Recovery in Channel-Hopping Cognitive Networks Under Random Primary User Activity

Jelena Mišić, *Senior Member, IEEE* and Vojislav B. Mišić, *Senior Member, IEEE*

Abstract—Collisions with primary user transmissions necessitate fast and robust recovery mechanisms in channel hopping cognitive networks. We describe a recovery algorithm based on a list of backup channels to be used as alternatives if primary user activity interrupts ongoing communication on the current channel. In case all backup channels are exhausted without success, the nodes revert to a blind rendezvous mechanism. We evaluate the performance of this algorithm using the tools of probabilistic analysis and renewal theory. We discuss the impact of various network and channel parameters on the performance of the recovery algorithm, and show the importance of accurate sensing information for successful recovery.

Index Terms—opportunistic spectrum access; wireless personal area networks; channel hopping cognitive networks; recovery after collision

I. INTRODUCTION

In the opportunistic (also referred to as cognitive or dynamic) spectrum access scheme, secondary (unlicensed) users can use the spectrum only in the absence of the primary (licensed) ones; when a primary user appears on a channel, secondary users should immediately move away to another channel [9]. This approach, often referred to as *overlay* architecture [1], allows the secondary (cognitive) network to protect its own communications whilst minimizing its impact on primary users' communication. However, protection is not perfect, and collisions with primary user transmissions will occasionally interrupt the operation of a secondary network.

In the absence of a central authority that would coordinate channel access for both primary and secondary networks, cognitive MAC protocols [8] try to avoid collisions by having the nodes that want to communicate perform sensing of the chosen channel before each packet transmission. This approach is well suited to an environment where primary user activity is changing slowly, as is the case with the so-called TV White Space [6]. However, it becomes inefficient when primary users exhibit fast activity changes, since each packet transmission necessitates sensing on one channel – or several of them, if the sensed channel is not free of primary user activity.

A better approach is to use dynamic channel hopping in which secondary users form piconets that switch channels according to a common hopping sequence, similar to Bluetooth [14]. In some solutions, the piconet simply hops sequentially through all available channels [13], but a better solution would

be to dynamically adapt the hopping sequence to avoid primary user activity [17].

Yet even dynamic channel hopping is unable to avoid all collisions and the piconet must be able to recover from a collision and continue its operation on a different channel quickly. In a Bluetooth piconet, the hopping sequence is pseudo-random but known beforehand to all the nodes in the piconet (it is derived from the master node ID). When a transmission is destroyed by interference from another network, nodes simply go to the next channel in the sequence, possibly several times, until they re-connect with the piconet. In a channel-hopping piconet, dynamic adaptation of the hopping sequence means that upon a collision nodes may not know where to hop next and synchronization may easily be lost. In many applications, including emergency response/disaster management and military communications, minimizing the duration of periods where the network is not operational is a requirement of highest priority.

The simplest solution would be to undertake a new rendezvous procedure where nodes attempt to form a new piconet, but it can take a long time during which there can be no data traffic in the piconet. A better solution is to devise a dedicated recovery protocol that would enable the piconet to *recover* from a collision by switching to another idle channel. Some MAC protocols that do include recovery rely on out-of-band coordination, e.g., the presence of a common control channel (CCC), to handle collisions and other interruptions [12], [18], [29]. While appealing, such solutions are not readily applicable in practice due to difficulties in ensuring that the CCC is always available and interference-free [4]. It should come as no surprise, then, that recovery has been listed among the problems that 'necessitate a different approach from the classical ad hoc networks' [1].

