Efficient Recovery Control Channel Design in Cognitive Radio Ad Hoc Networks

Brandon F. Lo, *Member, IEEE*, Ian F. Akyildiz, *Fellow, IEEE*, and Abdullah M. Al-Dhelaan

***Abstract*—A constantly available control channel facilitates control message exchange and spectrum coordination in cognitive radio (CR) ad hoc networks. When a dedicated control channel is unavailable, a control channel must be dynamically allocated in licensed channels and vacated for the presence of primary users (PUs). As a result, the establishment of such a control channel is a challenge. In this paper, an efficient recovery control channel (ERCC) design is proposed to address this challenge. This heuristic and distributed design approach is essentially based on the observed spectrum homogeneity in a neighborhood. By adaptively updating a list of channels commonly available to neighbors, each secondary user is able to efficiently establish new control channels among neighbors in response to PU activity changes. Therefore, a virtually "always on" control channel robust to PU activity can be realized by the proposed method. The contributions are summarized as follows: 1) The proposed method efficiently recovers control channels from PU activity changes and maintains network connectivity. 2) It extends the control channel coverage to facilitate broadcast and reduce control overhead and delay. 3) It minimizes the interference with PUs. Simulation results show that the proposed solution outperforms the classic group- and sequence-based solutions in the responsiveness to rapidly changing PU activity and the maintenance of connectivity. Furthermore, the increase in control channel coverage and the allocation of the highest quality channels to control channels can be well balanced with reliability and scalability in various network scenarios.**

***Index Terms*—Ad hoc networks, channel allocation, cognitive radio (CR) networks, common control channel (CCC), dynamic spectrum access, neighbor discovery, rendezvous.**

I. Introduction

COGNITIVE radio (CR) networks, having been paid a tremendous amount of research attention recently, is a promising solution to the spectrum underutilization problem [3]. As a dynamic spectrum access method, CR networks are overlaid with licensed networks for opportunistic spectrum access. Specifically, nodes equipped with cognitive radios (CRs), which are known as *secondary users* (SUs), are capable of detecting and utilizing "spectrum holes" in licensed bands when licensed users, which are known as *primary users* (PUs), are not present. In addition, SUs must avoid interfering with PUs when PUs appear in licensed bands. Nevertheless, when deployed to form a CR ad hoc network (CRAHN) without network infrastructure, SUs must cooperate to enhance the detection accuracy of PU activity and coordinate free spectrum access by control message exchange. Thus, establishing a common control channel (CCC) [1]–[3] that facilitates such control information exchange is essential.

While a control channel can be globally predefined outside PU's licensed bands, it may be infeasible in military, emergency relief, or other applications. Without a dedicated control channel, establishing a constantly available CCC in a CRAHN is a major challenge. The challenge arises from two aspects: 1) When a PU is present in the control channel, SUs must vacate the channel immediately and resume their control traffic in an available channel. However, negotiating a new control channel is difficult without the CCC. 2) Due to spectrum heterogeneity, a universally available control channel is unlikely to exist for all SUs. As a result, the coverage of a CCC is limited to a local area. Communication overheads may even increase if neighboring SUs can only be reached by multiple control channels. Therefore, the CCC problem that we address in this paper is maintaining the links with neighbors by dynamic control channel allocations in response to PU activity while maximizing the coverage of a CCC for reduced control signaling efforts.

The CCC design can generally be classified as three design approaches, i.e., *group-based* [6], [7], [15], *sequence-based* [4], [9], and *dedicated* [13], [18] CCC designs. For the group-based approach, a control channel can be selected from channels commonly available to SUs that form a group or a cluster in a local area [19]. Reference [7] proposed a clustering structure with a CCC selected by the cluster head of each cluster. Although the number of clusters and corresponding control channels can be minimized by the proposed cluster optimization algorithm [8], the coverage of a CCC is only limited to the area of a cluster. Moreover, when a PU occupies the control channel of a cluster, cluster reconfiguration is required with high overheads. Thus, the group-based approach by using clustering is generally less efficient in response to a high degree of PU activity. To address this problem, [6] proposed a distributed

Manuscript received December 2, 2009; revised April 10, 2010 and July 9, 2010; accepted August 9, 2010. Date of publication September 7, 2010; date of current version November 12, 2010. This work was supported by the U.S. National Science Foundation under Award ECCS-0900930. The review of this paper was coordinated by Dr. L. Li.

B. F. Lo is with the Broadband Wireless Networking Laboratory, School of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, GA 30332 USA (e-mail: brandon.lo@ece.gatech.edu).

I. F. Akyildiz is with the Broadband Wireless Networking Laboratory, School of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, GA 30332 USA, and also with the College of Computer and Information Sciences, King Saud University, Riyadh 11451, Saudi Arabia (e-mail: ian@ece.gatech.edu).

A. M. Al-Dhelaan is with the College of Computer and Information Sciences, King Saud University, Riyadh 11451, Saudi Arabia (e-mail: dhelaan@ksu.edu.sa).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TVT.2010.2073725

group-based algorithm for CCC assignment. By exchanging channel quality information, SUs in this method adaptively update their choice of control channels according to the decision of the majority of neighbors. As more SUs gradually agree upon the selected control channels, this method reduces the number of control channels in the network and increases the coverage of each CCC. However, when two neighboring nodes observe heterogeneous channel conditions, the CCC assignment may fluctuate with the updates of neighbors' choices. Thus, the performance of this method may not be consistent in all scenarios.

In addition to the group-based design, the sequence-based design aims at establishing multiple control channel links between different neighboring node pairs with predefined channel hopping sequences. For example, [9] constructs sequences by using permutations of available channels. Since the selected sequences have little adaptation to PU activity and new spectrum opportunities, the bounded time for neighboring nodes to locate each other is not guaranteed. Although the sequence-based design may be more robust to PU activity [17], they require significant time to re-establish a new control channel. Thus, this approach may not respond to PU activity and maintain the connectivity in a timely fashion. In addition, sequence-based design establishes connections on a link-by-link basis. Therefore, due to limited CCC coverage, broadcasting a message in a sequence-based CCC incurs high control overhead.

Motivated by research challenges of CCC design [1]–[3], we propose the efficient recovery control channel (ERCC) design solution to address the following issues:

1) *Responsiveness to PU activity:* When a PU is present in a control channel, a new control channel must be immediately established among SUs for network connectivity and seamless operations.
2) *CCC coverage:* The coverage of a control channel determines the range of a message broadcast in the same channel. Thus, a smaller number of control channels in the network reduces channel switching delay and control signaling overhead.
3) *Overhead of CCC establishment:* The overhead of establishing control channels needs to be minimized for reducing control information exchanges.

Our contributions to the CCC design problem are summarized as follows:

- *Efficient recovery from PU activity:* The proposed ERCC design enables the efficient recovery of CCCs upon PU's return to the CCC and is highly responsive to PU activity. With adaptively updated common channel lists (CCLs) at each SU, this method effectively recovers the lost links caused by changes in PU activity and maintains a high degree of connectivity.
- *Balance of maximizing coverage and minimizing interference:* The proposed ERCC design is capable of extending the coverage of a CCC while allocating a control channel of high quality to minimize the interference with PUs. Moreover, the tradeoff between coverage and interference facilitates broadcasts with reduced control efforts and increased broadcast throughput.

TABLE I
TABLE OF NOTATIONS

Notation	Description
A	PU and SU deployment area
a	Decaying coefficient in the exponential correlation model
α	death rate of PU activity (ON to OFF)
B	Channel bandwidth (control or data)
β	birth rate of PU activity (OFF to ON)
C	CCC allocation state
C_i/C_j	The i^{th}/j^{th} licensed channel
Ch_k	The CCC allocated to neighbor k
d_{ij}	Distance between SUs i and j
d_{pi}	Distance between the PU and the SU i
D_p	Density of PU deployment
D_s	Density of SU deployment
γ_i	Accumulated PU interference on channel C_i
γ_{pu}	Sensing threshold for PU receive power
γ_{su}	Sensing threshold for SU receive power
L_{Ci}	Common channel list of SU_i
L_{NB}	List of neighbors with allocated CCCs
L_P	Preferred channel list from local sensing
L_{CC}	The intersection of two broadcast CCLs L_{C_i} and L_{C_k}
L_R	List of channels for control radio channel hopping
N_{best}	Number of CCC links with the best allocated CCC
N_c	Number of licensed channels
N_{disc}	Number of established CCC links
N_r	Number of control capacity regions
N_k	Number of neighbors
N_m	Maximum number of channel switches during recovery
N_p	Number of PUs
N_s	Number of SUs
N_{tot}	Number of total available CCC links
p	Probability of selecting the common channel C_i
P_I	Interference power observed at a PU
P_{su}	SU transmit power
r_k	CCL broadcast rate of SU k
R	CCC recovery state
R_c	Sum-rate capacity of all control channels
$R_j^k(q)$	Maximum control throughput on C_j at SU k in region q
R_p	PU transmission range
R_s	SU transmission range
r_s	Periodic spectrum sensing frequency
S_m	The m^{th} channel in the channel hopping sequence
σ_{dB}	Log-normal shadow fading dB-spread
t_{disc}	Maximum duration for initial neighbor discovery
t_p	PU ON/OFF period
t_C/T_C	CCC allocation time
t_R/T_R	CCC recovery time
w_{jk}	The weight for the CCC link C_j to neighbor k
W_i	The weight of channel C_i for channel ordering

The remainder of this paper is organized as follows: Section II describes the system model for our CCC design. Section III details the ERCC method. Section IV provides a theoretical model for performance analysis and defines three metrics for performance evaluation. Section V evaluates the performance of our proposed method with simulation results. Section VI concludes this paper.

