Journal of Network and Computer Applications 113 (2018) 98–108

Contents lists available at ScienceDirect

Journal of Network and Computer Applications

journal homepage: www.elsevier.com/locate/jnca

Supporting fast and fair rendezvous for cognitive radio networks

Chih-Min Chao [*], Hsiang-Yuan Fu

Department of Computer Science and Engineering, National Taiwan Ocean University, Taiwan

ARTICLE INFO

Keywords:
Cognitive radio networks
Dynamic spectrum access
Channel hopping
Hadamard matrix
Hash function

ABSTRACT

A cognitive node in a cognitive radio network (CRN) temporarily and opportunistically uses a channel not being occupied by any licensed user. It is challenging to provide a rendezvous between any pair of nodes in CRNs. Enabling nodes to hop among channels is a dominant mechanism to solve the rendezvous problem without using a dedicated common control channel. Existing channel hopping solutions may suffer from poor or uneven rendezvous for different pairs of nodes. In this paper, we propose a novel distributed channel hopping protocol (HHCH) which efficiently provides rendezvous guarantee and fair rendezvous opportunity. The balanced rendezvous among nodes is provided by utilizing a Hadamard matrix while fair rendezvous opportunity is achieved by using a hash function. In addition to rendezvous guarantee and fairness, the HHCH protocol also provides high and even channel utilization. Analytical and simulation results verify that HHCH performs better in terms of time to rendezvous and network throughput when compared to existing representative protocols Jump-Stay, ACH, SYNC-ETCH, QLCH, and RQL.

1. Introduction

Wireless spectrum is a precious resource but a large portion of licensed spectrum is still underutilized (FCC, 2003). To increase spectrum utilization, using cognitive radios is considered as a promising solution since an unlicensed user (secondary user, SU node) is allowed to access the licensed spectrum not being used by any licensed user (primary user, PU node). The concept of cognitive radio was first proposed by Mitola J. in 1998 (Mitola, 1998) using the name of software-defined radio (SDR). A SDR is a fully reconfigurable wireless transceiver which automatically adapts its communication parameters to network and user demands. A Cognitive Radio Network (CRN), also known as dynamic spectrum access network, is considered evolved from SDR. In CRNs, SU nodes are able to recognize spectrum holes and can hop among them without causing operation interruption to PU nodes. CRNs attract a lot of attention recently (Khan et al., 2015; Kumar et al., 2016; Chao et al., 2017). Providing rendezvous opportunity to any pair of SU nodes are an essential and crucial requirement in a CRN. Some traditional wireless multi-channel protocols provide rendezvous guarantees but these protocols are not appropriate for CRNs because they do not handle the PU node occupancy issue. Several solutions use a single channel as the common control channel (CCC) to exchange control message to provide rendezvous (Cormio and Chowdhury, 2010; Hamdaoui and Shin, 2008; Kim and Yoo, 2012; Lo et al., 2010; Wang et al., 2011). A major concern of these methods is that a stable dedicated CCC may not be found due to PU node occupancy. Applying channel-hopping (CH) is one of the most popular techniques to achieve rendezvous guarantee without using a CCC. Channel-hopping solutions can be classified according to the following two factors:

- *with or without rendezvous guarantee*: Whether a rendezvous between any pair of SU nodes are guaranteed or not.
- *asynchronous or synchronous*: Whether all SU nodes are time-synchronized or not.