A promising approach to recovery is to maintain a list of backup channels to be used for recovery upon detecting a sudden onset of primary user activity. The list of backup channels, also referred to as the Preferred Channel List [18], is determined together with the next-hop channel by the coordinator [15] and kept up to date through periodic sensing. This is the essence of the Incumbent Detection Recovery Protocol (IDRP) described in [7] and later adopted in IEEE 802.22 standard [3] as well as in other MAC protocols for cognitive networks [16], [28]. However, IDRP and its derivatives focus on cognitive networks operating in a RF band (such as TV White Space) where the activity of primary users is slow compared to the activity of the cognitive network. Consequently, channel switch is performed only when necessary, i.e., as part of the recovery procedure through IDRP. Moreover, IDRP

J. Mišić and V. B. Mišić are with the Department of Computer Science, Ryerson University, Toronto, ON, Canada M5B 2K3.
E-mail: jmisic@scs.ryerson.ca, vmisic@ryerson.ca

assumes that some nodes may be able to communicate (and receive notifications about a collision) even after the onset of primary user activity. Both assumptions are rather specific and may be difficult to achieve in practice.

In this paper, we describe a backup channel-based recovery mechanism suitable for cognitive piconets in which

- Channel hopping is performed at a fast rate all the time without a dedicated, interference-free common control channel. As the result, both data and control communications occur on channels which are subject to primary user activity.
- Channels experience random primary user activity with a memoryless probability distribution of idle and active times. Moreover, primary users transmit at power levels that essentially preclude communication between nodes in the cognitive network.

The proposed recovery mechanism is distributed and needs no central authority or an always-available communication channel. Moreover, it is fully integrated with the transmission tax-based MAC protocol [26], which is one of the few proposals that integrate regular operation of a piconet (i.e., data transmission and reception, and bandwidth allocation) with sensing activities that aim to ensure smooth piconet operation under unpredictable primary user activity. In particular, cooperative sensing at the MAC level provides the necessary information for the selection of backup channels.

We analyze the performance of the proposed recovery mechanism, including the rendezvous mechanism used as a fallback in case recovery does not succeed [25], using the tools of probabilistic analysis and renewal theory, and demonstrate the impact of various primary user activity, piconet, and protocol parameters. Results are further validated through discrete-event simulation.

The paper is organized as follows: Section II begins with a brief description of the transmission tax protocol with particular emphasis on design choices to accommodate piconet recovery operation and rendezvous mechanisms. Section III presents the analytical models of the recovery process, while Section IV provides a detailed performance evaluation, including collision probability, algorithms for next-hop and backup channel selection, the recovery process itself, and the rendezvous process. Finally, Section V concludes the paper and highlights some future research. An Appendix provides the analytical model of the rendezvous mechanism.

II. MAC, RECOVERY, AND RENDEZVOUS PROTOCOLS

A. Transmission tax-based MAC protocol

We assume that the channel-hopping piconet consists of M nodes and a coordinator. The piconet hops through a RF band with N channels of arbitrary bandwidth, each of which is intermittently and randomly occupied by a licensed user.

The piconet uses a MAC protocol in which time is slotted into superframes which are further subdivided in sub-frames for data and administrative transmissions, such as reporting of sensing results, join/leave and bandwidth reservation requests, and also beacon and trailer frames, and a guard time for channel switching. We assume that the superframe lasts for s_f unit slots and that administrative communications take up Δ of these slots. Each superframe takes place on a distinct channel in the working RF band; between two successive superframes, all nodes in the piconet switch to the next-hop channel as instructed by the coordinator via the trailer frame. The structure of the superframe is schematically shown in Fig. 1(a), while the flowchart in Fig. 1(b) depicts the operation of the coordinator and ordinary nodes, in the left- and right-hand side, respectively.

At the beginning of the superframe, immediately after the channel switch, the coordinator emits a small beacon frame. The leading beacon allows piconet nodes to synchronize their transmissions, since different nodes might require different channel switching time, and also provides a convenient point in time for recovery attempts, as will be explained below.

At the end of the superframe, the coordinator emits a trailing beacon (trailer) frame. This frame serves to inform the nodes about the next-hop channel, as well as of bandwidth allocation in the next superframe (i.e., the network allocation vector), admission requests and other administrative information. In this manner, newly arrived nodes can receive a nearly immediate response and also learn about the next-hop channel. Moreover, the decision about the next-hop channel is made shortly upon receiving sensing reports, which improves the probability that the next-hop channel will be idle during the next superframe.