For clarity of notation usage, Table I lists the notations and their descriptions, and Table II lists abbreviations, acronyms, and their corresponding definitions.

II. System Model

For opportunistic spectrum access, a CRAHN is overlaid with a primary network where PUs operate in a set of licensed

TABLE II
TABLE OF ABBREVIATIONS AND ACRONYMS

Acronym	Definition
BCI	Best Channel Indicator
CCC	Common Control Channel
CCI	CCC Coverage Indicator
CCL	Common Channel List
CDF	Cumulative Distributed Function
CLI	CCC link indicator
CRAHN	Cognitive Radio Ad Hoc Network
ERCC	Efficient Recovery Control Channel
GRP	Group-based CCC
PCL	Preferred Channel List
PMF	Probability mass function
PU	Primary User
PUI	PU Interference
SEQ	Sequence-based CCC
STD	Standard Deviation
SU	Secondary User

channels. The number of licensed channels is denoted by N_c. Due to PU activity, the channels available to each SU in the CRAHN may only be a subset of all licensed channels. Thus, SUs rely on local spectrum sensing to observe channel conditions and identify spectrum opportunities.

For spectrum sensing and data transmission, each SU is equipped with two half-duplex transceivers that can be tuned to any licensed channel. One radio, which is called the control radio, is dedicated to allocated control channels, and the other, which is called the data radio, is used for data transmission. Each radio can transmit data, receive data, or sense a channel at a time but cannot perform more than one of these operations simultaneously.

The PU or SU transmit power decays with distance based on the free-space path loss model. When the shadow fading is considered, the combined path loss and shadowing model is used [11]. For correlated shadowing, we use the exponential correlation model [12], and the correlation function is given by $\rho_{ij} = e^{-ad_{ij}}$ [10], where a is the exponential decaying coefficient, and d_{ij} is the distance between SUs i and j. To determine the presence of PUs and neighboring SUs within the transmission range, SUs compare sensing thresholds γ_{pu} and γ_{su} with the receive power of PU and SU signals, respectively. In addition, given the accumulated PU interference (PUI) level γ_i on channel C_i, $i \in N_c$, the channel quality of C_i is *better* than that of C_j if $\gamma_i < \gamma_j$. As a result, the channel C_i is defined as the *best* channel or the channel of the *best* quality if $i = \arg\min_j \gamma_j, j \in N_c$. For simplicity, other types of fading and interference are not considered in this model.

When a PU returns to a control channel, the control radio ensures the detection of PU signals in a timely manner and switches to a new CCC based on the proposed control channel allocation method. Without the dedicated control radio, an SU with a single radio may be unaware of the control channel change because of the use of the only radio for data transmission. In addition, the synchronization of quiet periods for spectrum sensing is negotiated among neighboring nodes in the control channel. Thus, energy detection of PU transmit signals can be enforced at the link layer and performed by the data radio during quiet periods.

The PU activity is modeled as a two-state birth–death process [16], i.e., an ON–OFF model in which an ON-state represents the appearance of any PU, whereas an OFF-state represents the absence of all PUs. If the ON-state switches to the OFF-state with the probability α and the OFF-state switches back with the probability β, the steady-state probability of ON- and OFF-states are $\beta/(\alpha+\beta)$ and $\alpha/(\alpha+\beta)$, respectively. Thus, the state transition is a Poisson process, whereas PU interarrivals are exponentially distributed.

In the next section, we introduce our proposed efficient recovery CCC design.

III. EFFICIENT RECOVERY CONTROL CHANNEL

The ERCC design consists of three major components: 1) *neighbor discovery*; 2) *CCL update*; and 3) *efficient PU activity recovery*. The neighbor discovery process aims at increasing the probability of locating neighbors on common channels for the establishment of initial network topology. The CCL update focuses on maintaining a robust list of common channels by using local sensing and neighbor information on a regular basis. The efforts of CCL updates facilitate the efficient recovery from the PU activity in the event of PU's return to the CCC. These components are described in detail in the succeeding sections.

A. Neighbor Discovery

The system starts with a neighbor discovery process to establish initial network connectivity. During this process, all SUs, which are initially distributed in a set of predefined licensed channels, locate neighboring nodes within their transmission range. To locate any neighbor in the network, each SU obtains a list of available channels from local observations, follows a channel hopping sequence, and hops over available channels. A neighboring pair discovers each other and establishes a link when they hop to the same channel and exchange beacon messages. Thus, the initial network topology is formed after all links are established among neighboring nodes. Next, we describe the construction of channel hopping sequence, handshaking procedure, and neighbor list update in the neighbor discovery process.

1) Channel Hopping Sequence: The channel hopping sequence is a pseudorandom sequence of available channels for frequency hopping during neighbor discovery. To construct such a sequence, an SU starts with a channel list based on local observations of channel availability. The channels in the list are initially of order in decreasing channel quality (known as a *preferred channel list* (PCL) in Section III-B). To maximize the chance of locating neighbors in preferred common channels, the control radio is tuned to channels with the preference according to the channel order. For a CCL L_C of length n (which will also be defined in Section III-B), the probability of selecting $C_i \in L_C$, with bias toward lower index i, is given by

$$Pr(C_i) = \frac{n+1-i}{\sum_{j=1}^{n} j} = \frac{2(n+1-i)}{n(n+1)}, \qquad 1 \le i \le n. \quad (1)$$

Since $Pr(C_i)$ can be considered as the probability mass function (PMF) of a discrete random variable $C = C_i$, a discrete cumulative distribution function (CDF), which is denoted by $F_C(C)$, can be obtained accordingly. Thus, the value of the CDF at $C = C_i$ is given by

$$F_C(C_i) = \sum_{j=1}^{i} Pr(C_j), \qquad 0 \leq i \leq n \tag{2}$$

where $F_C(C_0) = 0$, and $F_C(C_n) = 1$. These values in (2) are used as thresholds in the mapping from the sequence of random numbers $r_m \in (0, 1]$ to the selected channel C_i. Thus, the channel hopping sequence $S_m, m = 1, 2, \ldots$ is generated as follows:

$$S_m = C_i \text{ for } F_C(C_{i-1}) < r_m \leq F_C(C_i),\ m = 1, 2, \ldots. \tag{3}$$

Since channels with higher preference in the CCL appear more often in the channel hopping sequence, SUs locate their neighbors in a channel common to more neighbors with higher probability.

2) Handshaking Procedure: In the neighbor discovery process, each SU follows its hopping pattern and tunes to one channel at a time. During the time interval, the SU broadcasts a beacon with random backoff and listens to the channel for any beacon broadcast. If the SU receives a beacon from a neighbor, it replies with an Ack message. Similarly, the SU receives an Ack if its neighbor receives the beacon. The beacon notifies neighbors of the SU's ID and its CCL while the Ack message ensures that the neighbor discovery between a neighboring pair is mutually recognized. Therefore, a link is established in a common channel between the neighboring pair after the beacon and Ack exchanges.

3) Neighbor List Update: After the neighboring pair completes the handshaking procedure, each SU's neighbor list is updated accordingly. The neighbor k is added to the neighbor list if it is new to the list. The control channel associated with this neighbor, which is denoted by Ch_k, is updated with the allocated control channel. The allocated CCC may be different from the channel that the neighboring pair meets because a channel that can reach more neighbors or has higher quality is preferred.

Since each end of the link obtains its own and neighbor's broadcast CCLs after beacon exchange, the neighboring pair can individually generate a set of channels, which are denoted by L_{CC}, from the intersection of those two broadcast CCLs. The best channel of L_{CC} is allocated as the CCC to the link. Thus, identical decision of this CCC allocation can be individually determined by the neighboring pair based on the same L_{CC}. No further control message exchange is required.

The neighbor discovery process is highlighted in Algorithm 1. In line 4, t_{disc} is the maximum duration for initial neighbor discovery. Lines 7–9 outline the handshaking procedure, whereas lines 10–15 show the neighbor list update and initial control channel assignment. The *OrderChannel* function in line 13 reorders the channels based on the ordering rules, which will be described in the next section.

Algorithm 1: Neighbor Discovery

1: PCL $L_{Pi} \leftarrow$ LocalSensing (γ_{pu})
2: CCL $L_{Ci} \leftarrow L_{Pi}$
3: $\{S_m\} \leftarrow$ SequenceGenerator (L_{Ci})
4: **while** NbrDiscoverTimer $\leq t_{\text{disc}}$ **do**
5: SwitchChannelTo (S_m)
6: RandomBackoff and SendBeacon (i, L_{Ci})
7: **if** ReceiveBeacon (k, L_{Ck}) from neighbor k **then**
8: SendAck (i, L_{Ci}, k)
9: **end if**
10: **if** ReceiveBeaconOrAck (k, L_{Ck}, i) **then**
11: $L_{NBi} \leftarrow L_{NBi} \cup \{k\}$
12: $L_{\text{CC}} \leftarrow L_{Ci} \cap L_{Ck}$
13: $L_{Ci} \leftarrow OrderChannel\ (L_{Ci}, L_{\text{CC}}, \gamma_j)$
14: $Ch_k \leftarrow \arg\min_j\{C_j | C_j \in L_{\text{CC}}\}$
15: **end if**
16: **end while**

B. CCL Update

Since neighboring SUs usually observe homogeneous channel availability in CRAHNs, each SU can individually obtain a similar list of available channels. Intuitively, channels in those lists common to a large number of neighboring nodes are the candidates for CCC allocations. Thus, the advantages of maintaining an ordered list of common channels are twofold: 1) selecting the channel common to the largest number of neighbors as the control channel increases the coverage of the CCC and, more importantly, and 2) when a PU occupies in the control channel, the common channel with the highest preference from the list can be immediately allocated as the new control channel. With such an allocation, most neighbors can immediately locate each other in the new control channel. Therefore, efficient recovery from PU activity can be achieved by CCL updates.

