannot transmit).

knowledge. A RDV problem in multichannel networks refers to the ability of two or more CRs to meet each other in the same channel.

Figure 1 illustrates an example of the channel occupancy by three primary users (hatched rectangles) and three secondary users (solid filled squares or rectangles) hopping through four channels (Ch1...Ch4) in a 16 slots of long cycle (for the sake of simplicity we illustrate the case where secondary users have a common time-slot system. This is done without losing generality, and, in fact, we have also presented asynchronous operations in [3]). Here, by a slot we understand some period of time within which SUs can communicate with each other (i.e., exchanging information, receiving and transmitting). The situation depicted in Figure 1 is very optimistic; nodes are able to meet once, twice or even three times (CR1 and CR3). However, while hopping or switching channels randomly it might happen that nodes never meet. In order to deal with the aforementioned RDV problem, we proposed in [4] the protocol which is based on the torus quorum system (tQS) concept.

Thanks to the use of the quorum system (QS) and also the difference set (DS) concept we are able to guarantee meeting on all r channels in the case of nodes with a symmetric channel view (SCHv), and almost on all channels in an asymmetric channel view (ACHv) case. We emphasize that CR nodes do not need to have mutual knowledge of their hopping sequences information because the use of QS properties is sufficient. Thanks to the QS intersection property guarantee cognitive radios will meet (intersect) when selecting quorum-based hopping sequences from the same QS. Quorums, satisfying the Rotation Closure Property, guarantee the intersection even if the cycles of the CRs are not aligned and, therefore, can be used in asynchronous protocols. The torus QS [5], exploited in this work, represents an intuitive and simple method to employ where a quorum set is chosen by selecting a column of an $r \times s$ rectangular array (r is the number of rows and s is the number of columns) and $\lfloor s/2 \rfloor$ elements from other columns. Other QS methods are much more complex, for example, cyclic QS (a cyclic QS is based on the cyclic block design and cyclic DSs in combinatorial theory [6]) proposed in [7]. The QS concept itself has already been used [8] in the context of operating systems. In the last decade, the use of QSs has been extended to other applications, for example, power-saving protocols (e.g., [9, 10]). Quorum systems are also adopted in order to overcome a rendezvous problem in multichannel networks (e.g., [11–14]). For additional review in the context of cognitive radios we refer the reader to [4].

The content of this work is as follows. For sake of understanding the analyzed algorithm, we give a short introduction to quorum systems focusing on a torus QS and Difference Sets. We describe the standard method of constructing a torus quorum [5] (forward manner) and the mirror method [15]. Afterwards, we shortly describe the construction rules of the MtQS-DSrdv protocol designed in our previous work [4], followed by a throughout performance evaluation. In a recent survey paper [4] we also discussed a possibility to use MtQS-DSrdv for RDV in the case of cognitive radio networks. However, that discussion was more of illustration of a possibility and contained only preliminary results. In this paper we will provide more in-depth analysis and consider also asymmetric channels. We provide additional analysis in terms of time-to-rendezvous (TTR) and the mean occurrence of rendezvous on a channel within a hopping sequence/period. In contrast to [4], we also consider the *asymmetric* channel view in terms of (maximal) (M)TTR, RDV on x channels, and RDV occurrence on a channel within a hopping sequence/period. Furthermore, as opposed to [4], we show the advantage of the MtQS-DSrdv protocol against the related work both in SCHv and/or ACHv in terms of (M)TTR metric and the probability of RDVs on different channels in one hopping sequence period.

Section 2 presents the related work in this area. In Section 3 we describe the QS and DS concepts with all relevant definitions and properties. The system model is presented in Section 4. We prompt the MtQS-DSrdv protocol in Section 5. The extensive evaluation of the protocol in the symmetric and asymmetric channel view is described in Section 6, with the comparison with the selected related work in the last subsection. Section 7 concludes this study.

2. Related Work

A strict coordination or some degree of synchronization between nodes is often assumed in dynamic spectrum access literature. Either a TDMA or FH-like access schemes are used with the assumption that nodes can synchronize or coordinate easily; see, for example, [16–19]. In [20] taxonomy, challenges, classification, and comparison of rendezvous approaches in CRNs can be found.

Here we summarize the most recent related work by categorizing methods into three branches. The first category comprises nonquorum-based solutions representing blind or pseudorandom RDV techniques [21–26]. The second branch contains either QS-based protocols proposed for a multichannel Medium Access Control (MAC) [11, 27, 28] or non-QS but sequence-based algorithms for CRNs [29–31]. Finally in the third branch we have a number of quorum system-based protocols proposed for CRNs [12–14, 32–34].

A-MOCH [30], the non-QS-based protocol for CRNs, is based on Latin Square (LS) (transmitter) and Identical-Row Square (IRS) maps (receiver). This approach guarantees RDV on all channels in a period, but only once on each channel. Moreover, while being in the reception mode receivers need to switch channels constantly, which is definitely not desirable for wireless resource-constrained systems. It might also happen (cf. [30]) that a transmitter will select such LS, which also implies switching channels constantly. In other words, in this approach the cost of channel switching time for a receiver should be taken into account.

In the asynchronous ASYNC-ETCH algorithm, presented in [31], there is no need of global clock synchronization, no matter how the hopping processes of nodes are misaligned. A hopping sequence S_i is composed of N frames (N denotes the number of available channels), where each frame is composed of a pilot slot and two subsequences sub S_i. The pilot slots of S_i, collected together, are the channels appearing in sub S_i in the same order. The hopping sequence is equal to $(2N + 1) \times N$; for example, with 5 channels, it is composed of 55 slots. If the pair of CRs selects the same hopping sequence, RDV is guaranteed in one slot per hopping period. However, while using different sequences, there is RDV guaranteed in N slots.

In order to have a RDV channel without the help of CCC or synchronization, and to guarantee rendezvous in at least one channel for each searching sequence, the Balanced Incomplete Block Design (BIBD) [35] has been used in [29]. Authors introduce single-sequence and multisequence MAC protocols, where the latter builds hopping sequences for the multichannel case. The protocol is compared to the permutation sequence proposed in [36] and blind rendezvous in [37].

In DSMMAC [13] all nodes create the same channel hopping sequence, and the only possible variation of this sequence is dependent on an offset (if cycles are not aligned). Moreover, the process of the forming of channel hopping sequences is not easy; namely, it is based on the Difference Sets which must be chosen in a very careful manner in order to ensure a high RDV probability.

Since a torus QS is a special case of a grid QS, we also consider two grid quorum-based rendezvous algorithms [14, 34]. The schemes do not guarantee RDV, although the percentage of missed RDV is very low.

In [4] we gave a comprehensive guidance on the use of quorum systems. We also addressed RDV issues in decentralized CRNs surveying exhaustively channel hopping approaches. Additionally, channel hopping requirements for cognitive radio ad hoc networks have been proposed, and the most suitable related work to the RDV problem has been appraised according to those requirements. The MtQS-DSrdv protocol rules were proposed in [4] along with its assessment according to the proposed requirements.

In [33] a sequence-based protocol has been implemented on Universal Software Radio Peripheral (USRP) boards and

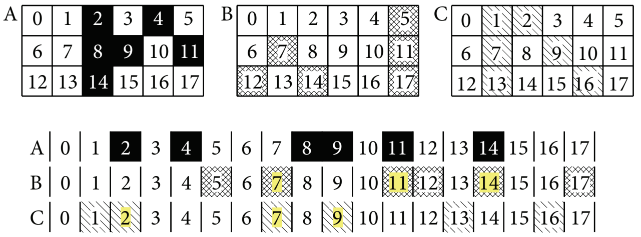


Figure 2: Standard torus QS example: node A is built by picking the third column as its head and 3 randomly chosen slots from succeeding columns. B is formed by selecting the sixth column and its tail from succeeding columns in a wrap-around manner. C's head is the second column. A and B intersect at slots 11 and 14, B and C at slot 7, and A and C at slots 2 and 9.

evaluated in terms of the time of the first encounter between two SUs and the time for encounter on all channels. The protocol itself is partly based on QS [38, 39]. The experiments show a big advantage of the sequence-based scheme over the random sequence scheme, with and without PU's presence.

In contrary to gQ-RDV method [34] or BIBD-based approach [29], the MtQS-DSrdv protocol guarantees rendezvous on all r available channels in a symmetric channel view. In comparison to A-MOCH [30] our scheme does not use two different sequences for receiver and transmitter, nodes switch channels less frequently, and nodes can meet more than once on the same channel during one period.

3. Torus Quorum System and Difference Set Properties

In this section we present the (torus) QS and DS concepts. Since QSs are not commonly used in wireless communications, we present first some relevant definitions in this section.

A torus-based QS (tQS) [5] adopts a rectangular array structure called *torus*, that is, wrap-around mesh, where the last row (column) is followed by the first row (column) in a wrap-around manner (later in this work, we call the standard tQS as the *forward* tQS, since its tail is selected going forward (to the next columns)). The height r (number of rows, i.e., entire column) and width s (number of columns, i.e., entire row) are defined where $n = r \times s$ and $s \geq r \geq 1$.

Definition 1 (Torus Quorum Systems). A torus quorum in a $r \times s$ torus is composed of $r + \lfloor s/2 \rfloor$ elements, formed by selecting any column C_j ($j = 1 \cdots s$) of r elements plus one element out of each of the $\lfloor s/2 \rfloor$ succeeding columns using end wrap-around. An entire column C_j portion is called the quorum's *head*, and the rest of the elements ($\lfloor s/2 \rfloor$) its *tail*.

Figure 2 gives an example for three nodes choosing three different torus quorums under $U = \{0, 1, \ldots, 17\}$ where $r = 3$ and $s = 6$, thus $n = 18$.

It is also possible to construct a tQ in a backward manner as shown in [40]. However, in order to select a torus tail in a more flexible manner, the *mirror* torus extension should be used [4, 15], which allows to alternate selecting tail's slot in a forward or backward manner.

Definition 2 (Mirror Torus Extension). A tail of a torus quorum, $\lfloor s/2 \rfloor$ elements, can be selected from any position of column C_{j+k_i*i} (one element from a column), where $k_i \in \{1, -1\}$ and $i = 1 \cdots \lfloor s/2 \rfloor$, in a wrap-around manner. Toruses of the same torus QS need to select elements in the same forward/backward order.

In other words, if an element was selected from column C_{j+1}, the next element cannot be selected from C_{j-1}, but needs to originate from the next succeeding (forward) column (C_{j+2}) or the preceding (backward) column (C_{j-2}). The parameter k_i needs to be the same for all quorums of the same torus QS; that is, the direction of the selection needs to be the same. Figure 3 shows the selection in a mirror way.

The intersection property of quorum systems is not sufficient when the cycle of nodes is not aligned or nodes are asynchronous. In order to have RDV guarantee in such case, a quorum must satisfy the Rotation Closure Property (RCP).

Definition 3 (Rotation Closure Property). For a quorum R in a quorum system Q under an universal set $U = \{0, \ldots, n-1\}$ and $i \in \{1, 2, \ldots, n-1\}$, one defines: rotate$(R, i) = (x + i) \bmod n | x \in R$. A quorum system Q has the Rotation Closure Property if and only if

$$\forall R', \quad R \in Q, R' \cap \operatorname{rotate}(R, i) \neq \emptyset \quad \forall i \in 1, 2, \ldots, n-1. \tag{1}$$

A quorum system, which satisfies the Rotation Closure Property, ensures that two asynchronous mobile nodes selecting any two quorums have at least one intersection in their quorums. The (forward, backward, and mirror) torus quorum satisfies the Rotation Closure Property.

The Difference Sets [6] concept is very close to QSs, being actually the basis of the cyclic QS.

Definition 4 (Cyclic Difference Set (DS)). A subset B, such as $B = \{a_1, a_2, \ldots, a_k\}$modulo n, for $a_i \in 1, 2, \ldots, n-1$, is called a cyclic (n, k, λ) difference set under Z_n (k and λ are positive integers such that $2 \leq k < n$ and $|B| = k$), if for every $b \not\equiv 0 (\bmod n)$ there are exactly λ ordered pairs (a_i, a_j), where a_i, $a_j \in B$ in such a way that $a_i - a_j \equiv b (\bmod n)$.

If at least one ordered pair (a_i, a_j) exists in (n, k) difference set, such set is called a relaxed DS.

