August 2015
Doctoral Degree Thesis

Energy-Efficient Network Management Protocols for Cognitive Radio Sensor Networks

Graduate School of Chosun University

Department of Computer Engineering

Shelly Salim

Energy-Efficient Network Management Protocols for Cognitive Radio Sensor Networks

인지 무선 센서 네트워크를 위한 에너지
효율적인 네트워크 관리 프로토콜

August 25, 2015

Graduate School of Chosun University

Department of Computer Engineering

Shelly Salim

Energy-Efficient Network Management Protocols for Cognitive Radio Sensor Networks

Advisor: Prof. Sangman Moh, PhD

A thesis submitted in partial fulfillment of the requirements for a Doctoral degree

April 2015

Graduate School of Chosun University

Department of Computer Engineering

Shelly Salim

살림 셀리의 박사학위논문을 인준함

위원장	조선대학교 교수	정 일 용 (인)
위 원	조선대학교 교수	신 석 주 (인)
위 원	조선대학교 교수	강 문 수 (인)
위 원	한국전자통신연구원 책임연구원	김 영 우 (인)
위 원	조선대학교 교수	모 상 만 (인)

2015 년 6 월

조선대학교 대학원

TABLE OF CONTENT

LIST OF FIGURES

LIST OF TABLES

ABSTRACT

Energy-Efficient Network Management Protocols for Cognitive Radio Sensor Networks

Shelly Salim
Advisor: Prof. Sangman Moh, Ph.D.
Department of Computer Engineering
Graduate School of Chosun University

A cognitive radio sensor network (CRSN) is a wireless sensor network (WSN) in which the sensor nodes are equipped with cognitive radio. CRSN is envisioned as a key technology to support the future wireless networks such as seamless telecommunication, the Internet of things, and the improvement of spectrum utilization. Despite of its merits, CRSN yields numerous challenges: some being inherent to cognitive radio properties and others to WSN characteristics. Cognitive radio possesses dynamic spectrum access capability. Thus, it requires a spectrum decision method to select its operating channel. Furthermore, clustered topology is mostly favored in a WSN. However, cluster formation becomes challenging in a CRSN environment. Last but not least, a WSN as an application-driven network needs to maintain an acceptable degree of reliability, which is supported by the transport

protocol. Above all, energy conservation is of the utmost importance because of the energy- and resource-constrained nature of sensor nodes.

In this thesis, energy-efficient network management protocols for CRSNs are proposed, which are composed of a spectrum decision framework, a clustering protocol, and a transport protocol. The spectrum decision framework is distributed and it contains two spectrum selection algorithms: random selection and game-theory based selection. This framework also incorporates two spectrum sensing schemes of full and partial spectrum sensing, a simple clustering, a spectrum characterization scheme implementing Markov chain, and a cluster member coordination scheme. The clustering protocol is compact and it efficiently achieves compact cluster formation by adopting two sub-phases (cluster head discovery and cluster member invitation) of cluster formation. By introducing a novel concept of temporary support nodes, the clustering enables sensor nodes to form clusters efficiently. The transport protocol is a content-aware data transmission and acknowledgment method that aims to increase the network lifetime while decreasing delay and maintaining reliability at the same time.

The performance of each proposed protocol is evaluated by means of computer simulations and compared with the existing works. The performance evaluation of the spectrum decision framework shows that the framework outperforms the existing work in terms of network lifetime and coordination overhead. The performance evaluation of the clustering protocol shows that it achieves outstanding energy savings that prolong the network lifetime and decreases both the clustering overhead and the average distance between cluster heads and their members, compared to the existing work. Finally, the performance evaluation of the transport protocol shows that it achieves remarkably longer network lifetime and shorter event-detection delay compared to the conventional transport protocol while preserving event-detection reliability.

한글요약

인지 무선 센서 네트워크를 위한 에너지 효율적인 네트워크 관리 프로토콜

살림 셀리

지도교수: 모상만

컴퓨터공학과

조선대학교 대학원

인지 무선 센서 네트워크(CRSN)는 인지 무선 기능을 갖는 센서 노드들로 구성된 무선 센서 네트워크이다. CRSN 은 단절 없는 통신, 사물 인터넷, 스펙트럼 이용률 증대 등과 같은 미래 무선 네트워크를 지원하는 핵심 기술로 전망되고 있다. 그와 같은 장점에도 불구하고, CRSN 에는 아직 많은 도전 분야가 존재한다. 그 중에는 인지 무선 고유의 특성에 의한 것도 있고 무선 센서 네트워크 특성에 기인한 것도 있다. 인지 무선 기술은 동적 스펙트럼 접근을 가능하게 하며 동작 채널을 선택하기 위한 스펙트럼 결정 방법을 필요로 한다. 무선 센서 네트워크에서는 클러스터 구성이 크게 선호되지만, 클러스터 구성은 CRSN 환경에서 많은 문제를 야기한다. 또한, 응용에 따라 구성되는 센서 네트워크 특성상 수용 가능한 수준의 신뢰성을 제공하는 트랜스포트 프로토콜이 요구된다. 무엇보다도 센서 노드는 에너지와 자원이 제한되므로 에너지 절감이 가장 중요한 요소이다.

본 연구에서는 CRSN 을 위한 에너지 효율적인 프로토콜로서 분산 스펙트럼 결정 프레임워크, 간결한 클러스터링 프로토콜, 견고한 트랜스포트 프로토콜을 제안한다. 첫째, 분산 스펙트럼 결정 프레임워크는 두 개의 스펙트럼 선택 알고리즘인 무작위 선택과 게임이론 기반 선택으로 구성된다. 이 프레임워크는 또한 두 가지의 스펙트럼 감지 기법, 간단한 클러스터링 프로토콜, 마코브 체인 기반 스펙트럼 특성화 기법, 클러스터 멤버 협력 기법 등을 포함한다. 둘째, 제안한 클러스터링 프로토콜은 두 단계의 클러스터 형성을 거쳐 간결한 클러스터 구조를 실현한다. 이 알고리즘은 임시 지원 노드의 개념을 적용하여 센서 노드들이 클러스터를 효율적으로 형성하게 해준다. 셋째, 제안한 트랜스포트 프로토콜은 내용 인지 데이터 전송 및 응답 프로토콜로서 네트워크 수명을 향상시킴과 동시에 전송 지연시간을 감소시키고 네트워크 신뢰성을 향상시킨다.

제안한 각 프로토콜의 성능은 컴퓨터 시뮬레이션을 수행하여 평가한다. 성능 평가 결과에 의하면, 분산 스펙트럼 결정 프레임워크는 네트워크 수명과 협력 오버헤드 측면에서 기존의 기법을 크게 능가한다. 제안한 클러스터링 프로토콜은 종래의 프로토콜에 비해서 두드러지게 에너지를 절감하여 네트워크 수명을 연장시키며, 클러스터링 오버헤드를 줄여주고 클러스터 헤드와 멤머들 사이의 평균 거리를 감소시킨다. 제안한 트랜스포트 프로토콜은 기존의 타 프로토콜과 비교하여 네트워크 수명을 현저하게 증가시키며, 이벤트 검출 지연시간을 감소시킴과 동시에 이벤트 검출 신뢰성을 보장해준다.

I. INTRODUCTION

Nowadays, wireless telecommunication technologies are getting more favored compared to their predecessor, wired technologies. The most appealing feature of wireless technology is that they do not require the installation of transmission media, which makes their deployment significantly faster, with lower cost and applicable for remote/challenging terrain. Wireless technologies are also the primary foundation for mobile and cellular communications and the promoters in realizing seamless and ubiquitous telecommunications. In April 2014, the International Telecommunication Union published a report that predicted the mobile-cellular subscriptions would reach 96% penetration rate by the end of 2014, in which more than three quarters of the subscriptions would come from developing countries [1]. Moreover, the same report stated that wired technologies growth in the developing countries is slowing down, which shows that wireless and mobile telecommunication technologies are preferred.

A. Wireless Sensor Network

In this thesis, the wireless sensor network (WSN) is of a particular interest. A WSN is a network of a large number of densely deployed sensor nodes [2, 3]. The sensor nodes are able to monitor various ambient conditions, such as temperature, humidity, movement, and so forth. WSN is a matured technology and it has vast applications in the field of environmental, industry, agriculture, healthcare, security, as well as commercial and military. Numerous ideas for WSNs implementations, namely 50 sensors applications for a "smarter world", are listed in [4] and real deployment cases are VigilNet, AlarmNet, Luster, etc [5]. WSN is envisioned as one of the essential foundations to realize Internet of Things (IoT) [6].

The sensor nodes send the environment-sensing data to a central repository entity called the sink node. They are battery-powered devices with very limited energy- and computational-resources. Thus, given the sensor nodes' resource-constraints nature, their energy consumption rates determine the lifetime of a WSN. The sensor nodes have to preserve their energy in order to extend the network lifetime by adopting energy-efficient protocols. Energy conservation strategies can be included at the node level, medium access control level, and network level [7]. Network-level energy conservation, particularly clustering approach, is of a particular interest. Clustering helps in minimizing routing activities, conserving bandwidth, stabilizing network topology, and preserving energy [8].

Clustering is a well-known strategy in WSNs, in which nearby nodes form a group called a cluster, and divides the data transmission activities into intra-cluster and inter-cluster transmissions. The sensor nodes belonging to a particular cluster do not send their data to the sink; instead, they send the data to their respective cluster head (CH). CHs are responsible for forwarding the data to the sink. A clustered WSN can significantly reduce energy consumption as well as network congestion and data collision, compared to non-clustered one [9]. Clustering can also reduce the transmission range needed by the sensor nodes to transmit their data (if transmission power can be adjusted, then energy consumption will be reduced). Adjacent sensor nodes might report similar data, thus, instead of sending entire data to the sink, CHs perform data aggregation to reduce the data volume and preserve energy.

Another important concern of a WSN is that it is an application-driven network. The WSN must supply the data to meet the application's objective in a reliable manner. However, the environment-sensing data are transmitted to the sink through wireless links, making the data transmissions prone to failure. In order to transmit the data in a reliable manner, the adaptation of an effective transport protocol is crucial. Especially in WSNs, the transport protocol must also ensure energy efficiency. Reliability is related to the provision of stable and error-free data transmissions. When a data transmission occurs, the sender should be able to confirm that the receiver has received the data correctly. The receiver should be able to notify the sender if it did not receive the transmitted data or if the received data are erroneous. A straightforward and common method to ensure reliability is by requiring the receiver to send an acknowledgment packet to the sender on correct reception of data.

B. Cognitive Radio

Even though wireless telecommunication technologies offer various advantages, they also possess some inherent challenges. Wireless devices communicate with each other by means of antennas that radiate within certain frequency, i.e. the operating spectrum band. Fundamentally, wireless devices broadcast their data, because of the shared-natured free-space medium. Thus, interferences between multiple wireless transmitters on the same or adjacent spectrum bands are inevitable, in which they have adverse effects on the data reception quality. The interferences in wireless transmission lead to higher data loss rate, lower transmission speed, higher delay, and lower protection against security attacks [10], compared to the wired telecommunication methods. In order to increase the quality of wireless data transmission, researchers are actively searching for innovative methods that aim to reduce either the interference sources or to mitigate the effects of interference [11, 12]. As the novel methods improve the quality of wireless data transmission, the users increase their usage and demand for higher data capacity and faster data transmission, creating the so called "The Virtuous Cycle of the Mobile Wireless Ecosystem" shown in Figure 1 [13]. However, in the earlier years, researchers were focused in improving the data transmissions quality and rather overlooked the fact that the radio spectrum is a limited resource and not always available. A certain spectrum band might be unavailable for use because of: (1) the transmissions are too crowded to allow acceptable-error data transmission; (2) the wireless devices do not support data transmissions on that spectrum band, and (3) the users do not have the license to transmit on that spectrum.

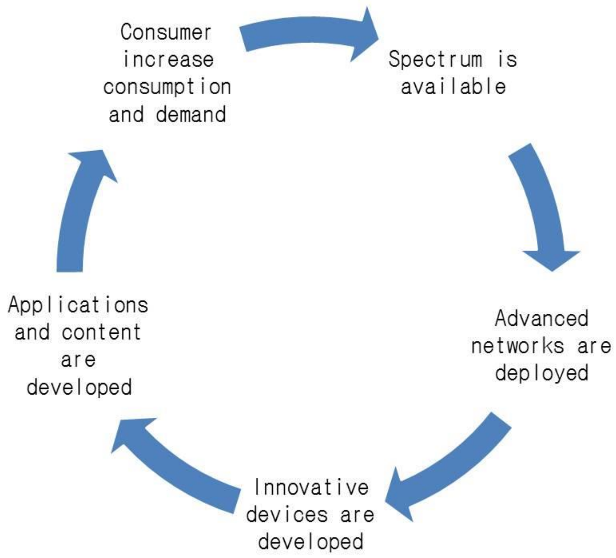


Figure 1. The virtuous cycle of the mobile wireless ecosystem.

The radio spectrum is considered as both non-renewable resource and non-depletable resource [14, 15], which means that new radio spectrum cannot be discovered as well as existing radio spectrum cannot be reduced (in volume). The radio spectrum bands have been divided into two groups: licensed spectrum bands and unlicensed spectrum bands. In order to facilitate internationally seamless wireless services, typically, licensed spectrum allocations are similar across the world. The usages of licensed spectrum bands are regulated by the governments who create spectrum/frequency allocations that divide the spectrum bands statically and assign the permissible utilizations. An example of frequency allocation can be obtained from [16]. Likewise, unlicensed spectrum bands assignments are similar for most countries. Some unlicensed spectrum bands are allocated for industrial, scientific, and medical (ISM) uses; however, they are also used for data transmission (non-ISM usage). Examples of the most popular unlicensed bands are the 2.4 GHz and 5 GHz.

For licensed spectrum band, not only services but also users are regulated. Usually, the government grants some wireless service providers the authority to utilize certain licensed spectrum bands for a specified long-term duration. During that duration, only those service providers and their customers are allowed to transmit on the spectrum. On the other hand, any user can transmit on unlicensed spectrum bands. As time goes by, the usages of unlicensed bands become very crowded because there are various usages on them, no utilization fee, and new ideas can be tested/implemented quickly. Meanwhile, there are several field measurements that showed low utilization on licensed spectrum bands. In late 2002, Federal Communications Commission (FCC) assessed that, on daily basis, the occupancy of a certain licensed spectrum band was 5 to 12%, with peak utilization at 85%, in some cities in the United States [17]. Newer surveys revealed similar results in different regions [18, 19]. The unbalanced situation makes the overall spectrum utilization low.

In order to improve the spectrum utilization and to accommodate the unlicensed spectrum users, an idea arose, to allow unlicensed users to transmit on the licensed band opportunistically. This approach introduces two types of users: incumbent/licensed/primary users and unlicensed/secondary users. The term primary user (PU) and secondary user (SU) are used throughout this thesis. The most fundamental requirement for SUs to be allowed to transmit in the licensed band is that their transmission must not impede the transmissions of PUs. SUs need wireless devices capable of detecting PUs transmission, operating on a wide spectrum band, and switching the operating spectrum/channel. Cognitive radio (CR) supports those capabilities and more (in some works, SU is also called CR user).

The term CR first emerged from a dissertation work of Mitola about advanced software-defined radio in 2000 [20]. However, the definition of CR adopted here follows the thorough description by Haykin in [21], in agreement to the more general description by FCC [22], that is: “A cognitive radio (CR) is a radio that can change its transmitter parameters based on interaction with the environment in which it operates.” The ability to adjust the transmission parameter is also called reconfigurability, which, together with cognitive capability, was added by Akyildiz as the two main characteristics of CR [23]. In agreement with the definition of cognitive/cognition in psychology, cognitive capability refers to the ability of learning and reasoning. Reconfigurability reacts on the outcome of the cognitive stage and modifies the transmission parameters accordingly.

Since CR’s introduction, it has received a lot of attention. Many features of CR and their implementations on existing wireless networks are studied extensively [24, 25]. International standardization bodies are developing standards to guide the implementations of CR, in which the first standard is the IEEE 802.22 published on 2008 [26, 27]. CR is also included in various types of wireless networks, resulting in new types of wireless networks: CR ad hoc networks [28, 29], CR sensor networks [30], CR mesh network [31], and so forth. Moreover, CR is foreseen to be the supporting technology for the fifth generation (5G) of cellular wireless standard [32].

Operations of CR follow the so called cognitive cycle, which consists of: spectrum sensing, spectrum management, spectrum mobility, and spectrum sharing, as shown in Figure 2. In spectrum sensing, CR device tunes to each channel within the spectrum band and senses the radio frequency to determine whether there are ongoing data transmissions. Licensed channels without any transmission detected are called vacant channels, spectrum holes, or white spaces, which can be utilized by SUs. In spectrum management, CR device analyzes the results of spectrum sensing and decides the operating channel by executing a set of rules or decision making methods. Spectrum mobility deals with changing operating channel to maintain uninterrupted data transmission and spectrum sharing considers transmission opportunity fairness between SUs.

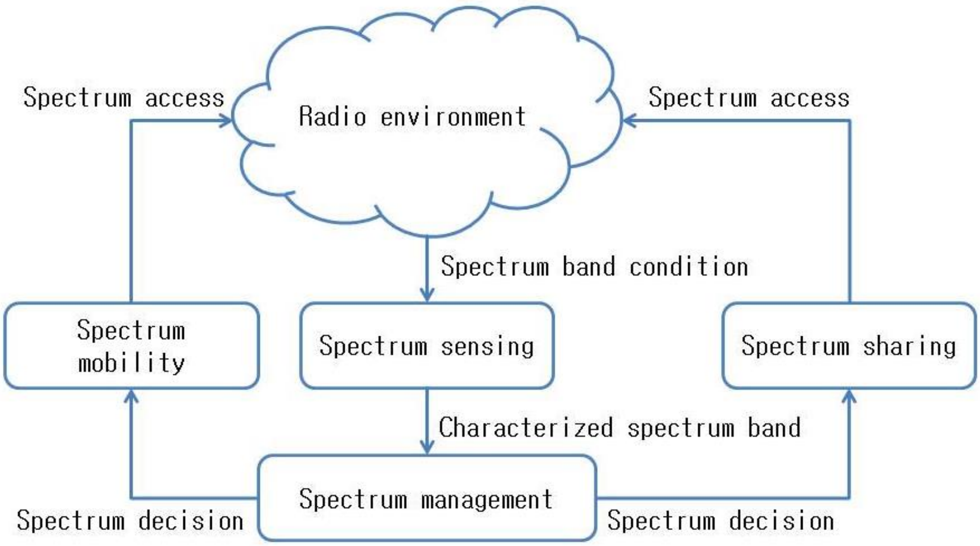


Figure 2. CR cycle.

The main trigger of enthusiasm in CR is to improve the spectrum utilization, especially in the licensed spectrum band. The expectation is, if PUs share their spectrum band with SUs, then spectrum utilization will be increase. This scheme is called spectrum sharing between PUs and SUs or inter-spectrum sharing, which can be supported, in the simplest manner, by setting a rule that SUs must not interfere with PUs' activity (in practical, this rule is rather difficult to follow [33]). At the same time, SUs should also share the transmission opportunities between themselves. Adopting the assumption that there is no prioritization between SUs (recent concepts introduce tertiary users [34]), the issue of spectrum sharing between SUs or intra-spectrum sharing should be addressed as well. In other words, spectrum utilization must be increased, hence, inter-spectrum sharing enabled by CR is encouraged and it is enabled by CR, however, as a consequence, intra-spectrum sharing issue arises. Intra-spectrum sharing could be accomplished by adopting an effective spectrum decision method.

Spectrum decision is a part of spectrum management which selects an operating channel among a number of possible channels. In spectrum decision, cognitive capability of the CR by means of machine learning and decision making algorithm is applied. The fundamental requirement of spectrum decision is to select a channel that is reported vacant by spectrum sensing. In the case of single SU, the spectrum decision may choose the best channel (in terms of signal to noise ratio, bit error rate, and so forth) as the operating channel. However, to communicate, there is no case of single user. In a CR network with multiple SUs, simply selecting the best channel as the operating channel may not turn out to be the best solution. For example, in Figure 3, sensor A selects channel 3 as its operating channel to transmit data to sensor C because channel 3 has the highest signal to noise ratio

(SNR). Meanwhile, sensor B also set its operating channel to channel 3 because of the same reason. When sensor A and B transmit their data at the same time, interference might occur on the receivers' side. Therefore, simply selecting the best channel may not be optimum, especially to support intra-spectrum sharing. Each SU has to consider and predict the actions of other SUs on its surroundings. SUs will perform better by selecting the most suitable channel. In the proposed spectrum decision framework, game theory is applied because it considers the interactions between multiple decision makers, i.e. the SUs.

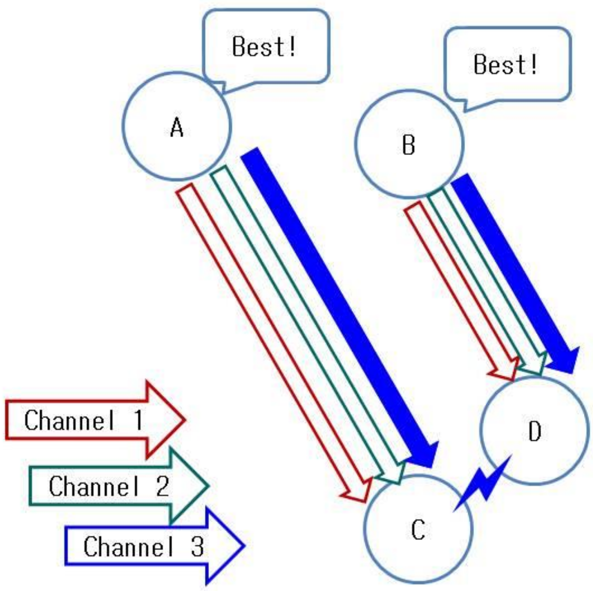


Figure 3. Spectrum decision may cause interference on the receivers' side.

Game theory is a decision-making theory under uncertain and interdependent situations; that is, the actions of the decision makers affect other decision makers [35]. It models and analyzes interactive decision conditions to predict the result of interactions among decision makers. It outperforms mathematical analysis of wireless networks [36]. In general, there are two types of games: cooperative game and non-cooperative game. In cooperative games, the players are able to communicate between them to arrange their strategies for achieving a social goal. In non-cooperative game, there is no communication between players and each player aims to maximize its own profit. A comparison of cooperative game and non-cooperative game could be found in [37]. The proposed spectrum decision framework belongs to non-cooperative game.

As the merits of WSN and CR have been presented, integration of CR to the WSN, i.e. a CR sensor network (CRSN), is of a particular interest.

C. Cognitive Radio Sensor Network

The idea to combine WSN with CR by integrating CR at the sensor nodes is a promising one. In the near future, we can expect multiple WSNs deployed within the same area. However, WSNs are only allowed to transmit in the unlicensed channels, where they share the channels with numerous other wireless devices and suffer from interferences. Considering that the data collected by the WSNs are crucial, WSNs have to be supported with a method to improve their channel access. CR allows the sensor nodes to access the vacant licensed channels opportunistically and its reconfigurability can be exploited to conserve the energy of sensor nodes. The resulting network of CR integration with WSN is called a CR sensor network (CRSN).

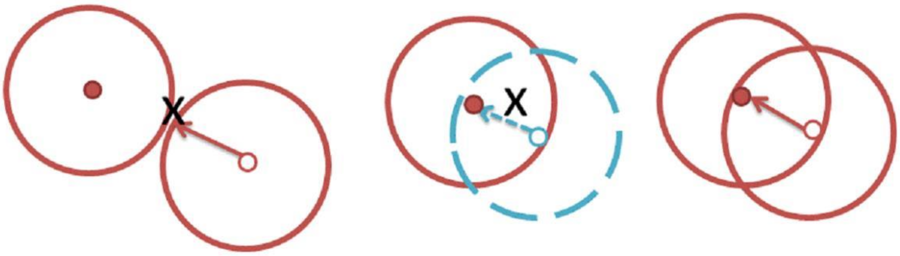
However, resource-constrained sensor nodes are required to perform the CR tasks in addition to their original tasks, resulting in increased energy consumption. Moreover, CRSN inherits the unique characteristics of WSNs, such as it is an application-driven network, the sensor nodes are energy-constrained, the data transmission is usually delay sensitive (real-time) and the data transmission flow is many-to-one, among other features. The SUs are the sensor nodes in a CRSN and the term SUs and sensor nodes are used interchangeably.

Spectrum decision alone is not sufficient for a CRSN. Thus, a framework that supports the entire operations of a CRSN is composed. The proposed spectrum decision framework is distributively carried out by each sensor node. It consists of three modules: spectrum sensing module, spectrum decision module, and data transmission module. Two spectrum sensing schemes and a simple residual energy-based clustering in spectrum sensing module are incorporated. The spectrum decision module contains a Markov chain-based spectrum characterization, a game theory-based spectrum selection method, a cluster member coordination scheme, and a spectrum access scheme. The data transmission module is a schedule-based one.

Another issue in a CRSN is clustering. Clustering in a CRSN is similar to clustering in an ordinary WSN. Each cluster consists of one cluster head (CH) and a number of cluster members (CMs). Clustering is considered a proper topology handling method for a CRSN, primarily because only the CHs need to perform CR management tasks instead of all the sensor nodes, which reduces the total energy consumption. However, clustering in a CRSN has an additional requirement: that is, to form a cluster, the sensor nodes not only have to be in the transmission range of one another but also have to operate in the same communication channel, as illustrated in Figure 4. This limitation might cause a poor cluster formation. Fundamentally, clustering in CRSNs should consider the energy consumption of the sensor nodes because this directly affects the network lifetime. In the same approach, clustering overhead has to be minimized in order to achieve efficient power consumption, while supporting both event-driven and regular data collection.