Based on this classification, existing solutions and the scheme proposed in this paper can be categorized in Table 1. In general, an SU node running a channel hopping protocol switches channels based on a predefined channel hopping sequence. Most channel hopping protocols guarantee a rendezvous between any pair of SU nodes. Some channel hopping protocols only provide *partial rendezvous guarantee* (Chuang et al., 2013; DaSilva and Guerreiro, 2008; Liu et al., 2010, 2012; Gandhi et al., 2012; Romaszko, 2012; Romaszko and Mähönen, 2011) in that rendezvous is guaranteed only at part of the available channels. Some other protocols (Altamimi et al., 2010; Bian and Park, 2011; Shin et al., 2010; Chao et al., 2015, 2016; Paul and Choi, 2016; Sali et al., 2016; Li and Xie, 2016; Yang et al., 2016; Sahoo and Sahoo, 2016; Bian et al., 2009; Zhang et al., 2011; Shih et al., 2010; Chao and Fu, 2016a) provide *complete rendezvous guarantee* in that any pair of SU nodes have a

* Corresponding author.
E-mail addresses: cmchao@ntou.edu.tw (C.-M. Chao), hyfu@ntou.edu.tw (H.-Y. Fu).

https://doi.org/10.1016/j.jnca.2018.03.032
Received 4 November 2016; Received in revised form 8 August 2017; Accepted 29 March 2018
Available online 31 March 2018

Table 1
Classification of CRN rendezvous protocols.

	w/o guarantee	w/ guarantee
Asynchronous	Theis et al., 2011	Chuang et al., 2013; DaSilva and Guerreiro, 2008; Liu et al., 2010, 2012; Gandhi et al., 2012; Romaszko, 2012; Romaszko and Mähönen, 2011; Altamimi et al., 2010; Bian and Park, 2011; Shin et al., 2010; Chao et al., 2016; Paul and Choi, 2016; Sali et al., 2016; Li and Xie, 2016; Yang et al., 2016; Sahoo and Sahoo, 2016
Synchronous	NA	Bian et al., 2009; Zhang et al., 2011; Shih et al., 2010; Chao et al., 2015; Chao and Fu, 2016a, ours

rendezvous at each of the available channels. The complete rendezvous guarantee solutions are considered more robust and flexible when some channels are occupied by PU nodes.

Because precise global clock synchronization is very difficult to achieve in CRNs, several asynchronous solutions provide complete rendezvous guarantee without global clock synchronization. Comparing to synchronous ones, asynchronous channel hopping schemes are more suitable for neighbor discovery during the network initialization phase. Here we briefly review two representative asynchronous channel hopping protocols. In the Jump-Stay channel-hopping protocol (JS) (Liu et al., 2012), each SU node's channel hopping sequence consists of a jump-pattern and a stay-pattern, lasts for $2P$ and P time slots respectively, where P is the smallest prime number larger than the number of available channels. An SU node switches among available channels in the jump-pattern and stay on a specific channel in the stay-pattern. The asynchronous channel hopping scheme (ACH) (Bian and Park, 2011) operates similar to a grid quorum system. An SU node running ACH is assigned to be either a sender or a receiver and uses an $n \times n$ grid to determine its channel hopping sequence, where n is the number of channels. A sender and a receiver use different grids. Instead of using the intersection between a column and a row in a grid, SU nodes running ACH utilize the intersection between a column and a span in a grid to provide rendezvous guarantee where a span consists of one element from each column. In general, the asynchronous solutions avoid time synchronization at the expense of *time to rendezvous (TTR)* and throughput, when compared to the synchronous ones.

To enhance network performance, many synchronous channel hopping protocols have been proposed. The QCH (Bian et al., 2009) scheme is a complete rendezvous guarantee protocol where a rendezvous between any pair of SU nodes is guaranteed because of the intersection property of quorum systems. M-QCH and L-QCH are two variations of QCH to minimize maximum time to rendezvous (MTTR) and channel load, respectively. A synchronous efficient channel hopping protocol (SYNC-ETCH) achieves complete rendezvous guarantee and have high channel utilization when the number of SU nodes is large enough (Zhang et al., 2011). However, it may suffer from low system throughput and large TTR. In general, existing complete rendezvous guarantee protocols may still suffer from high MTTR, low channel utilization, or energy waste.