Each SU constructs and maintains a CCL for periodic broadcasts to neighbors and dynamic CCC allocations. In general, a CCL is a list of channels commonly available to at least one neighbor. The order of the list is determined by the weight and the quality of the channels. The weight of a channel C_i, which is denoted by W_i, is the number of neighbors having C_i in their CCL. It indicates the number of neighbors an SU could reach if the channel is allocated as the CCC. Equivalently, it represents the preference of choosing the channel as a CCC in the neighborhood. Therefore, the channel order of a CCL follows two rules: 1) All channels in the CCL are of monotonically decreasing order according to W_i. 2) If two channels have the same weight, their order is determined by the PUI level γ_i. In other words, C_i is preferred to C_j, $i \neq j$, if $W_i > W_j$ or $W_i = W_j$, and $\gamma_i \leq \gamma_j$.

To construct a list of channels commonly available to neighbors, an SU requires its own observations of channel availability, i.e., a list called the *preferred channel list* (PCL), and neighbors' preference of available channels. Therefore, SUs update their CCL when they obtain a new PCL from local sensing or receive a CCL from its neighbor's broadcast.

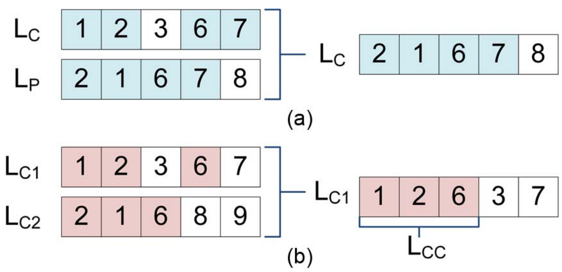

Fig. 1. Examples of CCL update. (a) CCL update with PCL. (b) CCL update with neighbor's CCL.

1) CCL Update With Local Sensing Information: To obtain a PCL, an SU senses each licensed channel, determines PU-occupied ones, and returns with a list of available channels on the order of the observed channel quality. A PCL, which is denoted by L_P, is a channel list of observed quality in monotonically decreasing order. Since the channel quality in channel C_i is inversely proportional to the total received power of PU transmit signals γ_i, a PCL of length n is defined as

$$L_P = \{C_i | 1 \leq i \leq n \text{ for } \gamma_1 \leq \cdots \leq \gamma_n \leq \gamma_{\text{pu}}\} \qquad (4)$$

where γ_{pu} is the PUI threshold that determines the PU's presence in a channel. Since PU-occupied channels are excluded from the PCL, all channels in L_P are presumably available unless PUs change their operating location or channel.

After obtaining a PCL from local sensing, an SU updates its CCL with the PCL. The CCL is initially set to the PCL and successively updated by new PCLs from periodical sensing. The update is essential for the following two reasons: 1) New PU-occupied channels that no longer exist in the PCL should be removed from the CCL. 2) Newly available channels should be added to the CCL for neighbor notification. Thus, a CCL after the update reflects the most up-to-date channel conditions.

Mathematically, given a CCL L_C and a PCL L_P, the removal of PU-occupied channels is given by

$$L_C \leftarrow L_C \setminus \{C_j | C_j \in L_C \text{ and } C_j \notin L_P\}. \qquad (5)$$

On the other hand, the addition of newly available channels is given by

$$L_C \leftarrow L_C \cup \{C_j | C_j \in L_P, C_j \notin L_C, \text{ and } W_j = 0\}. \qquad (6)$$

Notice that the weight W_j associated with the newly added channel C_j is initialized to zero. The channel order of the updated L_C follows the channel order rules.

Fig. 1(a) shows an example of the CCL update with a PCL. In the figure, channel 3 in L_C before the update is removed because it is unavailable in L_P. Furthermore, channel 8 in L_P is added to the CCL because it is a newly available channel that may also be available to neighbors. However, the weight associated with this channel is set to 0. This is represented by the box in white (no shade) that contains channel 8. Finally, channels in the CCL are sorted according to the preference in L_P.

2) CCL Update With Neighbor's Information: In addition to updating their CCL with sensing information, SUs update their CCL when they receive a CCL from a neighbor. The update is required for the following two purposes: 1) The update determines a list of common channels shared with neighbors. 2) The information of neighbors' common channel preference can be collected and combined by each SU for dissemination. Thus, the updated CCL reflects new preference of common channels in the neighborhood.

When an SU i updates its CCL L_C with its broadcast CCL L_{Ci} and neighbor k's CCL L_{Ck}, the SU first generates a list of common channels from L_{Ci} and L_{Ck} as follows:

$$L_{\text{CC}} \leftarrow L_{Ci} \cap L_{Ck}. \qquad (7)$$

For each C_j in L_{CC}, the corresponding weight in L_C is set for the neighbor k, i.e.,

$$L_C \leftarrow \{C_j | W_j : w_{jk} = 1 \text{ for } C_j \in L_{\text{CC}}\} \qquad (8)$$

where $W_j = \sum_{k=1}^{N_k} w_{jk}$, and N_k is the number of neighbors. As in the previous case, the order of L_C follows the channel order rules.

Fig. 1(b) shows the CCL update with a CCL from neighbor 2. As shown in the figure, the common channels of two CCLs are channels 1, 2, and 6. The weights associated with neighbor 2 are set accordingly. Since channels 3 and 7 are unavailable to neighbor 2, their weight remains 0. Thus, the resulting channel order reflects the new weights in the CCL.

The CCL update is listed in Algorithm 2. Lines 2–4 show the addition or the removal of channels for updates with sensing information (PCL). On the other hand, lines 7–11 outline the updates with neighbor's information (CCL).

Algorithm 2: CCL Update

1: Update with SU i's Preferred Channel List L_P:
2: $L_P \leftarrow$ LocalSensing (γ_{pu})
3: $L_C \leftarrow L_C \setminus \{C_j | C_j \in L_C \text{ and } C_j \notin L_P\}$
4: $L_C \leftarrow L_C \cup \{C_j | C_j \in L_P, C_j \notin L_C, \text{ and } W_j = 0\}$
5:
6: Update with Neighbor k's CCL L_{Ck}:
7: **if** ReceiveBeacon (k, L_{Ck}) from neighbor k **then**
8: $\quad L_{\text{CC}} \leftarrow L_{Ci} \cap L_{Ck}$
9: $\quad L_C \leftarrow \{C_j | W_j : w_{jk} = 1 \text{ for } C_j \in L_{\text{CC}}\}$
10: $\quad L_C \leftarrow OrderChannel\,(L_{Ci}, L_{\text{CC}}, \gamma_j)$
11: **end if**

C. *Efficient PU Activity Recovery*

In this section, we discuss the efficient recovery from the return of a PU to a CCC. The recovery consists of three steps: 1) new CCC allocation from the CCL; 2) neighbor list update for lost neighbors; and 3) control radio adaptation for recovering neighbors.

1) Control Channel Allocation: Due to their dynamic behavior, PUs are highly likely to occupy those established CCCs. Thus, the primary goal is to utilize CCLs for efficient recovery when PUs are present in the CCCs. When a PU in the vicinity occupies a CCC, SUs tuned to this control channel can immediately detect the change. Without sending any message that may cause interference, SUs choose the best channel in their CCL as the new CCC after the PU-occupied CCC is removed from the list. That is, for $C_j \in L_C, Ch_k \leftarrow \min_j C_j$, with $w_{jk} = 1$. Since SUs can reach all or most neighbors by the new CCC, most neighbors that detect the change in the neighborhood will switch to the new CCC and locate each other by beacon broadcasts. With the exchange of CCLs, most neighbors can be recovered in the new CCC to maintain the network connectivity to the maximum degree.

2) Neighbor List Update: The CCC links associated with each neighbor in the neighbor list show the status of this recovery. If a neighbor k is not yet recovered, the associated CCC Ch_k is a channel that is no longer available in the CCL. By using the criteria $Ch_k \notin L_C$, we can adaptively change the operating channel of the control radio to recover neighbors in other common channels. In addition, due to PU activity, some existing neighbors may be unable to reach in any available channel. In this case, those neighbors should be removed from the neighbor list after having no CCL arrival for a certain period of time.

3) Control Radio Adaptation: A control radio list, which is denoted by L_R, is a list of channels to which the control radio will be tuned based on the probability of channel selections. If the CCL or neighbor list updates approach a steady state, L_R only includes the allocated CCCs to reduce the switching overhead. In other words, L_R is simply the union of all channels in Ch_k, which is a small subset of L_C with the best case of single channel, as follows:

$$L_R \leftarrow \cup_{k=1}^{N_k} \{Ch_k\} \tag{9}$$

where N_k is the number of neighbors. For efficient recovery, the radio list is set to the CCL when the allocated CCCs no longer exist in the list, as follows:

$$L_R \leftarrow L_C \text{ for some } Ch_k \notin L_C. \tag{10}$$

Similar to the neighbor discovery process, the probability of selecting the channel from L_R is given in (1). Thus, the control radio is tuned to the CCC that reaches most neighbors with the highest probability.

The efficient PU activity recovery is listed in Algorithm 3. Lines 4 and 5 are new CCC allocations and control radio update in response to PU activity when neighbors can be recovered by the new CCL. Lines 8–11 show the neighbor list update and control radio adaptation when neighbors cannot be completely recovered by the new CCC. In this case, a neighbor recovery procedure similar to Algorithm 1 is required to locate lost neighbors or new ones.