Lastly, transport protocol plays an important role, especially in sensor networks, to provide reliability. However, studies on designing transport protocols to suit CRSNs are relatively few in number. The existing works on transport layer protocols for WSNs are not suitable for CRSNs because they do not consider the aspect of dynamic spectrum access [38, 39].

A content-aware transport protocol for CRSNs to preserve energy and maintain reliability is proposed.



(a) (b) (c)

Figure 4. Clustering requirements in a CRSN.

D. Research Objectives

The main objective of this thesis is to support CRSNs with energy-efficient network management protocols in order to prolong the network lifetime, to improve performance parameters, and to support the integration of CR and WSN. Thus, a spectrum decision framework, a clustering protocol, and a transport protocol are proposed. The spectrum decision framework is essential to achieve the fullest of CR's functionalities efficiently and to assist those functionalities to benefit the target application of CRSNs. The framework enables the sensor nodes, as SUs, to perform inter-spectrum sharing as well as intra-spectrum sharing. A clustering protocol is designed particularly to suit multiple and heterogeneous channel conditions in CRSNs, which could not be accomplished by traditional WSN clustering. The proposed clustering protocol is a refinement of popularly adopted clustering concept with an addition of a novel approach, namely temporary support nodes. Finally, CRSNs as application-driven networks needs to ensure the data transmission reliability, thus a content-aware and reliable transport protocol is proposed. The contributions of each protocol are presented in the respective protocol's chapter.

E. Thesis Layout

The rest of the thesis is composed as follows: in chapter 2, related works of the three proposed protocols are represented. Then, descriptions of the distributed spectrum decision framework, the compact clustering protocol, and the robust transport protocol are provided in chapter 3, 4, and 5, respectively. In each chapter of the proposed protocol, the composition is as follow: starting with preliminaries and basic assumptions adopted in the respective work, explaining the proposed protocol in detail, discussing the performance evaluation results, and lastly, delivering the conclusions. Finally, the overall conclusions of this thesis and consider a number of future works are provided in chapter 6.

II. RELATED WORKS

In this chapter, the works related with the proposed protocols are presented. To the best of our knowledge, there was no related work in the field of spectrum decision framework for CRSNs yet. Several spectrum decision methods were proposed for general/other types of CR network. As a matter of fact, there is no standardized definition for spectrum decision. In other works, the selection of operating channel might be referred to spectrum management, spectrum access, spectrum assignment, spectrum sharing, and so forth. Thus, a work is considered as a related work, if it performs similar operations as the proposed framework. Likewise, no transport layer protocols have yet been designed for CRSNs. Thus, some transport protocols for WSN and other types of CR network are presented. Several works have reported clustering algorithms for CRSNs. For each proposed protocol, one work is selected as a comparison work.

A. Spectrum Decision Methods

A survey paper about spectrum decision in CR networks mentions that there are three main functions of spectrum decision: spectrum characterization, spectrum selection, and CR reconfiguration [40]. Similar statement could be found for CRSNs, that the spectrum decision consists of three sub-processes: spectrum allocation, spectrum access, and spectrum handoff [41]. In the proposed framework, the most essential functions are extracted and the spectrum decision module consists of: spectrum characterization, spectrum selection, cluster member coordination, and spectrum access.

There are several works dealing with spectrum management for CRSNs. A naive approach, called ordered channel assignment [42] requires control over PUs channel decision, which is rather unrealistic. New parameters to support channel assignment are proposed in [43, 44], where they use R-coefficient to represent predicted residual energy and value of the spectrum usability to represent spectrum idle rate and spectrum quality, respectively. Both works showed improved performance but one work lacks what another work covered. In another words, a method that is energy-aware as well as spectrum-aware would be preferred. Channel assignment is combined with routing in [45] with packet-based channel assignment. While the consideration to routing might be one of the appeals of this work, it is rather not an essential issue in CRSNs, because fundamentally long distance data transmission is enabled by CR reconfiguration. Similar drawback could be inferred from grid-based channel assignment [46]. Markov decision process is adapted in channel allocation [47] and operation mode selection [48]. Markov decision process and game theory are both decision making algorithms in which the decision makers interact with opponents in a dynamic environment. However, as discussed in [49, 50], game theory is considered more suitable than Markov decision process in an environment with multiple agents (decision makers) and their actions contribute to the dynamics of the environment. A centralized spectrum allocation based on game theory is proposed in [51]. While centralized approach is plausible because there is at least one sink in a CRSN, it does not cope well with the spectrum heterogeneity over time and space. The result of centralized spectrum allocation might be not optimum for every sensor node, not to mention the additional overhead cost.

There are a lot more works of spectrum decision methods as well as general decision making algorithms for CR network and its varieties (except CRSNs). An analysis of load-based, interference-based, and joint of both showed that basic metrics were able to improve the performance of cognitive ad hoc network, though only for certain cases [52]. Novel parameters for spectrum decision were proposed, such as request index to represent the quality of service requirements of SUs [53], channel usage state [54], and outage probability [55]. While those works applying new parameters showed improved performance in various metrics, the definition of parameters were fixed. An online learning was used in spectrum decision algorithm based on predictions; however, the focus was merely to calculate the probability of handover [56]. Similarly, specific purpose spectrum decision methods are proposed in [57, 58, 59] in which they were focused on time minimization and security issues. An energy-aware spectrum decision framework is proposed in [60] where it includes an energy monitoring unit with a predefined threshold.

A spectrum decision framework for CR network proposed in [61] is selected as the comparison work of the proposed framework. The reasons are, similar with the proposed work, the comparison work proposed a framework, it has been highly cited by other papers, and it is a complete and detailed work with strong numerical analysis and improved performance evaluation results. The comparison work proposed two algorithms under its spectrum decision block: minimum variance-based spectrum decision (MVSD) and maximum capacity-based spectrum decision (MCSD). The proposed framework is compared only with MVSD because it was designed to support real-time applications. The admission control function is excluded because in the network scenario, there would be no new sensor nodes installed in the middle of the network operation. MVSD is a centralized approach applied in an infrastructure-base CR network in which the base station performs the spectrum decision. The CR nodes or sensor nodes perform spectrum sensing, send the results to the base station, and wait for the spectrum decision results.

B. Clustering Protocols for CRSNs

In some works in the field of CRSNs, the clustering methods are assumed or fixed [62, 63, 64]. Event-driven spectrum-aware clustering [65] creates temporal clusters for each event based on the position, node degree, available channels, and distance to the sink. The clusters are no longer available at the end of the event. Thus, event-driven spectrum-aware clustering is only suitable for WSNs intended for event-driven applications. Distributed spectrum-aware clustering (DSAC) [66, 67] uses the local minimum distance obtained by information exchanges to merge two nearby nodes or clusters that share the same available channels. The cluster formation process is repeated until the optimal number of clusters is reached. The adaptation of the low-energy adaptive clustering hierarchy (LEACH) protocol to suit CRSNs was reported in [68]. The proposed clustering protocol is compared to DSAC scheme because DSAC also focuses on clustering in general-purpose CRSNs, as in the proposed clustering protocol.

C. Transport Protocols for General CRN

Generally, transport protocols used in WSNs can be categorized depending on whether they focus on reliable transmission or on congestion control. Event-to-sink reliable transport (ESRT) [69], reliable multi-segment transport (RMST) [70], and “pump-slowly, fetch-quickly” (PSFQ) transport [71] are some protocols that have been proposed to achieve reliable transmission. The ESRT protocol reduces energy consumption by its low complexity but its transmission speed depends on the environment. The RMST protocol has a drawback of decreased energy efficiency because of its high complexity, but it has an advantage of highly efficient memory management. The PSFQ protocol reduces the transmission speed considerably, but it quickly restores reliability. The representative protocols for congestion control are the congestion detection and avoidance (CODA) protocol [72] and the sensor transmission control protocol (STCP) [73]. The CODA protocol controls network congestion by allowing the nodes to control the transmission rate after congestion is detected. The drawback of this protocol is that the loss of the ACK packet makes the transmission rate, delay, and response time longer because of network congestion.

The existing works on transport layer protocols for WSNs are not suitable for CRSNs mainly because they do not consider the aspect of dynamic spectrum access. Several transport protocols have been proposed for general CR wireless networks and CR ad hoc networks. These transport protocols do not consider the resource limitations of the sensor nodes, especially the energy constraint. One of the frequently-cited transport protocols is the transport protocol for CR ad hoc networks (TP-CRAHN) [74]. TP-CRAHN adapts TCP to suit the CR environment by creating six states, including spectrum sensing and spectrum change. A continuation work of TP-CRAHN can be found in [75]. The proposed transport protocol is compared with TP-CRAHN, which originally was developed for CR ad hoc networks. It is selected as a comparison work because it is one of the earliest and the most cited transport protocol in CR network environments.

III. DISTRIBUTED SPECTRUM DECISION

A. Introduction

Spectrum decision is a significant component in CR-based networks. In this Chapter, a spectrum decision framework suitable for a CRSN is designed and evaluated. Motivated to improve the performance of CRSNs, the contributions of the spectrum decision framework are as follows: (1) a complete framework for a time-slotted CRSN, (2) two types of spectrum decision algorithm, namely random selection and game-theory-based selection, and (3) simple yet effective supportive protocols for clustering, spectrum sharing, and spectrum access. The spectrum decision framework is called an energy-efficient distributed spectrum decision (EDSD) framework.

The EDSD framework is designed for CRSNs with numerous sensor nodes placed randomly in an area of interest and a sink located at the center of the area. The sensor nodes are battery-powered without energy-harvesting ability. The CRSN is able to access three spectrum bands: television (TV), ISM 2.4 GHz, and ISM 5 GHz. The TV band consists of 30 channels [76] and the first channel being the common control channel. The CRSN is located in an urban area where the incumbent users of the TV band (herein PU) exist. The maximum number of PUs is predetermined, but the number of PUs at a certain time is not fixed. The PUs can be either active or passive. A PU is active when it transmits or receives a data transmission; otherwise, it is passive. A passive PU can become active, and vice versa. PUs can change their operating channel and move their location while they are active. Mobility to PU is included because some smart phones are embedded with a TV receiver.

The CRSN's operations are performed in frames. It is a time-slotted network where the management and transmission activities are performed in a certain time slot during a frame. A frame is equal to 2 s and divided into 111 time slots where each time slot equals 18 ms [77] and the last time slot is equal to 20 ms.

Two channel models are included, each for licensed and unlicensed channels. Sensor nodes perform spectrum sensing in licensed channels to obtain the information on the channels' occupants. Sensor nodes are prohibited to use a licensed channel when it is sensed as not vacant. They can use any unlicensed channel regardless of the channel condition. The unlicensed channel is modeled in terms of peak interferences. Peak interferences take place on a channel, and they affect two adjacent channels. When a sensor node selects an unlicensed channel with high interference, transmission failure probability of is higher, whereas when it uses a licensed channel, transmission success is guaranteed unless the total number of sensor nodes transmitting on the channel is higher than a threshold. Otherwise, the probability of transmission failure increases with the number of channel occupants. A licensed channel is defined as the common control channel (CCC). The CCC is used exclusively to transmit control packets.

Each sensor node is equipped with one CR transceiver; thus, it is able to tune in one channel at a time. Meanwhile, the sink has two CR transceivers, in which one transceiver is always tuning in the CCC and another can switch its channel. The sink can broadcast its control packets on the CCC so that every sensor node can receive them. Fundamentally, the sensor nodes are also able to transmit to the sink directly, enabled by the CR's reconfiguration ability. However, direct transmission is not favorable because long-distance transmission requires high energy consumption.

EDSD framework is designed for CRSNs to suit environmental maintenance systems that require periodic data collection. In the long term, the expectation is that EDSD framework will contribute to the realization of the smart city concept or to support IoT.

B. Energy-Efficient Distributed Spectrum Decision (EDSD) Framework

The EDSD framework has two operation modes: coordination mode (C mode) and data transmission mode (D mode). A frame can be in either C mode or D mode. In C mode, coordination activities take most parts of the frame, whereas in D mode, environment-sensing data collections are encouraged. Both modes consist of the same modules: spectrum sensing, spectrum decision, and data transmission module. The duration of spectrum sensing (t_{ss}) is maximized in C mode to support full spectrum sensing, and it is minimized in D mode because only partial spectrum sensing is performed. On the other hand, the duration of data transmission (t_{dt}) is maximized in D mode. The duration of the spectrum decision (t_{sd}) for both modes is the same (see Fig. 5).

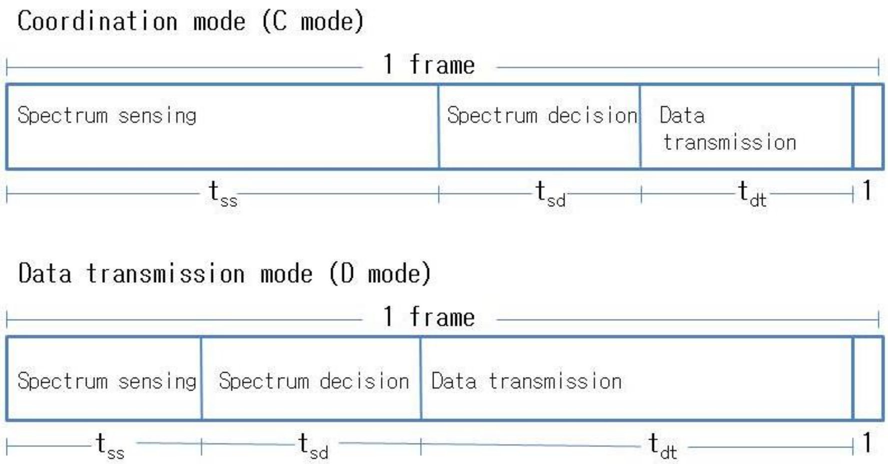


Figure 5. The modules of C mode and D mode.

The operation mode of the first frame is C mode and the operation mode of the following frames is decided by the sink. The last time slot in each frame is allocated for sensor nodes to report to the sink whether they require C mode. Two types of C-mode request are defined: normal request and urgent request. If the sink receives at least one urgent request, it decides that the next frame will be a C mode. When the sink receives normal requests, it does not immediately set the next mode to be a C mode, but waits until the number of frames associated with normal requests has exceeded a certain threshold. If there is no request at all, then the next mode is set to be D mode. The conditions in which a sensor node requests C mode are explained in the next section. Whenever a C mode is performed, the previous clustering topology is invalid and is resettled.

The sensor nodes are divided into three classes: sensor node (SN), cluster head (CH), and cluster member (CM). Class SN represents the initial sensor node class before the clustering protocol is carried out. The details of the EDSD framework are explained using following notations are used:

- A set of N sensor nodes acting SUs $S = \{S_1, S_2, \cdots, S_N\}$ with $status(S_i) \in \{active, passive\}$ and $class(S_i) \in \{CH, CM, SN\}$, $1 \leq i \leq N$.
- A set of M PUs $T = \{T_1, T_2, \cdots, T_M\}$ with $status(T_j) \in \{active, passive\}$, $1 \leq j \leq M$.
- A set of $(L+1)$ licensed channels $A = \{A_0, A_1, A_2, \cdots, A_L\}$ and its SNR observed by S_i $SNR_i = \{SNR_1, SNR_2, \cdots, SNR_L\}$.
- A set of K unlicensed channels $B = \{B_1, B_2, \cdots, B_K\}$ and its SNR observed by S_i $SNR'_i = \{SNR'_1, SNR'_2, \cdots, SNR'_K\}$.
- A common control channel $C_{CC} = A_0$.
- An operating channel of S_i on current frame $f = (C_i)_f = A_x$ or B_y or $\emptyset$, where $1 \leq x \leq L$, $1 \leq y \leq K$, and $\emptyset$ means empty set.

- A backup channel of S_i on current frame f = $(C'_i)_f$ = A_x or B_y or ∅, where $1 \leq x \leq L$, $1 \leq y \leq K$, and $(C'_i)_f \neq (C_i)_f$.
- $Status(A_x)_i \in \{avlb, not\ avlb, obsl, idle, busy\}$, $1 \leq x \leq L$, observed by S_i.
- $Status(B_y)_i \in \{clean, noisy, unknown\}$, $1 \leq y \leq K$, observed by S_i.
- CH type = $type(CH)_i \in \{0, 1, 2, 3, 4\}$, $class(S_i) = (CH)$, $1 \leq i \leq N$.

1. Spectrum Sensing Module

Full spectrum sensing (FSS) and partial spectrum sensing (PSS) are performed in C mode and D mode, respectively. The tasks in FSS and PSS are shown in Figure 6. In FSS, the sensor nodes perform spectrum sensing on entire licensed channels, that is, spectrum sensing set = $\{A_1, A_2, \cdots, A_L\}$ (A_0 is C_{CC}). Using one of the spectrum sensing methods, the sensor nodes obtain the information of whether there is an ongoing transmission on a channel, and record the channel's SNR. (The SNR calculation is explained later.) If there is no PU transmission detected by S_i on A_x, then

$$Status(A_x)_i = (avlb), \tag{1}$$

otherwise,

$$Status(A_x)_i = (not\ avlb), \tag{2}$$

for $1 \leq x \leq L$, $1 \leq i \leq N$, and $status(S_i) = (active)$.

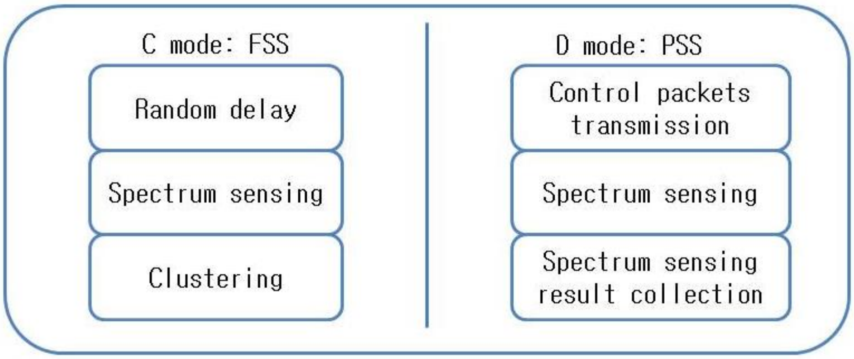


Figure 6. Spectrum sensing module.

To prevent false detection of PU transmissions, all sensor nodes must not perform any data transmission during FSS (quiet period). The activities in FSS are not only spectrum sensing but also sleeping for a random time (delay). The sensor nodes sleep for a random duration (which may be different per sensor) before they start to perform spectrum sensing. The main purpose of random delay is to support the clustering protocol. Moreover, the expectation is, that a random delay will increase the variability of spectrum sensing results compared with when all sensor nodes start their spectrum sensing at the same time. The random delay is brief, and its maximum duration is predetermined.

In PSS, the sensor nodes perform only spectrum sensing on their operating channels and backup channels selected from a previous frame. PSS is performed in a D mode in which the clustered topology is preserved and the sensor nodes are either CHs or CMs. First, the CMs check the connectivity to their CHs by sending a control packet. Meanwhile, the CHs examine the status of their backup channel (whether the backup channel is empty or not), and when they receive the control packets from their respective CMs, they reply with an acknowledgement and the status of the backup channel. After those control packet transmissions are finished, all sensor nodes (CHs and CMs) perform partial spectrum sensing. If there is no PU transmission detected by S_i on its operating channel selected from the previous frame ($(C_i)_{fprev}$), then

$$Status(C_i)_{fprev} = (avlb), \tag{3}$$

otherwise,

$$Status(C_i)_{fprev} = (obsl), \tag{4}$$

for $1 \leq i \leq N$ and $status(S_i) = (active)$.

Spectrum sensing on the backup channel is performed only when the backup channel is not empty, i.e., $(C'_i)_{fprev} \neq \varnothing$. Similarly, if there is no PU transmission detected by S_i on $(C'_i)_{fprev}$, then

$$Status(C'_i)_{fprev} = (avlb), \tag{5}$$

otherwise,

$$Status(C'_i)_{fprev} = (obsl), \tag{6}$$

for $1 \leq i \leq N$ and $status(S_i) = (active)$.

The CMs send their spectrum-sensing results to their CH. As the CRSN continues to operate, the sensor nodes eventually deplete its energy and become inactive. If a CH becomes inactive, its CMs would not receive acknowledgement packets; thus, they would go to sleep and wake up at the last time slot of the current frame to send urgent requests for C mode because an inactive CH causes the entire cluster to be inactive. If a CH does not receive any control packet (all of its CMs are inactive), it proceeds to the next activities and modules, and also sends a normal request for C mode at the last time slot. Aside from the energy depletion of the sensor nodes, the operation channel's quality degradation also causes the CHs or CMs to be unable to receive packets from each other. However, the protocol does not differentiate those causes of failed transmissions.

In FSS, the outcomes are a list of licensed channels, their status (available or not available), and variables to calculate SNR. In PSS, the outcomes are a list of operating channels and backup channels, their status (available or obsolete), and the variables to calculate the SNR. If the operation mode is C mode, then the sensor nodes continue to perform clustering; otherwise, they perform spectrum decision module directly.

2. Residual Energy-based Clustering

EDSD framework also includes a clustering protocol, called residual-energy-based clustering. This protocol is triggered during C-mode operation only. A sophisticated clustering protocol is avoided despite its performance improvement because in a CRSN there are additional energy-consuming spectrum-related activities compared to traditional WSNs. Therefore, to minimize energy consumption, a simple clustering method is included. The clustering protocol is included in the spectrum sensing module as a part of FSS. Some parts of the clustering protocol start before spectrum sensing, and the rest begin after spectrum sensing, as illustrated in Figure 7.

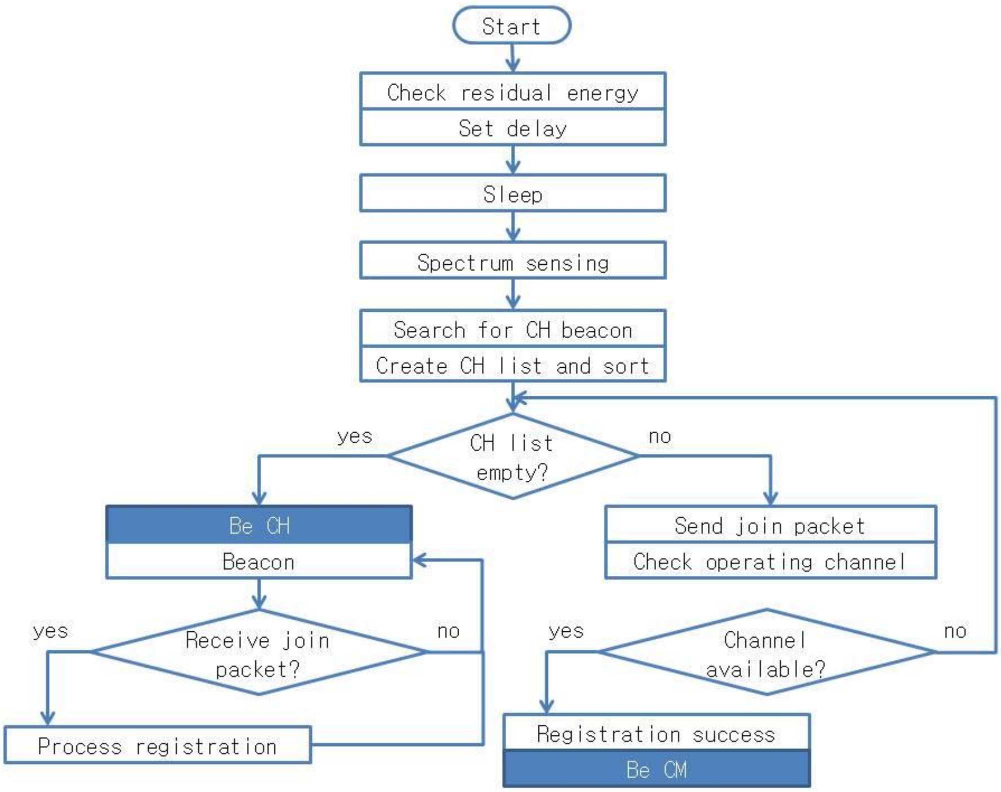


Figure 7. Residual-energy-based clustering.

Initially, all sensor nodes are in the class sensor nodes, class(S_i) = (SN), $\forall i$. First, the sensor nodes check the residual energy and set the maximum limit of the delay based on it. If the residual energy is high, then the maximum limit of the delay is low, and vice versa. Next, the actual delay duration is set randomly, following a uniform distribution with zero as the minimum value and the previously settled maximum limit as the maximum value. The sensor nodes go to sleep during this delay and wake up to perform spectrum sensing (as explained in the previous section). The sensor node S_i listens on the C_{CC} for CH beacons, creates a $CH\_list = \{CH_1, CH_2, \cdots, CH_r\}_i$, and sorts the CH_list based on the received signal strength indicator (RSSI) in a descending manner. The reason for adapting a random delay is that without a random delay, every sensor node would perform the same activity at the same time, which would result in no CH found when they try to find a CH because every sensor node is currently finding a CH. Later, when each of them decides to become a CH, probably no node is looking for a CH anymore because all of them have also become CHs.

The sensor nodes try to join the CH on the top of the list first. The cluster registration method is as follows: the sensor node sends a join packet to the CH, and the CH replies with operating channel information. The sensor node checks whether the assigned operating channel is idle on its side. (The spectrum sensing result might be different spatially.) If the assigned operating channel is available, then the sensor node S_i becomes a CM of that CH, i.e., $class(S_i) = (CM)$, sets its operating channel as the assigned operating channel, and goes into sleep state until the end of the spectrum sensing module. If the assigned operating channel is not available, then the sensor node checks the CH list and tries the next CH on the list, given that the remaining time is sufficient. Otherwise, the sensor node becomes a CH, i.e., $class(S_i) = (CH)$. A sensor node also becomes a CH if it cannot find any CHs from the first time.