Since tQs also form DSs we show an example based on the set from Figure 2. Node A has a set $\{2, 4, 8, 9, 11, 14\}$ where $n = 18$. The set is a relaxed DS, because there is at least one ordered pair (a_i, a_j):

$$\begin{gathered} 1 \equiv 9-8, \quad 2 \equiv 4-2, \quad 3 \equiv 11-8, \quad 4 \equiv 8-4, \\ 5 \equiv 9-4, \quad 6 \equiv 8-2, \quad 7 \equiv 9-2, \quad 8 \equiv 4-14, \\ 9 \equiv 11-2, \quad 10 \equiv 14-4, \quad 11 \equiv 2-9, \\ 12 \equiv 2-8, \quad 13 \equiv 9-14, \quad 14 \equiv 4-8, \\ 15 \equiv 11-14, \quad 16 \equiv 2-4, \quad 17 \equiv 8-9. \end{gathered} \quad (\bmod\ 18)$$

The reader should note that each cyclic DS satisfies the RCP.

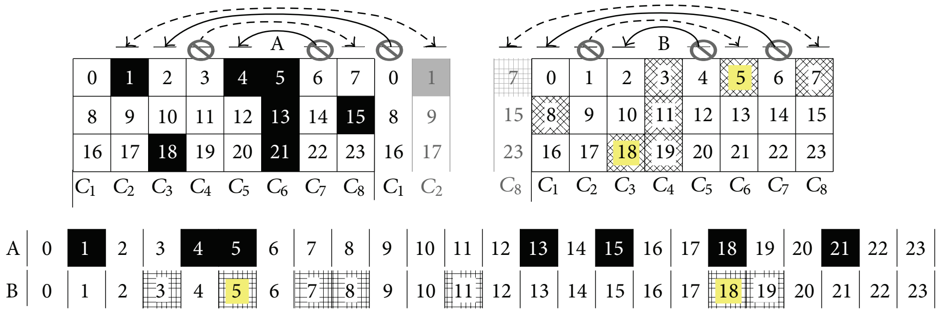

A
B
C1 C2 C3 C4 C5 C6 C7 C8 C1 C2
C8 C1 C2 C3 C4 C5 C6 C7 C8
A 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23
B 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23

Figure 3: Example of mirror torus QS: A and B initiate their tail selection in a backward manner; that is, the first element of node A is selected from column C_5 (a mirror of C_7), and the one of node B is selected from column C_3 (a mirror of C_5). Then, both nodes select the next tail element in a forward manner, thus from C_{j+2} column, which is C_8 in case of node A, and C_6 in case of node B. The third tail element is selected in a backward manner, which is C_3 in case of node A, and C_1 in case of node B. The last element of nodes A and B falls in the same (mirror) column. Nodes meet at slot 5 (they would meet in one element of column C_6 anyway) and additionally in slot 18. Hence, nodes can always alternate the manner, either forward or backward, while picking up elements; however, they need to have the same k_i, in this example $k_i = \{-1, 1, -1, \{1 \mid -1\}\}$, where $i = 1 \cdots \lfloor 8/2 \rfloor$.

4. System Model

We focus on Secondary Users in the presence of Primary Users in the network, where no central units for management of spectrum allocation are present. The rendezvous on the same channel (hereafter terms channel and frequency will be used interchangeably) between two SUs is crucial for the establishment of the communication. Each SU is equipped with a single tunable half-duplex radio transceiver which can switch between r different channels. In cognitive radio networks SUs must somehow identify spectrum holes, which vary in time and space, and after that select available frequencies. Based on a spectrum detection method (sensing, database) each CR recognizes a list of spectrum holes that can be used while respecting PU priority. It is assumed that channels are slowly time-varying, and that the system is slowly dynamic.

SUs should find each other periodically and as soon as possible (i.e., Time-to-Rendezvous should be small and bounded (TTR); is an amount of time, measured in slots, within which two or more CRs meet each other once they began hopping, or after the last RDV on a channel). Here, we emphasize that the need of rendezvous on *multiple* available channels in a hopping sequence period is paramount, as thanks to the guarantee of this property, the channel access delay will be minimized. If an RDV protocol cannot satisfy a periodic overlap between channel hopping sequences of cognitive radios on different channels, this raises an RDV problem, because a *single* rendezvous channel might become unavailable due to the (sudden) appearance of primary user signals. If an RDV protocol can guarantee a rendezvous on *every* available channel, it provides the maximum robustness. Therefore, we aim to guarantee rendezvous on every available channel in the case of the homogeneous spectrum availability (i.e., all SUs have the same channel set), and on multiple different available common channels in the case of the heterogeneous spectrum availability. The latter objective is challenging, not only due to the different channels sets of CRs, but also because of different lengths of channel hopping sequences.

In our previous work [4] we developed MtQS-DSrdv algorithm and showed that in the case of a symmetric channel view (homogeneous spectrum availability), CR nodes meet *periodically* on every available channel. As explained before, a periodic overlap is guaranteed thanks to the use of QS properties, and therefore there is no need of exchange of any information in order to meet in a hopping sequence period. In this work we extend the analysis of SCHv in terms of TTR and we evaluate the protocol for an asymmetric channel view (heterogeneous spectrum availability).

In the MtQS-DSrdv algorithm we consider that each SU hops from one channel to another (Figure 1) according to its frequency hopping sequence in order to discover another SU. Each CR determines its channel map for each of the channels making use of a torus array ($r \times s$) using torus QS and DS concepts and then combines them into a hopping sequence. This process requires no mutual knowledge of hopping sequence information and available channels from other CRs. The resulting hopping sequence is cyclic and it counts as many slots as there are elements in the torus array. Every element represents a time slot where a single channel is designated to be used. Here, we stress that a slot is an amount of time within which cognitive radio users can communicate, that is, discover each other by exchanging messages.

5. MtQS-DSrdv Epitome

In this section we describe MtQS-DSrdv algorithm. While forming the channel map (hopping sequence), two concepts are employed, namely, torus Quorum System and Difference Sets. The former is straightforward, since we just select a torus quorum algorithm while the head (column) should follow the construction rules of the algorithm, and tail can be chosen randomly. The reader should note that with $r > 4$ the first four channels have torus Quorum System-based maps, and the rest have Difference Set-based maps.

Figure 4 depicts how each node constructs its hopping sequence with a cycle of n slots. The column selection of Channel 1 (step 2.a.1) specifies the start point of a map

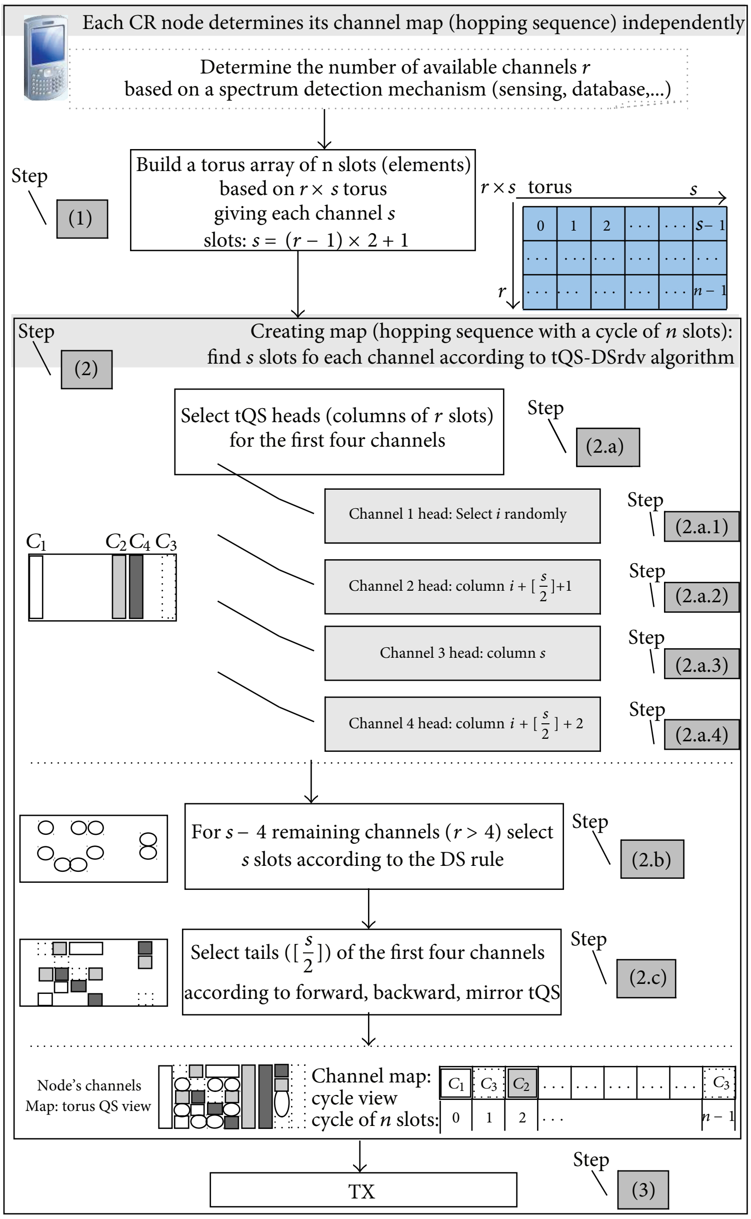

Each CR node determines its channel map (hopping sequence) independently
Determine the number of available channels r
based on a spectrum detection mechanism (sensing, database,...)
Step
(1)
Build a torus array of n slots (elements)
based on r × s torus
giving each channel s
slots: s = (r − 1) × 2 + 1
r × s torus
s
r
0
1
2
s− 1
n − 1
Step
(2)
Creating map (hopping sequence with a cycle of n slots):
find s slots fo each channel according to tQS-DSrdv algorithm
Select tQS heads (columns of r slots)
for the first four channels
Step
(2.a)
C1
C2 C4 C3
Channel 1 head: Select i randomly
Step
(2.a.1)
Channel 2 head: column i + [s/2]+1
Step
(2.a.2)
Channel 3 head: column s
Step
(2.a.3)
Channel 4 head: column i + [s/2] + 2
Step
(2.a.4)
For s − 4 remaining channels (r > 4) select
s slots according to the DS rule
Step
(2.b)
Select tails ([s/2]) of the first four channels
according to forward, backward, mirror tQS
Step
(2.c)
Node's channels
Map: torus QS view
Channel map:
cycle view
cycle of n slots:
C1
C3
C2
C3
0
1
2
n − 1
TX
Step
(3)

Figure 4: MtQS-DSrdv steps of each node.

construction. A node can select its head in s different ways because of s existing columns. In Figure 5 the head of Channel 1 is selected randomly to be C_1 (the first column). The remaining $s-4$ channels (step 2.b) search their map according to the DS rule (selecting a channel map by following the *DS rule* means that a relaxed DS set must be found of s slots; i.e., s remaining available elements are chosen so that at least one ordered pair (a_i, a_j) exists in (n, s) difference set, where $k = s$ from Definition 4) such that there are enough elements for tails of the first four channels.

For a node with eight available channels Figure 5 depicts two exemplary maps ($\text{Map}^{8\text{Chs}}(1)$ and $\text{Map}^{8\text{Chs}}(2)$) in order to show how to form another tQS-based map by just replacing the tail elements of the first four channels.

In addition, the reader should note that the selection of the channels is not strict; that is, we can replace tQS-based channels with DS based. Figure 6 illustrates a column selection of Channel 1 with different i (step 2.a.1 from Figure 4).

Thanks to the flexible use of tQS and DS concepts, the MtQS-DSrdv protocol allows automatically cognitive radio nodes to meet on each channel at least once in the symmetric channel view.