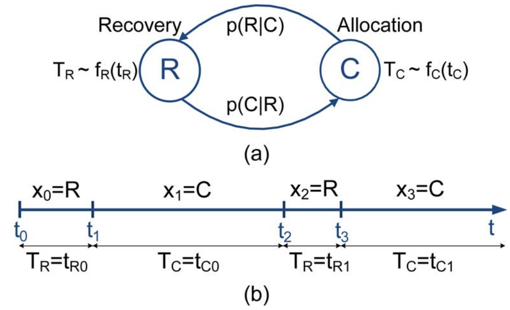

Fig. 2. (a) Semi-Markov chain and (b) alternating renewal process for ERCC performance analysis.

Algorithm 3: Efficient PU Activity Recovery

1: $L_P \leftarrow$ LocalSensing (γ_{pu})
2: $L_C \leftarrow$ UpdateCCL (L_P)
3: For neighbor k recovered by new CCL:
4: $Ch_k \leftarrow \min_j C_j \in L_C$ with $w_{jk} = 1$
5: $L_R \leftarrow \cup_{k=1}^{N_k} \{Ch_k\}$
6:
7: For neighbor k not recovered by new CCL:
8: **if** $Ch_k \notin L_C$ **then**
9: $L_{NB} \leftarrow L_{NB} \setminus \{k\}$
10: $L_R \leftarrow L_C$
11: **end if**
12: $C_j \leftarrow$ SelectChannel (L_R)
13: SwitchChannelTo (C_j)
14: Neighbor discovery as Algorithm 1

IV. Performance Analysis

In this section, we analyze the performance of the proposed scheme by utilizing a mathematical model for delay, throughput, and interference analysis. Moreover, we provide the overhead analysis by comparing our solution with existing grouping and clustering methods and cosite interference analysis to address the interference issue between the collocated control and data radios.

A. Analytical Model

To facilitate the performance analysis, we model the CCC recovery and allocation between a neighbor pair as a two-state semi-Markov process. Fig. 2(a) shows the state diagram of the semi-Markov chain with two states: *Recovery* and *Allocation*. The Recovery state, which is denoted by R, is the state when SUs are locating neighbors in the initial neighbor discovery phase or recovering from the lost of CCC upon the PU's return. The Allocation state, which is denoted by C, is the state when a CCC is allocated to the link between the neighbor pair. The sojourn time in state R, which is called the *CCC recovery time* and denoted by T_R, is a random variable with the distribution $f_R(t_R), t_R > 0$. Similarly, the CCC allocation time T_C is defined as the sojourn time in state C with distribution $f_C(t_C), t_C > 0$. The transition probabilities

$p(C|R)$ and $p(R|C)$ are unity in this model. As shown in Fig. 2(b), by alternatingly staying in each of the two states, the resulting process is essentially an alternating renewal process. For simplicity, we assume that the initial neighbor discovery has the same distribution as other recovery periods.

The average expected recovery time $E[T_R]$ is of great importance because it is the delay of recovering the lost CCC and the indicator of CCC recovery efficiency. To find $E[T_R]$, one needs to determine $f_R(t_R)$. The closed-form expression for $f_R(t_R)$, which is given in Proposition 1, is related to parameters such as PU activity, PU and neighbor locations, PUI, channel conditions, and number of channels. Here, we assume that the SU u and its neighbor SU k detect PU correctly and simultaneously when PU changes from inactive to active in channel C_i. We further assume that after C_i is removed from their CCLs, C_j is the only channel in common. If both SUs have more than one channel in common, the probability of meeting each other on a common channel is higher, and the recovery time is smaller. Thus, our assumption is the worst-case scenario.

When the CCLs of the neighbor pair have an identical best channel, e.g., C_j, the recovery is instant. Otherwise, the neighbor pair follows the neighbor discovery procedure and requires the recovery time to meet in C_j. If the probabilities of choosing C_i for SUs u and k are p_1 and p_2, respectively, the probability of SUs meeting on C_i is given by $p = p_1 p_2$. The probabilities p_1 and p_2 can be obtained from (1). They are, in general, not identical because the L_C or L_R of the neighbor pair are of different lengths and orders. Assume that the success of meeting each other at the mth channel switch is a discrete random variable that is denoted by M. If the neighbor pair experiences $m-1$ failures for previous $m-1$ channel switches and succeeds at the mth switch, the probability of successful rendezvous on channel C_i after the mth channel hopping is given by

$$P(M = m) = (1-p)^{m-1} p. \tag{11}$$

This is the PMF of random variable M, which is geometric distributed. Based on these observations, we obtain the distribution of T_R in Proposition 1. Based on the Proposition, one can obtain the average recovery time numerically.

Proposition 1: If the CCC recovery time T_R is the sum of M identically, independently, and exponentially distributed random variables with parameter λ, the distribution of CCC recovery time T_R, which is denoted by $f_R(t_R)$, is given by

$$T_R \sim f_R(t_R) = \sum_{m=1}^{\infty} \Gamma\left(t_R; m, \frac{1}{\lambda}\right) P(M = m) \tag{12}$$

where $\Gamma(t_R; m, \mu)$ is the gamma distribution with shape parameter m and scale parameter μ, and M is a geometric-distributed random variable with the PMF given by (11).

Proof: Consider that the CCC recovery time T_R is divided into M intervals, i.e., T_i, $1 \le i \le M$, where M is a discrete variable with the PMF given by (11) and denotes the number of channel switches required for the neighbor pair to successfully meet each other on one common channel. Assume that the duration of each interval is exponentially distributed with parameter λ, which is denoted by $T_i \sim Exp(\lambda)$. As a result, T_R is the sum of M exponentially distributed intervals given by $T_R = \sum_i^M T_i$. For each value of $M = m$, T_R, given that $M = m$ is gamma distributed with parameters m and $1/\lambda$, is denoted by $T_R(M = m) \sim f_R(t_R|M = m) = \Gamma(m, 1/\lambda)$. Therefore, by calculating the joint distribution $f_R(t_R, M) = f_R(t_R|M = m)P(M = m) = \Gamma(m, 1/\lambda)P(M = m)$ and summing over all m's, we obtain the marginal distribution (12). ■

In practice, the number of channel switches $M = m$ will not be infinite. For a large m, the probability $P(M = m)$ is negligible. Thus, the summation in (12) starts from 1 to the maximum number of channel switches N_m and $\sum_{m=1}^{N_m} P(M = m) \cong 1$. The resulting distribution is the linear combination of gamma distributions with different parameters m.

For the allocation time T_C, it is mainly determined by the PU activity, particularly the PUs' arrival rates. This is because, once a new CCC is allocated, the allocation will mostly remain unchanged until the PU's return to the channel. Even when an SU decides to change the CCC, the allocation time continues on the new CCC and thus has no state change in this case. Thus, we obtain the distribution of T_C in Proposition 2. Based on the Proposition, one can easily calculate the average allocation time as $1/(N_p \beta)$.

Proposition 2: Given N_p PUs with the rate of changing from inactive to active β, the distribution of CCC allocation time T_C, which is denoted by $f_C(t_C)$, is given by

$$T_C \sim f_C(t_C) = Exp(N_p \beta). \tag{13}$$

Proof: Given N_p inactive PUs on channel C_j, the SU selects C_j as the CCC and enters the Allocation state. Since each PU arrival follows Poisson distribution with rate β (becoming active with rate β), the arrival rate of N_p PUs is Poisson distributed with $N_P \beta$. As a result, the interarrival time between two PU arrivals is exponentially distributed with parameter $N_p \beta$. Based on the assumption that C_j is available, all active PUs must become inactive before the SU switches to C_j. Moreover, due to the memoryless property of exponential distribution, the PU inactive time before the SU switches to C_j is irrelevant. Thus, we obtain the distribution of allocation time $f_C(t_C)$ as in (13). ■

B. Delay, Throughput, and Interference

To find the delay and control throughput, we assume that SU i transmits on the CCC C_j to a neighbor k. The maximum achievable rate for the control transmission is given by

$$R_j^k = B \log\left(1 + \frac{P_{\mathrm{su}} |h_{ik}|^2}{N_0 B + \gamma_j^k}\right) \tag{14}$$

where B is the channel bandwidth, P_{su} is the SU transmit power, h_{ik} is the channel gain of the link between SUs i and k, $N_0/2$ is the power spectral density of additive white Gaussian noise, and γ_j^k is the accumulated interference power of PU transmit signals observed by SU k on C_j. If the SU has N_k neighbors within its transmission range R_s and all neighbors are tuned to C_j, we refer to the area covered by R_s as a *control capacity region*. Since the achievable throughput is limited by the rate of the weakest link, where the interference power γ_j^k is the largest and the channel gain h_{ik} is the smallest, the maximum achievable throughput in the capacity region is given by $R_j = \min\{R_j^k, k = 1, \ldots, N_k\}$. If the control packet is of length L bits, the transmission delay is L/R_j.

Now, if N_s SUs are uniformly deployed in the area A, there are approximately $N_r = A/(\pi R_s^2)$ control capacity regions. The CCC may be the same in each region, whereas the PUI levels and the channel conditions are different. Thus, the maximum control throughput of the CRAHN, which is called the sum-rate capacity, can be obtained by the maximum sum of rates from all regions $R_j(q)$, $q = 1, \ldots, N_r$, as follows:

$$R_c = \max_{R_j(1),\ldots,R_j(N_r)} \sum_{q=1}^{N_r} R_j(q), \qquad C_j \in \{1, \ldots, N_c\}. \tag{15}$$

ERCC intelligently selects the CCC C_j in each region such that the sum-rate capacity (15) is achieved.