When a sensor node becomes a CH, it selects one of its available channels as its operating channel randomly, transmits beacons on the C_{CC}, waits for any join packet, and responds to the join packet with information about its operating channel. However, the randomly selected operating channel is a temporary one because spectrum decision module is not executed yet. Temporary operating channel is included as a basic requirement in that both parties (CH and CM) find the operating channel as a vacant channel. At this stage, the CH type is type 0 (zero), i.e., $type(CH)_i = 0$. (Explanations about CH type are in the next section.) The outcome of the clustering protocol is that each sensor node has selected its class as either a CH or a CM. Afterward, the sensor nodes continue to the spectrum decision module.

3. Spectrum Decision Module

In the spectrum decision module, the CHs are responsible for selecting an operating channel and a backup channel, and managing their CMs.

The selected operating and backup channels may be licensed channels or unlicensed channels. There are four tasks in the spectrum decision module: spectrum characterization, spectrum selection, CM coordination, and spectrum access (Figure 8). These activities are similar in both operation modes except that in D mode some of the activities are shorter because the sensor nodes update the information regarding two channels only.

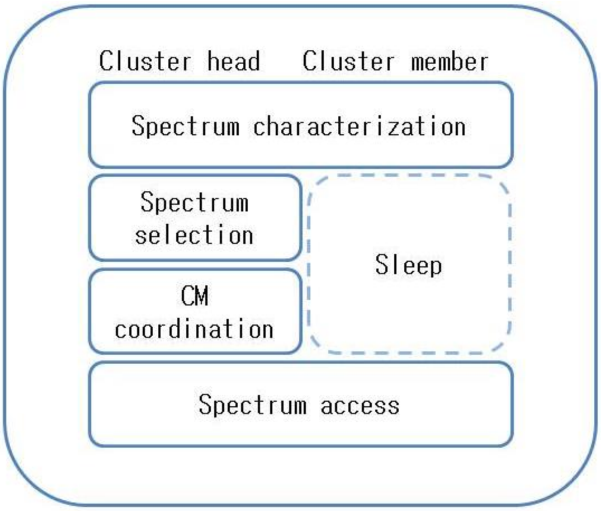


Figure 8. Spectrum decision module.

a. Spectrum Characterization

All sensor nodes (CHs and CMs) perform spectrum characterization, and the tasks are the same in both C mode and D mode. Although CMs are not involved in spectrum selection, they need to characterize the spectrum in order to update the spectrum history, which they will use if they become CHs on the next frame. The input of spectrum characterization is the outcome of the spectrum sensing module, i.e. the equation (1) - (6). The sensor nodes characterize the licensed channels in C mode, and characterize the operating channel and backup channel in D mode. The SNR calculations for licensed channels (SNR_x) and unlicensed channels (SNR'_y) are different, as follows:

$$SNR_x = (P_{PUsignal})_x / (P_{PUnoise})_x, \qquad (7)$$

and

$$SNR'_y = P_{highestnoise} - (P_{noise})_y, \qquad (8)$$

for $1 \le x \le L$ and $1 \le y \le K$ observed by S_i, where $(P_{PUsignal})_x$ is the power received from the closest PU on channel A_x inside the sensor node's transmission range, $(P_{PUnoise})_x$ is the power received from all PUs except for the closest one inside its interference range, $P_{highestnoise}$ is the maximum noise level, and $(P_{noise})_y$ is the actual noise level on channel B_y. (SNR'_y is not actually a ratio.)

Using these SNR calculations, the best SNR_x is equal to 0/0, because that value means that there is no PU at all. The next best SNR_x is equal to 0/V (V is an arbitrary value) because it means that all PUs are outside the sensor node's transmission range, even though there are PUs inside its interference range. In the protocol, the value 0/0 is replaced to 1 (one) and 0/V to 0 (zero). Using the SNR value, the status of the licensed channel is updated. If SNR_x = 1, then

$$Status(A_x)_i = (idle), \tag{9}$$

otherwise,

$$Status(A_x)_i = (busy), \tag{10}$$

for 1 ≤ x ≤ L, observed by S_i. (Note that this status is different from $status(A_x)_i$ = *(avlb)* and $status(A_x)_i$ = *(not avlb)*.)

The channel status update for an unlicensed channel is as follows: if SNR'_y is higher than a predetermined threshold, SNR'_{thres}, then

$$Status(B_y)_i = (clean), \tag{11}$$

otherwise,

$$Status(B_y)_i = (noisy), \tag{12}$$

for 1 ≤ y ≤ K, observed by S_i.

Spectrum sensing is not performed on the unlicensed channels, thus, the information about them only exists if the sensor nodes ever have to use unlicensed channels as their operating/backup channel (operating/backup channel sensing is performed in partial spectrum sensing). Otherwise,

$$Status(B_y)_i = (unknown), \tag{13}$$

for 1 ≤ y ≤ K, observed by S_i, and $(P_{noise})_y$ = 0 or $SNR'_y = P_{highestnoise}$.

A Markov chain to predict the channels' status and to update the channel holding time is incorporated. Channel holding time is defined as the expected "time" that the channel will remain idle or clean for licensed and unlicensed channels, respectively. The "time" refers to the number of frames. The Markov chain provides a simple ability to learn and predict. A Markov chain for each licensed channel and unlicensed channel is created, as shown in Figure 9.

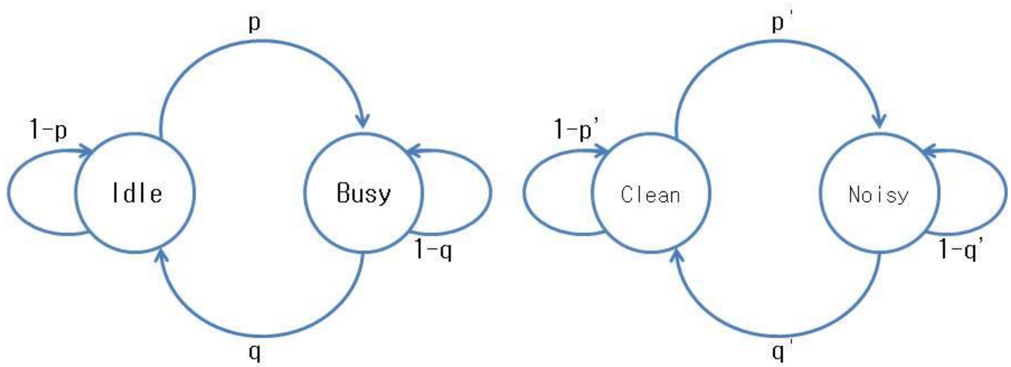


Figure 9. Markov chain for (a) licensed channels and (b) unlicensed channels (p = probability of an idle channel becomes busy, q = probability of a busy channel becomes idle, p' = probability of a clean channel becomes noisy, and q' = probability of a noisy channel becomes clean).

The values on the Markov chains are updated based on the spectrum characterization status. (Only the operating and backup channels are updated on D mode). The values ($1 - p$) and ($1 - p'$) are considered during the spectrum selection task for licensed channels and unlicensed channels, respectively. The channel holding time is calculated as follows:

If $(1 - p)^t > p_{thres}$ for a licensed channel and $(1 - p')^t > p'_{thres}$ for an unlicensed channel, $t \geq 1$, and $t_{max} \geq 1$, then

$$(CHT_x)_f = (CHT_x)_{fprev} + \sum_{t=1}^{t_{\max}} 1, \quad (14)$$

and

$$(CHT'_y)_f = (CHT'_y)_{fprev} + \sum_{t=1}^{t_{\max}} 1, \quad (15)$$

for $1 \leq x \leq L$, $1 \leq y \leq K$ observed by S_i at current frame f, where $(CHT_x)_f$ and $(CHT_x)_{fprev}$ are the channel holding times on channel A_x for the current frame and the previous frames, respectively. Similarly, $(CHT'_y)_f$ and $(CHT'_y)_{fprev}$ are the channel holding times on channel B_x for the current frame and the previous frames, respectively. The terms $(1 - p)$ and $(1 - p')$ are the probability of an idle channel to remain idle and a clean channel to remain clean, respectively. The terms $(1 - p)^t$ and $(1 - p')^t$ are the probabilities that an idle channel will remain idle or a clean channel will remain clean, for time t. The terms p_{thres} and p'_{thres} are predetermined probabilities. The term t_{max} is the highest integer for which the statements $(1 - p)^t > p_{thres}$ and $(1 - p')^t > p'_{thres}$ are still valid.

If $(1 - p) < p_{thres}$ for a licensed channel and $(1 - p') < p'_{thres}$ for an unlicensed channel, then

$$(CHT_x)_f = \frac{1}{2}(CHT_x)_{fprev}, \tag{16}$$

and

$$(CHT'_y)_f = \frac{1}{2}(CHT'_y)_{fprev}, \tag{17}$$

for $1 \le x \le L$, $1 \le y \le K$ observed by S_i at current frame f.

If $p = 0$ for a licensed channel and $p' = 0$ for an unlicensed channel, then

$$(CHT_x)_f = (CHT_x)_{fprev} + CHT_{\max}, \tag{18}$$

and

$$(CHT'_y)_f = (CHT'_y)_{fprev} + CHT'_{\max}, \tag{19}$$

for $1 \le x \le L$, $1 \le y \le K$ observed by S_i at current frame f, where CHT_{max} and CHT'_{max} are predetermined constants.

b. Spectrum Selection

Spectrum selection tasks are performed only by the CHs. CMs go to sleep to save energy during spectrum selection tasks, and they wake up to receive configuration settings at the beginning of CM coordination tasks. The input for spectrum selection is the characterized channels from spectrum characterization, whereas the output is the selection of one operating channel and one backup channel. The spectrum selection tasks of C mode and D mode are different because of the nature of the selection. In C mode, the goal is to select an operating channel and a backup channel, whereas in D mode the goal is to decide whether the current operating channel can be used or needs to be changed.

For spectrum selection in C mode, the CHs are divided into four types depending on the number of idle licensed channels (initially, all CHs are type 0). The CH types are:

$$Type(CH)_i = \begin{cases} 1 & \text{if } |(A_{idle})_i| = 0, \\ 2 & \text{if } |(A_{idle})_i| = 1, \\ 3 & \text{if } |(A_{idle})_i| = 2, \\ 4 & \text{if } 2 < |(A_{idle})_i| \le L, \end{cases} \tag{20}$$

for 1 ≤ i ≤ N and class(S_i) = (CH), where type(CH)$_i$ is the CH type of S_i, and $(A_{idle})_i$ is the set of idle licensed channels observed by S_i, i.e.,

$$\begin{aligned} (A_{idle})_i &= \{A_{x1}, A_{x2}, \ldots, A_{xl}\}_i \\ &= \{A_{xv} \mid A_{xv} \in A, status(A_{xv}) = (idle), 1 \le v \le L\}_i, \end{aligned} \tag{21}$$

where v is an arbitrary value, 1 ≤ v ≤ L, and $A_{x1} \neq A_{x2} \neq \cdots \neq A_{xl}$.

For S_i with $type(CH)_i$ = 1, because there is no idle licensed channel, both the operating channel and backup channel for the current frame, $(C_i)_f$ and $(C'_i)_f$, are selected from a set of unlicensed channels. Hence,

$$[(C_i)_f, (C'_i)_f] = [(C_i)_f \in (B_{clean})_i, (C'_i)_f \in \{(B_{clean})_i - (C_i)_f\}], \tag{22}$$

for 1 ≤ i ≤ N and class(S_i) = (CH), where the set $(B_{clean})_i$ is the clean unlicensed channels of S_i, i.e.,

$$(B_{clean})_i = \{B_{y1}, B_{y2}, \ldots, B_{yk}\}_i$$
$$= \{B_{yv} \mid B_{yv} \in B, status(B_{yv}) = (clean), 1 \le v \le K\}_i, \quad (23)$$

where v is an arbitrary value, $1 \le v \le K$, and $B_{y1} \neq B_{y2} \neq \cdots \neq B_{yk}$.

For S_i with $type(CH)_i = 2$, the operating channel is set to be the only idle licensed channel, and the backup channel is selected from $(B_{clean})_i$. For S_i with $type(CH)_i = 3$, one of its idle licensed channels is set as the operating channel, and the remaining one idle licensed channel is set as the backup channel. Hence,

$$[(C_i)_f, (C'_i)_f] = [A_{x1}, (C'_i)_f \in (B_{clean})_i], \quad (24)$$

for $1 \le i \le N$, $class(S_i) = (CH)$, and $type(CH)_i = 2$.

$$[(C_i)_f, (C'_i)_f] = [A_{x1}, A_{x2}] \; or \; [A_{x2}, A_{x1}], \quad (25)$$

for $1 \le i \le N$, $class(S_i) = (CH)$, and $type(CH)_i = 3$.

Additionally, for S_i with $type(CH)_i = 2$ or $type(CH)_i = 3$, the S_i broadcasts the licensed channel selection on the C_{cc}. For S_i with $type(CH)_i = 4$, it performs spectrum etiquette by allowing S_i with $type(CH)_i = 2$ or $type(CH)_i = 3$ to claim their operating channel first. Thus, S_i with $type(CH)_i = 4$ listens for broadcasts on the C_{CC} before it selects its operating channel. After listening to some declarations of operating channels, S_i with $type(CH)_i = 4$ eliminates the idle channels that have been claimed, and selects its operating and backup channels. Hence,

$$[(C_i)_f, (C'_i)_f] = [(C_i)_f \in \{(A_{idle})_i - (A_{excl})_i\}, \quad (C'_i)_f \in \{(A_{idle})_i - (A_{excl})_i - (C_i)_f\}], \tag{26}$$

for 1 ≤ i ≤ N and class(S_i) = (CH), where the set $(A_{exclude})_i$ contains the idle channels of S_i that have been claimed by other CHs, i.e.,

$$(A_{excl})_i = \{(C_{h1})_f, (C_{h2})_f, \ldots, (C_{h(R-1)})_f\}_i = \{\sum_{r=1}^{R-1}(C_{hr})_f\}, \tag{27}$$

where R is the number of total CHs in the network, and $(C_{hr})_f$ is the operating channel of CH r for current frame f.

Spectrum selection in D mode also divides the CHs into four types depending on the conditions of the operating channel and the backup channel. The CH types are:

$$Type(CH)_i = \begin{cases} 1 & \text{if } status(C_i)_{fprev} = (avlb) \text{ and } status(C'_i)_{fprev} = (avlb), \\ 2 & \text{if } status(C_i)_{fprev} = (avlb) \text{ and } status(C'_i)_{fprev} = (obsl), \\ 3 & \text{if } status(C_i)_{fprev} = (obsl) \text{ and } status(C'_i)_{fprev} = (avlb), \\ 4 & \text{if } status(C_i)_{fprev} = (obsl) \text{ and } status(C'_i)_{fprev} = (obsl), \end{cases} \tag{28}$$

for 1 ≤ i ≤ N, $class(S_i)$ = (CH), and operation mode is C mode. The values of $status(C_i)_{fprev}$ and $(C'_i)_{fprev}$ are obtained from (3), (4), (5), and (6).

For S_i with $type(CH)_i$ = 1, because both channels remain available, new channel selection is not needed. That is,

$$[(C_i)_f, (C'_i)_f] = [(C_i)_{fprev}, (C'_i)_{fprev}], \tag{29}$$

for 1 ≤ i ≤ N and class(S_i) = (CH).

For S_i with $type(CH)_i$ = 2, the backup channel becomes empty. Similarly, for S_i with $type(CH)_i$ = 3, the backup channel becomes the operating channel and the backup channel becomes empty. The S_i with $type(CH)_i$ = 2 or $type(CH)_i$ = 3 will send a normal request of C mode to the sink.

$$[(C_i)_f, (C'_i)_f] = [(C_i)_{fprev}, (\phi)], \tag{30}$$

for 1 ≤ i ≤ N, class(S_i) = (CH), and $type(CH)_i$ = 2. (ϕ indicates an empty set.)

$$[(C_i)_f, (C'_i)_f] = [(C'_i)_{fprev}, (\phi)], \tag{31}$$

for 1 ≤ i ≤ N, class(S_i) = (CH), and $type(CH)_i$ = 3.

For S_i with $type(CH)_i$ = 4, because both channels have become obsolete, S_i has nothing else to do but to request an urgent C mode to the sink.

$$[(C_i)_f, (C'_i)_f] = [(\phi), (\phi)]. \tag{32}$$

Two spectrum selection algorithms are provided: random selection (EDSD-R) and game theory-based selection (EDSD-G). One of the spectrum selection algorithms is performed on C mode to select an operating channel and backup channel, as outlined in (22), (24), (25), and (26). In EDSD-R, the operating and backup channels are selected randomly. For example, S_i with $type(CH)_i$ = 1 selects its operating channel and backup channel randomly from the set of clean unlicensed channels, but it must not be the same as the operating channel. Notice that, although the channels are selected randomly, they are selected inside the set of idle licensed channels or clean unlicensed channels. EDSD-R refines (22), (24), (25), and (26) into

$$[(C_i)_f,(C'_i)_f]=\begin{cases}[B_{yu1},B_{yu2}] & \text{if } type(CH)_i=1,\\ [A_{x1},B_{yu1}] & \text{if } type(CH)_i=2,\\ [A_{xv1},A_{xv2}] & \text{if } type(CH)_i=3,\\ [A_{xw1},A_{xw2}] & \text{if } type(CH)_i=4,\end{cases} \tag{33}$$

for 1 ≤ i ≤ N, class(S_i) = (CH), operation mode is C mode, and selection algorithm is EDSD-R, where

$$(B_{yu1},B_{yu2})\in(B_{clean})_i, \tag{34}$$

$$(A_{x1},A_{xv1},A_{xv2})\in(A_{idle})_i, \tag{35}$$

$$(A_{xw1},A_{xw2})\in\{(A_{idle})_i-(A_{excl})_i\}, \tag{36}$$

where the values of ($u1$, $u2$, $v1$, $v2$, $w1$, $w2$) are random values with boundaries of 1 ≤ $u1$, $u2$ ≤ yk, 1 ≤ $v1$, $v2$ ≤ xl, and 1 ≤ $w1$, $w2$ ≤ $h(R-1)$. For values yk, xl, and $h(R-1)$, see (21), (23), and (27).

In EDSD-G, a game theory solution for the spectrum selection problem is proposed, called mixed strategy with lowest payoff elimination (LPE). The payoff is the channel holding time obtained from the Markov chain in the spectrum characterization stage (Section 3.5.1). First, the payoff is sorted in a descending manner, with the top as the highest payoff:

$$\begin{aligned} payoff_i &= \{payoff_x, payoff'_y\}_i \\ &= \{sort((CHT_x)_f), sort((CHT'_y)_f) \mid \\ &(CHT_{x1}\geq CHT_{x2}\geq \ldots \geq CHT_{xl}),(CHT'_{y1}\geq CHT'_{y2}\geq \ldots \geq CHT'_{yk})\}_i, \end{aligned} \tag{37}$$

where $payoff_i$ is the payoff of S_i, $payoff_x$ and $payoff'_y$ are the payoff of licensed channels and unlicensed channels, respectively, observed by S_i. (The variable CHT written without frame information is CHT at the current frame, i.e., CHT_{x1} = $(CHT_{x1})_f$, CHT_{x1} ≠ $(CHT_{x1})_{fprev}$.) Then the lowest-payoff elimination is performed by deleting the channel with payoff lower than half of the maximum payoff:

$$\begin{aligned} cutpayoff_i &= \{cutpayoff_x, cutpayoff'_y\}_i \\ &= \{(CHT_{x1}, CHT_{x2}, \ldots, CHT_{xlpe} \mid CHT_{xlpe} \geq \tfrac{1}{2} CHT_{x1}, xlpe \leq xl), \\ &\quad (CHT'_{y1}, CHT'_{y2}, \ldots, CHT'_{ylpe} \mid CHT'_{ylpe} \geq \tfrac{1}{2} CHT'_{y1}, ylpe \leq yk)\}_i, \end{aligned} \tag{38}$$

where $cutpayoff_i$ is the payoff without the eliminated payoffs, and the payoffs for licensed channels and unlicensed channels are stored in $cutpayoff_x$ and $cutpayoff'_y$, respectively, observed by S_i. The remaining channels on $cutpayoff_i$, are called the candidate channels. Between these candidate channels, an operating channel and a backup channel are selected. As the payoff of a channel is higher, the probability of it getting selected as an operating/backup channel is higher.

S_i with $type(CH)$ = 1, 2, or 4 performs EDSD-G (except for S_i with $type(CH)$ = 3). S_i with $type(CH)$ = 3 has exactly two idle licensed channels; thus, it only needs to compare their CHT values, in which the channel with higher CHT is the operating channel and another is the backup channel. (If the CHT values are the same, then the CH performs a random selection.) For S_i with $type(CH)$ = 1, 2, or 4, channel selection depends on a probability distribution called selection game probability (SGP). The SGP of S_i is $(SGP)_i = \{(SGP)_x, (SGP')_y\}_i$, where $(SGP)_x$ and $(SGP')_y$ are the SGPs for licensed channels and unlicensed channels, respectively.

$$(SGP)_x = \{SGP(A_{x1}), SGP(A_{x2}), \ldots, SGP(A_{xlpe}) \mid SGP(A_{x1}) \geq SGP(A_{x2}) \geq \ldots \geq SGP(A_{xlpe})\}, \tag{39}$$

$$(SGP')_y = \{SGP'(B_{y1}), \ldots, SGP'(B_{ylpe}) \mid SGP'(B_{y1}) \geq SGP(B_{y2}) \geq \ldots \geq SGP'(B_{ylpe})\}, \tag{40}$$

where $SGP(A_{xv})$ and $SGP'(B_{yv})$ are the probability of channel A_{xv} and B_{yv} being selected as the operating or backup channel, respectively (v is arbitrary value). Moreover, $SGP(A_{x1}) + SGP(A_{x2}) + \cdots + SGP(A_{xlpe}) = 1$, as well as $SGP'(B_{y1}) + SGP'(B_{y2}) + \cdots + SGP'(B_{ylpe})$. The formulation of the EDSD-G spectrum selection is similar to that of EDSD-R, as shown in (33). However, the conditions are different, i.e.,

$$[(C_i)_f, (C'_i)_f] = \begin{cases} [B_{yu1}, B_{yu2}] & \text{if } type(CH)_i = 1, \\ [A_{x1}, B_{yu1}] & \text{if } type(CH)_i = 2, \\ [A_{xv1}, A_{xv2}] & \text{if } type(CH)_i = 3, \\ [A_{xv1}, A_{xv2}] & \text{if } type(CH)_i = 4, \end{cases} \tag{41}$$

for $1 \leq i \leq N$, class$(S_i) = (CH)$, operation mode is C mode, and the selection algorithm is EDSD-G, where

$$(B_{yu1}, B_{yu2}) \in (cutpayoff'_y)_i, \tag{42}$$

$$(A_{x1}, A_{xv1}, A_{xv2}) \in (cutpayoff_x)_i, \tag{43}$$

and the selection of (B_{yu1}, B_{yu2}, A_{xv1}, A_{xv2}) follows their respective probabilities $(SGP)_x$ and $(SGP)_y$ of S_i, as in (39) and (40). For S_i with $type(CH)$ = 3, $CHT_{xv1} \geq CHT_{xv2}$.

c. Cluster Member Coordination

After CHs perform spectrum selection (either EDSD-R or EDSD-G), they continue to carry out CM coordination tasks. The CM coordination tasks for C mode and D mode are identical. The CH starts by informing sink and their CMs about the selected operating channel and backup channel via C_{CC} and via the temporary operating channel (see previous section), respectively. The sink collects the operating channel and backup channel information from all CHs and creates an intercluster data transmission schedule (IE-DTS). Essentially, the sink considers the remaining time slots of the current frame and divides the time slots among the number of different operating channels selected by the CHs. (The value of time slots for data transmission is fixed, and it depends on the operation mode.) The sink assigns a disjoint time slot for each different channel; however, the CHs that select the same operating channel are assigned to the same time slot. IE-DTS contains the time slot and channel pairs, which are sent back to the CHs. Meanwhile, the CMs, upon receiving the information about the operating and backup channels, check on the operating channel's availability status on their side. (Notice that the availability requirement is looser than the idle requirement.) If the CMs find out that the assigned operating channel is stated as available, then they send an acknowledgement back to their respective CH on the assigned operating channel. Otherwise the CM goes into sleep state until the end of the frame and sends a normal C-mode request to the sink.

The CH receives IE-DTS from sink on the C_{CC} and receives acknowledgements from its CMs on the operating channel. Upon receiving IE-DTS, each CH extracts its own operating channel and its determined schedule. Then each CH creates its own intracluster data transmission schedule (IA-DTS) and determines the appropriate time slot for data collection activities from its CMs. The data collection activities include environment sensing, data transmission to the CH, and going into sleep state. The IA-DTS contains the time slot and data collection activity pairs, and it is sent to the CMs. The CMs receive IA-DTS from their CHs and set their timers to the scheduled data collection activities. Lastly, the CHs send an acknowledgement to the sink. The tasks in CM coordination are shown in Figure 10.

d. Spectrum Access

Both CHs and CMs perform spectrum access by reconfiguring their transmission power. The CMs reconfigure their transmission power so that minimum power is required in order to send data to their CHs. The CHs reconfigure their transmission power so that minimum power is required in order to send data to the sink.