6. Verification

In the symmetric channel view we assess MtQS-DSrdv as regards the TTR performance. In the asymmetric channel

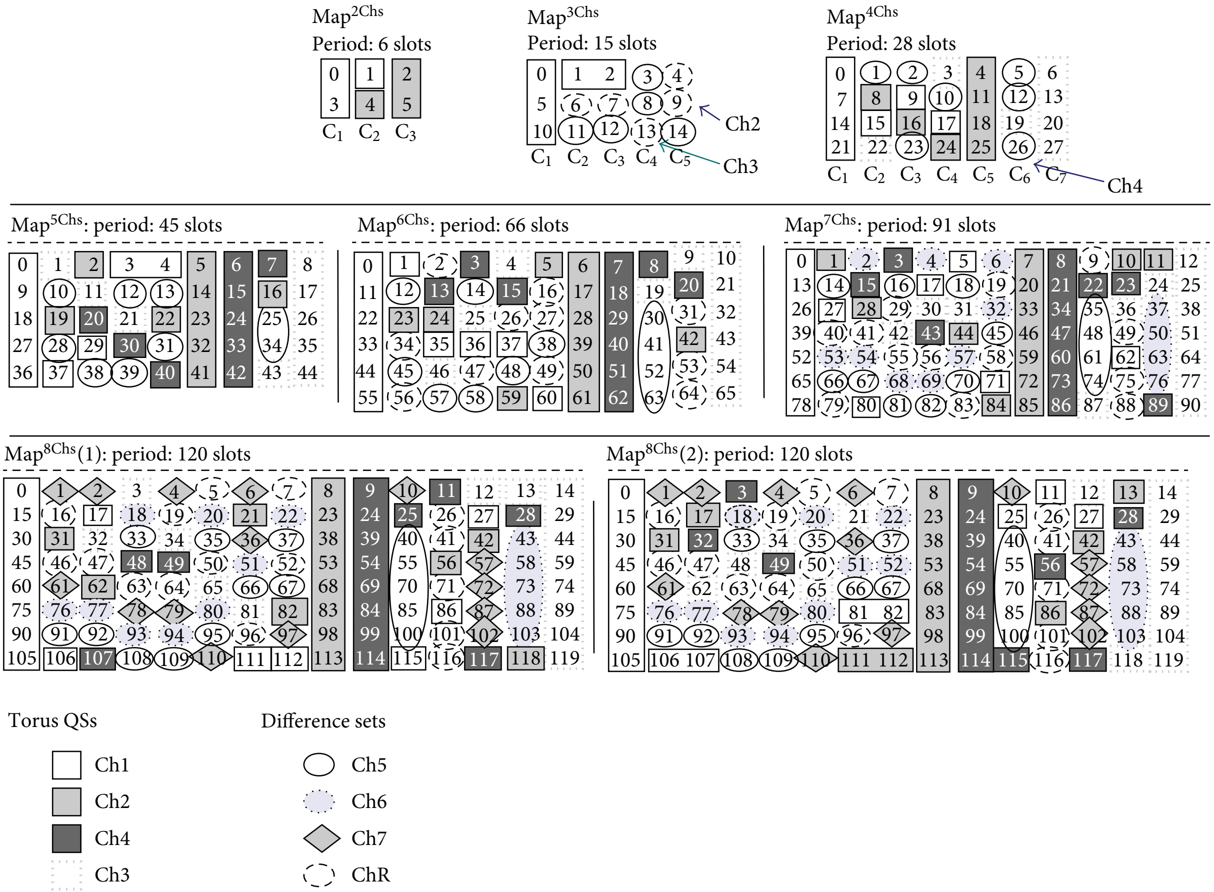


Figure 5: MtQS-DSrdv construction exemplary maps (Map$^{r\text{Chs}}$) for $r = 2 \cdots 8$ channels (Chs); Chr_i for $i = 1 \cdots r$ stands for a particular map of channel i. With three available channels, Channel 1 is selected according to the tQ forward manner, and the next two channels are formed from DSs (Map$^{3\text{Chs}}$), that is, DS_{Ch2}: $\{4, 6, 7, 9, 13\}$ and DS_{Ch3}: $\{3, 8, 11, 12, 14\}$. These DSs are exemplary, since others could also be found. While having four available channels, Channel 4 is selected according to the DS rule. Starting with four available channels (Map$^{4\text{Chs}}$) the mirror tQ is utilized, changing twice the direction (k_i). With five available channels (Map$^{5\text{Chs}}$), Channel 2 is built using a backward tQ except of one mirror element.

view we evaluate the protocol in respect to the RDV guarantee and occurrence on each channel and TTR performance. The reader should note that we also compare the performance of the pair of nodes using maps with the same number of channels against the pair of nodes using maps with a different number of available channels. Hence, we analyze whether it is better to use a smaller map with only available channels or a larger map also with unavailable channels. The latter case refers to the case where unavailable channels are also visited, but only for spectrum sensing (this is a frequently used approach while dealing with ACHv in the related work [13, 26, 30]).

We define the Rotation Closure Property for a complete channel map (*map-RCP*) as follows.

Definition 5 (Map-Rotation Closure Property). For map $R1$ and $R2$ of period (cycle) $\Theta = \{0, \ldots, n-1\}$ and for all $i \in \Theta$, one defines for all slot offset, $\exists i : R1_i \cap R2_{(i+\text{slotOffset}) \bmod N} \neq \emptyset$, where slot offset $\in \Theta$.

The frequency map of a node must be checked with each possible cycle shift (slot offset $\neq 0$). With a slot offset 0 (cycle alignment case) nodes will always meet on all channels at least once.

Proof. A hopping pattern for each channel is constructed according to Definitions 1, 2, and 4; that is, each channel set is a torus quorum or cyclic difference set as shown in Figure 5. A torus quorum and cyclic DS satisfy the Rotation Closure Property from Definition 3. Therefore, the set of all channels, map $R1$, and $R2$ with the same period Θ, composed of elements from a $r \times s$ torus, so that $rs = n$, satisfies the map-RCP from Definition 5. We do not need to check all r channels because the map-RCP definition is automatically satisfied due to the fact that each single channel map satisfies the RCP or DS.

In all considered cases we show statistical results for the complete set (equal or different channels) of results obtained by two maps. We analyze the ACHv case with maximum

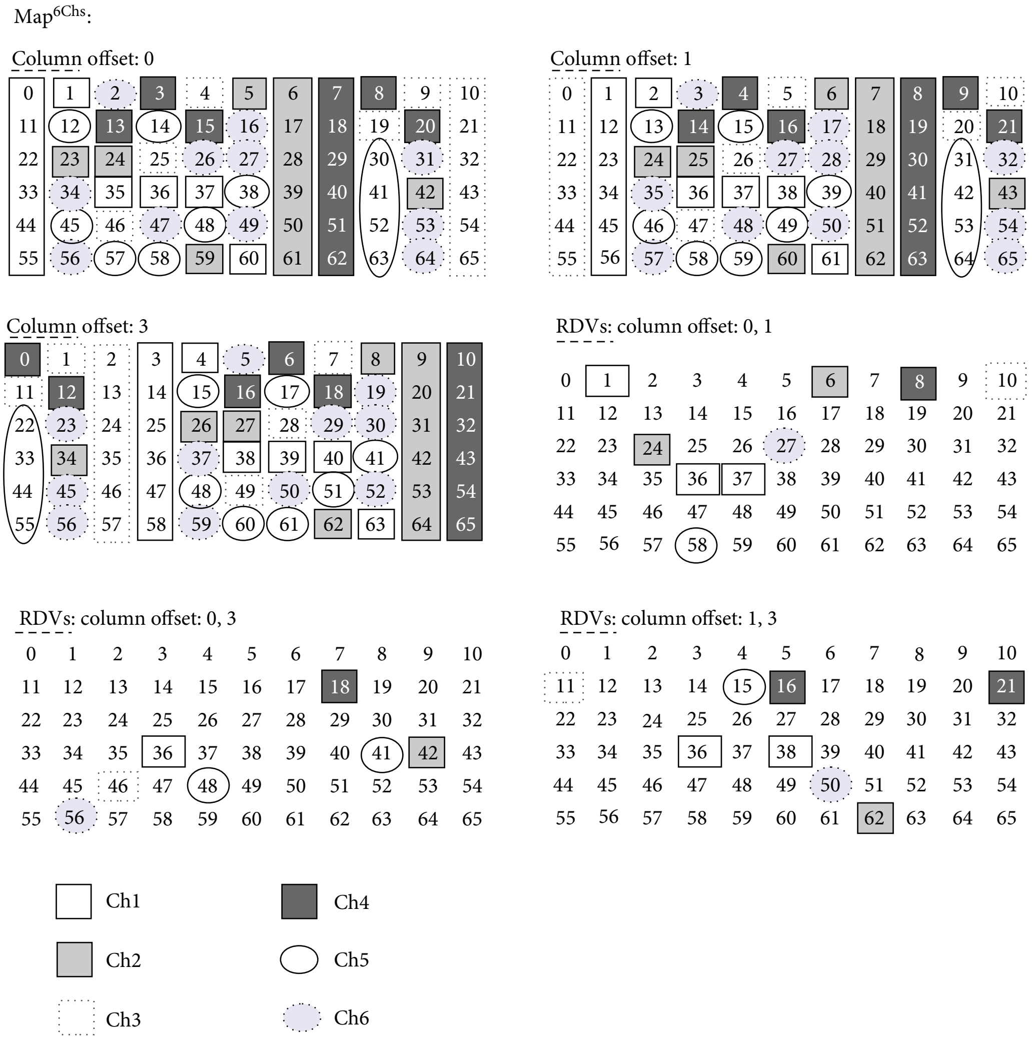

Map^6Chs:
Column offset: 0
Column offset: 1
Column offset: 3
RDVs: column offset: 0, 1
RDVs: column offset: 0, 3
RDVs: column offset: 1, 3
Ch1
Ch2
Ch3
Ch4
Ch5
Ch6

Figure 6: MtQS-DSrdv: different column offsets, CRs meet at least in r slots.

8 available channels, using channel maps from Figure 5. Moreover, while talking about the MtQS-DSrdv protocol, we refer to Figure 5, and Map$^{8\text{Chs}}(1)$ for a map with eight channels unless mentioned otherwise. □

6.1. MTTR in Symmetric Channel View. In this subsection we evaluate MtQS-DSrdv in SCHv in terms of maximum TTR. Table 1 shows TTR results in terms of the minimum of all TTR maxima (Υ), the maximum TTR (MTTR), the mean (μ) of MTTRs, the mean (μ) of TTRs, and the mean of occurrences on a channel ($\mu^{N\text{ch}}$) in a period.

In Table 1 we show the results for a map with 8 channels using both Map(1) ($8^{M(1)}$) and Map(2) ($8^{M(2)}$) from Figure 5. The reader should note that these are just two examples on how the tQS-based nodes can easily select their tail elements in a flexible manner.

In the case of SCHv it is clear that MTTR $< n$ ($n = |\Theta|$, where Θ is a period) for $r = 2-4$, and $\lesssim n/2$ for $r = 5-8$. One should note that we refer to MTTR per shift; that is, we take the maximum TTR in each shift, where μ^{MTTR} is an average of all MTTRs from all shifts while rotating the cycle. The average MTTR is considerably lower than MTTR. Note that the average TTR (μ^{TTR}) is significantly lower than MTTR and the average MTTR (μ^{MTTR}).

Table 1: MtQS-DSrdv in SCHv, so is *slot_offsets*; $(r, n) = \{(2,6), (3,15), (4,28), (5,45), (6,66), (7,91), (8,120)\}$.

r	Υ	MTTR	μ^{MTTR}	μ^{TTR}	$\mu^{N\text{ch}}$
2	1	4 (2 so)	3	1.5	1.2
3	4	9 (2 so)	5.7	2.5	1.4
4	3	18 (2 so)	10.3	3.5	1.6
5	4	21 (2 so)	12.6	4.5	1.6
6	5	35 (4 so)	20.6	5.5	1.7
7	6	47 (2 so)	23.6	6.5	1.7
$8^{M(1)}$	7	86 (2 so)	29	7.5	1.8
$8^{M(2)}$	7	57 (2 so)	30	7.5	1.8

In the worst case scenario while only one common channel is available and CRs can meet only once on the common channel, MTTR $= n$ (other less extreme cases

Table 2: MtQS-DSrdv in ACHv: RDV in a larger period of B; $_{\min}$RDV stands for the minimum number of RDVs (with the percentage).

A–B	$_{\min}$RDV	On all common CHs
2-3	1 (13%)	87%
2–4· · ·8	2	100%
3-4· · ·5	3	100%
3–6	2 (5%)	95%
3–7	3	100%
3–8	2 (7%)	93%
4-5	3 (16%)	84%
4–6	3 (5%)	95%
4–7· · ·8	4	100%
5-6	3 (5%)	79%
5–7	3 (2%)	88%
5–8	2 (1%); 3 (5%)	77%
6-7	4 (3%)	73%
6–8	4 (1%)	78%
7-8	5 (10%); 6 (33%)	58%

are discussed in Section 6.3). However, as soon as a CR becomes aware of the fact that some channels are not available anymore, it can adjust its hopping pattern.

6.2. Minimum RDV Occurrence in Asymmetric Channel View. In this section we evaluate the MtQS-RDV protocol in the heterogeneous spectrum availability case, where some of the available channels of one CR are unavailable to another. While analyzing the ACHv case one should consider whether it is better to use a map with a larger period or to use a map with available channels only. For instance, in the former case, a CR with 3 currently available channels can select a map with 4 channels instead of 3, and while visiting an unavailable channel only spectrum sensing is performed. MtQS-DSrdv with a symmetric channel set, using a map for 4 channels while having 3 available channels, still guarantees RDVs on all 3 available common channels.