If a PU is active on channel C_j and the area covered by PU's transmission range R_p is called the *protected region*, the interference with PUs results only from those transmitting SUs outside the protected region. Since there are only $N_r' = (A - \pi R_p^2)/(\pi R_s^2)$ possible control capacity regions and one transmitting SU in each region, the maximum accumulated interference from those SUs observed by the PU on C_j is

$$P_I = K \sum_{i=1}^{N_r'} P_{\mathrm{su}} \left(\frac{d_0}{d_{pi}}\right)^{\delta}, \qquad d_{pi} > R_p \quad \forall i \tag{16}$$

where K is an antenna-related constant, P_{su} is the SU transmit power, d_0 is the reference distance, d_{pi} is the distance between the PU and the SU i, and δ is the path loss exponent.

C. Overhead

The overhead of the ERCC algorithm is dominated by regular broadcasts of maintained CCLs. The frequency of the broadcasts determines the accuracy of channel conditions in the CCLs and the overhead incurred by ERCC. Thus, the choice of the broadcast rate is essential for reducing the unnecessary broadcast overhead.

The CCL updates and broadcasts are related to PU activity and the broadcast rate of neighbors since CCLs are updated with the following: 1) the PCL from the sensing results and 2) the CCL received from neighbors. Assume that an SU has N_k neighbors and that neighbor k, $k = 1, \ldots, N_k$, broadcasts its CCL with the rate r_k. The death and birth rates of the PU model are α and β, respectively. Since the periodic sensing frequency r_s must be no less than the PU activity change rate and SUs need to broadcast CCLs with new channel conditions, we assume that $r_s = \max\{\alpha, \beta\}$, where α, $\beta \le r_k$ $\forall k$. Thus, the rule of thumb for choosing the broadcast rate r_i is formulated as follows:

$$r_s = \max\{\alpha, \beta\} \le r_i \le \min\{r_1, \ldots, r_{N_k}\}. \tag{17}$$

By adaptively selecting the broadcast rate based on (17), the SU and its neighbors broadcast CCLs with the rates gradually approaching the PU activity change rate α or β, whichever is the largest, to minimize the unnecessary broadcasts.

D. Cosite Interference

As described in Section II, each SU is equipped with two radios dedicated to control and data channels, respectively. Due to their collocation in each SU, the out-of-band (OOB) emission [20] from a transmitting radio (control or data) can block the transmission or corrupt the reception at the other radio operating in a different channel within the same band [14]. This phenomena, which is called *cosite interference*, results in degraded performance: an unreliable CCC and data transmissions with compromised data rate.

Contrary to the setting in [14], where all radios are used for data transmissions, ERCC utilizes separate radios for control and data. The key differences in ERCC are twofold: 1) To ensure the reliability of the CRAHN, any operation in a CCC has higher priority than those in data channels. 2) Since all channels have the same bandwidth, CCC transmissions are short, compared with data transmissions. Thus, medium access control (MAC) techniques [20] such as prioritized time sharing, power control, and dynamic channel allocation power control can be utilized to ensure the reliability of CCC and mitigate the cosite interference, whereas the throughput of the data channel is not considerably compromised as follows:

1) Prioritized Time Sharing: When both control and data radios are transmitting, only one radio is active at a given time [14]. Due to its high priority, whenever the control radio transmits, it temporarily refrains the transmission in the data channel. The throughput of data transmissions is only slight for short control traffic. Similarly, the data radio temporarily stops transmitting whenever the control radio starts receiving control data. If the control radio transmits when the data radio receives data from others, the control radio notifies the transmitting neighbor to temporarily stop the data transmission. Moreover, the SUs can broadcast their regular control traffic schedule to neighbors so that their schedulers automatically refrain data transmissions during the scheduled control traffic period.

2) Power Control and Rate Adaptation: To reduce the power leakage from the receive data channel to the control channel, the control radio notifies the transmitting neighbor to perform transmission power control and adjust the transmission rate in the data channel for cosite interference mitigation.

3) Dynamic Channel Allocation: The data channel can be dynamically reallocated to the one far separated from the CCC, if possible, to further reduce the cosite interference. The control channel can also be dynamically changed if the CCC quality degrades.

E. Performance Metrics

The performance of CCC establishment can be evaluated in a variety of ways. For convenience of performance evaluation, we define four metrics in this section: 1) CCC link indicator (CLI); 2) CCC coverage indicator (CCI); 3) best channel indicator (BCI); and 4) PUI.

1) CLI: A link is said to be available between two SUs if they are located within their transmission range and observe at least one channel in common but have not located each other in any channel. When neighboring nodes operate and exchange information in a common channel, a CCC link is established. Thus, we define a CLI as the percentage of established CCC links over all available links in the network as follows:

$$\text{CLI} = N_{\text{disc}}/N_{\text{tot}} \tag{18}$$

where N_{disc} is the number of current established CCC links, and N_{tot} is the number of total available links. Since topology and neighbors change as PU activity changes, the CLI also indicates how fast SUs establish links with new neighbors and recover links with old neighbors. Moreover, the CLI value achieves unity when all neighbors are discovered and CCC links are established. Therefore, this indicator is an alternative way of evaluating neighbor discovery rate and the responsiveness to PU activity.

2) CCI: The coverage of a CCC refers to an area covered by links allocated to a control channel. Since obtaining exact footprint of those links is nontrivial, a CCI is used as an alternative to evaluate the coverage of CCC distribution in the network. The CCC distribution refers to the number of CCC links distributed in all licensed channels. Thus, the CCI is defined as follows:

$$\text{CCI} = STD(p_{\text{dist}})/STD(p_{\text{best}}) \tag{19}$$

where $STD(p_{\text{dist}})$ is the standard deviation (STD) of current CCC distribution over all licensed channels, and $STD(p_{\text{best}})$ is the STD of the CCC distribution in the best case. If p_i is the number of CCC links in channel C_i, $\overline{p}$ is the average number of all p_i's, and N_c is the number of licensed channels, the STD of CCC distribution p is defined as $STD(p) = \sqrt{(1/N_c)\sum_{k=1}^{N_c}(p_i - \overline{p})^2}$. For different numbers of licensed channels or available links, the STD of the distribution can vary significantly. Thus, for easy comparison of different test cases, $STD(p)$ can be normalized by the STD of CCC distribution in the best scenario. The best case can be achieved when all SUs use a single CCC. That is, $p_i = N_{\text{tot}}$, and $p_j = 0$ for $i \in \{1, \ldots, N_c\}$, $1 \leq j \leq N_c$ and $i \neq j$. Evidently, the STD of CCC distribution in the best scenario is the maximum for a given number of channels N_c. Therefore, the CCI indicates how close current CCC distribution to the distribution in the best case and the CCI value achieves unity when all CCC links in the network are established in the same channel.

3) BCI: The BCI indicates the percentage of CCC links to which the best quality channel observed at each SU is allocated. Thus, the BCI is defined as follows:

$$\text{BCI} = N_{\text{best}}/N_{\text{tot}} \tag{20}$$

where N_{best} is the number of CCC links to which the best quality channel in the PCL is allocated, and N_{tot} is the number of total available links. The BCI value achieves unity when all the CCC links are of the highest observed channel quality.

4) PUI: The PUI indicates the average accumulated PU transmit signal power observed on channel C_j at each SU when C_j is used for control transmission and is given by

$$\text{PUI} = \sum_{i=1}^{N_s} \gamma_j^i / N_s, \qquad C_j \in \{1, \ldots, N_c\} \tag{21}$$

where N_s is the number of SUs, N_c is the number of channels, and γ_j^i is the accumulated interference power on the control channel C_j at SU i. Since higher PUI level implies the higher possibility of PUs in the surrounding area, this metric indicates the average level of interference with PUs per SU during control transmission. Moreover, due to the interference at the SU, this metric can also be used to evaluate the level of achievable control throughput. In general, the higher the PUI level, the lower the control throughput.

In the next section, we evaluate the performance of our proposed method with the metrics defined in this section.

V. Performance Evaluation

In this section, we discuss the simulation setups and evaluate the performance of our proposed ERCC method under several test scenarios. We first introduce our simulation environment, compare the analytical model with the simulation model, and then describe six test cases for performance evaluation.

A. Simulation Environment

In our simulation environment, we assume that a number of SUs are randomly deployed in a square area of 500 m $\times$ 500 m sharing a set of licensed channels with PUs in the 5.2-GHz frequency band. Both PU and SU transmit powers are set to 0.1 W. The PU and SU interference thresholds are set to γ_{pu} and γ_{su}. These settings correspond to PU and SU transmission ranges R_p and R_s, respectively. For example, for $\gamma_{\text{pu}} = -72.7$ dBm, $\gamma_{\text{su}} = -66.7$ dBm, and wavelength $\lambda = 0.058$, the PU and SU transmission ranges are approximately 200 and 100 m, respectively. The noise floor is set at -101 dBm. For correlated shadowing, the decaying coefficient a in the exponential correlation model is set to 0.002 for suburban settings [10]. This corresponds to the decorrelation distance of approximately 346 m, where the correlation drops to 0.5 and ensures that the observation of neighbors are highly correlated ($\rho_{ij} > 0.8$). For convenience, the number of PUs, SUs, and licensed channels are denoted by N_p, N_s, and N_c, respectively. In addition, the PU density, PU ON/OFF period, PU transmission range, SU transmission range, and lognormal shadowing decibel spread are denoted by D_p, t_p, R_p, R_s, and σ_{dB}, respectively.