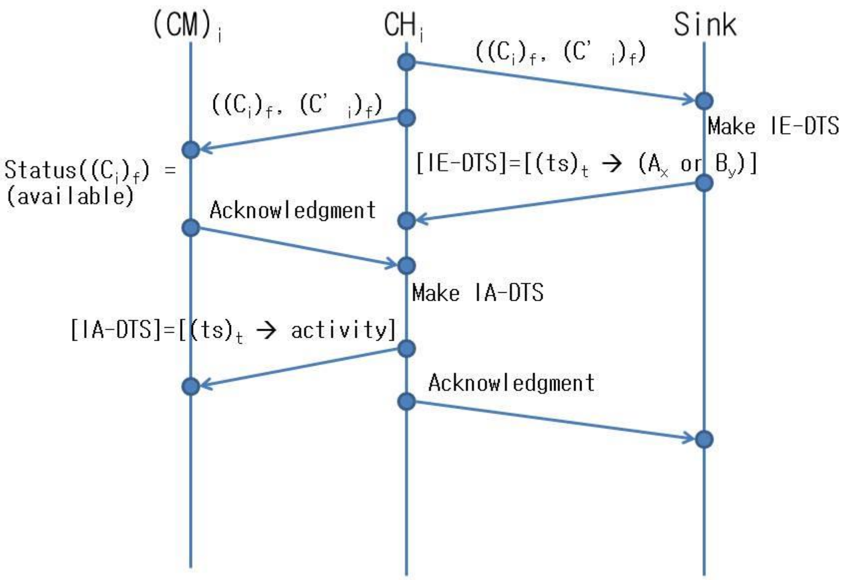


Figure 10. CM coordination tasks ($(ts)_t$ means time slot t).

4. Schedule-based Data Transmission Module

The data transmission in this module is the environment-sensing data collection (not spectrum sensing results). The tasks for the data transmission modules for C mode and D mode are identical. The data transmissions from the CMs to the CHs follow the IA-DTS, whereas the data transmissions from the CHs to the sink follow the IE-DTS. However, after a number of simulations, sometimes the operating channels selected by the CHs were mostly the same, particularly when the idle licensed channels were limited. In that case, the sink received only a few different sets of operating channels from a larger set of CHs. In other words, multiple CHs were selecting the same operating channels. If this kind of configuration is allowed, the sink would end up creating IE-DTS where the intervals of data transmission of each CH were brief, not allowing the CMs to go to sleep state. Therefore, a threshold for a minimum active channel is included. If the set of operating channels reported by the CHs is less than the minimum active channel, then the sink inserts some gaps between data collections.

C. Performance Evaluation

The performance of EDSD-R and EDSD-G is evaluated and the results are compared with MVSD. The tool used is MATLAB with the simulation settings presented in Table 1. The network topology is shown in Figure 11. Two evaluation parameters are selected: network lifetime, defined as the time until half of the sensor nodes are alive; and coordination overhead, defined as number of time-slot spends for coordination divided by the total number of time slots.

Table 1. Simulation Parameters

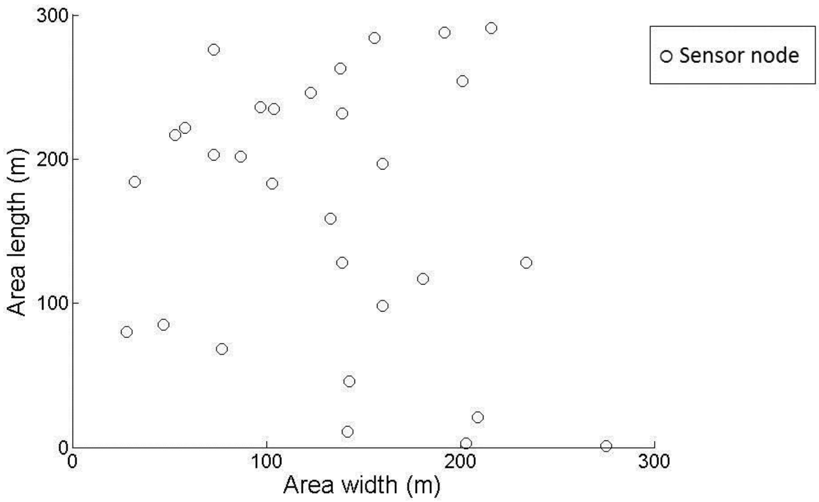
Parameter	Value
Network topology	
Network area	300 m x 300 m

Number of sensor nodes (as SU)	30 nodes
Installation method	Random
Number of sink	1 sink
Location of sink	150 m, 150 m
PU model and channel model	
PU protection range	50 m
PU active probability = passive probability	0.5
PU location mobility = channel mobility	0.5
Number of licensed channel	29 channels
Number of unlicensed channel	29 channels
Maximum noise level	10
Maximum peak interference	14
SNR'_{thres}	5
Common control channel frequency	474MHz
Licensed channel frequencies	482MHz - 546 MHz (bandwidth 8 MHz) 536 MHz - 787 MHz (bandwidth 13MHz)
Unlicensed channel frequencies	ISM 2.4 GHz and 5 GHz
Sensor nodes properties	

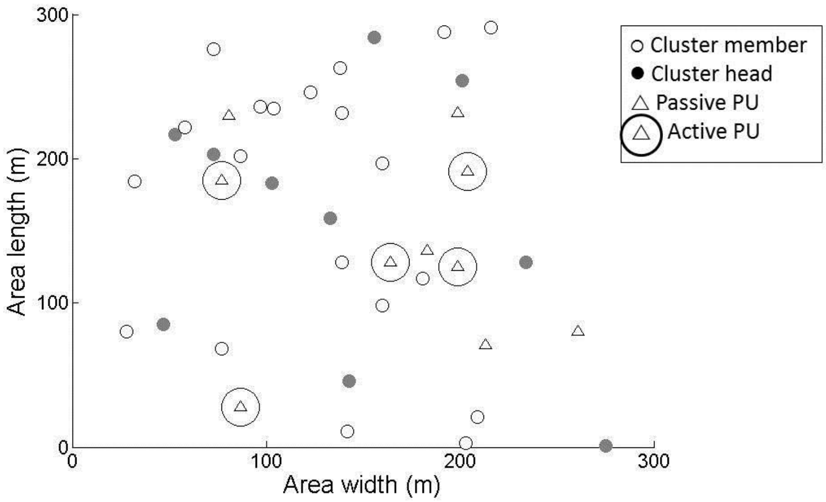
Power supply (2 AA alkaline batteries)	9360J * 2 [78]
Sensing range (environment)	50 m
Transmission range (initial)	100 m
Interference range	150 m
Energy consumption [79]	
Transmit (initial)	459 µJ
Beacon	45.9 µJ
Receive	378 µJ
Active	432 µJ
Idle	172.8 µJ
Sleep	540 nJ
Sensing environment	1031.4 µJ
Partial spectrum sensing	236.9 µJ
Configuration	207.29 µJ
Spectrum switching	296.13 µJ
Residual-energy-based clustering	
Maximum limit of random delay	25 time slots
Residual energy levels	5 levels
Markov chain properties	
$p_{thres} = p'_{thres}$	0.5

CHT_{max} = CHT'_{max}	2
Other settings	
Minimum active channels	5 channels
C mode request threshold	3 normal requests or 1 urgent request

Figure 12 shows the network lifetime results with a varied number of maximum PUs. The lifetimes of EDSD-G and EDSD-R are relatively consistent, whereas the lifetimes of MVSD increase with an increasing maximum number of PUs. These results show the merit of a centralized method where the central entity has global knowledge of the network and is therefore able to optimize network performance even when the number of PUs increases. However, both EDSD-G and EDSD-R outperform MVSD. The reason is that MVSD requires multiple control-packet exchanges to the sink for each frame, while EDSD requires fewer transmissions of control packets, especially in the D mode. On average, EDSD-R and EDSD-G have 25.48% and 19.75% longer lifetimes, respectively, compared with MVSD. The best performance is obtained when the number of PUs is 30.

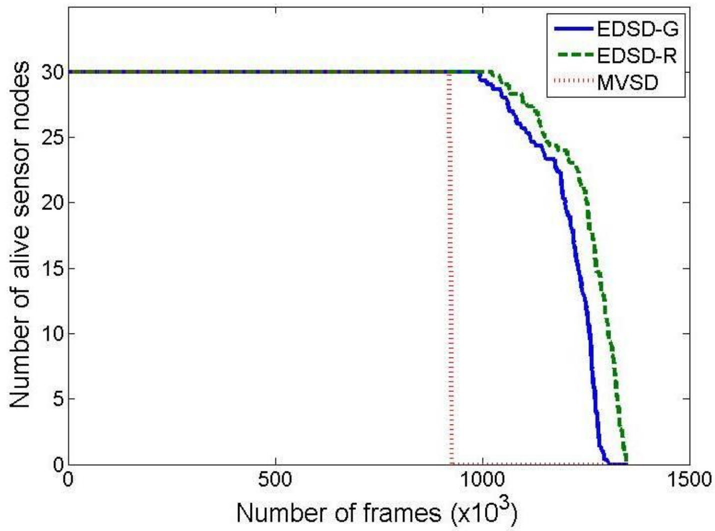


(a)

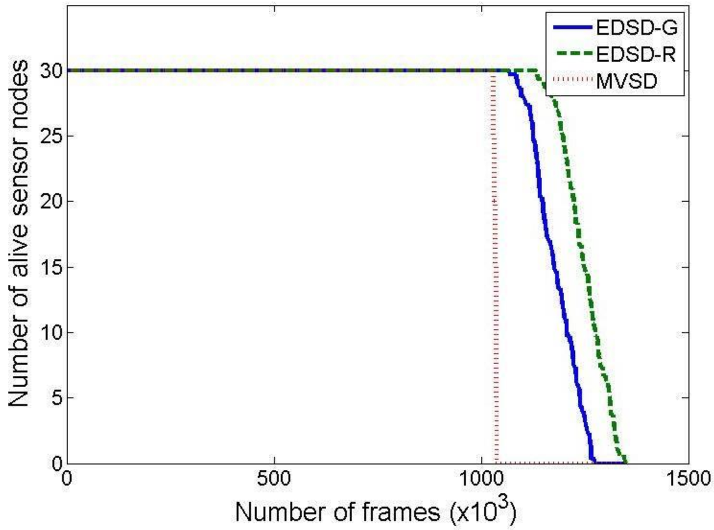


(b)

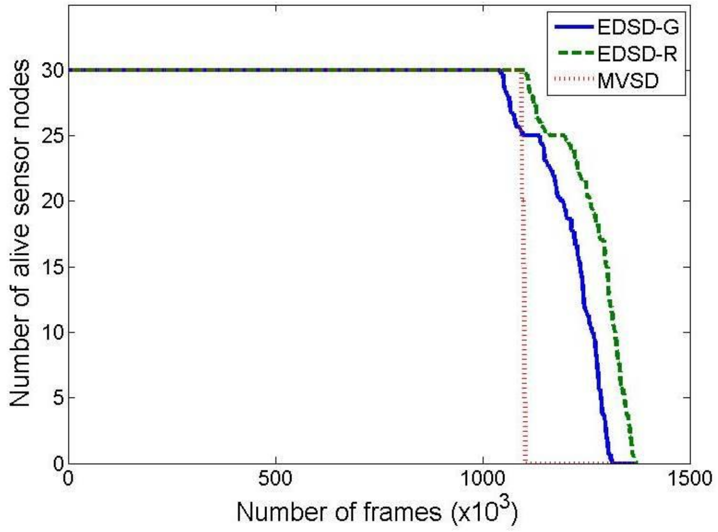
Figure 11. Network topology of (a) sensor nodes only and (b) sensor nodes after clustering and PUs.


EDSD-G
EDSD-R
MVSD
Number of alive sensor nodes
Number of frames (x10^3)
0
5
10
15
20
25
30
0
500
1000
1500

(a)


EDSD-G
EDSD-R
MVSD
Number of alive sensor nodes
Number of frames (x10^3)
0
5
10
15
20
25
30
0
500
1000
1500

(b)

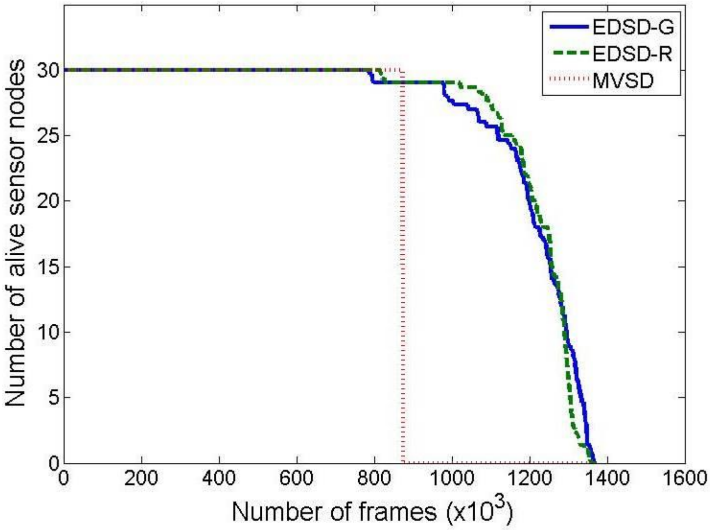


(c)

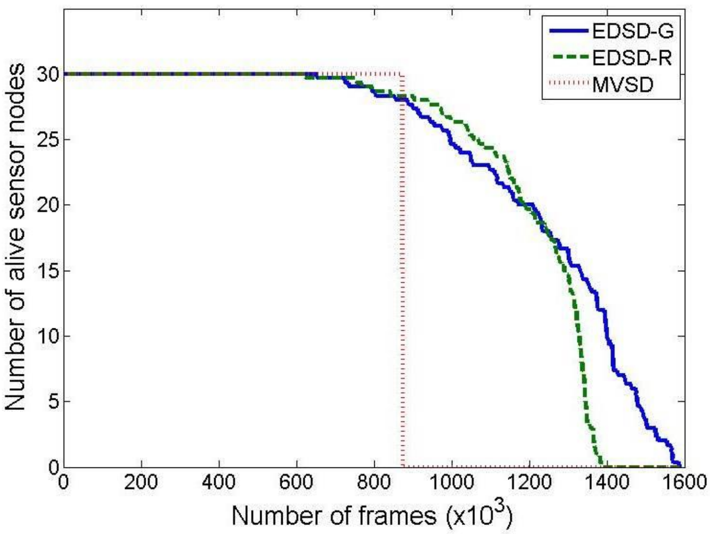
Figure 12. Network lifetime when the maximum number of PUs is (a) 30, (b) 60, and (c) 90, where PUs do not have a favorite channel.

Despite the additional complexities of EDSD-G compared with EDSD-R, the performance of the latter turned out to be better. EDSD-R outperforms EDSD-G by 4.64%, on average. This result occurred because the PUs were also selecting their channels randomly. Random channel selection of PUs renders prediction by a Markov chain less optimal. Therefore, another scenario where the PUs have a certain favorite channel and would likely select it as their operating channel is included. Notice that this selection is not fixed but probabilistic, i.e., the PUs do not always select the favorite channel on each occasion. This assumption is acceptable especially if the PUs are TV viewers, who may have some preferred TV shows during certain times.

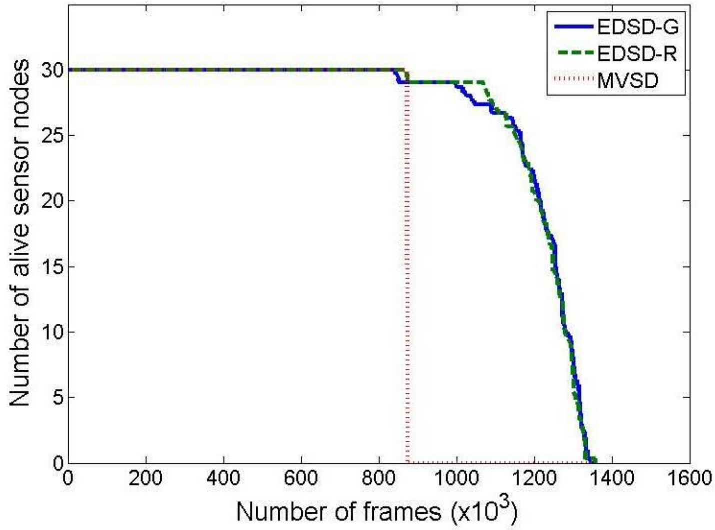
The results of the network lifetime where the PUs have favorite channels are shown in Figure 13. The expectation is that the performance of EDSD-G to be better than the others; however, the simulation results did not fully support the expectation. When the numbers of maximum PU are 30 and 90, the EDSD-G reached almost the same lifetime as EDSD-R, with a minor difference of two and seven frames, respectively. However, when the number of PUs is 60, EDSG-G had a 3.18% longer lifetime compared with EDSD-R. Another observation is that the last sensor node depletes its energy after a longer time (14.19% longer) in EDSD-G than in EDSD-R. Thus, the number of PUs affects the performance of EDSD-G relative to EDSD-R. When the number of PUs is low, the prediction by the Markov chain did not perform optimally because there was not enough data. Nevertheless, when the number of PUs was high, the values on the Markov chain tended to be similar. Thus the prediction also did not perform optimally. However, in this scenario, EDSD-G and EDSD-R also outperform MVSD by 46.62% and 44.86%, respectively.


EDSD-G
EDSD-R
MVSD
Number of alive sensor nodes
Number of frames (x10^3)
0
5
10
15
20
25
30
0
200
400
600
800
1000
1200
1400
1600

(a)


EDSD-G
EDSD-R
MVSD
Number of alive sensor nodes
Number of frames (x10^3)
0
5
10
15
20
25
30
0
200
400
600
800
1000
1200
1400
1600

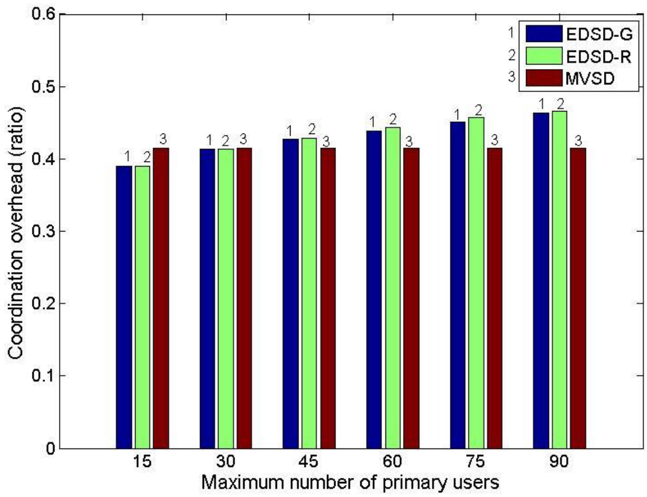
(b)



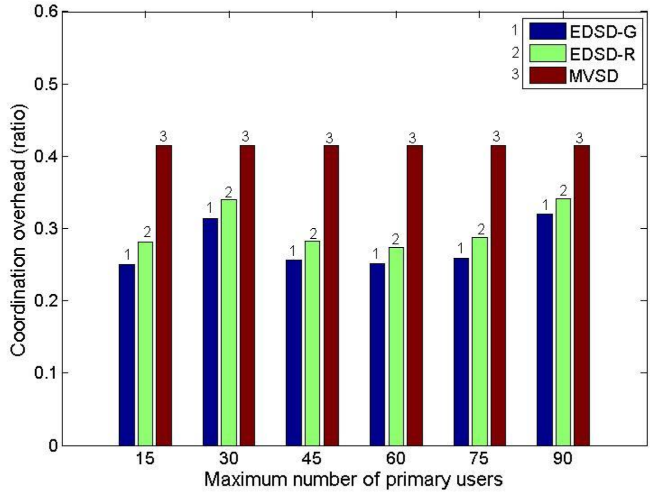
(c)

Figure 13. Network lifetime when the maximum number of PUs is (a) 30, (b) 60, and (c) 90, where PUs have a favorite channel.

The scenario where PUs had a favorite channel is also included in the coordination overhead evaluation. Figure 14 shows the overhead when the numbers of maximum PUs are varied. When PUs have no favorite channel, the overheads of EDSD-G and EDSD-R are similar and they increase as the number of PUs increases. On average, EDSD has higher overhead than MSDV by 4.88%, that is caused by MVSD's centralized approach with fixed overhead. EDSD is a distributed approach where its overhead depends on the network condition and number of PUs. When the number of PUs increases, the probability of the operating channel becoming obsolete also increases, resulting in more frequent C modes. (In C mode, the time spent on coordination is higher than in D mode.) However, when the PUs have a favorite channel, the EDSD-G and EDSD-R outperform MVSD by 31.7% and 26.83%, respectively. The reason for this improvement is that when the PUs have a favorite channel, the probability of an arbitrary operating channel of a sensor node being claimed by the PUs is lower, except that the operating channel used is the favorite channel. Hence, the portion of D mode is higher than that of C mode. Moreover, in this result, a slight improvement of EDSD-G over EDSD-R is observed. Last but not least, there is a case where the distributed method may lead to lower overhead than the centralized method.



(a)



(b)

Figure 14. Coordination overhead when (a) PUs do not have a favorite channel and (b) PUs have a favorite channel.

D. Conclusions

In this chapter, a spectrum decision framework–called an energy-efficient distributed spectrum decision (EDSD) framework–for a CR sensor network is proposed. EDSD framework has two operation modes: coordination mode and data collection mode. The core of EDSD is the spectrum selection algorithms, in which there are random selection (EDSD-R) and game-theory-based selection (EDSD-G). EDSD, with both spectrum selection algorithms, was compared with a minimum-variance-based spectrum decision (MVSD), a centralized spectrum decision framework for general CR networks. The simulation results show that EDSD has a longer lifetime and lower overhead compared with MVSD. In the best scenario, EDSD outperforms MVSD by a 46.62% longer lifetime and 31.7% lower coordination overhead. It is observed that the reason for the improvements is mainly because EDSD is a distributed method that requires fewer control packet exchanges to the sink. Other contributors to the performance improvement are the simple yet energy-aware clustering method, the predictions by a Markov chain (for EDSD-G), and a data collection mode that consumes less energy than the coordination mode. Nevertheless, a weak point in the spectrum selection algorithms is discovered: EDSD-G performs slightly better than EDSD-R, which means that the properties of the Markov chain and/or game theory are not optimized. For future works, the intention is to (1) combine the game theory with other machine-learning techniques to exploit the spectrum usage pattern of the PUs, (2) incorporate real measurements of PU spectrum usage, and (3) consider application-specific sensor placements.

IV. COMPACT CLUSTERING

A. Introduction

Clustering is considered a suitable topology in a CRSN because only the CHs need to perform CR management tasks instead of all the sensor nodes, that reduce the total energy consumption. However, clustering in a CRSN has an additional requirement: the sensor nodes not only have to be in the transmission range of one another but also have to operate in the same communication channel. This limitation might cause a poor cluster formation.

Motivated by providing suitable clustering method for CRSNs, a novel energy-efficient and compact clustering scheme called clustering with temporary support nodes (CENTRE) is designed. CENTRE aims to improve the network performances with the deployment of temporary support nodes. Here, the main features of CENTRE are presented:

- Temporary support node: CH assigns a sensor node as the temporary support (TS) node to support cluster formation. The TS node broadcasts an invitation packet on each channel. Because the CH should stay on its operating channel to accept sensor nodes' registration, it needs the help of the temporary support node to send out the invitation. The invitation packet contains the information about the existence and operating channel of the CH of the TS node. As each sensor node might tune to a different channel, the sensor node cannot possibly discover a CH even though the CH is located within the sensor node's transmission range. Therefore, the role of the TS node is to alert the sensor nodes that have not joined any cluster about the presence of the TS node's CH.

- Two sub-phases of cluster formation: The cluster formation process consists of two sub-phases: CH discovery and cluster member invitation. In CRSNs, it is difficult for sensor nodes to find a CH. The two sub-phases enable the sensor nodes to find a CH effectively.
- Partial spectrum sensing: The sensor nodes do not carry out the spectrum sensing process on all the channels but only on some part of the channels to conserve energy and time. This is not a usual approach because the sensor nodes are usually required to perform full spectrum sensing or even cooperative spectrum sensing [80, 81]. In CENTRE, the sensor nodes intentionally perform partial spectrum sensing to save energy because a CRSN is an energy-constrained network.
- Communication frequency selection: Intra-cluster transmissions are assigned a high frequency and the inter-cluster transmissions are assigned a low frequency. High frequency channels support higher data transmission rates but shorter transmission ranges, thus, they are suitable for intra-cluster data transmission. Low frequency channels support longer transmission ranges and consume less energy and, thus, they are suitable for one-hop data collections from CHs to the sink.

The principal contribution of CENTRE is the development of a clustering method that suits CRSNs with low energy consumption. Besides energy conservation, the CENTRE is shown to have low clustering overhead and short distance between CHs and their members. Another contribution is that a novel approach of clustering is developed by introducing the concept of temporary support node. With the help of temporary support node, the CENTRE is able to perform well under CRSNs environment.

The deployment of sensor nodes is random, dense, and redundant. This assumption implies that a number of sensor nodes might be excluded (put to sleep) during the data collection activity, without affecting the WSN's sensing coverage functionality. Each sensor node is equipped with one CR transmitter. When a sensor node (of any class) simultaneously receives more than one packet, it receives one packet successfully while discarding the others. The medium access control protocol is based on time division multiple access (TDMA). The CRSN application requires periodic data collection and there is at least one reserved, low-frequency common control channel between the sink and the cluster heads. However, there is no common control channel among the high-frequency channels.

The CRSN is deployed in a remote location where no PU is present. Even though there is no PU, the network could be considered a CRSN because the sensor nodes are equipped with CR capabilities such as dynamic spectrum access and transmission parameters reconfigurability. Wireless networks employing CRs should consider the interference to the PUs. However, in this section, the main objective is to introduce the novel approach for clustering that involves a particular cluster member to be a temporary support node and to evaluate its performance. Even though PUs are not included, the underlying network still can be categorized as a CRSN because it is a WSN and the sensor nodes are equipped with CR capability, according to the definition of CR adopted in this thesis.