While using MtQS-DSrdv maps for a different number of channels than available, and the number of available channels is not large, there is also RDV guarantee on all available common channels most of the time (in a larger period). For instance, we compare Map$^{3\text{Chs}}$ (map with 3 available channels from Figure 5) of one node with Map$^{4\text{Chs}}$ of another node, but in a larger period of 28 slots as it happens when both nodes use Map$^{4\text{Chs}}$. In this case, nodes also meet on all 3 available channels.

The following definition considers the case where node A, which has a smaller period Θ, still meets with node B, which has a larger period Φ.

Definition 6 (nM-Maps-Rotation Closure Property). For map $R1$ with period $\Theta = \{0, \dots, n-1\}$ and for all $i \in \Theta$, and map $R2$ with period $\Phi = \{0, \dots, m-1\}$ and for all $j \in \Phi$, and where $n < m$, there is defined for all Φ $\exists i, \exists j : R1_i \cap R2_j \neq \emptyset$.

Table 2 shows the analysis of the combination of maps (with minimum $r = 2$ and maximum $r = 8$) while checking the number of RDVs in a larger period Φ. The results show that in many cases there is still RDV on all common channels of a node with fewer available channels. The probability to have an RDV on all shared channels is very high; for example, node with three available channels meets node with six available (3–6 case) at least on two channels in a period ($_{\min}$RDV = 2), and this can happen with only 5% probability, otherwise nodes meet on three common channels.

Let us next analyze an example where a node has four available channels in a set and another node has three available channels. In Figure 7 the first one (node $A1$) uses the map Map$^{4\text{Chs}}$ from Figure 5, and the second (node $A2$) uses Map$^{3\text{Chs}}$. While both nodes are aligned at slot 0, they meet on each channel in a smaller period on Channels 2 and 3 twice. We also show RDVs in a larger period; that is, apart from R1 rendezvous (in Φ we have $R1+R2$ RDVs) nodes meet in R2 slots (15, 18-19, and 22) on each channel. While shifting the alignment of a smaller period (Θ) map, we show RDVs only in a smaller period. The minimal number of channels on which a RDV occurs in a smaller period is one. Nodes usually meet on all three channels in a smaller period. Note that while analyzing RDVs in a larger period Φ of 28 slots, nodes have RDVs on *all* available channels of A2 in all shifts.

For the sake of clarity, we show in Figures 8 and 9 the RDV occurrence of Map$^{4\text{Chs}}$ in SCHv and a combination of maps with 3 and 4 channels. In both cases CR nodes meet on all three available channels. We can additionally notice that the combination of 3-4 channels maps is better than that using maps with 4 channels. Single RDV on a particular channel in a period is decreased significantly, since nodes have multiple RDVs in a period on all three channels. If we take a mean of RDV occurrence in a period for each channel, we obtain 1.8 with Map$^{4\text{Chs}}$ (SCHv) and 2.3 with the combination of maps (ACHv). In other words, it is better to use an asymmetric channel set in spectrum availability heterogeneity in this particular case.

In Figures 11 and 13 we show an example where nodes do not meet on each channel, namely, the combination of maps with 4 and 6 channels, and the combination of maps with 5 and 7 channels. In the former case, nodes do not have RDV on four channels ($_{\min}$RDV = 3) in 3 shifts (once on the first channel, and twice on fourth channel). However, as one can see in Figure 11, they meet very often, at least three times, on each channel in a period (a mean of RDV occurrence is circa 2.8). Contrary, while using a symmetric channel set map (Map$^{6\text{Chs}}$ in Figure 10) nodes meet on all channels, but most often once or twice in a period (a mean of RDV occurrence is circa 1.7). We also show an example while the pair of nodes uses Map$^{7\text{Chs}}$ (symmetric channel set) against the combination of maps with 5 and 7 channels (Figures 12 and 13). While using different maps (combination case), nodes have RDV guarantee on three channels ($_{\min}$RDV = 3), where RDV on three channels only happens in 2 shifts. RDV on four and five channels is the most frequent.

Although nodes do not meet on all channels, there is still a question whether this map should not be used, since nodes meet more often (mean of RDV occurrence is circa 2.6) than in the case of a symmetric channel set with 7 channels map (mean of RDV occurrence is circa 1.7).

Table 3: MtQS-DSrdv in ACHv: RDV in a larger period of B.

A–B	Υ	MTTR	μ^{MTTR}	σ^{MTTR}	μ^{Nch}
2-3	2	11 (3 so)	6.7	2.6	2.5
2–4	4	10 (5 so)	8.1	1.7	3.5
2–5	4	17 (7 so)	11	5	4.5
2–6	11	16	13	2	5.5
2–7	10	18	15.8	2.6	6.5
Map (1): 2–8	7	29	21.5	10.3	7.5
Map (2): 2–8	7	29	21	10	7.5
3-4	4	14 (4 so)	9.4	3.7	2.3
3–5	6	22 (3 so)	13	4.5	3
3–6	7	22 (4 so)	15	3	3.7
3–7	8	33 (6 so)	20.7	7	4.3
Map (1): 3–8	6	70 (8 so)	21.5	17	5
Map (2): 3–8	6	65 (8 so)	23	18	5
3^*–8^*	14	54 (8 so)	25	10	5
4-5	6	31 (2 so)	14	6	2.3
4–6	9	29 (2 so)	18	5.7	2.8
4–7	13	42 (4 so)	24.6	7	3.3
4–7^*	9	31 (3 so)	19	5	3.3
Map (1): 4–8	12	51 (4 so)	28.8	9	3.8
Map (2): 4–8	12	45 (4 so)	27.6	8	3.8
4–8^*	12	57 (4 so)	25	9	3.8
5-6	9	39 (1 so)	18	6.8	2.2
5-6^*	6	32 (3 so)	17	6	2.2
5–7	8	45 (2 so)	22.7	7	2.6
5–7^*	9	47 (2 so)	21	9	2.6
Map (1): 5–8	12	65 (3 so)	26.7	9.4	3
Map (2): 5–8	10	70 (3 so)	27	11	3
5–8^*	10	56 (3 so)	24	10	3
6-7	10	66 (1 so)	22.9	8.8	1.7
6^*-7^*	10	43 (2 so)	19	7	2.2
Map (1): 6–8	11	51 (2 so)	26	7.6	2.5
Map (2): 6–8	10	49 (2 so)	25.8	8	2.5
6^*–8^*	8	46 (2 so)	25	8	2.5
Map (1): 7-8	13	55 (2 so)	27.7	9.5	2.2
Map (2): 7-8	9	64 (1 so)	25	9	2.2
7^*-8^*	12	58 (1 so)	26	10	2.2

6.3. MTTR in Asymmetric Channel View. In Table 3 MTTR statistics are presented for the asymmetric channel view. One should note that we refer to MTTR per shift (slot offset, "so" in table); that is, we take the maximum TTR in each shift, where μ^{MTTR} is an average of all MTTRs from all shifts while rotating the cycle.

In ACHv $A - B$ denotes combinations of maps of node A with node B, where the number of channels of node A is smaller than the number of channels of node B, as defined before. In this table we also show the results for a map with 8 channels using both $\text{Map}^{8\text{Chs}}(1)$ and $\text{Map}^{8\text{Chs}}(2)$ from Figure 5. In addition, in some of the combinations we have shown two cases, with or without $(^*)$. The case without $(^*)$ stands for the maps as proposed in Figure 5; that is, the first channels are assigned to tQS maps, and the next to DS maps. In the case with $(^*)$, in the map where there are more DSs than one (with $r = 3$ and $r = 6 \ldots 8$), the first channels are assigned to DSs (following the order from Figure 5, thus, map of Ch1 becomes Ch5, then Ch2 $\leftarrow$ Ch6, Ch3 $\leftarrow$ Ch7, and Ch8 $\leftarrow$ Ch4), where the remaining channels are assigned to tQS-based maps, also following the order from Figure 5 (e.g., map Ch5 $\leftarrow$ Ch1). Note that the described results below are without $(^*)$, unless mentioned otherwise in the text.

For ACHv the situation is naturally different than that for SCHv (Table 1) due to different period sizes of the compared channel sets. The results below are shown from the perspective of larger period (Φ) in order to compare it with the SCHv case. Depending on the combination of the

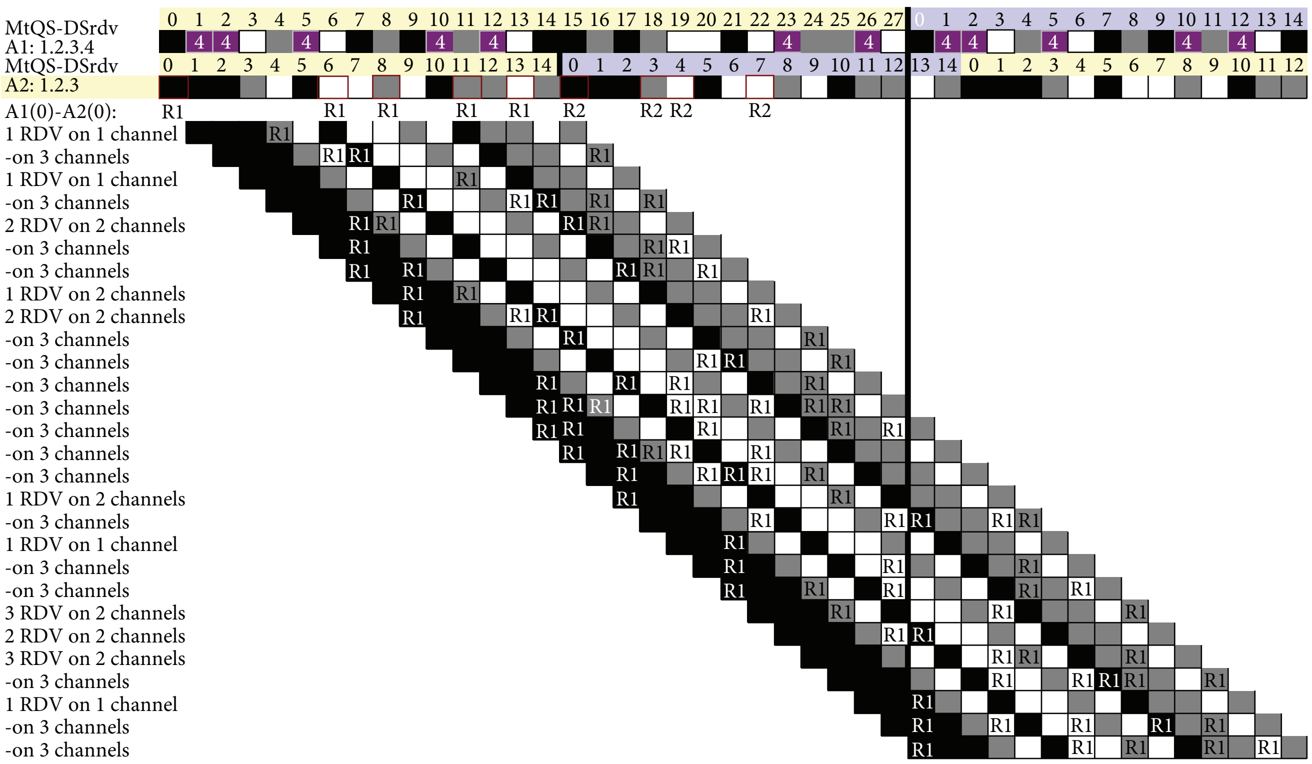


Figure 7: Asymmetric channel view: combination of maps with 3 and 4 channels, while rotating the cycle; RDV consideration in Θ; *black* square stands for channel 1, *gray* square for channel 2, and *white* square for channel 3; R stands for Rendezvous.