Upon successful transmission of a data packet, the sender node is obliged to perform sensing for k_p superframes – the transmission tax or penalty [26]. Each of the sensing nodes independently and randomly selects which channels to sense during the data subframe of a superframe, and reports the results back to the coordinator in the reporting subframe. The penalty coefficient is a small positive integer $k_p \geq 1$, hence sensing duty may extend over one or more superframes. (The value of k_p may be dynamically adjusted to maintain the desired sensing errors under varying network and traffic conditions [21].)

The coordinator maintains a list of idle and busy channels— the channel map—and decides on the channel to be used for the next hop. Decision about the next hop channel should be made with the goal of selecting the channels most likely to remain idle throughout the next superframe [23].

Due to the discrete character of sensing and the delay needed to collect sensing results, the information in the channel map differs from the actual state [20]. Sensing results could be reported in 'bulk' at the end of the sensing duty, or periodically in each of the k_p superframes. The latter approach is preferred on account of the following. First, frequent reporting improves the accuracy of the sensing reports and of the coordinator's channel map; under bulk reporting, sensing results are older and less accurate. Second, sensing nodes need to hear every trailer because it announces the next-hop channel, and also because they be required to suspend sensing for in order to receive data from another node. (In the latter case, sensing resumes in the next superframe.) Preemption of sensing by reception improves throughput since transmissions need not wait until the receiving node finishes sensing duty [26]. Moreover, it simplifies bandwidth allocation since the coordinator need not keep track of the nodes which
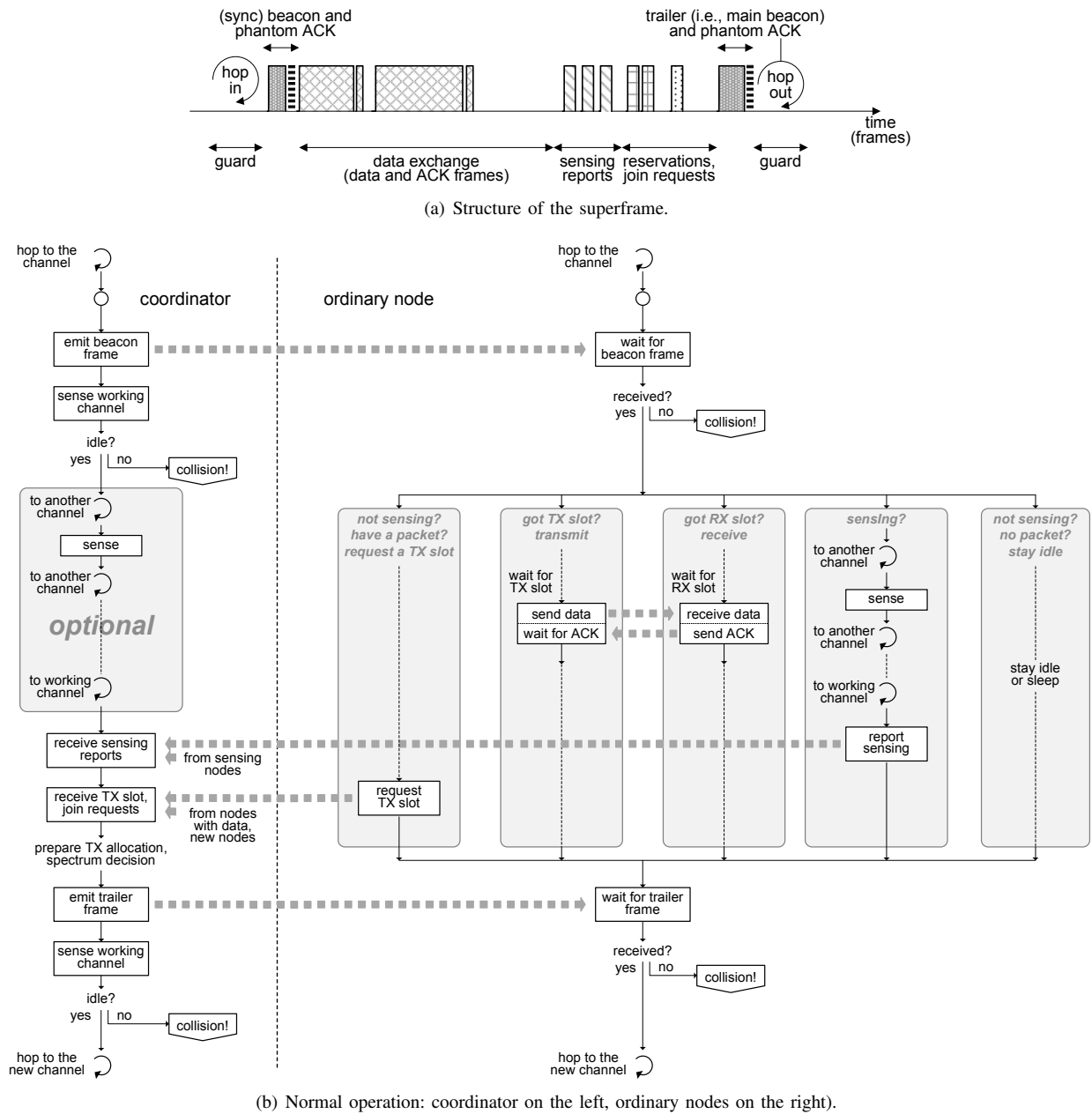
(a) Structure of the superframe.