B. Clustering with Temporary Support Nodes (CENTRE)

CENTRE is performed in rounds where a round consists of: the cluster formation phase and the data transmission phase. The cluster formation phase consists of two sub-phases: CH discovery (henceforth called sub-phase 1) and cluster member invitation (henceforth called sub-phase 2). The durations of each phase and sub-phase are fixed and predetermined. In sub-phase 1, the sensor nodes search for the CH. In a CRSN, however, the sensor nodes might not be able to locate a CH even though the CH is inside the transmission range because the sensor nodes and the CH use different channels. This condition is anticipated in sub-phase 2 when each CH actively search for sensor nodes that can become its members, with the help of a TS node. Figure 15 illustrates the aim of cluster formation sub-phases 1 and 2. The procedures of the CENTRE rounds will be shown in Figure 16.

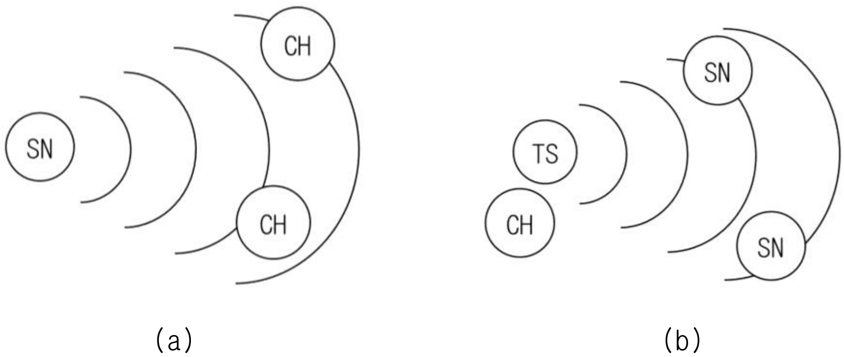


Figure 15. Cluster formation phase. (a) Sub-phase 1: CH discovery, (b) Sub-phase 2: cluster member invitation.

After the completion of the cluster formation phase, the clustering process is completed with one CH in each cluster. The data transmission phase that follows involves multiple pairs of intra-cluster and inter-cluster data transmission. During the intra-cluster data collection, the cluster members send the sensed data (related not to spectrum sensing, but to application-related environment sensing) to their CHs using one of the high-frequency channels agreed upon with the CH. During the inter-cluster data transmission, the CHs send the data to the sink by one-hop transmission using one of the low-frequency channels assigned by the sink. The sensor nodes are divided into four classes: sensor node (SN), cluster head (CH), cluster member (CM), and temporary support (TS). Class SN, CH, and CM are similar with those of EDSD’s (Chapter III. B).

For the class TS, during cluster formation sub-phase 2, a number of CMs might be selected as TSs. When the tasks of the TSs are completed, they return to function as CMs. However, at the end of sub-phase 2, some sensor nodes might still be SNs; in other words, they do not belong to any cluster. These SNs will be unable to participate in the following data transmissions. However, by assuming a dense and redundant deployment of the sensor nodes in the network, the sensing coverage is expected to be tolerable. The five major activities in CENTRE are given below. These activities are performed regularly during the CENTRE rounds, as illustrated in Figure 16.

- Partial spectrum sensing and CH discovery: The sensor nodes carry out the spectrum sensing process on some of the channels to find CHs.
- CH declaration: The sensor nodes declare themselves as CHs after they fail to discover any CH on their current operating channels by following a predetermined probability.
- Registration to a CH: The sensor nodes that find a CH proceed to join the cluster.
- TS node assignment and cluster member invitation: The CHs might assign the closest cluster member as a support node temporarily. The TS node broadcasts invitation packets on each channel.
- Data transmission: Data transmission includes both intra-cluster and inter-cluster data transmissions.

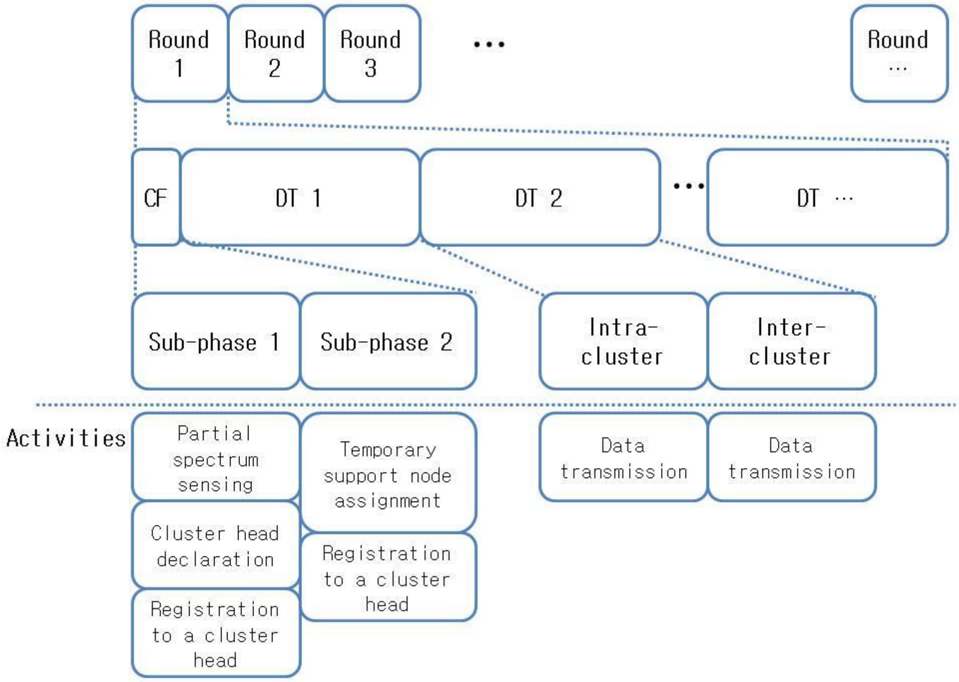


Figure 16. The procedures in CENTRE rounds and the activities performed during the rounds. (CF: cluster formation, DT: data transmission).

1. Cluster Formation Phase

The cluster formation phase aims to build an optimal cluster topology in a distributive manner. As mentioned earlier, this phase consists of two sub-phases: CH discovery (sub-phase 1) and cluster member invitation (sub-phase 2). In sub-phase 1, the sensor nodes search for CHs, whereas, in sub-phase 2, the CHs search for new cluster members, with the help of TS nodes. Figure 17 shows the flow chart of the cluster formation phase.

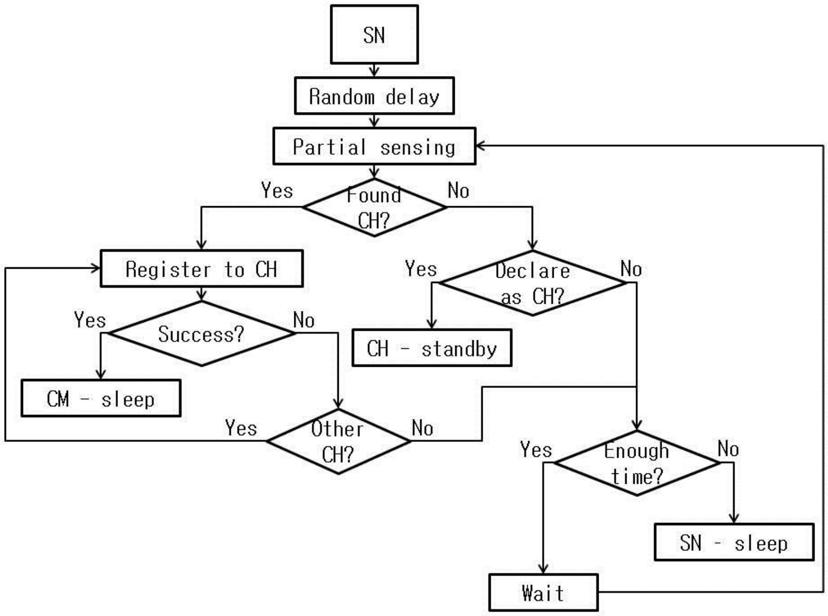


Figure 17. Procedural flow of the cluster formation phase.

Three activities take place in sub-phase 1: (1) partial spectrum sensing and CH discovery, (2) registration of sensor nodes to a CH, and (3) CH declaration. The first step in cluster formation is partial spectrum sensing, in which the sensor nodes perform spectrum sensing on a part of the entire spectrum. The purpose of partial spectrum sensing is to save energy and time, because sensing on the entire spectrum requires considerably higher processing tasks and time, but the sensor nodes have only limited resources. The sensor nodes start partial spectrum sensing after a random delay. The random delay is applied to increase the probability of discovering a CH. If all the sensor nodes start spectrum sensing at the same initiation time, then no CH would be found because all the sensor nodes are performing spectrum sensing. If a sensor node waits, then by the time it starts spectrum sensing, there is a possibility of it finding a CH as some sensor nodes might have declared themselves as CHs earlier. However, the random delay is kept short to minimize the cluster formation duration.

Each sensor node keeps a list of CHs that it has found during partial spectrum sensing and it registers to the first-listed CH. No additional computation is performed to select among CHs from the list to reduce energy consumption. The packet exchanges that take place between a sensor node and the CHs during the registration period are shown in Figure 18.

The sensor node sends a join packet to the CH first on the list and sets its timer. The join packet may fail to reach the CH or collide with other packets at the CH. Hence, if the timer expires but the sensor node has not received a response packet, it sends a join packet to the same CH one more time. When the second join packet also fails to elicit a response from the first CH, the sensor node sends a join packet to the next CH on the list. The CH that receives the join packet sends back a response packet to the sensor node. Here, it is assumed that when a CH receives many join packets simultaneously, one join packet is received successfully and the other join packets are dropped. Moreover, the CH does not send any notification to the sending nodes about the dropped join packets.

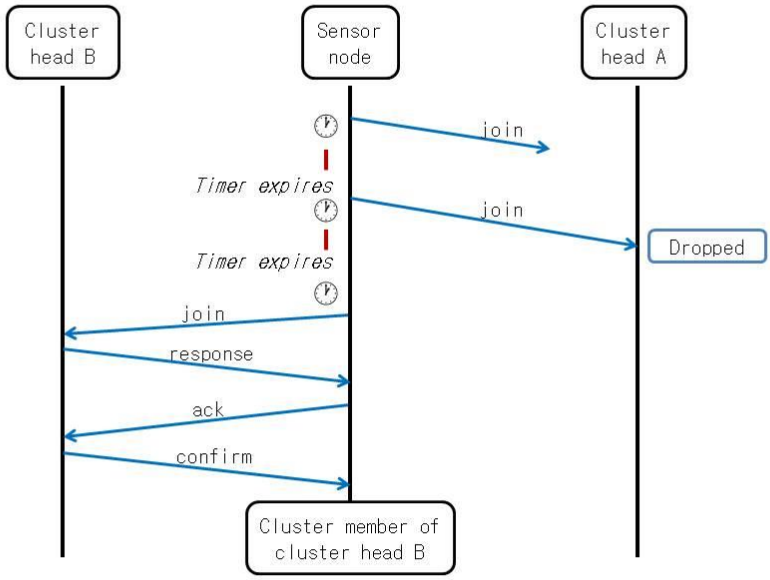


Figure 18. Procedure for registration to a CH.

Once the sensor node receives a response packet from the CH, it calculates the distance and required transmission power based on the received power level. The sensor node reconfigures its transmission power level to the minimum required power level (reconfiguration is enabled by CR) and sends an acknowledgement packet to the CH at the new transmission power level. The acknowledgement packet also contains the distance information. The purpose of the acknowledgement packet is to ensure that the CH can successfully receive and decode a packet transmitted at the new transmission power level. The distance information is used during sub-phase 2. When the CH receives the acknowledgement packet, it replies with a confirmation packet, sets the sensor node as its cluster member, and records its distance information. Similarly, the sensor node sets itself as the cluster member of the respective CH after it receives the confirmation packet. Then, the cluster member goes to the sleep state until the end of sub-phase 1.

In case the registration fails and there is no CH on the list, the sensor node checks the remaining time. If the remaining time is sufficient to support another round of partial spectrum sensing and registration trial, then the sensor node waits for a random delay period and repeats partial spectrum sensing. If the remaining time is insufficient, then the sensor node goes to the sleep state until the end of sub-phase 1.

If no CH is found at the end of partial spectrum sensing, then the sensor node declares itself as a CH with a certain predetermined probability. When the sensor node becomes a CH, it beacons about its presence periodically on its operating channel and waits for registration requests. When the sensor node fails to become a CH, it again checks the remaining time. If the time is sufficient, it waits and repeats partial spectrum sensing; otherwise, it goes to the sleep state until the end of sub-phase 1.

Sub-phase 2 starts with the assignment of TS nodes. Each CH computes the number of cluster members. If the CH does not have any member, then it becomes a sensor node. If the number of members in the cluster is less than the predefined threshold, then the CH assigns the closest cluster member as a TS node; otherwise, the CH and its members go to the sleep state until the end of sub-phase 2. As the CH has recorded the distance information between itself and each of its cluster members during the registration procedure, it easily decides the closest cluster member. The CH then sends a TS node assignment packet to that cluster member and stays ready to process registration requests from sensor nodes until the end of sub-phase 2.

The consideration to assign the closest cluster member as the temporary support node is as follow: As the closest cluster member/temporary support node broadcasts an invitation, the sensor nodes within its transmission range could receive it. However, these sensor nodes need to transmit their registration packet to the CH, not to the temporary support node. Because the temporary support node is the closest node to the CH, if a sensor node can receive a packet form a temporary support node, then it is highly probable that it can send a packet successfully to the CH.

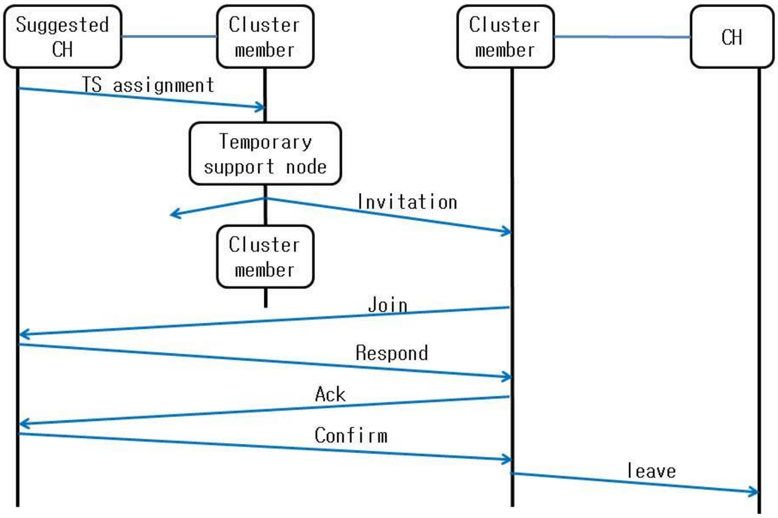
Each cluster member wakes up from the sleep state and waits for the TS node assignment packet from the CH. If the cluster member receives a TS node packet, then it becomes a TS node. Otherwise, the cluster member recalls its distance to the CH. The cluster members that are relatively closer to the CH go to the sleep state until the end of sub-phase 2, whereas the cluster members that are farther from the CH stay in the active state.

The TS node sets its transmission power level to the default setting (maximum) and broadcasts an invitation packet on each available channel. The invitation contains the address and operating channel of the TS node’s CH.

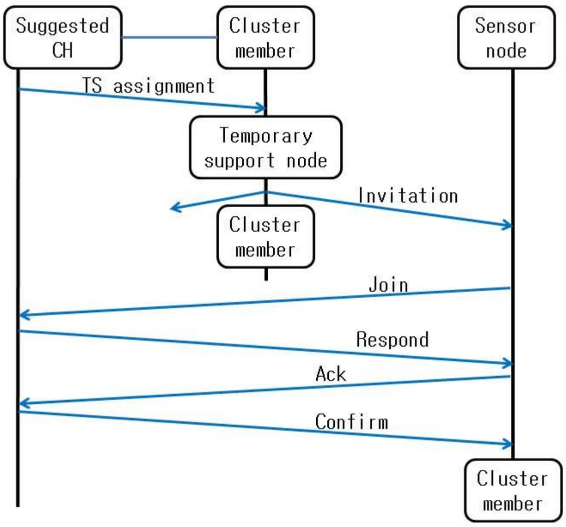
After the TS node finishes the broadcasts, it returns to its original class (i.e., becomes a cluster member) and goes to sleep until the end of sub-phase 2. The receivers of the invitation packets are active cluster members and sensor nodes. The CH of the TS node is called the suggested CH. The receivers of the invitation packets record up to two invitation packets. The exchange of packets during sub-phase 2 is shown in Figure 19.

In case that the receiver is a CM, it computes the distance to the TS node based on the received power level. The cluster member considers this distance as the distance to the suggested CH because the TS node is the closest node to the suggested CH. The cluster member compares the distance to the suggested CH with the distance to the current CH. If the suggested CH is closer than the current CH, then the cluster member tries to register to the suggested CH by following the registration procedures. When the registration to the suggested CH is approved, the cluster member joins the cluster of the suggested CH and sends a leave packet to the previous CH to inform that it has left that cluster. Otherwise, the cluster member stays with the current CH and goes to the sleep state until the end of sub-phase 2.

The sensor node who receives an invitation packet immediately tries to register to the suggested CH. If the registration is approved, then the sensor node becomes a cluster member and goes to the sleep state until the end of sub-phase 2. If the registration is unsuccessful, then the sensor node goes to sleep until the end of the current round.



(a)



(b)

Figure 19. Packet exchanges during sub-phase 2 when the receivers of the invitation packets are (a) active cluster members and (b) sensor nodes.

2. Data transmission Phase

The data transmission phase consists of pairs of intra-cluster and inter-cluster data transmission repeated multiple times. In intra-cluster data transmission, each cluster member sends its sensed data to the CH using a high-frequency channel, whereas in inter-cluster data transmission, each CH sends the aggregated data to the sink using a low-frequency channel. By transmitting in low-frequency channel, the CHs are able to transmit data to the sink in one-hop transmission.

At the beginning of the data transmission phase, the sink monitors the number of available channels on the low-frequency channels and decides which channels are to be used. The sink creates an inter-cluster schedule, includes the channel information on the schedule packet, and broadcasts the schedule on the low-frequency common control channel. The CHs synchronize with each other by receiving the inter-cluster schedule from the sink. After synchronization, the CHs switch back to their operating channel, configure the intra-cluster schedule, and broadcast this schedule to their cluster members. The intra-cluster schedule defines the time when a cluster member should report its sensed data to the CH. The CHs collect the sensed data from the cluster members, aggregate them, and send them to the sink by following its schedule. The inter-cluster and intra-cluster schedules are not updated and are kept unchanged until the end of the round. The packet exchanges during the data transmission phase are shown in Figure 20.

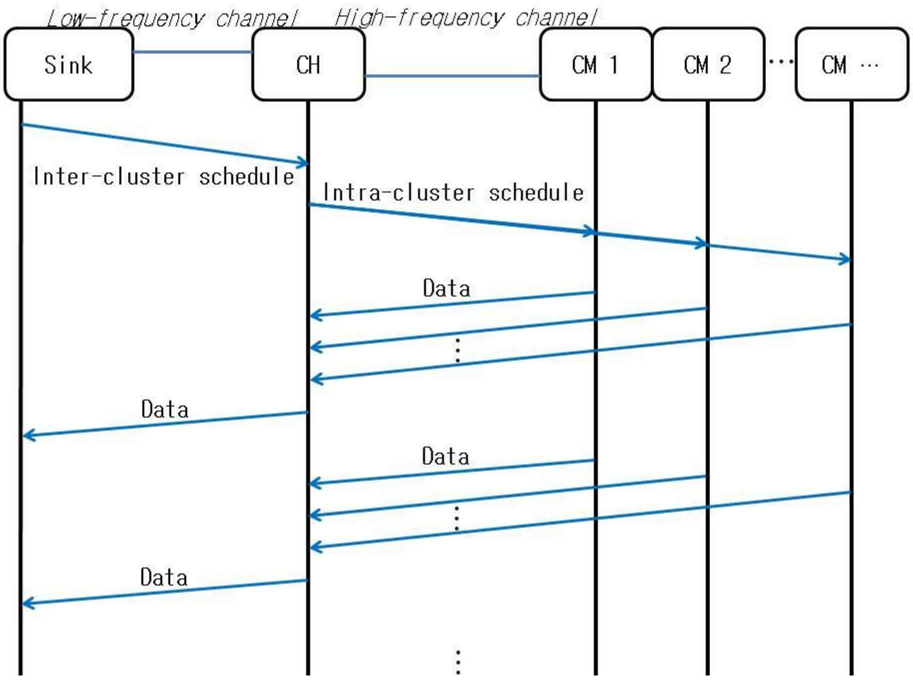


Figure 20. Packet exchanges during the data transmission phase illustrated with one CH only.

C. Performance Evaluation

CENTRE is evaluated by computer simulation using MATLAB and compared with a distributed spectrum-aware clustering (DSAC) scheme. DSAC is selected for comparison because it focuses on clustering in general-purpose CRSNs as in CENTRE. CENTRE is implemented based on time slots, where one time slot equals 18 ms. Hence, DSAC is adjusted to enable simulation based on time slots so that its performance can be compared with that of CENTRE (the adjustment did not alter the main principles of DSAC). Moreover, CENRE is implemented based on rounds, where one round consist of cluster formation and data transmission. For CENTRE, a round equals to one time cluster formation and 1000 times data transmission or 42100 time slots. Again, DSAC is adjusted to this time framing. To simulate one round, simulations were repeated 1000 times for both intra-cluster and inter-cluster transmissions. The simulation settings are presented in Table 2 (settings for power supply and energy consumptions are the same with Table 1).

Table 2. Simulation Parameters

Network topology	
Number of sensor nodes	120 nodes
Network area	600 m × 600 m
Sensor node’ s transmission range	300 m
Sensor nodes deployment	Random
Clustering setting	Probability of a sensor node becoming a CH (for CENTRE): 5% Optimal number of clusters (for DSAC): 5 clusters
Communication frequency	
Frequency	Intra-cluster: IEEE 802.11 2.4 GHz

	Inter-cluster: IEEE 802.22 TV band
Bandwidth	Intra-cluster: 22 MHz Inter-cluster: 6 MHz
Number of channels	Intra-cluster: 3 channels (any three non-overlapping channels) Inter-cluster: 1 channel or more (determined by the sink)
Partial spectrum sensing width (for CENTRE)	1 channel
Timing	
CENTRE cluster formation time	50 time slots (sub-phase 1) and 50 time slots (sub-phase 2)
CENTRE maximum delay (sub-phase 1)	25 time slots

Four network performance parameters are analyzed: network lifetime, energy consumption per round, normalized clustering overhead, and average clustering distance. In CRSNs, the network lifetime is the utmost importance parameter because there is no constant supply of energy. Energy consumption per round is also analyzed to evaluate the energy consumption trend, in which low and stable energy consumption is desired. Because a clustering method is proposed, the measure of its effectiveness by measuring the normalized clustering overhead and average distance between CHs and the cluster members is necessary. To achieve superior performance, the clustering method should have not only low clustering overhead to reduce the energy consumption during cluster formation but also compact clustering (short distance between the CHs and the cluster members) to reduce the energy consumption in intra-cluster data transmission.

The performance parameters are measured at the end of each round. However, as DSAC requires iteration to reach the optimal number of clusters, its cluster formation duration is varied per round. Therefore, the performance parameters of both CENTRE and DSAC are evaluated based on the CENTRE's round duration. The network topology in which sensor nodes are randomly deployed is shown in Figure 21.

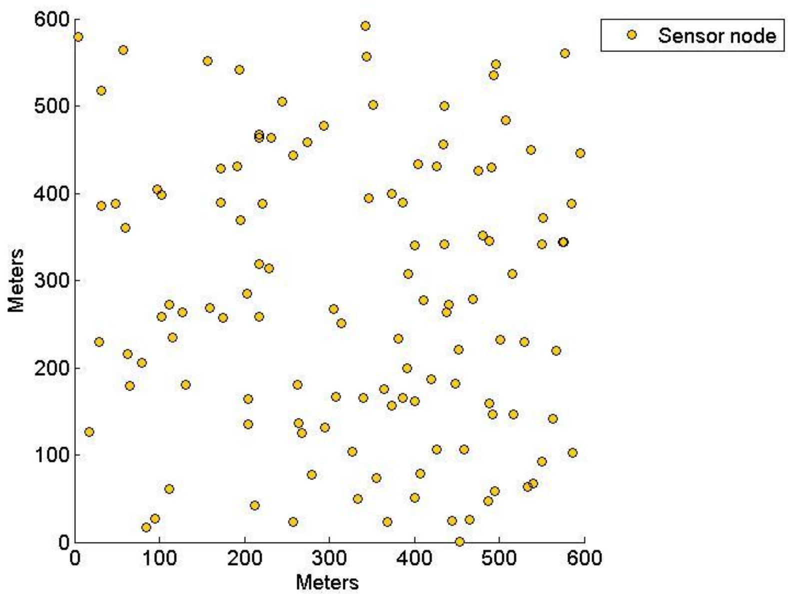


Figure 21. The CRSN topology under evaluation.