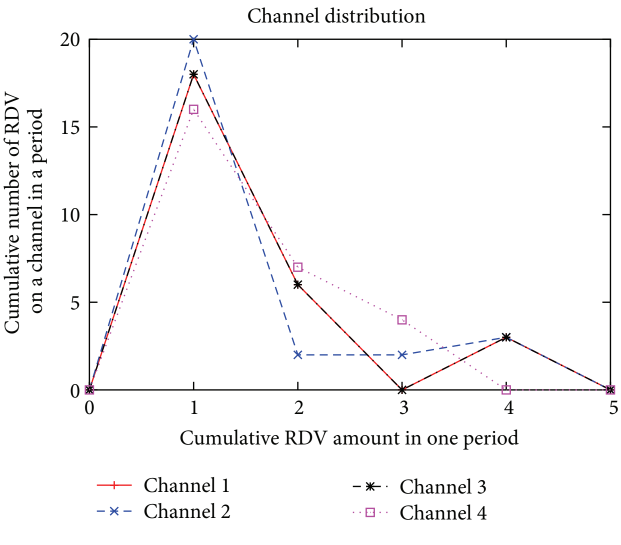


Figure 8: Symmetric channel set: $\text{Map}^{4\text{Chs}}$ (28 slots).

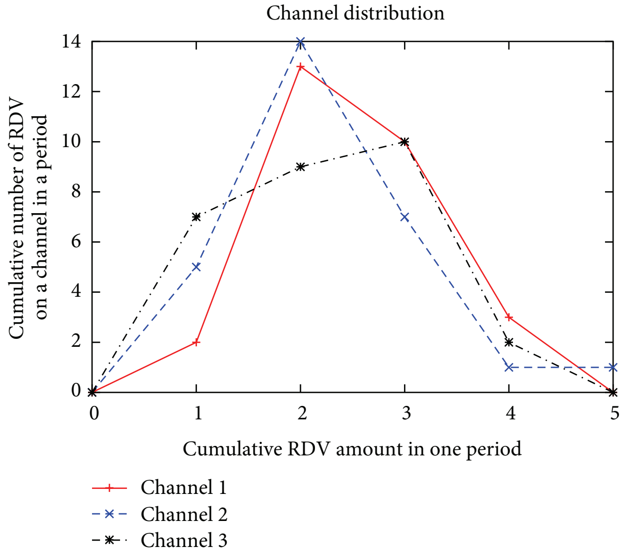


Figure 9: Asymmetric channel set: combination of maps with 3 and 4 channels.

channels of nodes A and B, the maximum TTR varies usually between $m/2$ and $(m/3)$ $(m = |\Phi|)$, where the average MTTR is lower than 30 slots in all cases, varying usually between $m/3$ and $m/4$. We also show some of the combinations where DS channel maps are reordered with tQS channel maps, the case with $(^*)$. One can see that assigning first DS maps and then tQS maps to channels usually decreases MTTR and/or MTTR mean. We observe that the selection of different tail elements (as exemplary shown with maps $\text{Map}^{8\text{Chs}}$) can improve MTTR or μ^{MTTR}.

Cognitive radio nodes that use a different map increase significantly RDV occurrences ($\mu^{N\text{ch}}$) on a channel in a period. The reader should note that in comparison with the performance in SCHv, RDV occurrence only once in a period

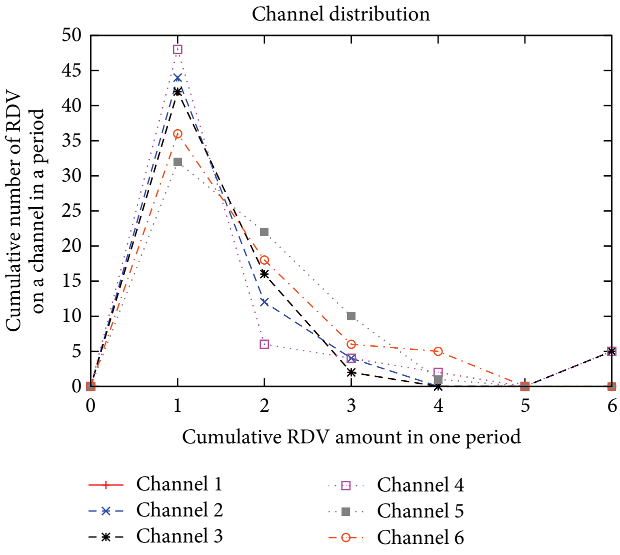


Figure 10: Common channel set: Map$^{6\text{Chs}}$ (66 slots).

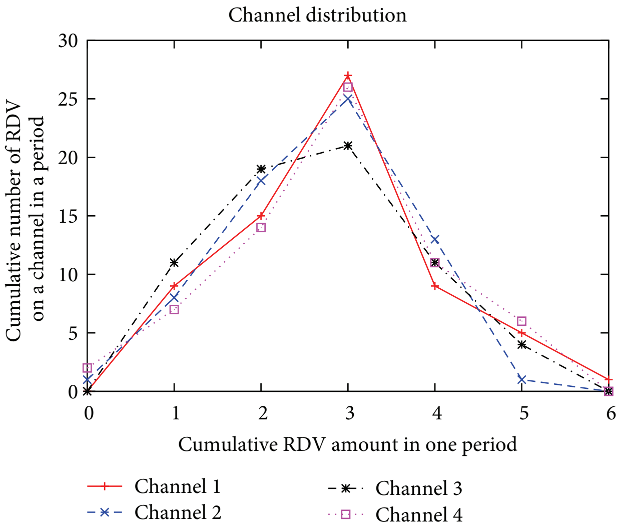


Figure 11: Asymmetric channel set: combination of maps with 4 and 6 channels.

is insignificant, since multiple RDVs are the dominant case. However, in order to answer the question which map one should use, the one imposing a symmetric channel view, or the one using an asymmetric channel set, the comparison of Tables 1 and 3 will not provide conclusive answer. Therefore, in Table 4 we show the MTTR statistics in respect to a smaller number of available common channels κ, although using maps with a larger number of channels.

Note that in this implicit way the related work papers handle an asymmetric view case [13, 26, 30]. In this case it is also easy to verify an upper bound of MTTR of the MtQS-DSrdv design (due to the RCP guarantee CRs always meet on every channel in a sequence period), which will be MTTR $\leq (m - \kappa)$ in the worst case scenario (the worst possible chosen maps). For example, with $r = 8$ and $\kappa = 2(\mathbf{8} \Rightarrow 2)$ such

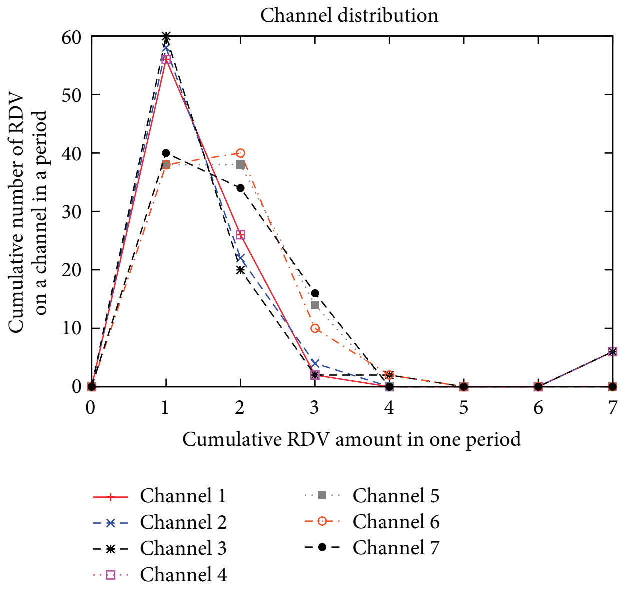


Figure 12: Common channel set: Map$^{7\text{Chs}}$ (91 slots).

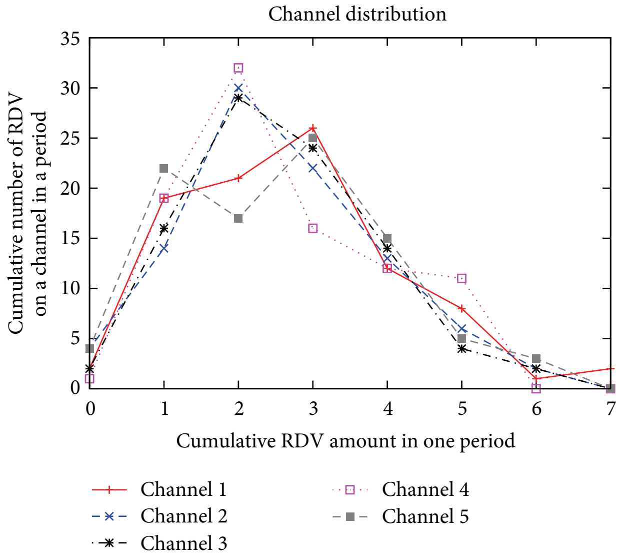


Figure 13: Asymmetric channel set: combination of maps with 5 and 7 channels.

upper bound is met in two cycle shifts (2 so in the table); since MTTR = 118 which is exactly $(m - \kappa)$, the same situation is in cases, $r = 3$ and $\kappa = 2$, $r = 6$ and $\kappa = 2$, and $r = 7$ and $\kappa = 2$. In other cases MTTR is lower or significantly lower.

Combining the results from Tables 3 and 4 it is clear that in terms of (M)TTR it is more advantageous to use a map with a number of locally available channels instead of using a larger map. MTTR of asymmetric channel set maps is always lower except for a few cases and even in those exceptional cases μ^{MTTR} of asymmetric channel set maps is better than that of symmetric channel set maps. Nevertheless, an investigation should be done combining the selection of a map including the duration of nonavailability of channels. It might be better

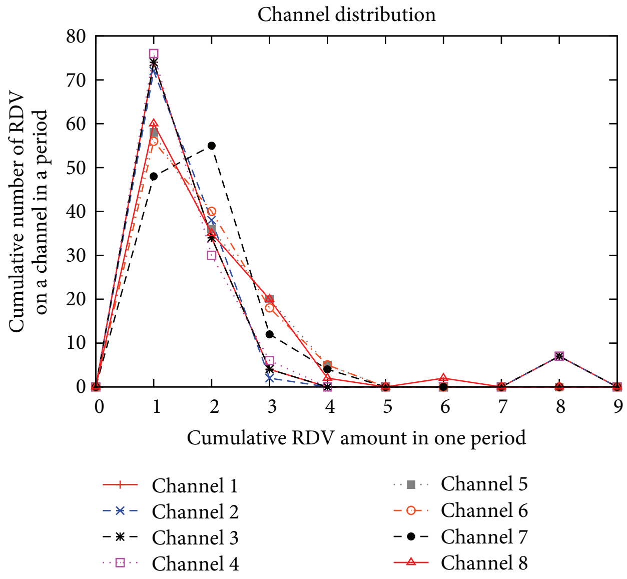


FIGURE 14: MtQS-DSrdv: 8 channels (120 slots).

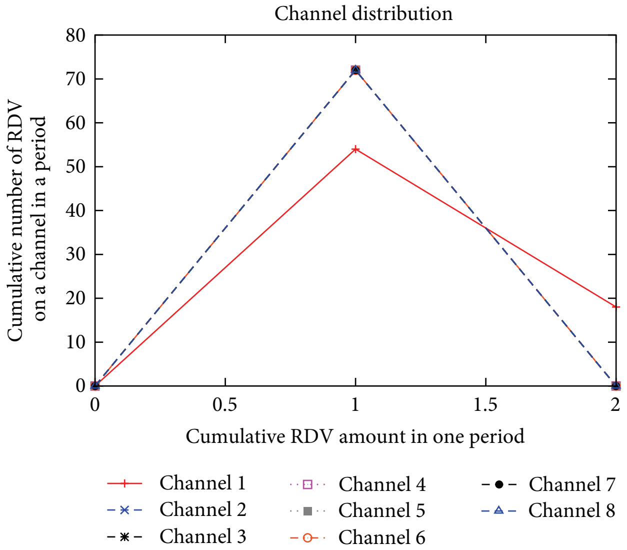


FIGURE 15: DSMMAC [13]: 8 channels (73 slots).

to use a larger map keeping in mind a probability of the channel release (predicting channel availability based on past observations) of the channel occupied by a PU recently. Hence, an RDV protocol should closely cooperate with a spectrum decision mechanism, which will be the topic of interest in our future work.

6.4. Comparison with the Related Work. In the subsection we compare the MtQS-DSrdv protocol with the related work where the designed approaches guarantee RDV either on all channels or in a very small TTR time. Hence, we consider the DSMMAC (*Difference-Set*-based asynchronous Multichannel MAC) [13], A-MOCH (Asynchronous Maximum Overlapping CH protocol) [30], ASYNC-ETCH (Asynchronous Efficient Channel Hopping) [31], and Balanced Incomplete Block Design- (BIBD-) based [29] algorithms in the synchronous channel view and consider also asynchronous channel view if applicable. In all considered cases we show results including the rotation of a whole cycle; that is, all possible cases without cycle alignment are checked.