For performance comparison, we select a group-based CCC design approach from [6], which is denoted by *GRP*, and a sequence-based approach from [9], which is denoted by *SEQ*, as references. These two selected reference approaches are summarized as follows:

1) *GRP:* SUs exchange quantized channel quality information by sending Hello messages to neighbors. Based on the channel quality values received from neighbors, SUs adaptively update a probability list for control channel

selection. The probability for a channel is higher if more neighbors select that channel as the control channel. The channel with the highest probability is selected as the CCC. Thus, control channels are selected according to the decisions of the majority of neighbors. The settings used in GRP are given as follows: $A = 0.1$, $B = 1.5$, and $C = 4$ for probability list update. The number of quantized receive power levels for determining quality values is 128.

2) *SEQ:* Each SU constructs a channel hopping sequence by using permutations of available channels. A neighboring SU pair establishes a control link after both SUs hop to the same channel and exchanges information. To establish other control links, both SUs hop to other channels based on their own sequence. If the channel is occupied by a PU, the channel is removed from the hopping sequence. New sequence is generated for new channel availability obtained from local sensing information.

The PU activity follows the two-state birth–death process with a birth rate of 0.3 and a death rate of 0.2. In this case, PUs fix their location and operating channels but may be active or inactive based on the state of the process. When a PU is inactive, the PU-occupied channel is considered free until the PU is active. The degree of PU activity is determined by the ON/OFF period t_p.

The observation time for each topology is set for 10 min. For neighbor discovery and message exchange, each SU is tuned to a channel for 200 ms. During the time interval, SUs perform local spectrum sensing, broadcast channel, and neighbor information; determine new CCLs; and allocate available channels to CCCs accordingly. The metrics are collected every 200 ms after SEQ changes its hopping channel. All results are averaged over the observation time and ten random network topologies, in which PUs and SUs are uniformly distributed in the deployment area. Although the synchronization of SUs is not required, all nodes are activated simultaneously in the test cases.

B. Comparison of Analytical and Simulation Models

To compare the analytical model introduced in Section IV-A with the simulation model, we focus on a neighboring SU pair and their average CCC recovery time. In the analytical model, the average recovery time is obtained by calculating the expected recovery time numerically using the distribution from (12) with the maximum channel switches N_m set to 50. In the simulation model, the recovery time of a neighboring pair is averaged over all occurrences of CCC recovery during the entire observation time and the random network topologies under testing.

Fig. 3 shows the comparison of the average recovery time from the analytical model and the simulation model under various degrees of PU activity characterized by the probability of PU ON states P_{ON}. In general, the CCC recovery time is linearly increased with the number of available channels. This is because the SU may choose other available channels that are not common to the neighbor of interest, resulting in the increase in the average recovery time, even though the probability of choosing the common channel in the CCL is the largest. In addition, given a number of available channels, the average recovery time does not vary significantly under different levels of PU activity. Although the PU activity affects the channel availability and the probability of channel selections, the recovery time is dominated by the number of channel switches once the available channels are determined in the CCL. Thus, the probability of selecting the common channel [p in (11)] for recovery remains constant if there is no CCL update, owing to the PU activity. More importantly, the figure shows that the empirical values from the simulation model closely follow the analytical values as the number of channels varies. Therefore, the analytical model provides the first-order analysis and prediction of the average CCC recovery time.

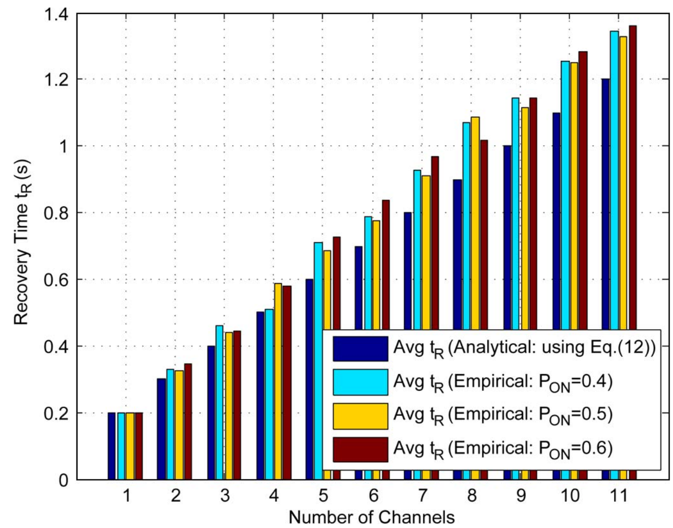

Fig. 3. Comparison of the average CCC recovery time in analytical and simulation models.

C. Test Cases

To evaluate the performance, we test our proposed ERCC solution in the following test cases: 1) PU ON/OFF period; 2) PU transmission range; 3) PU density as the number of PUs per channel; 4) SU transmission range; 5) the scalability or the density of SU population; and 6) shadow fading for a range of decibel spread. These test cases will show how our solution performs under the impacts of PU activity, network topology changes, and channel impairments. The configuration used in each test case for cross reference is $N_p = 10$, $N_s = 60$, $N_c = 10$, $D_p = 1$, $t_p = 4$ s, $R_p = 200$ m, $R_s = 100$ m, and $\sigma_{\mathrm{dB}} = 0$ dB. In each test case, we evaluate the performance of all three methods by varying one of the parameters and illustrate the average and STD of metric values versus the parameter of interest. In the figures of this section, the top left, top right, bottom left, and bottom right subfigures show the CCC links, CCC coverage, best channel, and PUI metric values, respectively.

1) PU ON/OFF Period: The PU ON/OFF period is the smallest duration of a PU being active or inactive. Based on the states in the birth–death process, a PU may be consecutively active

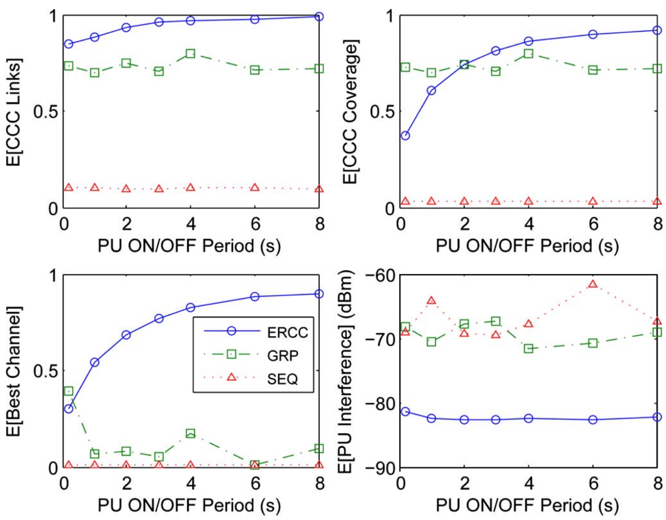

Fig. 4. Expected metric values versus PU ON/OFF period t_p.

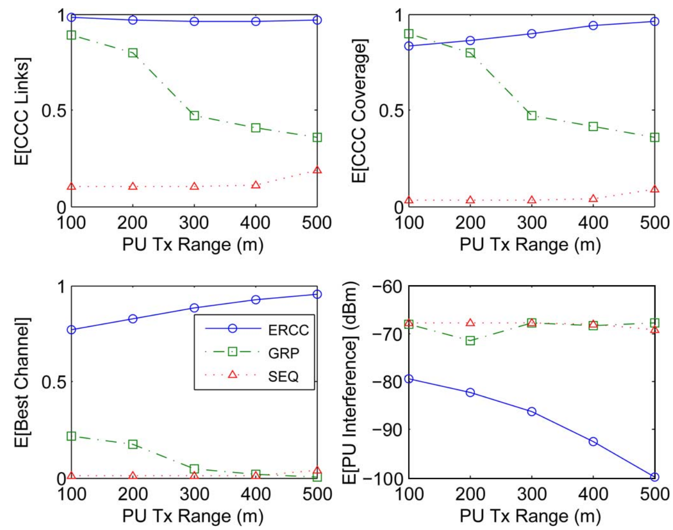

Fig. 5. Expected metric values versus PU transmission range R_p.

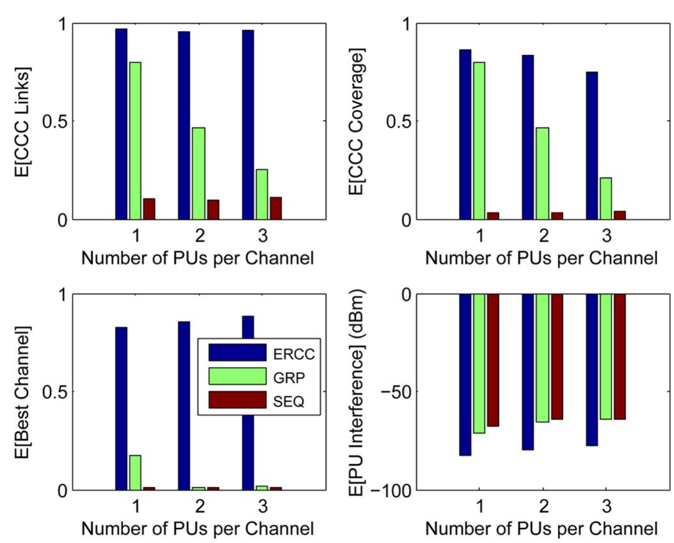

Fig. 6. Expected metric values versus the number of PUs per channel D_p.

or inactive for several periods. During these periods, the PU activity can be considered stationary. Thus, increasing the period reduces the frequency of dynamic changes in PU activity. Fig. 4 shows the four expected metric values of three methods under testing in the range of PU ON/OFF period from 0.2 to 8 s. As can be seen in the figure, ERCC steadily improves the connectivity with neighbors, increases the CCC coverage, and selects more channels of the best quality while maintaining the lowest interference with PUs among all three methods, as the PU activity appears to be less dynamic on the average. This proves its capability of efficient recovery from high PU activity. Specifically, ERCC maintains *at least* 80% of CCC links when PU activity is most dynamic and also improves its connectivity to close to 100% when the activity is less intense while GRP can only achieve *at most* 80% of connectivity. Even though GRP has better CCC coverage than ERCC under highly dynamic PU activity, it is achieved at the expense of causing more interference. Moreover, SEQ appears to be less susceptible to PU active periods. However, it achieves low indicator values and causes more interference than ERCC because SEQ selects channels for CCC links based on hopping sequences that do not consider channel quality and neighbor information. Thus, ERCC makes better tradeoffs between increasing coverage and choosing a channel of best quality for minimizing the interference.