Figure. 22 shows the number of sensor nodes that are alive per round. Because the sensor nodes are randomly and redundantly deployed, the network lifetime is defined as the time period during which more than half of the sensor nodes are alive. In other words, the CRSN and its application are considered no longer active when more than half of the sensor nodes have depleted their energy. Because the number of sensor nodes in the simulation is 120, the network lifetime is the time duration from the initial network configuration to the death of the 61st node. As shown in Figure 22, the network lifetime of CENTRE is longer than that of DSAC. The main reasons for this improvement are that CENTRE does not require multiple beacon broadcasts for the nodes/clusters merging iteration during the cluster formation and CENTRE has short, fixed-duration cluster formation. Another reason is that CENTRE enables the adjustment of transmission power to reduce energy consumption. The network lifetime of CENTRE is 34.2% longer than that of DSAC. However, CENTRE has a minor drawback; only a very few number of sensor nodes are alive for a long time. This is because CENTRE is a distributed clustering protocol without any local information exchange and, thus, the sensor nodes are not aware of the condition of other sensor nodes. Therefore, the sensor nodes would simply follow the protocol and go to the sleep mode even though the network is no longer active. In real time units, the lifetime of CENTRE is around 74 days (8441 rounds) whereas the lifetime of DSAC is around 55 days (6289 rounds).

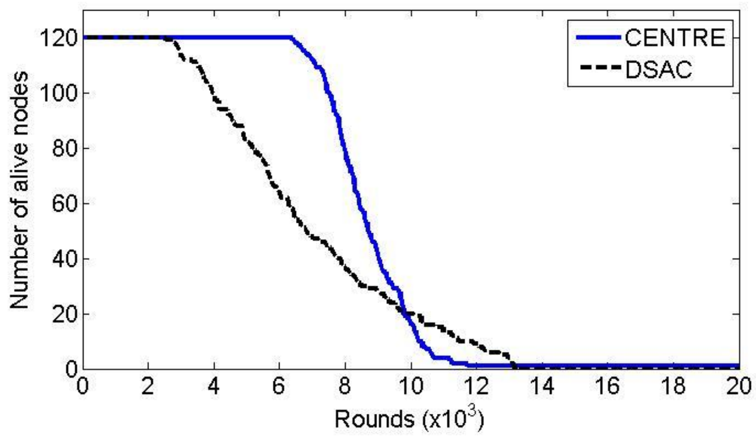
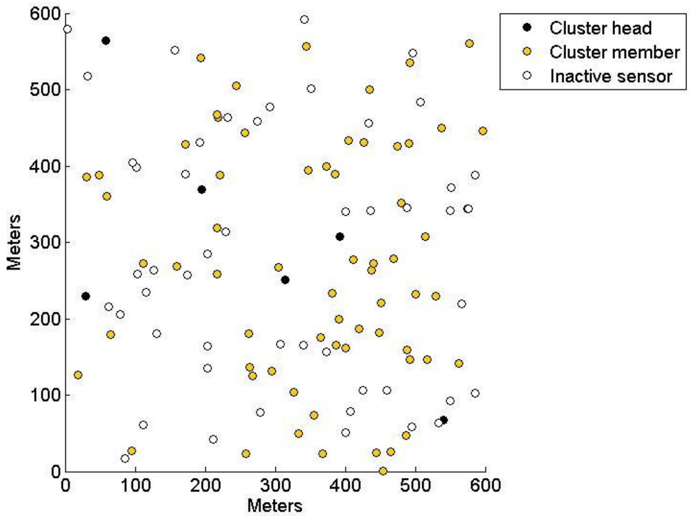
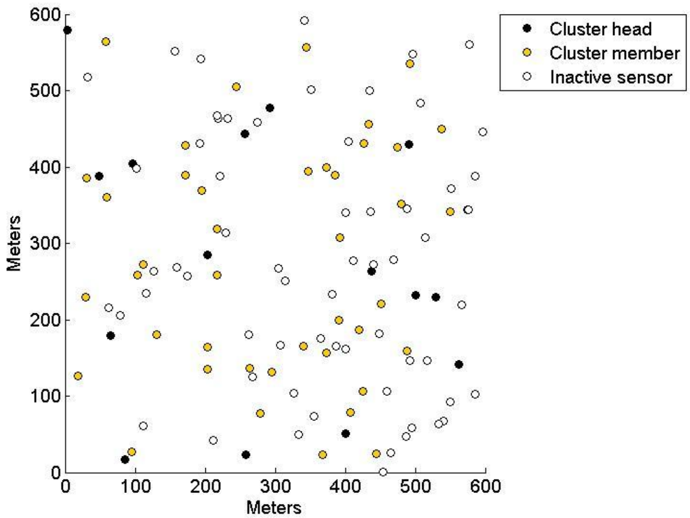


Figure 22. Network lifetime.

To further observe the energy consumption and network lifetime, the snapshots of network composition when half of the sensor nodes are alive are shown in Figure 23. The number of CHs in DSAC is far higher than that in CENTRE. In DSAC, the number of CHs is predetermined and the sensor nodes only have the knowledge of their surroundings (not the global knowledge) by receiving the beacons that their neighboring nodes transmit. In CENTRE, the desired number of clusters is implemented in each sensor node as the probability of becoming a CH. Therefore, the number of clusters is approximately the same with the predetermined setting, which is 5% of remaining nodes in the simulation.



(a)



(b)

Figure 23. Network composition when half of the sensor nodes are alive for (a) CENTRE and (b) DSAC.

Figure 24 shows the energy consumption per round for both CENTRE and DSAC schemes. The energy consumption of CENTRE is relatively constant mainly because of the fixed cluster formation duration. Moreover, the organized sleep mode protocol of CENTRE helps in balancing the energy consumption among the nodes. Until approximately round 2500 (almost half of its lifetime), DSAC consumes 52% more energy than CENTRE. This is mainly due to the fact that, in DSAC, the cluster formation protocol is based on iterations and multiple beacons are required. When the number of active sensor nodes decreases, the energy consumption of DSAC also decreases. After round 4000, DSAC has lower energy consumption compared to CENTRE. This is because the energy consumption of DSAC is highly dependent on the number of sensor nodes, especially during the cluster formation. On the other hand, the energy consumption of CENTRE is stable throughout the network lifetime. During the network lifetime, on average, the energy consumption per round for CENTRE (185.53 Joule) is 21.1% less than that for DSAC (235.16 Joule).

To further evaluate the clustering methods, the normalized clustering overhead and the average distance between the CHs and their cluster members are measured. The normalized clustering overhead is defined as the clustering time divided by the data transmission time per round. A small ratio of overhead is desired because it means that the time spent in creating the cluster topology is negligible compared to the actual data transmission time. Figure 25 shows that the normalized clustering overhead of CENTRE is much lower than that of DSAC. On average, the normalized clustering overhead of CENTRE is 0.0024 but that of DSAC is 0.0306 which is 12.8 times higher compared to CENTRE. DSAC has much higher clustering overhead because, again, it is based on iterations requiring higher packets exchanges (beacons). Both schemes have relatively low normalized clustering overheads (less than 0.035). The duration of cluster formation in CENTRE is fixed, which is 100 time slots or equal to 1.8 seconds. By analyzing the normalized clustering overhead, the required time for clustering in DSAC could also be obtained. Given the normalized clustering overhead and the data transmission time, the average duration of cluster formation in DSAC can be calculated, which is 1250 time slots or 22.5 seconds according to the definition of the normalized clustering overhead.

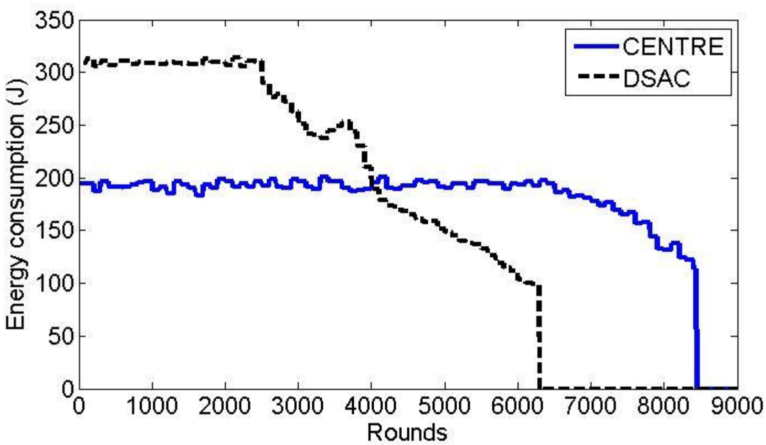


Figure 24. Energy consumption per round (showed until the lifetime of the two schemes).

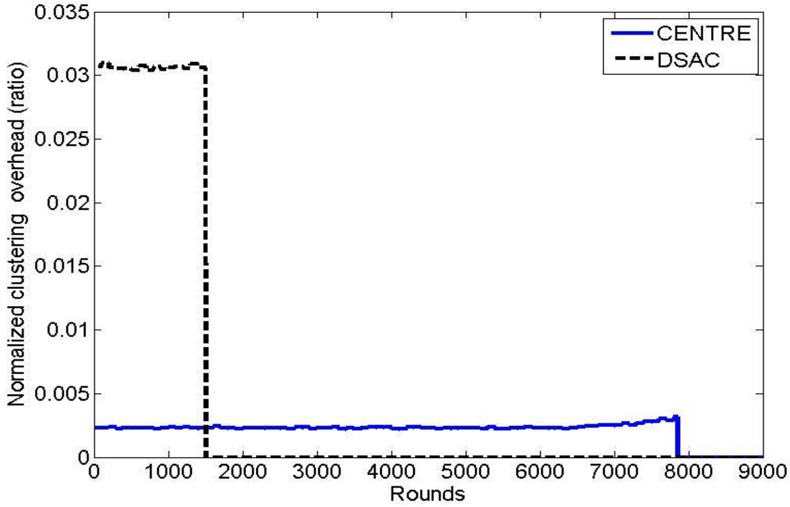


Figure 25. Normalized clustering overhead per round (showed until the lifetime of the two schemes).

Figure 26 shows the average distance between the CHs and their cluster members. The transmission range of sensor nodes is 300 m. A smaller average distance indicates better cluster formation because it means that each sensor node joins the closest CH. Here, CENTRE outperforms DSAC by having about 10% lower average distance. Even though the improvement is only 10%, it is an interesting result because CENTRE performs better than DSAC even though DSAC merges two closest sensor nodes/clusters into a cluster. This result confirms the efficiency of CENTRE's TS nodes that invite sensor nodes within the transmission range and allow cluster members to switch to another CH that is closer.

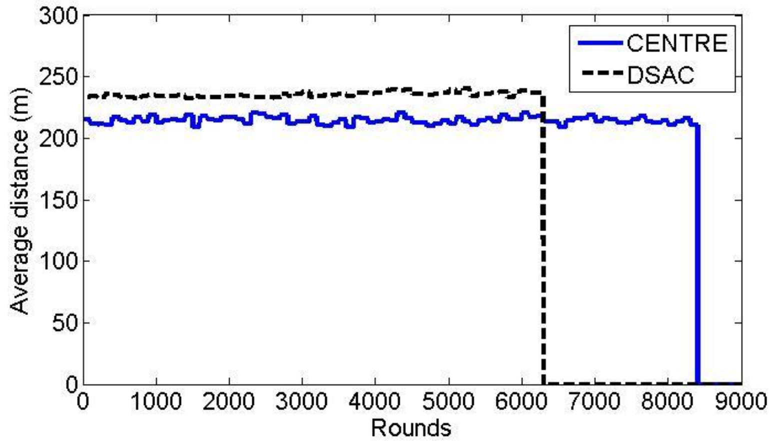


Figure 26. Average distance between the CHs and their cluster members (showed until the lifetime of the two schemes).

The simulation results have shown the superiority of CENTRE against DSAC in terms of network lifetime, energy consumption, clustering overhead, and distance between the CHs and their members. These performance improvements are due to the properties of CENTRE, which are: (1) it does not require multiple beacon/data broadcasts during the cluster formation, (2) it has fixed cluster formation duration and it does not depend on multiple iterations, (3) it enables transmission power adjustment, (4) it has organized sleep mode, and (5) it adopts temporary support nodes which results in compact clustering. These properties make CENTRE an effective and efficient clustering method for CRSNs. In practice, the expectation is that CENTRE would perform the best under the condition of dense and random deployment of sensor nodes and the sensing applications that require regular data collection. However, CENTRE would need a common control channel.

D. Conclusions

A novel energy-efficient and compact clustering scheme called CENTRE with temporary support nodes is designed for CRSNs. The CENTRE's cluster formation process has two sub-phases: CH discovery and cluster member invitation. Even though it is difficult for sensor nodes to find a CH in CRSNs, the two sub-phases enable the sensor nodes to find a CH efficiently. CENTRE also decreases the average distance between CHs and their members, resulting in compact clustering. In addition, adopting a fixed duration for cluster formation results in remarkable energy saving. The performance evaluation shows that CENTRE achieves 34% longer network lifetime with less clustering overhead. The average distance between of CHs and their cluster members is also decreased. The main reasons for the performance improvement of the CENTRE scheme include the following: the fixed cluster formation duration, the adjustment of the transmission power of the cluster members based on the distance to the CH, the refinement of the cluster formation process by the use of temporary support nodes, and the use of the sleep mode when the sensors are not active.

V. ROBUST TRANSPORT PROTOCOL

A. Introduction

Transport protocol is a crucial part of a CRSN because it provides reliability. However, there is no transport protocol designed for CRSNs in the literature. A transport layer protocol called the robust and energy-efficient transport protocol (RETP) is designed for CRSNs. The motivation for developing RETP is to improve the network lifetime of CRSNs while achieving low event-detection delay without any degradation of reliability level. The main features of RETP are:

- There are two operation modes: management mode and data collection mode. In the management mode, sensor nodes perform spectrum management in addition to data collection whereas in data collection mode, sensor nodes perform only data collection. Spectrum management activities include spectrum sensing and spectrum decision, which are enabled by CR. The spectrum sensing method is not specified; however, RETP is compatible with general spectrum sensing methods for CR networks [82, 83] or even cooperative spectrum sensing.
- Every sensor node determines its operating channel distributively and sends data to the sink according to a specified schedule. This feature

provides more accurate spectrum sensing and spectrum decision, as well as saves energy by adopting a coordinated duty cycle.

- The interchange of ACK packets and negative ACK (NACK) packets depends on the sensor data type. This feature ensures the reliability of delay-sensitive data transmission.
- The sensor nodes collect the data and send them to the sink regularly. This method is suitable for CRSNs applications that require regular data collection, but it can also be applied to event-based CRSNs.

The main contribution of RETP is the provision of a transport protocol with high energy-efficiency that leads to prolonged network lifetime while simultaneously preserves event-detection reliability in CRSNs. The performance study shows that the RETP not only prolongs network lifetime significantly but also decreases event-detection delay while preserving event-detection reliability compared with the conventional protocol. The underlying CRSN is modeled as follows:

- Each sensor node is equipped with a CR transmitter.
- Sensor nodes are installed manually following a predetermined topology (not random) and the sink is aware of the location of each sensor node.
- Each sensor node can reach the sink in one hop (by reconfiguring its transmission parameters).
- There is one dedicated common control channel (CCC).
- The CRSN is deployed in an urban environment where PUs are present.

- The CRSN applications being considered are related to the realization of the smart city concept, particularly in terms of disaster avoidance mechanisms such as structural health monitoring, air pollution control, and earthquake/landslide/flood warnings.

The CRSN network adopts time-division multiplexing in which time is divided into frames and each frame is divided into timeslots. The multiple access method used is code-division multiple access coordinated by the sink.

B. Robust and Energy-Efficient Transport Protocol (RETP)

1. Operation Modes

In RETP, there are two operation modes: management mode and data collection mode. Initially, the network starts in the management mode, and subsequently the sink manages the next mode based on the condition of the sensor nodes. Each mode is performed during a frame. The management mode includes spectrum management and data transmission activities, whereas the data collection mode consists only of data transmissions. The activities related to data transmission in both the modes are similar.

The management mode starts with spectrum management activities. The sensor nodes perform spectrum sensing on the entire spectrum bands assigned. Based on the spectrum sensing result, each sensor node performs spectrum decision to select an operating channel and a back-up channel. The spectrum decision method assumed in this section is a simple one; that is, the sensor node randomly chooses one of the available spectrums. Next, each sensor node reports its preferred operating channel to the sink on the common control channel and waits for further coordination. The sink collects all the control packets containing the operating channels selected by the sensor nodes and constructs a schedule called the sink schedule (S-schedule). The S-schedule contains three elements: time, channel, and reporting nodes. The sink broadcasts the S-schedule on the common control channel and follows this schedule during the data transmission activities. For instance, at time t_i, the sink waits on channel c_i for data transmissions from the sensor nodes that uses c_i as their operating channel. After that, at time t_{i+1}, the sink switches to another channel, c_j, and waits for data transmissions from the sensor nodes. The sink continuously switches the channel and collects the data from the sensor nodes. The sensor nodes receive the S-schedule on the common control channel and create their own schedule. The sensor nodes extract the time information in which the sink is expected to listen on their operating channel. The sensor nodes then go to the sleep state and wake up when they need to perform environment sensing and send the data to the sink according to the S-schedule. The spectrum management activities are illustrated in Figure 27.

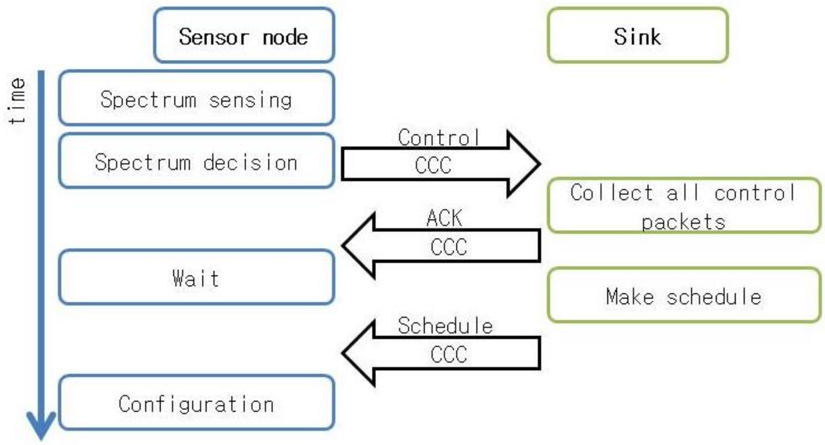


Figure 27. Spectrum management activities (CCC: common control channel).

After spectrum management activities are completed, the sensor nodes and the sink perform data transmission activities during the remaining time of the management mode (the same frame that starts with spectrum management). The data transmission activities of the sensor nodes and the sink simply follow the S-schedule. The sink switches and listens to different channels and collects the data transmitted by the sensor nodes. The sensor nodes wake up, perform environment sensing (data reading from the environment), and send the data to the sink on their operating channel at predetermined schedules. Once the transmission is completed, the sensor nodes go to the sleep state. The data transmission activities are illustrated in Figure 28.

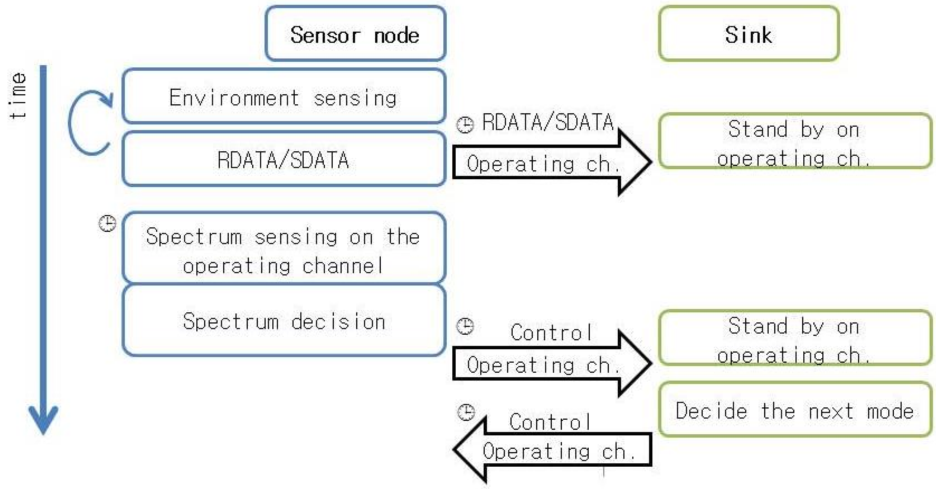


Figure 28. Data transmission activities. (clock image: the activity is performed by following the S-schedule, ch.: channel).

At the end of data transmission activities, the sensor nodes are required to perform spectrum sensing on their operating channel. Depending on the result of spectrum sensing, each sensor node performs spectrum decision. The possible outcomes of spectrum decision are: (1) the operating channel remains unchanged; (2) owing to the detection of PUs' transmission on the operating channel, the operating channel for the next frame is changed to the back-up channel and the back-up channel set becomes empty; and (3) when the sensor node decides to change its operating channel to the back-up channel but the back-up channel set is empty, the sensor node will request management mode on the next frame to the sink. The outcome of the spectrum decision stage is forwarded to the sink on the common control channel. The sink decides the operation mode for the next frame based on the reported spectrum decision from the sensor nodes. If there is at least one sensor node that requests management mode, then the sink announces the operation mode for the next frame as management mode. Otherwise, the sink creates a new S-schedule, announces that the operation mode for the next frame is data collection mode, and piggybacks the S-schedule with the announcement.

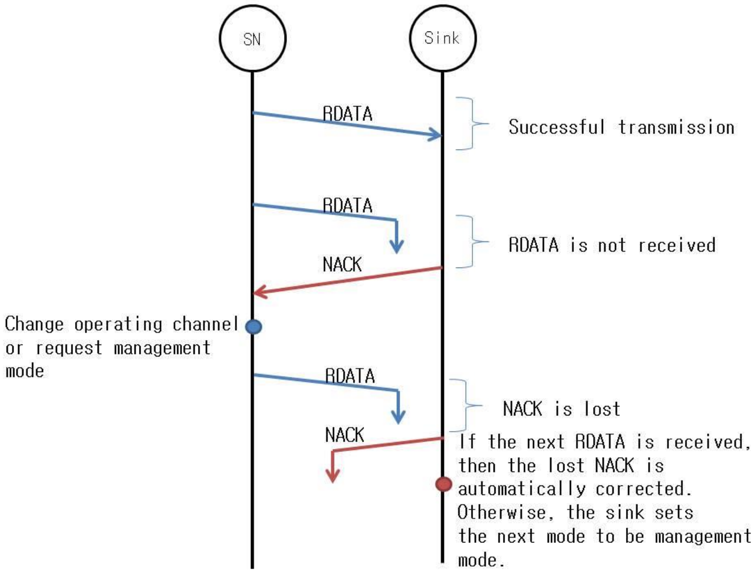
The data collection mode consists of data transmission activities that follow the S-schedule broadcasted by the sink at the end of the previous frame. The S-schedule created by the sink consists of the activities until the end of the current frame that includes spectrum sensing and spectrum decision at the end of the frame.

2. Interchange of ACK and NACK

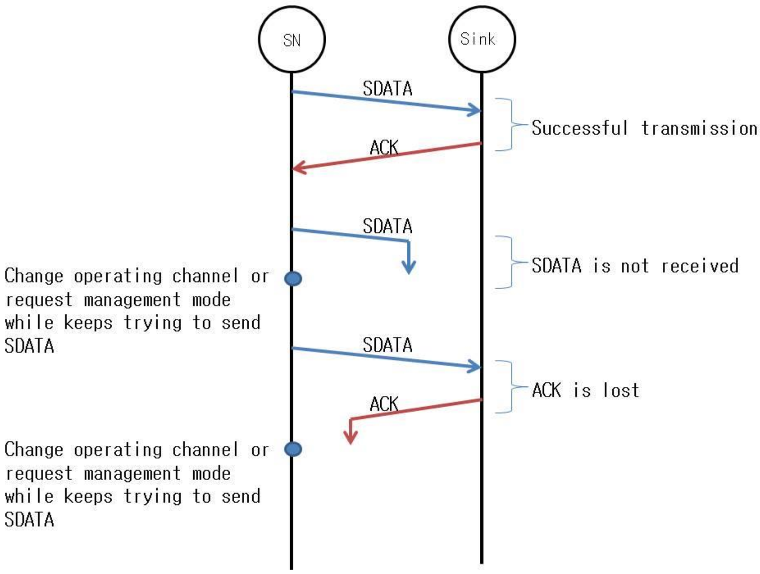
There are two types of data: sensitive data (SDATA) and regular data (RDATA). SDATA are the data that reflect critical events, such as the rapid spreading of cracks on a building wall or a bridge, dangerous levels of air pollution/water/temperature, etc. Thus, SDATA must be sent to the sink immediately. On the other hand, RDATA are collected periodically for data monitoring. Transmission of RDATA can be postponed if SDATA is present. In the case of SDATA transmission, ACK method is adopted; for RDATA transmission, NACK method is used.

By default, the sink expects regular RDATA transmissions from sensor nodes. The sink anticipates one RDATA transmission during a predetermined period. If the sink receives RDATA successfully, then it stores the data; otherwise, the sink sends a NACK packet to the sensor node whose RDATA is not received and waits for RDATA retransmission on the next scheduled transmission. The sensor node that receives the NACK packet retransmits RDATA on the next transmission schedule (following the S-schedule from the sink). If the sensor node keeps receiving NACK, then it assumes that the current operating channel's quality has been degraded and changes its operating channel or requests management mode on the next frame. As RDATA is not delay-sensitive data, the transmission of RDATA can be postponed.

When the sensor node detects an event in the environment such that its reading exceeds the predetermined threshold, it sends SDATA and waits for an ACK from the sink. If an ACK is not received, the sensor node sends SDATA once more at the next data transmission schedule. If the sensor node still does not receive an ACK, it checks the S-schedule and changes its operating channel to follow the sink's channel until it receives an ACK from the sink. After that, the sensor node changes its operating channel or requests management mode on the next frame. This interchange of ACK and NACK is illustrated in Figure 29.