6.4.1. Comparison with DSMMAC. In Figure 14 we depict results of MtQS-DSrdv with 8 channels according to the map presented in Figure 5. Figure 15 illustrates DSMMAC results with 8 channels. We remind the reader that the map from [13] is constructed entirely from Difference Sets. MTTR of MtQS-DSrdv equals 86 slots (using Map(1), but 57 slots with Map(2)), which happens in two shifts only. The average MTTR (μ^{MTTR}) is 28.8 with a standard deviation of 12. However, one should note that nodes meet often multiple times ($\mu^{N\text{ch}} = 1.8$) in a period as it can be seen in the figure. In this particular example the MTTR of DSMMAC is better than that of MtQS-DSrdv, since it equals 44 slots, with a mean of 23 slots and a standard deviation of 8.1. However, as can be clearly seen in Figure 15, on DSMMAC channels $2 \cdots 8$, nodes always meet only *once* in a period and *no more*. In DSMMAC, Channel 1 gets an extra slot, since all DSs have been chosen from a period of 73 slots, having 9 elements in a set. The remaining slot (slot 0) has been assigned to Channel 1, and therefore on Channel 1 nodes also have RDV twice in a period (Figure 15).

The fact that nodes meet only once in a period on $2 \cdots 8$ channels diminishes the chance of the protocol to be adaptable in ACHv, which was dismissed in this work. In [13] it is assumed that every node should use the same hopping sequence. Therefore, we also check the MTTR statistics while both CRs apply 8 channel maps, but they can meet only on 2 common channels because only these two are available for one of the CRs. The MTTR of DSMMAC will increase until 71 slots (which happens twice), whereas μ^{MTTR} is increased to 50.4 with $\sigma^{\text{MTTR}} = 10.4$. Those MTTR results are clearly much worse than using the 2–8 combination of maps of MtQS-DSrdv (Table 3).

We also analyze the possibility of the combination of maps with 2 and 8 channels for this approach. Figures 16 and 17 show the results for both protocols.

With both protocols nodes meet on each channel. The MTTR of DSMMAC is slightly lower (25 slots) than that of MtQS-DSrdv (29 slots, see Table 3). μ^{MTTR} is similar (20.5 of DSMMAC), but $\mu^{N\text{ch}}$ of DSMMAC is lower than that of MtQS-DSrdv (for both channels 7.5), since the average RDV occurrence on Channel 1 equals to 4.4 and on Channel 2 is 5.1, which can also be observed from the figures.

The DSMMAC map with eight available channels has seven neighboring pairs of slots, one of each channel ($2 \cdots 8$), except of Channel 1 having once three neighboring slots. Whereas, the MtQS-DSrdv map has more neighboring slots and can have even more thanks to the flexible design of the protocol. This characteristic is paramount with asynchronous nodes (without slot alignment), since nonoverlapping slots influence badly the chance for actual RDV

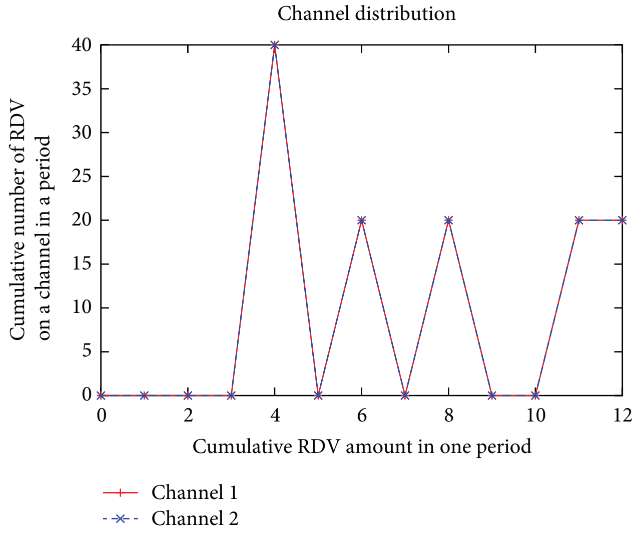


Figure 16: MtQS-DSrdv in ACHv: the combination of maps with 2 and 8 channels (120 slots).

on a channel. Hence, the MtQS-DSrdv map predominates DSMMAC in this context. However, further comparison of both approaches is difficult due to the unavailability of maps with other number of channels.

6.4.2. Comparison with A-MOCH. In Table 5 we show the results of the A-MOCH protocol [30] for $r = 2\cdots 8$ channels. We remind that in the A-MOCH algorithm a sender generates its hopping sequence based on the LS array and the receiver map is based on the IRS array. Since according to the algorithm nodes may randomly select a permutation of $\{0, 1, \ldots, N-1\}$, where N denotes the number of channels, we created all the maps in the same way, that is, rotating each row in a forward wrap-around manner to receive easily an LS square. For instance, with 5 channels the sender channel hopping sequence in the example below is $\{0, 1, 2, 3, 4,\ 4, 0, 1, 2, 3,\ 3, 4, 0, 1, 2,\ 2, 3, 4, 0, 1, 1, 2, 3, 4, 0\}$. The receiver with 5 channels adopts the following sequence: $\{0, 1, 2, 3, 4, 0, 1, 2, 3, 4, 0, 1, 2, 3, 4, 0, 1, 2, 3, 4, 0, 1, 2, 3, 4\}$. This map construction gave us the worst possible map cases, since the mean MTTR is always equal to the MTTR and is constant as shown in the table; that is, there are no differences between MTTR, Υ (minimum of MTTR maxima), and μ^{MTTR}. Thus, the meeting points will always have the same occurrence ($\mathrm{MTTR} = N^2 - N + 1$ as reported in [30]). In addition, nodes will always meet *once* on each channel in each period (μ^{Nch}). In comparison with MtQS-DSrdv, the MTTR of A-MOCH is slightly better, we say slightly, since MTTR of MtQS-DSrdv happens to be maximum in 4 cases (usually 2 cases), whereas for A-MOCH for every shift. Of course as a consequence, the mean (μ^{MTTR}) of A-MOCH is significantly worse than of MtQS-DSrdv.

As mentioned above, the table shows the worst cases of MTTR. If we select other sequences we can improve the MTTR, for example, with 5 available channels and the sender adopting the following sequence: $\{0, 1, 2, 3, 4,\ 1, 2, 0, 4, 3,\ 2, 3, 4, 0, 1,\ 3, 4, 1, 2, 0,\ 4, 0, 3, 1, 2\}$, and the receiver with the sequence:$\{4, 0, 1, 2, 3,\ 4, 0, 1, 2, 3,\ 4, 0, 1, 2, 3,\ 4, 0, 1, 2, 3,\ 4, 0, 1, 2, 3\}$, we can improve μ^{MTTR} to 15 with $\sigma^{\mathrm{MTTR}} = 5.4$ and $\Upsilon = 8$. The MTTR stays 21, but it occurs 10 times.

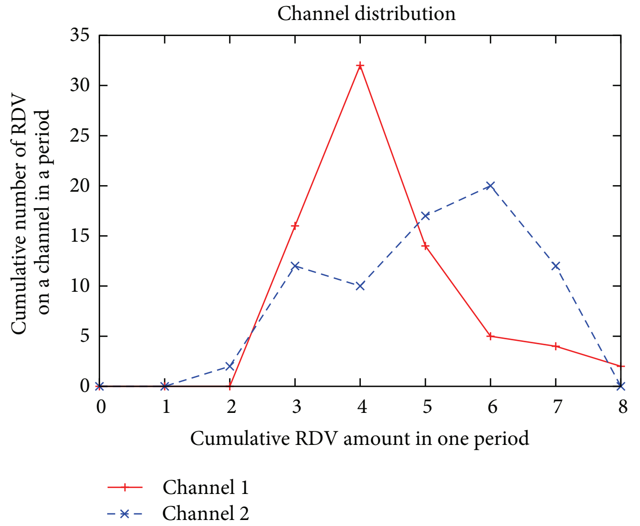


Figure 17: DSMMAC [13]: the combination of maps with 2 and 8 channels (73 slots).

The reader should note that μ^{Nch} can never be changed (independently of chosen sequence) and remains always *one* RDV on each channel in a period, which diminishes the RDV chances of asynchronous nodes (partial overlap of a single slot might not be sufficient to have a successful RDV).

Moreover, the protocol does not work explicitly with the heterogeneous spectrum availability, since if we take the sender map with, for example, 5 channels and the receiver with, for example, 8 channels, then there is no RDV guarantee, even in a larger period where there are *eight* shifts with no matches. Therefore, the only possible option is to use the map with 8 channels, that is, also hopping on unavailable channels. For instance, if the $\kappa = 5$, but $r = 8$, then the MTTR of A-MOCH equals 60 (occurring 32 times), with $\mu^{\mathrm{MTTR}} = 58.5$, which is automatically worse than MtQS-DSrdv with the combination of maps with 5 and 8 channels (5–8 in Table 3).

If we take $\kappa = 2$ and $r = 8$ for A-MOCH, then we obtain MTTR = 63 (56 times) with $\mu^{\mathrm{MTTR}} = 62.3$, which is definitely worse than the results of MtQS-DSrdv with the combination of maps with 2 and 8 channels (2–8 in Table 3).

Taking a better (aforementioned) map with 5 channels, and checking the least extreme case, namely, with $\kappa = 4$, then we get MTTR = 22 with a mean of 15.4. The MTTR is somewhat better than that of MtQS-DSrdv, but the mean is similar. However, with $\kappa = 3$, MTTR is 23 (5 times) and mean 19, which is already worse than the asymmetric channel set with 3 and 5 channels' combination of maps of MtQS-DSrdv (3–5 in Table 3).

In [30], the metric Maximum Conditional TTR (MCTTR) has been defined which refers to the maximum TTR between two hopping nodes when PU traffic is present and at least one channel is available for the two nodes.

Table 4: MtQS-DSrdv: maximum time-to-RDV performance of κ.

$r \Rightarrow \kappa$	Υ	MTTR	μ^{MTTR}	σ^{MTTR}	μ^{Nch}
3$\Rightarrow 2$	4	13 (2 so)	7.3	3.2	1.4
4 $\Rightarrow 2$	3	26 (2 so)	15.6	6.6	1.6
4 $\Rightarrow 3$	3	22 (2 so)	12.4	5.3	1.6
5 $\Rightarrow 2$	4	41 (2 so)	25.2	9.5	1.6
5 $\Rightarrow 3$	4	33 (2 so)	19	6.6	1.6
5 $\Rightarrow 4$	4	27 (4 so)	16.4	6.3	1.6
6 $\Rightarrow 2$	5	64 (2 so)	41	15.9	1.7
6 $\Rightarrow 3$	5	55 (8 so)	34	13.8	1.7
6 $\Rightarrow 4$	5	51 (2 so)	30	11.8	1.7
6 $\Rightarrow 5$	5	42 (4 so)	24.2	9.3	1.7
7 $\Rightarrow 2$	6	89 (2 so)	57.7	20.5	1.7
7 $\Rightarrow 3$	6	82 (2 so)	44.4	16.3	1.7
7 $\Rightarrow 4$	6	63 (2 so)	38.3	13.9	1.7
7 $\Rightarrow 5$	6	57 (2 so)	31	11.7	1.7
7 $\Rightarrow 6$	6	53 (2 so)	26.8	9.9	1.7
8 $\Rightarrow 2$	7	118 (2 so)	73	25.6	1.8
8 $\Rightarrow 3$	7	101 (2 so)	57	21.4	1.8
8 $\Rightarrow 4$	7	101 (2 so)	47.8	19.8	1.8
8 $\Rightarrow 5$	7	93 (2 so)	39.7	16.4	1.8
8 $\Rightarrow 6$	7	93 (2 so)	35.6	15.3	1.8
8 $\Rightarrow 7$	7	86 (2 so)	33	14.9	1.8

Table 5: A-MOCH [30] in SCHv $(r, n) = \{(2,4), (3,9), (4,16), (5,25), (6,36), (7,49), (8,64)\}$.

#	Υ	MTTR	μ^{MTTR}	σ^{MTTR}	μ^{Nch}
2	3	3 (4 times)	3	0	1
3	7	7 (9 times)	7	0	1
4	13	13 (16 times)	13	0	1
5	21	21 (25 times)	21	0	1
6	31	31 (36 times)	31	0	1
7	43	43 (43 times)	42	0	1
8	57	57 (57 times)	57	0	1

MCTTR of A-MOCH is at least N^2 (N denotes the number of channels), thus, with 5 channels MCTTR = 25, and with 8 channels MCTTR = 64. Coming back to our protocol, if we know that only one channel is incumbent-free for a long period of time (such information can be obtained for instance from Radio Environmental Maps), and we cannot visit other channels, then the MCTTR will equal s, for example, for 5 channels $s = 9$, for 8 channels $s = 15$.