(b) Normal operation: coordinator on the left, ordinary nodes on the right).

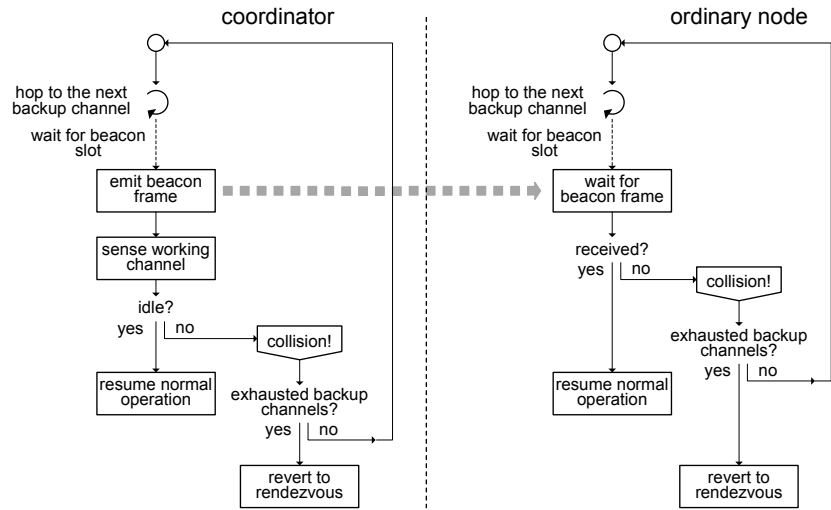Fig. 1. Operation of the transmission tax-based MAC protocol.

are currently doing sensing duty.
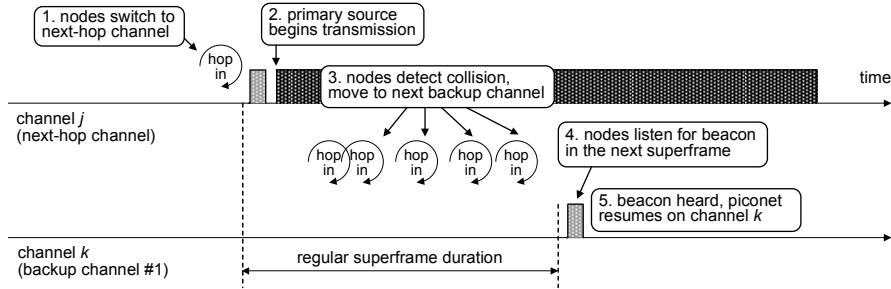
B. Recovery using backup channels