Utilizing quorum systems and latin squares, two existing synchronous channel hopping protocols, Quorum and Latin square Channel Hopping (QLCH) (Chao et al., 2015) and Randomized Quorum and Latin square Channel Hopping (RQL) (Chao and Fu, 2016a) have been proposed recently. Enabling an SU node to calculate its neighbors' channel hopping sequences using their IDs, both protocols avoid redundant transmissions and reduce TTR significantly. In QLCH and RQL, a quorum system is used to provide balanced rendezvous among SU nodes and a latin square is used to share the rendezvous among channels. Specifically, an SU node partitions its time slots into default slots and switching slots (which is called a *pattern* hereafter) based on a quorum selected by its SU node ID. An SU node is assigned an initial channel to indicate the channel to be switched to at a time slot. An SU node waits for transmission requests at default slots and sends requests at switching slots. An SU node tunes to the initial channel at a default slot and may switch to a channel different from the initial channel at a switching slot. Both protocols reduce TTR to a very low value (around two). This is a significant improvement since the TTR generated by other channel hopping protocols is proportional to the number of channels being used. An issue of QLCH is that the channel utilization is always 50% and thus the system throughput can be improved. An SU node uses different patterns randomly in RQL instead of sequentially in QLCH, which enables RQL to achieve better network throughput. A flaw of RQL is the high MTTR. A weakness of both protocols is that the variance of pairwise rendezvous may be large. That is, some pairs of SU nodes have an obvious higher number of rendezvous when compared to the others.

In this paper, a novel efficient channel hopping scheme denoted as HHCH is proposed. Similar to QLCH and RQL, HHCH utilizes the default slots and switching slots to provide pairwise rendezvous with constant TTR. HHCH avoids the issues occurred in QLCH and RQL by using the concepts of Hadamard matrix and hash function. A Hadamard matrix is utilized to provide balanced rendezvous guarantee among different SU nodes while a hash function is applied to share such rendezvous among different channels. SU nodes running HHCH do not need to exchange their channel hopping sequences with other SU nodes. An SU node's channel hopping sequence is determined by its node ID. Thus, it is easy for an SU node i to calculate another SU node j's channel hopping sequence if SU node j's ID and time slot offset (time slot difference) between SU nodes i and j are available. Note that a SU node running HHCH only needs its neighbors' IDs and the time slot offsets to its neighbors. SU nodes do not need to be globally time-synchronized. In order to obtain neighbor information, HHCH can cooperate with an existing asynchronous rendezvous-guaranteed channel hopping protocol (such as JS (Liu et al., 2012) or ACH (Bian and Park, 2011)) at the initialization phase. After that, because HHCH performs well, SU nodes can apply HHCH to enjoy improved performance. An attractive feature of HHCH is fairness: All different pairs of SU nodes have the same number of rendezvous during a given time interval. The HHCH protocol provides complete rendezvous guarantee with low TTR, low TTR variance, high channel utilization, and high system throughput. To the best of our knowledge, HHCH is the first scheme that aims to minimize the variance of pairwise rendezvous. We have extended our earlier work (Chao and Fu, 2016b) by proving the correctness of HHCH and analyzing the performance of HHCH theoretically. Specifically, we have added two properties and two theorems to prove that HHCH provides both rendezvous guarantee and complete rendezvous. We have proposed the scrambled reduced Hadamard matrix to further reduce MTTR. We have also provided a complete performance analysis of HHCH and conduct simulation comparison of MTTR for different protocols.

The main contributions of this paper are summarized as follows.

1. A CRN rendezvous protocol HHCH is proposed to provide higher numbers of rendezvous among SU nodes without using a CCC (Section 3).
2. The pairwise rendezvous probability is identical for all pairs of SU nodes. That is, each SU node has the same rendezvous opportunity with any of its intended receivers. In addition, rendezvous among SU nodes are evenly distributed among all channels. (Section 3).
3. Prove that HHCH provides complete rendezvous guarantee (Theorem 2 in Section 3).
4. Provide theoretical analysis to verify the superiority of HHCH in TTR (Table 5 in Section 4).