6.4.3. Comparison with ASYNC-ETCH. In Figures 18 and 19 we show the cumulative channel distribution of A-ETCH [31] in a hopping period for all shifts with 5 available channels. We remind that in A-ETCH a hopping sequence S_i is composed of N frames (N denotes the number of available channels), where each frame is composed of a pilot slot and two subsequences sub S_i. The pilot slots of S_i, collected together, are the channels appearing in sub S_i in the same order, thus $|S_i| = (2N + 1) * N$; for example, with 5 channels a period is composed of 55 slots in total since $|S_i| = (2 * 5 + 1) * 5 = 55$. Figure 18 depicts the case when nodes select the same sequence, where we show the following sequence (S_0): $\{0,\ 0,1,2,3,4,0,1,2,3,4,\ 1,\ 0,1,2,3,4,0,1,2,3,4,\ 2,\ 0,1,2,3,4,0,1,2,3,4,\ 3,\ 0,1,2,3,4,0,1,2,3,4\ ,4,\ 0,1,2,3,4,0,1,2,3,4\}$. Figure 19 depicts the case when one CR selects (S_0) and another (S_1): $\{0,\ 0,2,4,1,3,0,2,4,1,3,2,\ 0,2,4,1,3,0,2,4,1,3,\ 4,\ 0,2,4,1,3,0,2,4,1,3,\ 1,\ 0,2,4,1,3,0,2,4,1,3,3,\ 0,2,4,1,3,0,2,4,1,3\}$.

The average TTR (μ^{TTR}) is reported to be $(2N^2+N)/(N-1) \approx 2N$. Hence, with $N = 5$ we have $\mu^{\mathrm{TTR}} \approx 11$, which is indeed the case but with two different maps (see Figure 19). If the pair of CRs selects the same hopping sequence, RDV is guaranteed in one slot in a hopping period. Using different sequences RDV is guaranteed in N slots.

The reader should note that this guarantee does not happen on all channels, which is clearly noticeable in the figures. With *different* maps, there are shifts where RDV happens on one or two channels. As a consequence, it might happen that there is indeed RDV guarantee on even 11 or 12 slots of 55 slots, but only on 1, 2, or 3 channels, which is highly undesirable, especially in the case when these channels are of bad quality. Moreover, RDV on particular channels is more random, since it depends on the shift and sequence. To be more precise, there are 40 shifts where nodes meet on *three* channels, but they will *never* meet on *four* or *five* channels in one hopping period. What Figure 19 also shows is that the RDV frequency on a channel is alternating from very low (once or twice) to moderate (5 or 6 times) or to very high (10 times) with this scheme.

In the case that the nodes select the *same* hopping sequence (Figure 18) MTTR is 54 slots with the mean value of

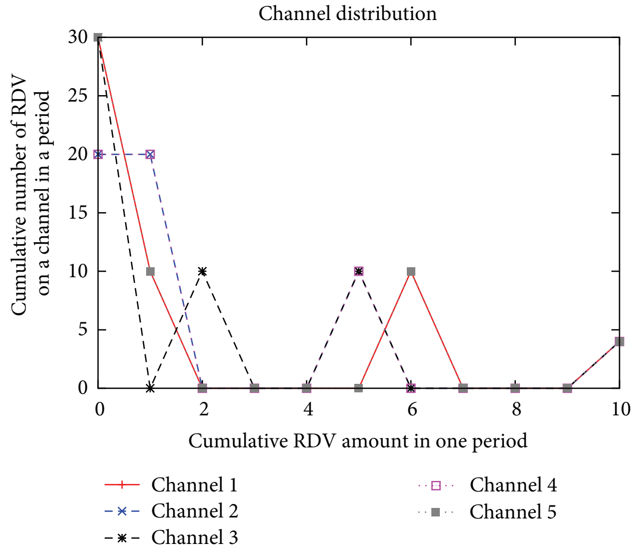


FIGURE 18: A-ETCH with the same sequences (5 channels).

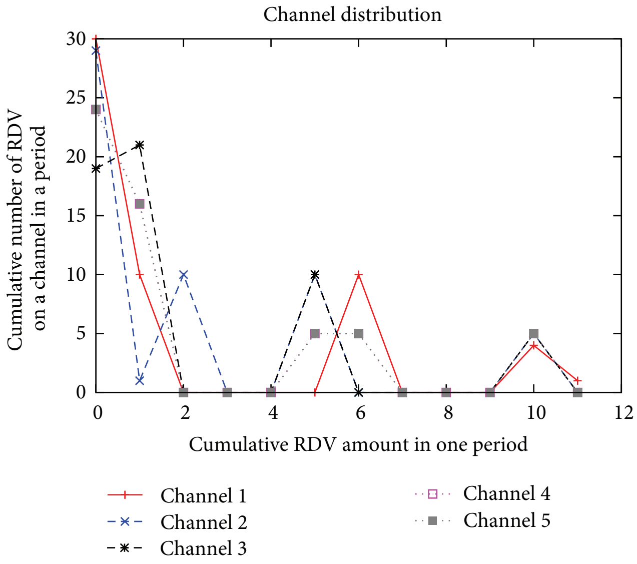


FIGURE 19: A-ETCH with the different sequences (5 channels).

32 slots. An interesting observation is that in this case there are 14 shifts in which nodes meet on all *five* channels in one hopping period, but on the remaining shifts (40) they meet only on 1 or 2 channels.

We also compare A-ETCH in ACHv, using the order of maps as in S_0 and S_1 (i.e., $i = 0, 1$ in Algorithm 2 from [31]) in Table 6. The rows without $(^*)$ stand for the case when nodes select the same maps S_0 (but of course for a different number of channels N). The rows with $(^*)$ show results with the same S_1 map, and finally information with $(^{**})$ is about selection of different sequences (S_1 and S_0).

One observes that having maps with a different number of channels, allow nodes to meet always on all channels in a period. However, the choice of the maps is extremely influential on MTTR statistics, since it can be much worse than that of MtQS-DSrdv (the case without star, i.e., S_0 maps combinations), similar to that of MtQS-DSrdv (the case with $(^*)$, i.e., S_1 maps combinations), and better than that of MtQS-DSrdv (combination of S_1 and S_0 maps). Hence, the algorithm from [31] has room for improvement, especially in the case while nodes have the same channel view (SCHv), since TTR ≈ $2N$ is definitely not sufficient if nodes meet on one or two channels while having much more in a set. In ACHv there is a need for a kind of avoidance of the selection of the same sequence in order to improve TTR statistics. Finally, the algorithm assumes that N must be prime, but it can be also generalized for a nonprime N.

TABLE 6: A-ETCH in ACHv with S_0 sequence combinations for $N = 3, 5, 7$; $(r, n) = \{(3, 21), (5, 55), (7, 105)\}$.

#	Υ	MTTR	μ^{MTTR}	σ^{MTTR}	μ^{Nch}
3–5	8	36 (9 times)	21	9	3.6
$(^*)$ 3–5	8	18 (20 times)	14	4	3.6
$(^{**})$ 3–5	6	12 (19 times)	10	2	3.6
3–7	15	51 (25 times)	38	10	5
$(^*)$ 3–7	15	26 (60 times)	24	3	5
$(^{**})$ 3–7	9	24 (25 times)	17.5	4	5
5–7	19	79 (14 times)	44	20.4	3
$(^*)$ 5–7	19	40 (8 times)	26.5	8	3
$(^{**})$ 5–7	10	17 (48 times)	14	3	3

6.4.4. Comparison with Balanced Incomplete Block Design-Based RDV Approach. In order to guarantee RDV the Balanced Incomplete Block Design (BIBD) (a BIBD [35] is an arrangement of v distinct objects into b blocks, so that each block contains exactly k distinct objects, each object occurs in exactly r different blocks, and every pair of distinct object a_i, a_j occurs together in λ blocks. A BIBD has five parameters $\{v, b, r, k, \lambda\}$ so that $bk = vr$ and $r(k - 1) = \lambda(v - 1)$. While $v = b$, and then $k = r$ then such BIBD is called *symmetric* using three parameters $\{v, k, \lambda\}$) feature has been used in [29]. With two available channels a BIBD of $\{7, 3, 1\}$ is used in [29]. In order to have an RDV guarantee for multiple channels, all channels C are divided into two groups, $G1 = \{1, 2, \ldots, C/2\}$ and $G2 = \{C/2 + 1, (C/2) + 1, \ldots, C\}$. Each group is assigned to a different BIBD state; for example, BIBD of $\{7, 3, 1\}$ has $\{1, 1, 0, 1, 0, 0, 0\}$. Hence, we obtain $\{G1, G1, G2, G1, G2, G2, G2\}$. Each group is divided into two subgroups, for example, $G1 = \{G1.1, G1.2\}$ and $G1 = \{G2.1, G2.2\}$. Afterwards, each subgroup is assigned again to a different BIBD state; for example, using the same BIBD sequence, $G1$ is assigned into $\{G1.1, G1.1, G1.2, G1.1, G1.2, G1.2, G1.2\}$ and $G2$ into $\{G2.1, G2.1, G2.2, G2.1, G2.2, G2.2, G2.2\}$. The steps (iterations) are repeated until each channel has a subgroup. This algorithm is indeed very simple, and it is very easy to form an RDV sequence using, for example, the same BIBD sequence. However, one must note than there are also shortcomings of this scheme; first, it is not a problem to assign 2^s channels where $s = \{1, 2, 4, 5, \ldots\}$, since this number of channels is easy to divide multiple times in groups according the aforementioned algorithm. However, if

$n = \{3, 5, 6, 7, \ldots\}$, some of the channels must be assigned to both leaf groups (the final last subgroups) in order to form a symmetric BIBD satisfying intersections of all channels in all blocks (RCP using QS context). Moreover, the *cycle* of such sequences increases very fast. In [29] it is stated that MTTR = v^i, where v is the number of building blocks (slot offsets or shifts), where i is the number of building iterations of this sequence and $i = \log_2 C$ (C number of channels). Therefore, with *two* channels we have MTTR = 7 for aforementioned example of BIBD ($\{7, 3, 1\}$), with four channels MTTR = 7^2 = 49, with *eight* channels MTTR = 7^3 = 343, and with *sixteen* channels MTTR = 7^4 = 2401. MTTR is also equivalent to the size of the whole sequence, that is, the size of a *cycle*; that is, with *sixteen* channels the sequence has 2401 slots!

Both protocols, MtQS-DSrdv and BIBD based, guarantee RDV on all channels in the period defined by the corresponding approach. If we compare in terms of MTTR, for example, the case with available 4 channels, the cycle of MtQS-DSrdv is much lower (28 instead 49 slots) and MTTR is also much lower, 18 slots with a mean of 10 slots, instead of 49 slots; the case with 8 available channels, the cycle of MtQS-DSrdv is significantly lower (120 instead of 343 slots) and MTTR significantly lower, 86 slots with a mean of 29 slots, instead of 343 slots.

Analyzing BIBD approach in the case with heterogeneous spectrum availability, for example, with one node with 4 available channels and the other with 8 available channels, we observe than nodes still meet on four channels of the first node, with MTTR equal to 172 slots, whereas with MtQS-DSrdv we have MTTR of 51 slots with a mean of 29 slots, which is significantly better than the BIBD approach.

7. Conclusion and Future Work

In this work we evaluated a torus-QS- and DS-based rendezvous protocol while having symmetric and asymmetric channel views. We showed that the heterogeneous spectrum availability does not decrease the performance of our RDV protocol, since nodes either can still meet on all available common channels in each period, or they meet multiple times on the visited common channels. MTTR is bounded and usually being smaller than half of a larger period. We have also shown that our algorithm is more efficient and more stable in comparison with the related approaches. We pointed out that a small MTTR value is not sufficient if CR nodes do not meet on multiple channels in a sequence period, since we cannot provide the reliable performance in CR ad hoc networks due to an easier link breakage caused by the appearance of PU signals. In the future work we will address underlying MAC protocol and will implement the proposed protocol on a CR platform.

Acknowledgments

The authors thank Professor Petri Mähönen for fruitful discussions. The authors thank the financial support from Deutsche Forschungsgemeinschaft and RWTH Aachen University through UMIC Research Centre. This work has benefited significantly from discussions with participants of European Union funded ACROPOLIS Network of Excellence (Grant ICT-257626).