(a)



(b)

Figure 29. (a) NACK method for RDATA and (b) ACK method for SDATA.

3. Analysis of Energy Consumption

In this subsection, the analysis of energy consumption is derived. The energy consumption during data transmission is analyzed first. The data transmission contains the environment sensing data (RDATA or SDATA), which is sent from the sensor nodes to the sink regularly. Then, the energy consumption in the management and data collection modes is comparatively analyzed. Because data collection is the main objective in CRSNs, the number of data collection mode occurrences should be higher than that of management mode occurrences. Furthermore, to preserve energy, the energy consumption in the data collection mode should be lower than that in the management mode. In this analysis, a boundary condition where the energy consumption in the data collection mode is lower than that in the management mode is derived.

a. Energy Consumption during Data Transmission

In this section, the energy consumption during the transmission of sensed data from the sensor nodes to the sink is derived. As described earlier, there are two types of data: SDATA and RDATA. In each data transmission, either SDATA or RDATA is sent to the sink by following the protocols described in Section 5.2.2. Hence, the energy consumption during data transmission, E_{TR}, is derived as follows:

$$E_{TR} = (1-q)E_{RDATA} + qE_{SDATA}, \tag{44}$$

where q is the probability of SDATA transmission, and E_{RDATA} and E_{SDATA} are the energy consumption of RDATA transmission and SDATA transmission, respectively. First, the E_{RDATA} and E_{SDATA} are derived and then incorporated to Equation (44). If p is the probability of successful data transmission and E_{TX} is the energy consumption of transmitting data as well as receiving and decoding data, then by following the protocols of RDATA and SDATA transmission,

$$E_{RDATA} = [p(1-p)](E_{TX})^2 + [p^2 + 1 - p](E_{TX}), \tag{45}$$

and

$$E_{SDATA} = p(E_{TX})^2 + (1-p)(E_{TX}). \tag{46}$$

Hence,

$$E_{TR} = [p(1-p+pq)](E_{TX})^2 + [(1-p) - p^2(1-q)](E_{TX}). \tag{47}$$

If E_{TR} is calculated using the E_{TX} value used in the simulation, then

$$mean(E_{TR}) \cong 1.25 E_{TX}. \tag{48}$$

b. Energy Consumption in the Management and Data Collection Modes

To achieve better spectrum selection and lower overhead, the number of management mode occurrences should be lower than that of data collection mode occurrences. This means that there are more data transmissions than management activities. Moreover, to save energy, the energy consumption in the data collection mode (E_{DC}) should be lower than that in the management mode (E_M). E_{DC} and E_M can be defined by

$$E_{DC} = (E_{ES} + E_{TR})d_{DC} + E_{LM}, \tag{49}$$

and

$$E_M = C_T E_{SS} + E_{SM} + (E_{SS} + E_{TR})d_M + E_{LM}, \tag{50}$$

respectively, where E_{ES} is the energy consumption of environment sensing, E_{SS} is the energy consumption of spectrum sensing, E_{SM} is the energy consumption of spectrum management in the management mode, E_{LM} is energy consumption of the last spectrum management performed at the end of a frame, C_T is the total number of channels, d_{DC} is the number of data collection cycles in the data collection mode, and d_M is the number of data collection cycles in the management mode. Because of the energy consumption values are fixed, both E_{DC} and E_M greatly depend on d_{DC} and d_M, respectively. There are three possible conditions between d_{DC} and d_M:

(1) If $d_{DC} = d_M$, then $E_{DC} < E_M$.

(2) If $d_{DC} < d_M$, then $E_{DC} \ll E_M$.

(3) If $d_{DC} > d_M$, then there should be a limitation to make $E_{DC} < E_M$.

The duration of a frame is fixed as 100 time slots in the performance evaluation. Furthermore, certain activities are occurred determinately and their time consumption is predetermined. Hence,

$$d_{DC} = 93/(3C_{DC}), \tag{51}$$

and

$$d_M = (87 - C_T)/(3C_M), \tag{52}$$

where C_T is the total number of channels, and C_{DC} and C_M are the numbers of active channels in the data collection mode and in the management mode, respectively. Furthermore, $1 \leq C_{DC} \leq C_T$ and $1 \leq C_M \leq C_T$. Let $E = E_{ES} + E_{TR}$, $E' = C_T E_{SS} + E_{SM}$, and E_{LM} be eliminated in both E_{DC} and E_M, then, to show that $E_{DC} < E_M$, $\forall\ d_{DC} > d_M$, the following equation should be true:

$$E(d_{DC} - d_M) < E', \tag{53}$$

where $(d_{DC} - d_M) > 1$.

To check the absolute truth of Equation (10), the maximum($E(d_{DC} - d_M)$) is compared with minimum(E'). Further analysis is: maximum($E(d_{DC} - d_M)$) = E × maximum($d_{DC} - d_M$) = E × (maximum(d_{DC}) − minimum(d_M)). The d_{DC} is maximized when C_{DC} equals to 1. Hence, d_{DC} equals to 31 whereas the minimal d_M equals to 1. Using the same variables used in the simulation in Section 5.3, $E_{ES} \fallingdotseq E_{TX}$ and, using the result from Equation (48),

$$E \cong 2.25 E_{TX}. \tag{54}$$

Similarly, $E_{SS} \fallingdotseq 0.5E_{TX}$ and $E_{SM} \fallingdotseq 4.67\ E_{TX}$. From the setting of $d_M = 1$ and maximal C_M, C_T equals to 21 can be obtained. Hence,

$$E' \cong 15.17 E_{TX}. \tag{55}$$

By substituting Equations (54) and (55) into Equation (53), the statement (10) turns out to be not true. This means that $\exists\ d_{DC}$ and d_M such that $E_{DC} < E_M$. Now, the boundary of d_{DC} and d_M is to be found. The boundary of d_{DC} and d_M can be simplified into the boundary of C_T, C_{DC} and C_M by referring to Equations (51) and (52). These are known: $1 \leq C_{DC} \leq C_T$ and $1 \leq C_M \leq C_T$, $\forall$ C_{DC}, C_M, and C_T. First, the condition when $C_{DC} = C_M$ is evaluated. Using Equations (51) and (52), $d_{DC} > d_M$, $\forall$ C_{DC}, C_M and $C_T > 1$. Then, when $C_{DC} \neq CC_M$, by setting minimal $d_{DC} = 2$ and maximal d_M happens when $C_M = 1$, $d_{DC} > d_M$, given $d_{DC} \geq 2$ and $C_T > 81$. However, only the first condition ($C_{DC} = C_M$) is considered because, in the simulation in Section 5.3, $C_T < 81$ and the case of $C_T > 81$ is rare in the practical situation. Therefore,

$$1 \leq C_T \leq 80, \tag{56}$$

for $C_{DC} = C_M = C$, $1 \leq C \leq C_T$.

Thus, Equations (51) and (52) can be represented as

$$d_{DC} = 31/C, \tag{57}$$

and

$$d_{DC} = (87 - C_T)/(3C), \tag{58}$$

respectively. Also, referring to Equations (54) and (55) and eliminating the term E_{TX} at both equations:

$$E_{DC} = E \times d_{DC} = 279/(4C), \tag{59}$$

and

$$E_M = E \times d_M = \frac{2.25(87 - C_T)}{3C} + 0.5C_T + 4.67. \tag{60}$$

Now, by setting $E_{DC} < E_M$, the limit of C is obtained as

$$C \geq \lceil (54 + 9C_T)/(56 + 6C_T) \rceil, \tag{61}$$

which is the boundary condition where the energy consumption in the data collection mode is lower than that in the management mode. When C_T is the maximum of 80, C should be ≥ 2. In the simulation, C_T is set to be equal to 30. So, using Equation (61), $C \geq 2$, which means that the minimum number of active channels in the management and data collection modes is 2. Therefore, if $d_{DC} > d_M$, then $E_{DC} < E_M$, given the boundary of C as in (61), which satisfies the three possible conditions between d_{DC} and d_M.

C. Performance Evaluation

The performance of RETP is evaluated by a computer simulation using MATLAB. The RETP is compared with TP-CRAHN. Originally, TP-CRAHN was developed for CR *ad hoc* networks. In the performance study, TP-CRAHN is selected as a comparison work because it is one of the earliest and the most cited transport protocol in CR network environments. The sensor nodes are deployed inside a building following a predetermined topology as shown in Figure 30. The application of the CRSN might be structural health monitoring, temperature monitoring, *etc.* The simulation settings are presented in Table 3 (settings for power supply and energy consumptions are the same with Table 1). RETP is compared with TP-CRAHN in terms of the (I) number of alive nodes; (II) delay in event detection; and (III) reliability of event detection.

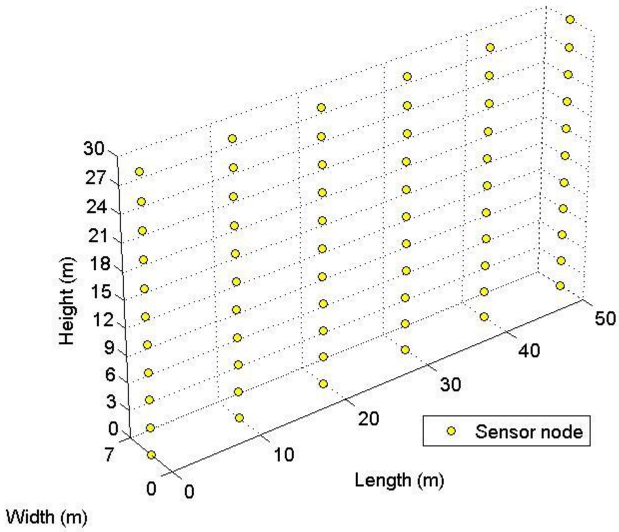


Figure 30. Sensor node deployment.

Table 3. Simulation Parameters

Parameter	Value
Topology	50 m × 30 m × 7 m
Number of sensor nodes	60
Number of PUs	6
Number of sink nodes	1
Locations of sink node	(25 m, 15 m, 3.5 m)
Sensing range (environment sensing)	10 m
Number of channels	30
Number of timeslots per frame	100

Duration of a timeslot	20 ms

Figure 31 shows the number of alive nodes per frame during the network active time. As the network topology is predefined and the sensing coverage is not redundant, the exhaustion of even one sensor node means that the CRSN coverage is disrupted. Therefore, the network lifetime is defined as the energy depletion of the first sensor node. With this definition, RETP has 53.77% longer lifetime compared with TP-CRAHN. The main reason for low energy consumption in RETP is that it follows the schedule from the sink, according to which a sensor node can go to the sleep state if it has no scheduled activity.

In TP-CRAHN, because it was designed for *ad hoc* networks, the sensor nodes are required to perform frequent control channel exchanges and relay data packets. Nevertheless, both protocols perform spectrum decision in a distributive manner, making the spectrum decision results (operating and backup channel selection) more accurate compared to centralized spectrum decision. In this simulation, the frame duration is equal to 2 s; thus, the lifetime of RETP is about 53 days and the lifetime of TP-CRAHN is about 34 days.

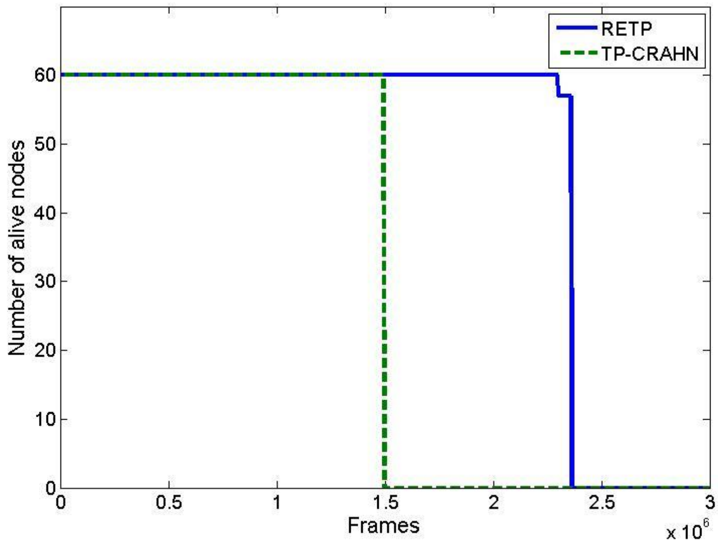


Figure 31. Number of alive nodes.

Delay in event detection is defined as the time elapsed between the occurrence of a real event and the detection of the event by the sink. Figures 32, 33, and 34 show the event detection reliability in three scenarios: varied channel condition, varied probability of event occurrence, and varied probability of PUs' channel change, respectively.

The channel condition is varied from good to very poor, representing the packet error probability. The probability of occurrence of an event is varied from 20% to 80%. The PUs' channel change is defined as the occasion of PUs changing their operating channel, and it is varied from 20% to 80%. For all settings, the simulation results show the same trend; RETP has shorter delay than TP-CRAHN by 53.52%, 51.18%, and 51.33% in the scenarios of varied channel condition, varied probability of event occurrence, and varied probability of PUs' channel change, respectively. These simulation results show that even though the performance of each of the two protocols is stable under these varying conditions, RETP always has the shorter delay. The reason is, in RETP, data transmission activities are scheduled by the sink. Therefore, if the data is transmitted successfully, then the delay is fixed. On the other hand, in TP-CRAHN, data transmission occurs in a sporadic manner, resulting in inconsistent and longer delays (in the case of route failure).

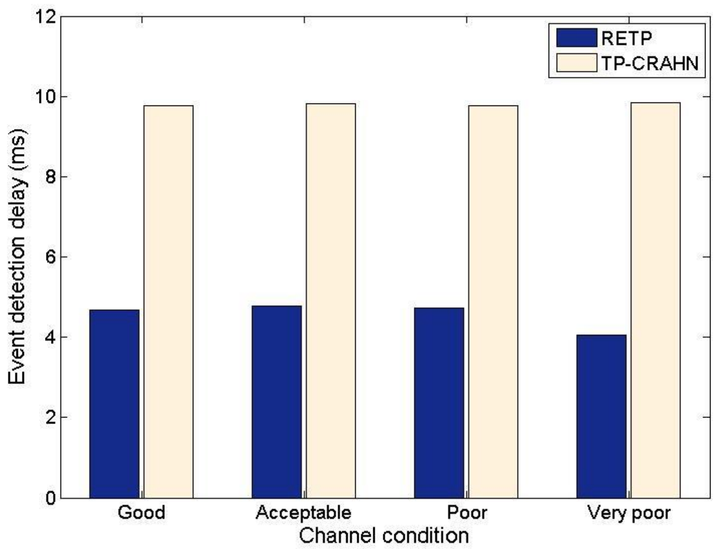


Figure 32. Event detection delay against channel condition.

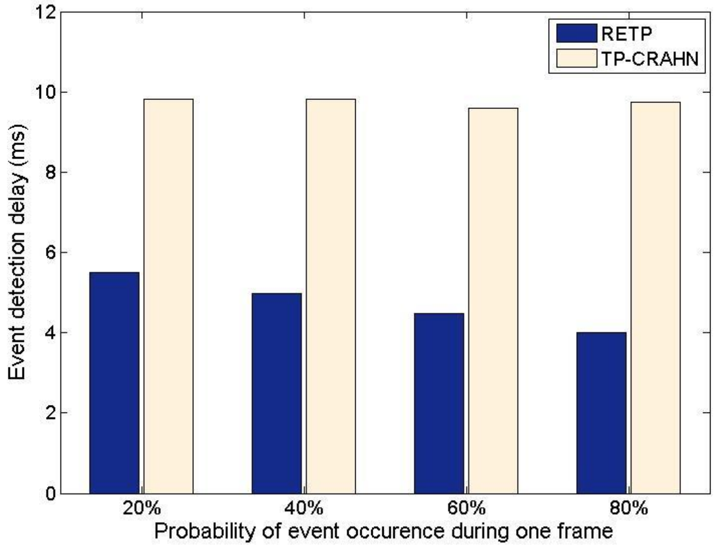


Figure 33. Event detection delay against probability of event occurrence.

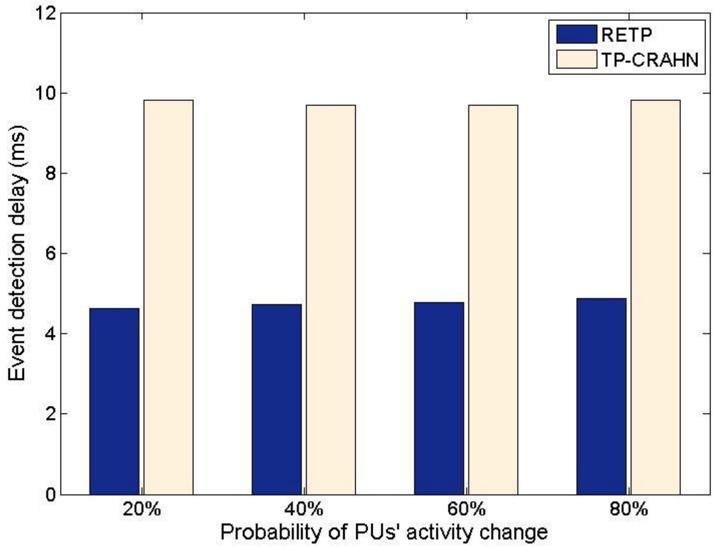


Figure 34. Event detection delay against probability of PUs' activity change.

High reliability of event detection is one of the most important requirements of a sensor network. The reliability of event detection is the capability of the sensor nodes to detect the occurrence of a real event and deliver the information successfully to the sink. In other words, the sink needs to be informed about every event detection. When an event such as a rise in the temperature occurs, a sensor node is able to sense the event if it is inside the sensor node's sensing coverage limit. It is possible that an event is sensed by more than one sensor node. In this case, the sensor node creates a package containing the event information and sends it to the sink. Sometimes, if the quality of the channel is poor, data transmission from the sensor node to the sink might fail. In this case, the event is detected by the sensor node, but it is not detected by the sink, which makes the CRSN fails to deliver the data to the users. Figures 35, 36, and 37 show the event detection reliability in three situations: varied channel condition, varied probability of event occurrence, and varied probability of PUs' channel change, respectively.

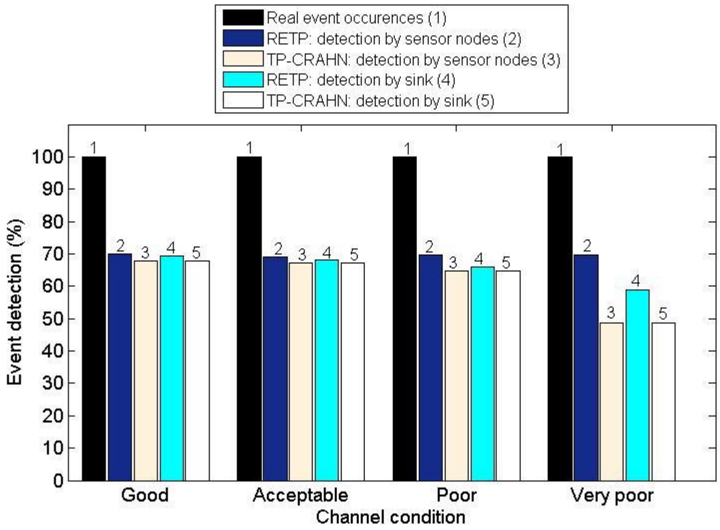


Figure 35. Event detection probability against channel condition.

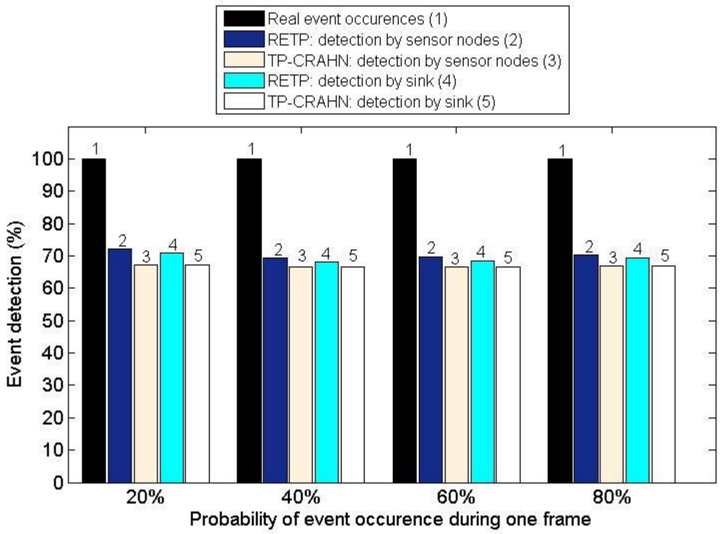


Figure 36. Event detection probability against probability of event occurrence.

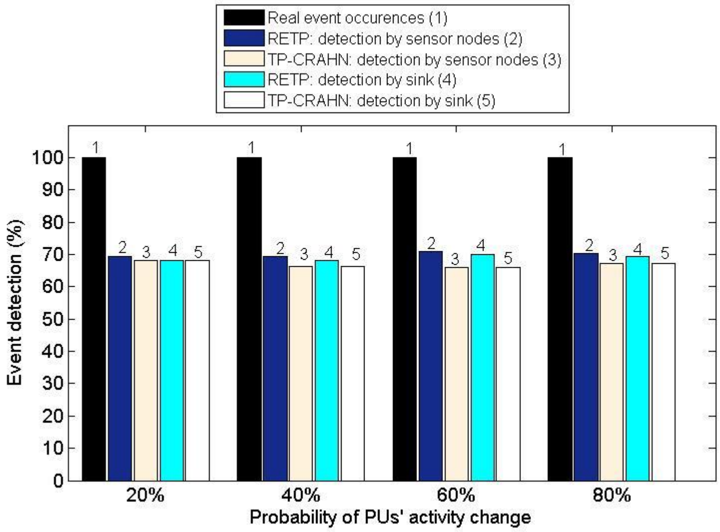


Figure 37. Event detection probability against probability of PUs' activity change.

Figure 35 shows the event detection probability against the channel condition. The real event occurrence is the actual event simulated; thus its value is 100%. The event is directly detected by the sensor nodes (detection by sensor nodes) and the sensor nodes send the data to the sink (detection by the sink). The simulation results show that not all the events are detected by the sensor nodes. Indeed, the highest detection rate by the sensor nodes is 69.85%. Furthermore, in most cases, not every detection by the sensor nodes is successfully informed to the sink, *i.e.*, the detection by the sink is almost always lower than the detection by sensor nodes. The detection by sink is varied from 48.66% to 69.37%, relative to the real event occurrence, or from 84.38% to 100% relative to the event detection by sensor nodes. The reason that the sensor nodes fail to detect an event is that they are occupied with other tasks such as spectrum sensing and spectrum management. Nevertheless, the detection by sensor nodes in RETP remains stable (average 69.58%) whereas it decreases in TP-CRAHN from 67.75% to 48.67% as the channel quality degrades. The detection by sensor nodes in RETP is higher than in TP-CRAHN by a minimum of 2.84% when the channel condition is acceptable and a maximum of 30.28% when the channel condition is very poor. These results show that the performance of RETP remains stable even if the channel condition becomes worse.

Some events detected by sensor nodes might not be successfully received by the sink. Thus, the rate of event detection by the sink is at most the same as that of detection by sensor nodes. When both the rates are same, it means that every data transmission from sensor nodes to the sink is successful. Naturally, as the channel condition becomes worse, the rate of detection by the sink would become lower than that by sensor nodes, because the probability that the data transmission is lost or erroneously received becomes higher. The rate of detection by the sink relative to that by sensor nodes in RETP is 94.20% on average, whereas in TP-CRAHN, it is 99.98% on average. TP-CRAHN shows better performance compared to RETP by 5.78%. The reason is, in RETP, for each event detection, a sensor node sends a SDATA package and waits for an ACK. If it fails to receive an ACK, it retransmits the data on the next schedule during the same frame. When the current frame ends, the sensor node starts the next frame by following the operation mode and ignores its pending SDATA, making the sink fail to detect the event. In TP-CRAHN, the network activities are not divided in frames and the sensor nodes do not follow a centralized schedule; therefore, when an event occurs, the sensor nodes send this data to the sink immediately and the route failure method is ready to repair link failures.

Figure 36 shows the event detection probability against event occurrence probability. The results show that both protocols perform satisfactorily under these variations. Overall, RETP outperforms TP-CRAHN by 3.00% and 1.88% for detection by sensor nodes and detection by sink, respectively. These results show that both protocols are able to handle frequent event occurrences. Figure 37 shows the event detection probability against PUs' channel change probability. Similar to previous results, the performance of both protocols remain stable, even though the probability of PUs' activity change increases. Overall, RETP outperforms TP-CRAHN by 3.61% and 2.53% for detection by sensor nodes and detection by sink, respectively. This result shows that both protocols are able to adapt to frequent PUs' activity changes. Table 5tb quantitatively summarizes the improvement of the performance metrics measured.

D. Conclusions

A transport protocol for CRSNs called the robust and energy-efficient transport protocol (RETP) has been proposed. RETP focuses on prolonging the network lifetime of CRSNs while simultaneously reducing event detection delay and maintaining reliability. The protocol operates in two modes: management mode and data collection mode. In RETP, channel sensing and channel decision are performed in a distributive manner by the sensor nodes, whereas data transmission is governed by the sink. The sink broadcasts a schedule for each frame in which it is followed by the sensor nodes. RETP has two types of data. SDATA has to be transmitted immediately for which an acknowledgment from the sink is required. RDATA does not require an acknowledgement from the sink, but if the sink does not receive any data, it sends a NACK packet. The performance of RETP has been evaluated and compared with the performance of TP-CRAHN. Simulation results show that RETP achieves 53.8% longer network lifetime compared to that of TP-CRAHN while achieving shorter event detection delay and preserving stable event detection probability.