Regular operation of the piconet may be disrupted by random activity of primary users. The onset of primary user activity is referred to as a collision, which may occur in two scenarios: a type 1 collision occurs when the piconet hops to a channel which is thought to be idle but is, in fact, busy; a type 2 collision occurs when the primary user begins its activity during an ongoing superframe. Either way, immediate remedial action is needed. The simplest solution would be for nodes to re-form the piconet anew by undertaking the rendezvous procedure but this process can take considerable time during which the piconet is not operational [22]. A better way is to

have all the nodes, including the coordinator, attempt recovery by switching to a backup channel as soon as the collision is detected [1].
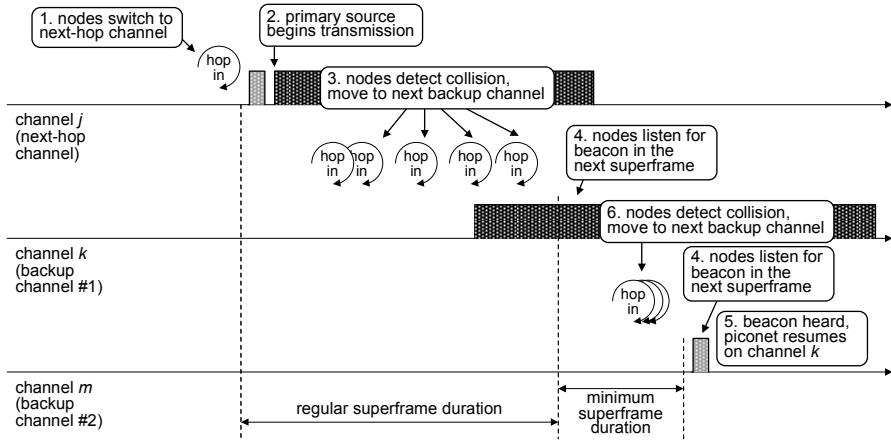
However, not all nodes will detect the collision at the same time: some nodes will detect it by the absence of a scheduled transmission, others by missing an acknowledgment (ACK) frame requested by the preceding data frame. Furthermore, some nodes may have been absent on sensing duty, while others that have no traffic and no sensing duty may have temporarily switched off their radios to conserve energy. As the MAC protocol requires all nodes to join the piconet in time to hear a trailer frame [26], a missing trailer frame will lead the nodes to conclude there has been a collision. They will, then, attempt to switch to a backup channel and synchronize at

(a) Operation during recovery. Left: coordinator, right: ordinary nodes.

(b) Recovery succeeds on the first backup channel.

(c) Recovery succeeds on the second backup channel.

Fig. 2. Recovery in case of a collision.

the earliest suitable time, which would be the time scheduled for the beacon frame in the next superframe. As the trailer frame was missing, there can be no new next-hop channel, and the nodes should switch to the backup channel instead. This situation is schematically shown in Fig. 2(a).

But which channel is designated as the backup channel? This information is announced by the coordinator in the trailer, together with the regular next-hop channel. To increase the likelihood of successful recovery, since the designated backup channel may also turn out to be busy, the coordinator announces not just one but several candidate backup channels chosen among the channels currently sensed as idle. (This information is readily available in the coordinator's channel map.) Then, backup channels from the list in the last trailer received during regular piconet operation are probed in sequence; failure is declared, and rendezvous procedure undertaken, only when all of the backup channels are exhausted. Figs. 2(b) and 2(c) schematically show recovery procedures that succeed in the first and second attempt, respectively, upon detecting a type 2 collision. A type 1 collision would be

handled in a similar way, except that the collision would be detected by all nodes in the piconet around the beginning of superframe on channel j.