References

[1] S. Haykin, *Fundamental Issues in Cognitive Radio*, Edited by S. Haykin, SpringerLink, 2007.

[2] I. F. Akyildiz, W. Y. Lee, M. C. Vuran, and S. Mohanty, “NeXt generation/dynamic spectrum access/cognitive radio wireless networks: a survey,” *Computer Networks*, vol. 50, no. 13, pp. 2127–2159, 2006.

[3] S. Romaszko, D. Denkovski, V. Pavlovska, and L. Gavrilovska, “Asynchronous rendezvous protocol for cognitive radio ad hoc networks,” in *Proceedings of the International Conference on Ad Hoc Networks (ADHOCNETS ’12)*, October 2012.

[4] S. Romaszko and P. Mähönen, “Quorum systems towards an asynchronous communication in cognitive radio networks,” *Hindawi Journal of Electrical and Computer Engineering*, vol. 2012, Article ID 753541, 22 pages, 2012.

[5] S. Lang and L. Mao, “A torus quorum protocol for distributed mutual exclusion,” in *proceedings of the International Conference on Parallel and Distributed Systems (ICPADS ’98)*, December 1998.

[6] J. M. Hall, *Combinatorial Theory*, John Wiley & Sons, 1986.

[7] W. S. Luk and T. T. Wong, “Two new quorum based algorithms for distributed mutual exclusion,” in *Proceedings of the 17th International Conference on Distributed Computing Systems (ICDCS ’97)*, pp. 100–106, May 1997.

[8] M. Singhal and N. G. Shivaratri, *Advanced Concepts in Operating Systems*, McGraw-Hill, New York, NY, USA, 1994.

[9] J. R. Jiang, Y. C. Tseng, C. S. Hsu, and T. H. Lai, “Quorum-based asynchronous power-saving protocols for IEEE 802. 11 ad hoc networks,” in *Proceedings of the IEEE International Conference on Parallel Processing (ICPP ’03)*, October 2003.

[10] Z. T. Chou, Y. H. Lin, and R. H. Jan, “Optimal fully adaptive power management protocols for asynchronous multi-hop ad hoc wireless networks,” in *Proceedings of the 11th IEEE Singapore International Conference on Communication Systems (ICCS ’08)*, pp. 569–573, November 2008.

[11] E. K. Lee, S. Y. Oh, and M. Gerla, “Randomized channel hopping scheme for anti-jamming communication,” in *Proceedings of the IFIP Wireless Days (WD ’10)*, October 2010.

[12] K. Bian, J. M. Park, and R. Chen, “A quorum-based framework for establishing control channels in dynamic spectrum access networks,” in *Proceedings of the 15th Annual ACM International Conference on Mobile Computing and Networking (MobiCom ’09)*, pp. 25–36, September 2009.

[13] F. Hou, L. X. Cai, X. Shen, and J. Huang, “Asynchronous multichannel MAC design with difference-set-based hopping sequences,” *IEEE Transactions on Vehicular Technology*, vol. 60, no. 4, pp. 1728–1739, 2011.

[14] S. Romaszko and P. Mähönen, “Grid-based channel mapping in cognitive radio ad hoc networks,” in *Proceedings of the 22nd Annual IEEE International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC ’11)*, September 2011.

[15] S. Romaszko and P. Mähönen, “Torus quorum system and difference setbased rendezvous in cognitive radio ad hoc networks,” in *Proceedings of the International Conference on Cognitive Radio Oriented Wireless Networks and Communications (CrownCom ’12)*, June 2012.

[16] W. Hu, D. Willkomm, M. Abusubaih et al., "Dynamic frequency hopping communities for efficient IEEE 802.22 operation," *IEEE Communications Magazine*, vol. 45, no. 5, pp. 80–87, 2007.

[17] H. Su and X. Zhang, "Channel-hopping based single transceiver MAC for cognitive radio networks," in *Proceedings of the 42nd Annual Conference on Information Sciences and Systems (CISS '08)*, pp. 197–202, March 2008.

[18] L. Jiao and F. Y. Li, "A single radio based channel datarate-aware parallel rendezvous MAC protocol for cognitive radio networks," in *Proceedings of the IEEE 34th Conference on Local Computer Networks (LCN '09)*, pp. 392–399, October 2009.

[19] S. Geirhofer, J. Z. Sun, L. Tong, and B. M. Sadler, "Cognitive frequency hopping based on interference prediction: theory and experimental results," *ACM SIGMOBILE Mobile Computing and Communications*, vol. 13, pp. 49–61, 2009.

[20] H. Liu, Z. Lin, X. Chu, and Y. W. Leung, "Taxonomy and challenges of rendezvous algorithms in cognitive radio networks," in *Proceedings of the International Conference on Computing, Networking and Communications (ICNC '12)*, January2012.

[21] M. D. Silvius, F. Ge, A. Young, A. B. MacKenzie, and C. W. Bostian, "Smart radio: spectrum access for first responders," in *Wireless Sensing nand Processing III*, vol. 6980 of *Proceedings of the SPIE*, 2008.

[22] C. Cormio and K. R. Chowdhury, "Common control channel design for cognitive radio wireless ad hoc networks using adaptive frequency hopping," *Elsevier Ad Hoc Networks*, vol. 8, no. 4, pp. 430–438, 2010.

[23] C. Cormio and K. R. Chowdhury, "An adaptive multiple rendezvous control channel for cognitive radio wireless ad hoc networks," in *Proceedings of the 8th IEEE International Conference on Pervasive Computing and Communications Workshops (PERCOM WS '10)*, pp. 346–351, April 2010.

[24] N. C. Theis, R. W. Thomas, and L. A. DaSilva, "Rendezvous for cognitive radios," *IEEE Transactions on Mobile Computing*, vol. 10, pp. 216–227, 2010.

[25] H. Liu, Z. Lin, X. Chu, and Y. W. Leung, "Ring-walk based channelhopping algorithms with guaranteed rendezvous for cognitive radio networks," in *Proceedings of the International Workshop on Wireless Sensor, Actuator and Robot Networks (WiSARN-FALL '10), in conjunction with IEEE/ACM CPSCom*, China, December 2010.

[26] Z. Lin, H. Liu, X. Chu, and Y. W. Leung, "Jump-stay based channel-hopping algorithm with guaranteed rendezvous for cognitive radio networks," in *Proceedings of the IEEE International Conference on Computer Communications (INFOCOM '11)*, pp. 2444–2452, April 2011.

[27] C. M. Chao, H. C. Tsai, and K. J. Huang, "A new channel hopping MAC protocol for mobile ad hoc networks," in *Proceedings of the International Conference on Wireless Communications and Signal Processing (WCSP '09)*, November 2009.

[28] C. M. Chao and Y. Z. Wang, "A multiple rendezvous multi-channel MAC protocol for underwater sensor networks," in *Proceedings of the IEEE Wireless Communications and Networking Conference (WCNC '10)*, April 2010.

[29] M. Altamimi, K. Naik, and X. Shen, "Parallel link rendezvous in ad hoc cognitive radio networks," in *Proceedings of the 53rd IEEE Global Communications Conference (GLOBECOM '10)*, December 2010.

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

[31] Y. Zhang, Q. Li, G. Yu, and B. Wang, "ETCH: efficient Channel Hopping for communication rendezvous in dynamic spectrum access networks," in *Proceedings of the IEEE International Conference on Computer Communications (INFOCOM '11)*, pp. 2471–2479, chn, April 2011.

[32] K. Bian and J. M. Park, "Asynchronous channel hopping for establishing rendezvous in cognitive radio networks," in *Proceedings of the IEEE International Conference on Computer Communications (INFOCOM '11)*, pp. 236–240, April 2011.

[33] Y. S. Hsieh, C. W. Lien, and C. T. Chou, "A multi-channel testbed for dynamic spectrum access (DSA) networks," in *Proceedings of the 7th ACM workshop on Wireless multimedia networking and computing (WMUNEP '11)*, Miami, Fla, USA, October 2011.

[34] S. Romaszko and P. Mähönen, "Quorum-based channel allocation with asymmetric channel view in cognitive radio networks," in *Proceedings of the 6th ACM Performance Monitoring, Measurement and Evaluation of Heterogeneous Wireless and Wired Networks Workshop (PM2HW2N '11)*, October 2011.

[35] D. R. Sinson, *Combinatorial Designs: Constructions and Analysis*, Springer, 2004.

[36] L. A. DaSilva and I. Guerreiro, "Sequence-based rendezvous for dynamic spectrum access," in *Proceedings of the 3rd IEEE Symposium on New Frontiers in Dynamic Spectrum Access Networks (DySPAN '08)*, pp. 440–446, October 2008.

[37] Y. R. Kondareddy, P. Agrawal, and K. Sivalingam, "Cognitive radio network setup without a common control channel," in *Proceedings of the IEEE Military Communications Conference (MILCOM '08)*, November 2008.

[38] R. Zheng, J. C. Hou, and L. Sha, "Asynchronous wakeup for ad hoc networks," in *Proceedings of the 4th ACM International Symposium on Mobile Ad Hoc Networking and Computing (MOBIHOC '03)*, pp. 35–45, June 2003.

[39] J. R. Jiang, Y. C. Tseng, C. S. Hsu, and T. H. Lai, "Quorum-based asynchronous power-saving protocols for IEEE 802.11 ad hoc networks," *Mobile Networks and Applications*, vol. 10, no. 1, pp. 169–181, 2005.

[40] S. Romaszko, "Making a blind date the guaranteed rendezvous in cognitive radio ad hoc networks," in *Proceedings of the 18th European Wireless Conference (EW '12)*, April 2012.

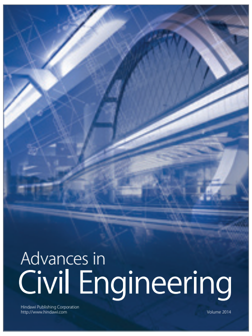
Journal of
Engineering

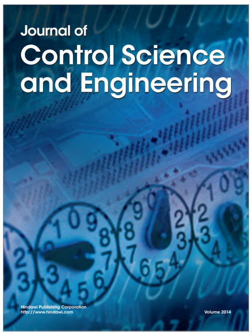
The Scientific World Journal

International Journal of
Rotating Machinery

Journal of
Sensors

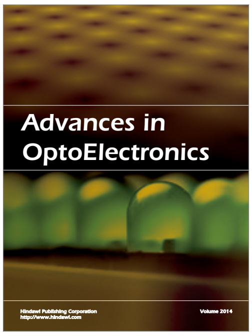
International Journal of
Distributed Sensor Networks


Advances in
Civil Engineering


Journal of
Control Science and Engineering

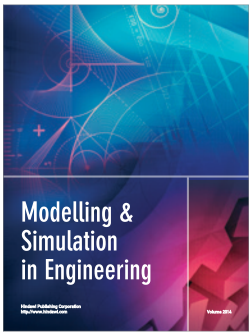
Journal of
Robotics

Journal of
Electrical and Computer Engineering

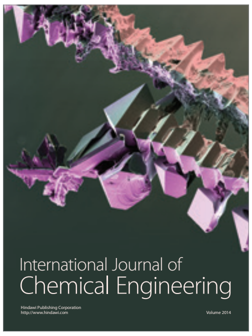

Advances in
OptoElectronics

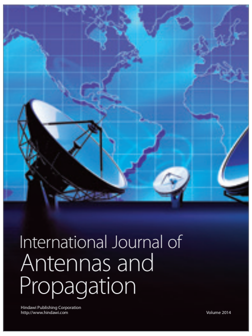
VLSI Design

International Journal of
Navigation and Observation

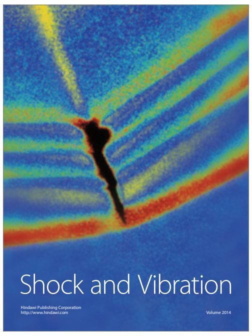

Modelling & Simulation in Engineering

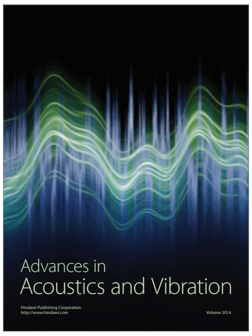
International Journal of
Aerospace Engineering


International Journal of
Chemical Engineering


International Journal of
Antennas and Propagation

Active and Passive Electronic Components


Shock and Vibration


Advances in
Acoustics and Vibration