VI. CONCLUSIONS AND FUTURE WORKS

A. Conclusions

In this thesis, energy-efficient protocols for a CR sensor network (CRSN) are proposed, as the CRSN is envisioned to be the key network to support the requirements of future wireless networks, such as seamless telecommunication, Internet of things, and improvement of spectrum utilization. However, there are numerous challenges in CRSNs. Some of the challenges are inherent to the CR devices properties whereas the others to the WSNs characteristics. Those difficulties are overcome by: (1) a distributed spectrum decision framework, (2) a compact clustering protocol, and (3) a robust transport protocol. The performance evaluation for each proposed protocol shows an outstanding energy savings and other network parameters improvements.

An effective spectrum decision framework is necessary to support inter-spectrum sharing as well as intra-spectrum sharing. In Chapter III, an energy-efficient distributed spectrum decision (EDSD) framework is proposed with two operation modes where each mode has three modules: spectrum sensing, spectrum decision, and data transmissions. Two spectrum selection algorithms, namely EDSD random selection (EDSD-R) and EDSD game theory-based selection (EDSD-G), are proposed and their performances are evaluated against a related work. The performance evaluation shows that EDSD-R and EDSD-G have longer network lifetime in both scenarios of PUs did not have favorite channel and they have a favorite channel. Moreover, the proposed works also have lower overhead. The reason of the improvements is mainly because EDSD is a distributed method which requires less control packet exchanges to the sink. Other contributors of the performance improvement is the simple yet energy-aware clustering method, the predictions by Markov chain (in EDSD-G), and the data collection mode that has lower energy consumption than coordination mode.

Similar with WSNs, clustering is one of the energy conservation strategies in CRSNs. In Chapter IV, a clustering with temporary support nodes (CENTRE) with two sub-phases of cluster formation is proposed to solve the additional clustering requirement in the CR environments. A novel concept of temporary support nodes to assist the cluster coordination is introduced. The performance evaluation of CENTRE shows that it achieves up to 34% longer network lifetime with lower clustering overhead compared to a related work. CENTRE also decreases the average distance between CHs and their members, resulting in compact clustering. The main reasons for the performance improvement of the CENTRE scheme include the following: the fixed cluster formation duration, the adjustment of the transmission power of the cluster members based on the distance to the CH, the refinement of the cluster formation process by the use of temporary support nodes, and the use of the sleep mode when the sensors are not active.

Last but not least, a transport protocol is the key to reliable data collection in wireless sensor applications. In chapter V, a robust and energy-efficient transport protocol (RETP) that differentiates the data type based on their content and interchange the acknowledgment methods based on the data type is proposed. Performance evaluation shows that RETP achieves up to 53.8% longer network lifetime compared to a related work while achieving shorter event detection delay and preserving stable event detection probability, simultaneously. The performance improvements are caused by RETP methods of distributive spectrum sensing and decision which leads to more accurate spectrum selection and less spectrum switching. Moreover, the interchange of acknowledgment method gives a prioritization to delay-sensitive data.

The three proposed protocols are designed based on different application scenarios. EDSD is suitable in an environment where the PUs are crowded and their channel access behaviors reveal certain patterns that can be exploited in order to select the most stable operating channel in terms of channel holding time by training the Markov chain. A suitable application for EDSD is an environment monitoring application in an urban area, such as air pollution monitoring in a city. On the contrary, CENTRE works best when the PUs are sparse or none. The reason is because, rather than anticipating

PUs appearances on the operating/backup channels, CENTRE aims at creating compact clusters and maintaining the formation for as long as possible. However, CENTRE can cope well with large-scale, dense, and random deployment of cognitive sensor nodes in a wide-band CRSN (comprises of wide selection of channels). A suitable application for CENTRE is an environmental monitoring application in a rural area, such as forest fire detection. The third protocol, RETP, can be implemented in any PUs condition, because it monitors the operating channel's condition periodically to anticipate PUs appearance. However, RETP can only handle small-scale and pre-determined deployment of cognitive sensor nodes, where clustering is not needed. A suitable application for RETP is a modest and pre-planned sensor network, such as structural health monitoring and smart homes. Table 4 lists the features of each protocol.

Table 4. Features of EDSD, CENTRE, and RETP

	EDSD	CENTRE	RETP
Network size	Moderate	Large	Small
Population of sensor node	Any	Dense	Sparse
Deployment of sensor nodes	Any	Random	Pre-determined
Number of CR per sensor node	Single CR	Single CR	Single CR

Clustered sensor nodes	Supported	Yes	No
Number of PUs	Crowded	Sparse/none	Any
Condition of PUs	Have certain patterns	Any	Any
Coexistence with other wireless systems	Very suitable	Suitable	Suitable
Number of licensed channels	Many	Many	Several
Data collection type	Periodic	Periodic	Periodic
Focus	PUs' spectrum usage prediction	Compact clustering with partial spectrum sensing	End-to-end reliability with data prioritization
Suitable scenario	Environment monitoring in urban area	Environment monitoring in rural area	Structural health monitoring or other pre-planned network

B. Future Works

The essential issues of CRSNs have been covered in this thesis; however, this work is still far from CRSNs realization. Aside from regulation and standardization issues, the most suitable protocols and scenarios for CRSNs could be prepared. The future works are:

1. To refine the proposed protocols to achieve higher energy saving

For spectrum decision framework, the game theory-based spectrum selection might be combined with other machine learning or artificial intelligent algorithm to exploit PUs spectrum usage pattern. The goal is to allow sensor nodes to select an operating channel with lowest channel switching probability (or proactive spectrum mobility) to reduce energy consumption.

2. To refine the PU activity model based on real measurement

   PU channel usage might reveal a particular pattern during specific time span. Real measurements on PU activity, if any, are to be incorporated to measure the performance of the spectrum decision especially.

3. To match the sensor nodes deployment to the application scenarios

   Two of the three proposed protocol assumed that the sensor nodes are placed randomly over the interested region. However, some wireless sensor applications might have predetermined sensor placement, such as the case considered in Chapter V. In this case, the clustering protocol proposed in Chapter IV might be refined to achieve higher energy conservation.

BIBLIOGRAPHY

[1] International Telecommunication Union, "The World in 2014: ICT Facts and Figures." http://www.itu.int/en/ITU-D/Statistics/Documents/facts/ICTFactsFigures2014-e.pdf. Accessed on December 2014.

[2] I. F. Akylidiz, W. Su, Y. Sankarasubramaniam, E. Cayirci, "Wireless Sensor Networks: A Survey," *Computer Networks*, Vol. 38, pp. 393-422, 2002.

[3] J. Yick, B. Mukherjee, and D. Ghosal, "Wireless sensor network survey," *Computer networks*, Vol. 52, No. 12, pp. 2292-2330, 2008.

[4] Libelium, "50 Sensor Applications for a Smarter World," http://www.libelium.com/top_50_iot_sensor_applications_ranking/download. Accessed on December 2014.

[5] J. A. Stankovic, A. D. Wood, and T. He, "Realistic applications for wireless sensor networks," *Theoretical Aspects of Distributed Computing in Sensor Networks*, Springer Berlin Heidelberg, pp. 835-863, 2011.

[6] S. Tozlu, M. Senel, W. Mao, A. Keshavarzian, "Wi-Fi enabled sensors for internet of things: A practical approach," *IEEE Communications Magazine*, Vol. 50, No. 6, pp. 134-143, 2012.

[7] C. Wei, J. Yang, Y. Gao, and Z. Zhang, "Cluster-based routing protocols in wireless sensor networks: a survey," *Proc. of International Conf. on Computer Science and Network Technology*, Vol. 3, pp. 1659-1663, Dec. 2011.

[8] H. Shin, S. Moh, and I. Chung, "Clustering with One-Time Setup for Reduced Energy Consumption and Prolonged Lifetime in Wireless Sensor Networks," *International Journal in Distributed Sensor Networks*, 2013.

[9] S. Salim and S. Moh, "Energy-Efficient Clustering Based on Game Theory for Wireless Sensor Networks," *International Journal of Engineering and Industries*, Vol. 15, No. 3, pp. 31-37, Sept. 2014.

[10] S. Moh, "14897 Topics in Mobile Computing: Introduction," Lecture material, Source: Chosun University, spring 2014.

[11] C. C. Cavalcante, R. F. Colares, and P. C. Barbosa, "Telecommunications: Advances and Trends in Transmission, Networking and Applications," Universidade de Fortaleza, Brazil, 2006.

[12] S. Salim, C. H. W. Oey, and S. Moh, "Are Heterogeneous Cellular Networks Superior to Homogeneous Ones?" *Lecture Notes in Computer Science*, Vol. 7523, pp. 61-68, Sept 2012.

[13] Federal Communication Commission, "Comments of CTIA - The Wireless Association" , 2009.

[14] P. Zhang, "In the development of wireless cognitive science," *Chinese Science Bulletin*, Vol. 57, pp. 3661-3661, 2012.

[15] C. A. Herter Jr, "Electromagnetic Spectrum: A Critical Natural Resource," *The National Resources Journal*, Vol. *25*, 1985.

[16] National Telecommunications and Information Administration, "United States Frequency Allocations", http://www.ntia.doc.gov/page/2011/united-states-frequency-allocation-chart, accessed on December 2014.

[17] Federal Communication Commission, "Report of the Spectrum Efficiency Working Group," November 2002.

[18] V. Valenta, R. Marsalek, G. Baudoin, M. Villegas, M. Suarez, F. Robert, "Survey on spectrum utilization in Europe: Measurements, analyses and observations," *Proceedings of the Fifth International Conference on Cognitive Radio Oriented Wireless Networks & Communications (CROWNCOM),* June 2010.

[19] R. Urban, T. Korinek, and P. Pechac, "Broadband spectrum survey measurements for cognitive radio applications," *Radio engineering*, Vol. 21, No. 4, pp. 1101-1109, 2012.

[20] J. Mitola, "Cognitive Radio - An Integrated Agent Architecture for Software Defined Radio", Ph. D Dissertation, Royal Institute of Technology, Sweden, 2000.

[21] S. Haykin, "Cognitive radio: brain-empowered wireless communications, "*IEEE Journal on Selected Areas in Communications*, Vol. 23, No. 2, pp. 201-220, 2005.

[22] Federal Communication Commission, "Notice of Proposed Rule Making and Order," December 2003.

[23] I. F. Akyildiz, W. Y. Lee, M. C. Vuran, S. Mohanty, "NeXt generation/dynamic spectrum access/cognitive radio wireless networks: a survey," *Computer Networks*, Vol. 50, No. 13, pp. 2127-2159, 2006.

[24] B. Wang and K. R. Liu, "Advances in cognitive radio networks: A survey," *IEEE Journal of Selected Topics in Signal Processing,* Vol. 5, No. 1, pp. 5-23, 2011.

[25] S. Salim and S. Moh, "On-demand Routing Protocols for Cognitive Radio Ad Hoc Networks," *EURASIP Journal on Wireless Communications and Networking*, Vol. 13, No. 102, pp. 1-10, April 2013.

[26] C. R. Stevenson, G. Chouinard, Z. Lei, W. Hu, S. J. Shellhammer, W. Caldwell, "IEEE 802.22: The first cognitive radio wireless regional area network standard," *IEEE Communications Magazine*, Vol. 47, No. 1, pp. 130-138, 2009.

[27] S. Salim and S. Moh, "Busy Source Prioritization for Improving On-demand Frame Contention in IEEE 802.22 Networks," *Proc. of 2012 KISM Fall Conference*, Vol. 1, No. 2, pp. II-B.5.1-II.B.5.3, 2012.

[28] I. F. Akyildiz, W. Y. Lee, K. R. Chowdhury, "CRAHNs: Cognitive radio ad hoc networks," *Ad Hoc Networks*, Vol. 7, No. 5, pp. 810-836, 2009.

[29] S. Salim, S. Moh, and I. Chung, "Transmission Power Control Aware Routing in Cognitive Radio Ad Hoc Networks," *Wireless Personal Communications*, Vol. 71, No. 4, pp. 2713-2724, August 2013.

[30] O. B. Akan, O. Karli, and O. Ergul, "Cognitive radio sensor networks," *IEEE Network,* Vol. 23, No. 4, pp. 34-40, 2009.

[31] T. Chen, H. Zhang, G. M. Maggio, I. Chlamtac, "CogMesh: a cluster-based cognitive radio network," *2nd IEEE International Symposium on New Frontiers in Dynamic Spectrum Access Networks,* pp. 168-178, April 2007.

[32] C. I. Badoi, N. Prasad, V. Croitoru, R. Prasad, "5G based on cognitive radio," *Wireless Personal Communications*, Vol. 57, No. 3, pp. 441-464, 2011.

[33] S. Salim and S. Moh, "Interference-based Routing Metric for Cognitive Radio Ad Hoc Networks," *Proc. of 2012 KISM Spring Conference*, pp. 52-55, 2012.

[34] B. Fette, "Fourteen Years of Cognitive Radio Development," *IEEE Military Communications Conference (MILCOM)*, pp. 1166-1175, 2013.

[35] R. Machado and S. Tekinay, "A survey of game-theoretic approaches in wireless sensor networks," *Computer Networks*, vol. 52, pp. 3047-3061, Nov. 2008.

[36] V. Srivastava, J. Neel, A. B. Mackenzie, R. Menon, L. A. Dasilva, J. E. Hicks, J. H. Reed, and R. P. Gilles, "Using game theory to analyze

wireless ad hoc networks," *IEEE Communications Surveys & Tutorials*, vol. 7, pp. 46-56, 2005

[37] S. Salim, C. Baek, S. Moh, I. Chung, "Cooperative and Non-Cooperative Games for Spectrum Sharing in Cognitive Radio Networks: A Comparative Study," *The Fourth International Conference on Convergence and its Application (ICCA)*, Nov. 2014.

[38] A. O. Bicen and O. B. Akan, "Reliability and congestion control in cognitive radio sensor networks," *Ad Hoc Networks*, Vol. 9, pp. 1154-1164, 2011.

[39] A. O. Bicen, V. C. Gungor, and O. B. Akan, "Delay-sensitive and multimedia communication in cognitive radio sensor networks," *Ad Hoc Networks*, Vol. 10, pp. 816-830, 2012.

[40] M. T. Masonta, M. Mzyece, and N. Ntlatlapa, "Spectrum decision in cognitive radio networks: A survey," *IEEE Communications Surveys & Tutorials*, Vol. 15, No. 3, pp. 1088-1107, 2013.

[41] J. G. Jia, Z. W. He, J. M. Kuang, H. F. Wang, "Analysis of Key Technologies for Cognitive Radio Based Wireless Sensor Networks," *6th International Conference on In Wireless Communications Networking and Mobile Computing (WiCOM)*, pp. 1-5, Sept. 2010.

[42] M. Askari, Y. S. Kavian, H. Kaabi, H. F. Rashvand, "A channel assignment algorithm for cognitive radio wireless sensor networks", *IET Conference on Wireless Sensor Systems (WSS)*, 2012.

[43] X. Li, D. Wang, J. McNair, J. Chen, "Residual energy aware channel assignment in cognitive radio sensor networks," *IEEE Wireless Communications and Networking Conference (WCNC)*, pp. 398-403, March 2011.

[44] Z. Hu, Y. Sun, and Y. Ji, "A Dynamic spectrum access strategy based on real-time usability in cognitive radio sensor networks," *Seventh International Conference on Mobile Ad-hoc and Sensor Networks (MSN)*, pp. 318-322, Dec. 2011.

[45] C. Wu, S. Ohzahata, and T. Kato, "Dynamic channel assignment and routing for cognitive sensor networks," *IEEE International Symposium on Wireless Communication Systems (ISWCS)*, pp. 86-90, August 2012.

[46] M. Cardei and A. Mihnea, "Channel assignment in cognitive wireless sensor networks," *International Conference on Computing, Networking and Communications (ICNC)*, pp. 588-593, Feb. 2014.

[47] A. Jamal, C. K. Tham, and W. C. Wong, "Event detection and channel allocation in cognitive radio sensor networks," *IEEE International Conference on Communication Systems (ICCS)*, pp. 157-161, Nov. 2012.

[48] J. A. Han, W. S. Jeon, and D. G. Jeong, "Energy-efficient channel management scheme for cognitive radio sensor networks," *IEEE Transactions on Vehicular Technology*, Vol 60, No. 4, pp. 1905-1910, 2011.

[49] J. Y. Yu and S. Mannor, "Online Learning in Stochastic Games and Markov Decision Processes," Not yet published, http://www.cim.mcgill.ca/~jiayuan/survey08.pdf, Accessed on December 2014.

[50] M. Bowling and M. Veloso, "An analysis of stochastic game theory for multiagent reinforcement learning," No. CMU-CS-00-165, Carnegie-Mellon Univ, 2000.

[51] S. S. Byun, I. Balasingham, and X. Liang, "Dynamic spectrum allocation in wireless cognitive sensor networks: Improving fairness and energy efficiency," *IEEE 68th Vehicular Technology Conference*, Sept. 2008.

[52] L. Marin and L. Giupponi, "Performance evaluation of spectrum decision schemes for a cognitive ad-hoc network," *IEEE 19th International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC)*, Sept. 2008.

[53] B. Canberk, I. F. Akyildiz, and S. Oktug, "A qos-aware framework for available spectrum characterization and decision in cognitive radio networks," *IEEE 21st International Symposium on Personal Indoor and Mobile Radio Communications (PIMRC)*, pp. 1533-1538, Sept. 2010.

[54] Y. Yao, S. R. Ngoga, and A. Popescu, "Cognitive Radio spectrum decision based on channel usage prediction," *8th EURO-NGI Conference on Next Generation Internet (NGI)*, pp. 41-48, June 2012.

[55] Y. Yang, G. A. Zhang, Y. C. Ji, "Impartial spectrum decision under interference temperature model in cognitive wireless mesh networks," *International Journal of Embedded Systems*, Vol. 5, No. 3, pp. 134-141, 2013.

[56] Y. Li, Y. Dong, H. Zhang, H. Zhao, H. Shi, X. Zhao, "Qos provisioning spectrum decision algorithm based on predictions in cognitive radio networks," *6th International Conference on Wireless Communications Networking and Mobile Computing (WiCOM)*, Sept. 2010.

[57] L. C. Wang, C. W. Wang, F. Adachi, "Load-balancing spectrum decision for cognitive radio networks," *IEEE Journal on Selected Areas in Communications*, Vol 29, No 4, pp. 757-769, 2011.

[58] Z. Jin, S. Anand, and K. P. Subbalakshmi, "Robust spectrum decision protocol against primary user emulation attacks in dynamic spectrum access networks," *IEEE Global Telecommunications Conference (GLOBECOM)*, Dec. 2010.

[59] G. Jakimoski and K. P. Subbalakshmi, "Towards secure spectrum decision," *IEEE International Conference on In Communications (ICC)*, June 2009.

[60] V. Mishra, L. C. Tong, S. Chan, A. Kumar, "Energy aware spectrum decision framework for cognitive radio networks," *International Symposium on Electronic System Design (ISED)*, pp. 309-313, Dec. 2012.

[61] W. Y. Lee and I. F. Akyildiz, "A spectrum decision framework for cognitive radio networks," *IEEE Transactions on Mobile Computing*, Vol. 10, No. 2, pp. 161-174, 2011.

[62] C. H. W. Oey, I. Christian, and S. Moh, "Energy- and Cognitive-Radio-Aware Routing in Cognitive Radio Sensor Networks," *Internatioal Journal on Distributed Sensor Networks*, 2012.

[63] K. L. A. Yau, P. Komisarczuk, and P. D. Teal, "Cognitive Radio-Based Wireless Sensor Networks: Conceptual Design and Open Issues," *Proceedings of the 34th IEEE Conference on Local Computer Networks*, pp. 955-962, October 2009.

[64] P. T. A. Quang, S. Kim, and D. Kim, "A throughput-aware routing for distributed industrial cognitive radio sensor networks," *Proceedings of the 9th IEEE International Workshop on Factory Communication Systems*, pp. 87-90, May 2012.

[65] M. Ozger and O. B. Akan, "Event-Driven Spectrum-Aware Clustering in Cognitive Radio Sensor Networks," *Proceedings of the 32nd IEEE International Conference on Computer Communications*, pp. 1483-1491, April 2013.

[66] H. Zhang, Z. Zhang, H. Dai, R. Yin, X. Chen, "Distributed Spectrum-Aware Clustering in Cognitive Radio Sensor Networks," *Proceedings of the 2011 IEEE Global Telecommunications Conference*, pp. 1-6, Dec. 2011.

[67] H. Zhang, Z. Zhang, and C Yuen, "Energy-efficient spectrum-aware clustering for cognitive radio sensor networks," *Chinese Science Bulletin*, Vol. 57, pp. 3731-3739, 2012.

[68] N. Panahi, A. Payandeh, H. O. Rohi, M. S. Haghighi, "Adaptation of LEACH Routing Protocol to Cognitive Radio Sensor Networks," *Proceedings of the 6th International Symposium on Telecommunications*, pp. 541-547, Nov. 2012.

[69] Y. Sankarasubramaniam, O. B. Akan, and I. F. Akyildiz, "ESRT: Event-to-sink reliable transport in wireless sensor networks," *Proceedings of the 4th ACM International Symposium on Mobile Ad Hoc Networking & Computing*, pp. 177-188, June 2003.

[70] F. Stann and J. Heidemann, "RMST: Reliable data transport in sensor networks," *Proceedings of the 1st IEEE International Workshop on Sensor Net Protocols and Applications (SNPA)*, pp. 102-112, May 2003.

[71] C. Wan, A. T. Campbell, and L. Krishnamurthy, "PSFQ: A reliable transport protocol for wireless sensor networks," *Proceedings of the 1st ACM International Workshop on Wireless Sensor Networks and Applications*, Sept. 2002.

[72] C. Wan, S. B. Eisenman, A. T. Campbell, "CODA: Congestion detection and avoidance in sensor networks," *Proceedings of the 1st International Conference on Embedded Networked Sensor Systems*, pp. 266-279, Nov. 2003.

[73] Y. G. Iyer, S. Gandham, and S. Venkatesan, "STCP: A generic transport layer protocol for wireless sensor networks," *Proceedings of the International Conference on Computer Communications and Networks*, pp. 449-454, Oct. 2005

[74] K. R. Chowdhury, M. D. Felice, and I. F. Akyildiz, "TP-CRAHN: A Transport Protocol for Cognitive Radio Ad-Hoc Networks," *Proceedings of the IEEE on INFOCOM*, pp. 2482-2490, April 2009.

[75] K. R. Chowdhury, M. D. Felice, and I. F. Akyildiz, "TCP CRAHN: A Transport Control Protocol for Cognitive Radio Ad Hoc Networks," *IEEE Transactions on Mobile Computing*, Vol. 12, pp. 790-803, 2013.

[76] X. Lin and Y. Fang, "A survey on the available digital TV bands for cognitive radio operations," *IEEE Fourth International Conference on In Communications and Networking in China (ChinaCOM)*, August 2009.

[77] S. Shellhammer, G. Chouinard, "Spectrum Sensing Requirements Summary," June 2006. Available online: http://www.ieee802.org/22/Meeting_documents/2006_July/22-06-0089-01-0000-Spectrum-Sensing-Requirements-Summary.doc. Accessed on January 2014.

[78] Find the energy contained in standard battery sizes", http://www.allaboutbatteries.com/Energy-tables.html. Accessed on January 2014.

[79] V. Shnayder, M. Hempstead, B. Chen, G. W. Allen, M. Welsh, "Simulating the Power Consumption of Large-Scale Sensor Network Applications," *Proc. of 2nd international conference on Embedded Networked Sensor Systems*, pp. 188-200, 2004.

[80] R. Deng, J. Chen, C. Yuen, P. Cheng, Y. Sun, "Energy-Efficient Cooperative Spectrum Sensing by Optimal Scheduling in Sensor-Aided Cognitive Radio Networks," *IEEE Trans. on Vehicular Technology*, Vol. 61, No. 2, pp. 716-725, 2012.

[81] Y. Liu, S. Xie, R. Yu, Y. Zhang, "An Efficient MAC Protocol with Selective Grouping and Cooperative Sensing in Cognitive Radio Networks," *IEEE Trans. Vehicular Technology*, pp. 3982-3941, 2013.

[82] H. Sun, A. Nallanathan, C. Wang, Y. Chen, "Wideband spectrum sensing for cognitive radio networks: A survey," *IEEE Wireless Communication*, Vol. 20, pp. 74-81, 2013.

[83] Y. Liu, S. Xie, R. Yu, Y. Zhang, X. Zhang, C. Yuen, "Exploiting Temporal and Spatial Diversities for Spectrum Sensing and Access in Cognitive Vehicular Networks," *Wireless Communication and Mobile Computing*, 2013.

ACKNOWLEDGEMENT

I would like to express my sincere gratitude to Prof. Sangman Moh, who has supported me from the very beginning to the end. Without his guidance, this five-year journey would not be fruitful. I also would like to thank the thesis committee: Prof Ilyeong Chung, Prof Seokjoo Shin, Prof Monsoo Kang, and Dr. Yeongwoo Kim for the insightful comments and discussions.

Countless thanks to families, friends, professors, colleagues, and university staffs that have helped me directly or indirectly in completing this thesis. I wish the works presented here may benefit our society. My apologies for any inconvenience I may have caused to anybody during my study and research activities.

This thesis is dedicated to

Mama, 내 인생에 가장 빛나는 버석,
박성철, my biggest supporter,
Cita and Harni, no truer friends.