Note that ordinary nodes detect a collision if a beacon or trailer frame is missing – but the coordinator transmits those frames and can't know if they haven't been properly received. The solution is to have the coordinator perform a quick sensing of the working channel immediately upon transmitting the beacon and trailer frames. (Ordinary nodes need not do this.) If the coordinator detects primary user activity during phantom ACK intervals, a type 1 or 2 collision has occurred and recovery is needed. As the time after transmission is usually reserved for an ACK frame, we refer to these periods as phantom ACK in Fig. 1(a).

We note that the phantom ACK period after the beacon frame incurs no performance penalty since the time delay needed for the coordinator to switch from transmission to reception is also needed for the first node that has been granted permission to transmit. On the other hand, the phantom ACK period after the trailer frame requires the coordinator to switch to reception and then sense the channel, which does incur an extra delay. Yet this is a small price to pay for the ability to keep the piconet synchronized; moreover, the guard interval during which the nodes hop to the next channel must be set to a large enough value anyway, because nodes with different hardware might need different time to perform the switch.

During recovery, regular sensing and data transmission activities are suspended since the piconet is not operational and the superframe duration need not stay the same. Remember that all backup channels were idle at the time of last sensing; as we move farther away from the trailer in which they were announced, some of them may become busy and, thus, unusable for recovery. Consequently, setting the superframe duration to a lower than normal value $s_{fm} < s_f$ leads to faster recovery because it reduces the time between successive recovery attempts. Moreover, shorter time between attempts translates into higher probability that one of the backup channels will remain idle and, consequently, the probability of successful recovery.

C. Selecting the next-hop and recovery channels

Let us now discuss the selection of next-hop and backup channels. We assume that the coordinator maintains a list that includes current (perceived) state for all channels, as well as the times of last perceived state change (esp. busy to idle), compiled on the basis of sensing reports sent by the nodes. We assume that sensing results are reported truthfully, although the accuracy of the channel map depends on the number of sensing reports as well as on the dynamics of primary user activity [20]. In any case, the temporal granularity of channel map information is equal to one superframe duration, since the sensing reports are sent to the coordinator once in each superframe.

The next-hop channel can be selected at random from the set of idle channels, or perhaps from the set of channels that have turned idle most recently. The remaining channels from the chosen set can be used as backup channels. Alternatively,

the coordinator may estimate the probability that the channels which are currently idle will remain idle throughout the next superframe. To this end, it can use the recorded values of channel idle times to build a histogram of idle period duration, one for each channel; the histogram approximates the probability distribution of channel idle time. The width of each bin of the histogram equals the shortest possible duration of the superframe; this corresponds to the temporal granularity of the recovery process. The number of bins is slightly harder to determine as having too few bins would reduce the usability of the histogram, while having too many would increase the complexity without the corresponding increase in accuracy. In our calculations, we have set the number of bins so that the histogram covers a predefined percentage (90%) of all recorded values.

Now, for each channel that is currently idle, say i, the coordinator calculates the index of the bin that corresponds to the time interval $\tau_i + s_f$, which is the duration of the current idle period augmented by the calculated duration of the next superframe. Different channels may have turned idle at different times, and thus the intervals tau_i (and the corresponding bin numbers) will differ. The coordinator then finds the channel for which $\pi_i = \pi_i^* / \sum_i \pi_i^*$, the probability of being idle at the end of the next superframe, is the largest. (Strictly speaking, the actual bin value, π_i^*, is not a probability, hence the need to normalize the value by the sum of values in all the bins.) The remaining idle channels, sorted in descending order of π_i, can be announced by the coordinator as backup channels.

If the set of candidate channels is empty, either because there really are no idle channels or the coordinator erroneously assumes there are none – which may occur due to the inertia of the sensing process and the resulting errors in the channel map [20] – the coordinator may decide to continue operation on the current channel, without announcing any backup channels at all. In this case, a collision would force the nodes to repeat the piconet formation (rendezvous) process [22].