2) PU Transmission Range: The PU transmission range is determined by the path loss model with specified PU transmit power and the receive PU signal threshold. We can change transmit power to obtain different transmission ranges. Alternatively, with the fixed transmit power, we assume that CR users change the thresholds for different levels of tolerable PUI and PU transmission range. The larger the PU transmission range is, the more homogeneous the spectrum availability is in a neighborhood. Fig. 5 shows the expected metrics from the PU transmission range of 100–500 m. These ranges correspond to PU threshold γ_{pu} values from -92.72 to -110.70 dB. As shown in the figure, ERCC utilizes the local spectrum homogeneity to improve all metrics as the PU transmission range increases. For the same reason, SEQ slightly improves its performance. As the range increases, the hopping sequences chosen by neighbors in SEQ are more similar for better chances of rendezvous. Conversely, the performance of GRP drops significantly as the range increases. This is because, as the range of the PU on each channel increases, the probabilities for selecting control channels in GRP appear to be more comparable. As a result, SUs using GRP in a neighborhood cannot easily agree upon their control channel selection. This test case shows that ERCC is more consistent and reliable than the other two methods as PU adapts its transmit power and range.

3) PU Density: In this test case, we increase the PU density by increasing the number of PUs per licensed channel within the testing area. This will increase the observed PUI level and reduce the observed channel quality if more than one PU is active. Fig. 6 shows the expected metric values versus one, two, and three PUs occupying each licensed channel. As expected, the interference level increases for all methods as the density of PUs increases. Even though channel quality deteriorates,

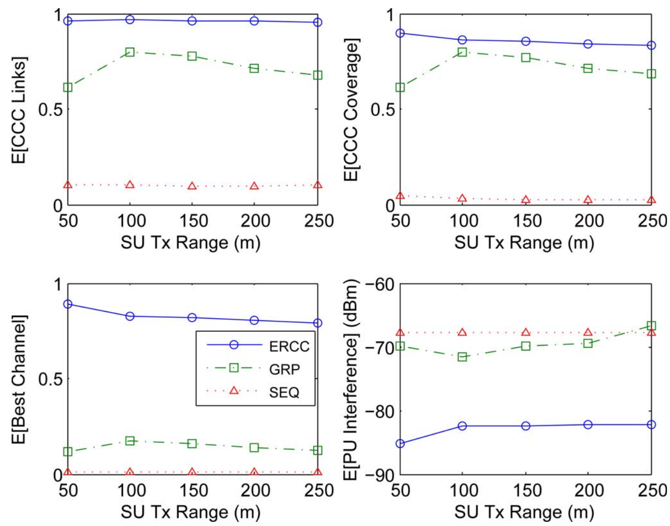

Fig. 7. Expected metric values versus SU transmission range R_s.

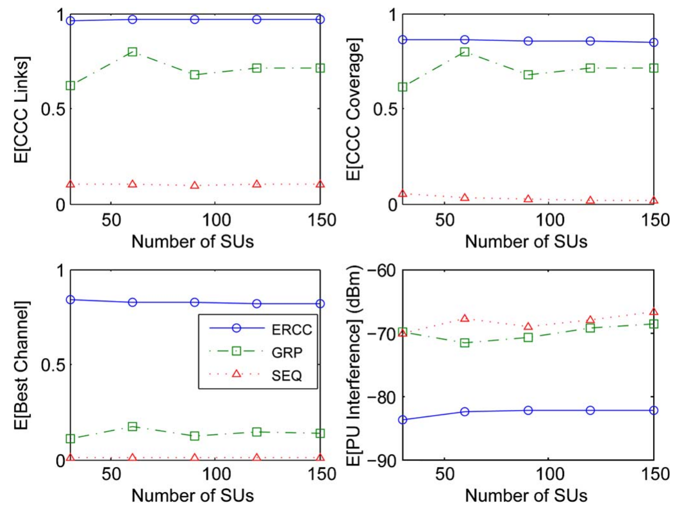

Fig. 8. Expected metric values versus the number of SUs in deployment N_s.

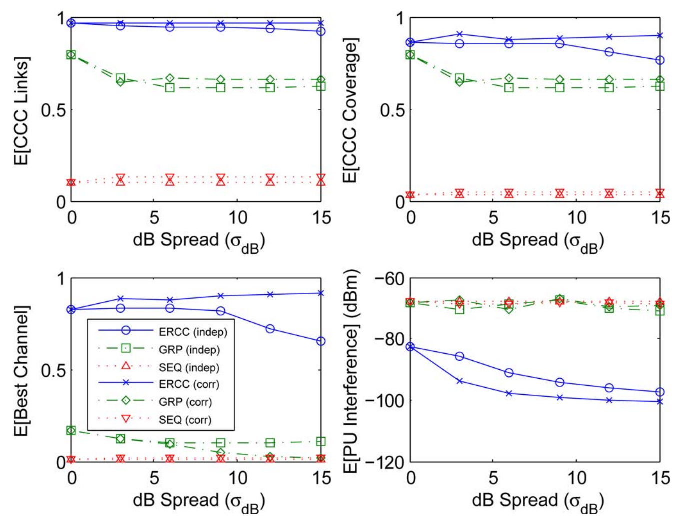

Fig. 9. Expected metric values in shadow fading σ_{dB}.

ERCC maintains a high percentage of links with neighbors and best channel selections partially based on the ordering of the channel quality. On the contrary, the performance of GRP is considerably affected by the reduced channel quality since it updates channel selection probabilities with channel quality values. As before, SEQ is insensitive to PU parameter changes, even though it does not achieve high coverage and connectivity as the other two methods. Therefore, these results show that ERCC is capable of adapting to high-interference environment with minor coverage reduction.

4) SU Transmission Range: Similar to the *PU transmission range* test case, we vary the SU transmission range by changing the SU sensing threshold γ_{su}. As the range increases, more neighbors are covered, resulting in an increased number of neighbors. Fig. 7 shows expected metrics versus SU transmission ranges from 50 to 250 m. In general, the performance of both ERCC and GRP slightly decreases as the SU transmission range increases. Since more neighbors away from the neighborhood contribute to the message exchange, the channel list may not reflect the real channel conditions in the surrounding area. Interestingly, the performance of GRP also degrades when the range is small. Since CCC allocation in GRP relies on the updates from the majority of neighbors in the neighborhood, a small SU range covers only a few neighbors that may not represent the true majority of neighbors for correct channel selection. This test case shows that a few benefits can be obtained from increasing the SU range to a large value, not to mention the waste of transmit power and higher interference incurred. However, proper transmission range is still essential to methods such as GRP for achieving good performance.

5) Scalability of SU Deployment: The scalability of SU deployment is evaluated by varying the number of SUs in the testing area. This also changes the density of SU population in the fixed area. Similar to the *SU transmission range* test case, the change in SU density affects the number of neighbors in the neighborhood. Fig. 8 shows the expected metric values versus the number of SUs ranging from 30 to 150. As shown in the figure, the performance of ERCC and SEQ is consistent and thus scalable in the range under testing. GRP, in general, is also scalable. However, too many or too few neighbors degrade its performance. Thus, GRP is more sensitive to SU parameters and neighbor updates, whereas ERCC and SEQ exhibit the scalability for a variety of different SU deployment sizes.

6) Shadow Fading: Unlike all previous test cases wherein PU signal quality is deteriorated only by the path loss model, this test case evaluates the performance of CCC solutions with the addition of independent and correlated shadow fading to reflect more realistic channel conditions. With the increase in the lognormal shadowing decibel spread σ_{dB} in the channels, the received PU signal power greatly varies such that the SUs are more susceptible to incorrect detection of PUs and channel availability. In this test case, we assume that all packets for message exchange between neighbors are protected by upper layer error control schemes and are correctly received.

Fig. 9 shows the expected metrics versus the decibel spread values in both independent and correlated shadow fading.

ERCC outperforms GRP and SEQ in terms of all the metrics. However, the performance of ERCC and GRP gradually degrades in independent shadowing as σ_{dB} increases. Unlike ERCC and GRP, SEQ is less susceptible to σ_{dB} changes. Interestingly, ERCC maintains better CCC links and coverage in correlated shadowing than those in the independent case. This is because, when the neighbors' observations are correlated, their CCLs tend to be similar, even with large decibel spreads, which facilitates the CCC allocation and improves the CCC coverage in a deep shadow. However, due to inaccurate received PU power levels, it is possible to incur the interference with PUs in this case. Thus, any cooperative spectrum sensing scheme [5], [10] can be incorporated into ERCC to mitigate the effects of channel impairments. By using the established CCC links between neighbors, neighboring SUs in ERCC can exchange spectrum sensing information to improve the detection of PUs and obtain fading-independent CCLs for robust CCC establishment and better CCC coverage.