A detailed performance comparison of these algorithms that the histogram-based channel selection offers the best performance of the three, in particular when primary users are heterogeneous, i.e., when their mean active and idle periods differ [23]. However, when the probability distribution of active and idle times on a channel is memoryless (i.e., Erlang $k = 1$ which corresponds to exponential distribution), all three algorithms perform about the same.

D. Rendezvous

If the recovery procedure does not succeed, the nodes must undertake a full rendezvous procedure, i.e., they will have to find each other and again form the piconet from scratch. As random hopping was shown to be the most efficient approach to rendezvous [2], a number of recently described rendezvous procedures rely on predefined pseudo-random channel sequences that guarantee a finite upper bound for time-to-rendezvous (TTR) [10], [5]; however, this property does not hold in the presence of primary user activity [22].

A better solution – and one that is compliant with the transmission tax-based MAC protocol – involves random
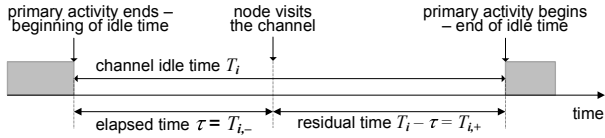
Fig. 3. Pertaining to the definition of elapsed and excess idle time on a channel.

hopping based on sensing [25], [22]. In this mechanism, a node that wants to find the piconet (hereafter referred to simply as the node) hops randomly through the channel set (we assume that it is aware of all N channels used by the piconet). If the node finds the channel to be busy, it stays for a short time period only; if it hops to an idle channel, it stays there for a longer time awaiting the piconet which hops through the same channel set. Maximum values of the wait intervals, T_{wb} and T_{wi}, will be referred to as the busy and idle timeout, respectively. The choice of suitable values for these two timeout intervals is discussed in Section IV-C below.

Rendezvous will be unsuccessful if the piconet and node sojourns on a channel don't overlap, or if an attempt by the node to join the piconet is destroyed by the onset of primary source activity. Rendezvous will succeed if the node and piconet sojourns overlap, regardless of who came in first, for as long as necessary for the node to send its join request. This request can be granted immediately, and the coordinator will announce the admission in the trailer frame; if not, the node can follow the piconet to the subsequent superframe(s) until it is admitted. Rendezvous will succeed even if the node hears only the trailer frame, in which case it will follow the piconet to the next-hop channel and request admission in the succeeding superframe.

Note that the presence of the piconet takes precedence over the idle timeout T_{wi} – if the node hears a transmission from a piconet within the time T_{wi}, it may overstay to send its join request and hear the admission decision in the trailer.

III. Modeling the recovery process

Let us assume that, for any channel, the durations of active and idle times, T_a and T_i, follow random probability distributions with the probability density functions (pdf) $t_a(x)$ and $t_i(x)$, respectively. As active and idle times alternate, the total cycle time on the channel will have the probability density function $t(x) = t_i(x) * t_a(x)$. Then, the probability that the channel is busy or idle can be calculated as $p_{on} = \overline{T_a}/\left(\overline{T_a} + \overline{T_i}\right)$ and $p_{off} = 1 - p_{on}$. Mean cycle time will be $\overline{T_{cyc}} = \overline{T_a} + \overline{T_i}$. Mean number of idle and busy channels is $\overline{N_i} = p_{off}N$ and $\overline{N_a} = p_{on}N$, respectively.

The performance of recovery and rendezvous mechanisms can be analyzed using renewal theory. Renewal process is a random process which counts the number of some general cycles where cycle durations X_i, are independent, identically distributed nonnegative random variables [11]. Beginning of a new cycle period is a renewal point at which a new probabilistic replica of the original renewal process begins.