VI. Conclusions

The CCC problem in CRAHNs aims at dynamically allocating in-band control channels in response to PU activity and extending control channel coverage for reducing control efforts. In this paper, we have proposed an ERCC method to address this challenge. By adaptively updating and periodically exchanging CCLs among neighboring nodes, our proposed method is capable of efficiently responding to PU activity change with dynamic control channel allocation. It also balances the tradeoff between extending the coverage of CCCs for reducing control signaling efforts and selecting channels of best quality for minimizing the interference with PUs. With CCLs constructed at each ad hoc node, the work can be extended to consider the adaptation of control channel bandwidth for a variety of control traffic loads. Furthermore, the optimal utilization of the CCLs for both control and data channel allocations can be developed for throughput and quality-of-service analysis. Finally, cooperative spectrum sensing schemes can be incorporated into ERCC for performance improvement in realistic channel conditions.

References

[1] I. F. Akyildiz, W.-Y. Lee, and K. R. Chowdhury, "CRAHNs: Cognitive radio ad hoc networks," *Ad Hoc Netw.*, vol. 7, no. 5, pp. 810–836, Jul. 2009.

[2] I. F. Akyildiz, W.-Y. Lee, and K. R. Chowdhury, "Spectrum management in cognitive radio ad hoc networks," *IEEE Netw.*, vol. 23, no. 4, pp. 6–12, Jul./Aug. 2009.

[3] I. F. Akyildiz, W.-Y. Lee, M. C. Vuran, and S. Mohanty, "NeXt generation/dynamic spectrum access/cognitive radio wireless networks: A survey," *Comput. Netw.*, vol. 50, no. 13, pp. 2127–2159, Sep. 2006.

[4] K. Bian, J.-M. J. Park, and R. Chen, "A quorum-based framework for establishing control channels in dynamic spectrum access networks," in *Proc. MobiCom*, Sep. 2009, pp. 25–36.

[5] D. Cabric, S. M. Mishra, and R. W. Brodersen, "Implementation issues in spectrum sensing for cognitive radios," in *Proc. 38th Asilomar Conf. Signals, Syst., Comput.*, Nov. 2004, pp. 772–776.

[6] T. Chen, H. Zhang, M. D. Katz, and Z. Zhou, "Swarm intelligence based dynamic control channel in CogMesh," in *Proc. IEEE ICC*, May 2008, pp. 123–128.

[7] T. Chen, H. Zhang, G. M. Maggio, and I. Chlamtac, "CogMesh: A cluster-based cognitive radio network," in *Proc. IEEE DySPAN*, Apr. 2007, pp. 168–178.

[8] T. Chen, H. Zhang, G. M. Maggio, and I. Chlamtac, "Topology management in CogMesh a cluster-based cognitive radio mesh network," in *Proc. IEEE ICC*, Jun. 2007, pp. 6516–6521.

[9] L. A. DaSilva and I. Guerreiro, "Sequence-based rendezvous for dynamic spectrum access," in *Proc. IEEE DySPAN*, Oct. 2008, pp. 1–7.

[10] A. Ghasemi and E. S. Sousa, "Collaborative spectrum sensing for opportunistic access in fading environments," in *Proc. IEEE DySPAN*, Nov. 2005, pp. 131–136.

[11] A. Goldsmith, *Wireless Communications*. New York: Cambridge Univ. Press, 2005, ch. 2, pp. 51–52.

[12] M. Gudmundson, "Correlation model for shadowing fading in mobile radio systems," *Electron. Lett.*, vol. 27, no. 23, pp. 2145–2146, Nov. 1991.

[13] B. Hamdaoui and K. Shin, "OS-MAC: An efficient MAC protocol for spectrum-agile wireless networks," *IEEE Trans. Mobile Comput.*, vol. 7, no. 8, pp. 915–930, Aug. 2008.

[14] S. Kakumanu and R. Sivakumar, "Glia: A practical solution for effective high datarate wifi-arrays," in *Proc. MobiCom*, Sep. 2009, pp. 229–240.

[15] L. Lazos, S. Liu, and M. Krunz, "Spectrum opportunity-based control channel assignment in cognitive radio networks," in *Proc. IEEE SECON*, Jun. 2009, pp. 135–143.

[16] W.-Y. Lee and I. F. Akyildiz, "Optimal spectrum sensing framework for cognitive radio networks," *IEEE Trans. Wireless Commun.*, vol. 7, no. 10, pp. 3845–3857, Oct. 2008.

[17] P. Pawelczak, S. Pollin, H.-S. W. So, A. Motamedi, A. Bahai, R. V. Prasad, and R. Hekmat, "State of the art in opportunistic spectrum access medium access control design," in *Proc. Int. Conf. CrownCom*, May 2008, pp. 1–6.

[18] H. Su and X. Zhang, "Cross-layer based opportunistic MAC protocols for QoS provisionings over cognitive radio wireless networks," *IEEE J. Sel. Areas Commun.*, vol. 26, no. 1, pp. 118–129, Jan. 2008.

[19] J. Zhao, H. Zheng, and G.-H. Yang, "Distributed coordination in dynamic spectrum allocation networks," in *Proc. IEEE DySPAN*, Nov. 2005, pp. 259–268.

[20] J. Zhu, A. Waltho, X. Yang, and X. Guo, "Multi-radio coexistence: Challenges and opportunities," in *Proc. ICCCN*, Aug. 2007, pp. 358–364.

Brandon F. Lo (M'10) received the B.S. (honors) degree in computer science from Tunghai University, Taichung, Taiwan, in 1992 and the M.S. degree in electrical engineering from the University of Michigan, Ann Arbor, in 1995. He is working toward the Ph.D. degree in electrical and computer engineering with the Broadband Wireless Networking Laboratory, School of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta.

Before his doctoral study, he designed processors and application-specific integrated circuit chips for broadband communications and networking in the semiconductor industry. His research interests include cognitive radio networks, mobile ad hoc networks, and wireless communications.

Ian F. Akyildiz (M'86–SM'89–F'96) received the B.S., M.S., and Ph.D. degrees in computer engineering from the University of Erlangen-Nurnberg, Erlangen, Germany, in 1978, 1981, and 1984, respectively.

He is currently with Georgia Institute of Technology (Georgia Tech), Atlanta, as the Ken Byers Chair Professor with the School of Electrical and Computer Engineering and the Director of Broadband Wireless Networking Laboratory and Chair of the Telecommunication Group. Since June 2008, he has been an Honorary Professor and Director of the Center for Nanonetworking in Catalunya (N3Cat) with the School of Electrical Engineering, Universitat Politècnica de Catalunya, Barcelona, Spain. Since March 2009, he has been an Honorary Professor and the Director of the Advanced Sensor Networks Laboratory, Department of Electrical, Electronic, and Computer Engineering, University of Pretoria, Pretoria, South Africa. Since January 2010, he has been a Visiting Professor with the College of Computer and Information Sciences, King Saud University, Riyadh, Saudi Arabia. From 1989 to 1998, he served as a National Lecturer for the Association for Computing Machinery (ACM). He is or has been the Editor-in-Chief for the *Computer Networks Journal* (Elsevier) and the founding Editor-in-Chief for the *Ad Hoc Networks Journal* (Elsevier), the *Physical Communication (PHYCOM) Journal* (Elsevier), and the *Nano Communication Networks (NANO-COMNET) Journal* (Elsevier) in 2003, 2008, and 2010, respectively. His current research interests are nanonetworks, cognitive radio networks, and wireless sensor networks.

Dr. Akyildiz has been an ACM Fellow since 1997. He serves on the advisory boards of several research centers, journals, conference proceedings, and publication companies. He was the recipient of the "Don Federico Santa Maria Medal" for his services to the Universidad of Federico Santa Maria, Valparaiso, Chile, in 1986; the ACM Outstanding Distinguished Lecturer Award in 1994; the 1997 IEEE Leonard G. Abraham Prize Award (IEEE Communications Society) for his paper entitled "Multimedia Group Synchronization Protocols for Integrated Services Architectures," which was published in the IEEE JOURNAL OF SELECTED AREAS IN COMMUNICATIONS in January 1996; and the 2002 IEEE Harry M. Goode Memorial Award (IEEE Computer Society) with the citation "for significant and pioneering contributions to advanced architectures and protocols for wireless and satellite networking." He was also the recipient of the 2003 IEEE Best Tutorial Award (IEEE Communication Society) for his paper entitled "A Survey on Sensor Networks," which was published in the *IEEE Communications Magazine* in August 2002; the 2003 ACM Sigmobile Outstanding Contribution Award with the citation "for pioneering contributions in the area of mobility and resource management for wireless communication networks;" the 2004 Georgia Tech Faculty Research Author Award for his "outstanding record of publication of papers between 1999 and 2003;" and the 2005 Distinguished Faculty Achievement Award from the School of Electrical and Computer Engineering, Georgia Tech.

Abdullah M. Al-Dhelaan received the B.S. (Honors) degree in statistics from King Saud University, Riyadh, Saudi Arabia, in 1982 and the M.S. and Ph.D. degrees in computer science from Oregon State University, Corvallis, in 1986 and 1989, respectively.

He is currently an Associate Professor of computer science, Chairman of the joint Ph.D. program, and Director General for the Center for International Collaboration and Visiting Professors, College of Computer and Information Sciences, King Saud University. He has guest edited several special issues for the *Telecommunication Journal* (Springer) and the *International Journal for Computers and Their Applications*. Moreover, he is currently on the editorial boards of several journals, such as *Computer Network* (Elsevier) and *The International Journal of Computers and Their Applications*. His current research interest includes mobile ad hoc networks, sensor networks, cognitive networks, network security, and high-performance computing.