The process that counts cycles of primary source activity, beginning with the onset of idle period on some channel σ, is a renewal process. Consider a node that visits an idle channel at a time, relative to the beginning of idle time (a renewal point), denoted with τ. According to renewal theory, $\tau = T_{i,-}$ is considered to be the deficit (or elapsed) idle time on that channel, while $T_i - \tau = T_{i,+}$ is the residual (or excess) idle time on the channel, as shown in Fig. 3. Deficit idle channel time has the probability distribution function (PDF) defined as $A(x) = P(\tau \leq x)$, while its pdf is $a(x) = \frac{dA(x)}{dx}$. Let us also define $P(T_i > x) = T_i^c(x) = \int_{y=x}^{\infty} t_i(y)dy$. Then, the PDF of the deficit idle channel time can be calculated as

$$A(x) = \frac{1}{\overline{T_i}} \int_0^x T_i^c(y)dy \qquad (1)$$

and the corresponding pdf as

$$a(x) = \frac{d}{dx}A(x) = \frac{T_i^c(x)}{\overline{T_i}} \qquad (2)$$

The process that counts the number of sensing events on some channel σ is also a renewal process since time periods between two consecutive sensing events follow same probability distribution derived from the fact that selection of channels to sense is randomly performed by each sensing node, independently of any central authority and the selection of other nodes [20], [24]. For this process, the onset of activity of primary user between two sensing points is a random point in the sensing cycle. If we denote the duration of sensing period on channel σ as R and the moment of onset of primary user activity relative to previous sensing point as ξ_R, then $\xi_R = R_-$ is elapsed sensing time and $R - \xi_R = R_+$ is residual sensing time, analogous to the definitions above and shown in Fig. 3. Since sensing periods are synchronized to piconet activity, they are multiples of the basic time slot used for the MAC protocol and the probability distribution of the sensing time is discrete, unlike the distribution of activity times of primary users which is continuous and independent of piconet activities.

In [24], probability distribution of residual sensing time with respect to the start of idle period was calculated in form of Probability Generating Function (PGF) of $R_+(z) = \sum_{i=0}^{\infty} R_i z^i$, where mass probabilities R_i depend on the number of nodes in the piconet, traffic load, and the number of data frames that a node is allowed to transmit in a single superframe.

The process that counts superframes on any given channel is also a renewal process, although a trivial one. In this case, the onset of activity of primary user on channel σ is a random point in the superframe currently taking place on some channel μ. If we denote the duration of superframe as $C = s_f$ and the moment of onset of primary user activity, relative to the beginning of the superframe, as ξ_C, then $\xi_C = C_-$ is elapsed superframe time and $C - \xi_C = C_+$ is residual superframe time.

Probability density function of the residual superframe time has the form $c = 1/s_f$ which can be obtained if (2) is applied to constant variable s_f. This result holds for both discrete and continuous versions of the superframe residual time.

A. Delay in accessing an idle channel

The piconet will not access a channel as soon as it becomes idle. Instead, the channel can be selected for the next hop only upon being sensed and recorded as idle in the channel map. Since the selection of channels to sense is randomly performed by each sensing node, independently of any central authority and the selection of other nodes [24], the time between the moment when the channel becomes idle (i.e., the primary user transmission ends) and the moment when the piconet can access it is a random variable, and change of channel state is a random point in the sensing period. Therefore, the time between a change of channel state and next sensing of that same channel is, in fact, residual sensing time [11]. More specifically, change of state of the channel ch occurs at a random point of a piconet superframe, so the time period between that point and the moment of reporting of sensing results has the probability distribution of residual superframe time [11]. Therefore, we can say that a channel state transition will be sensed in the ongoing superframe only if residual sensing time is shorter than the residual superframe time; it will be detected in the immediately following superframe if the residual sensing time is longer than the residual superframe time but shorter than the sum of residual superframe time and superframe length; and so on for the second, third, … following superframe.

For simplicity, however, we calculate only the probabilities that an active-to-idle channel transition is detected in the on-going superframe and two following superframes, Ps_0, Ps_1, and Ps_2, respectively, and ignore the cases where detection occurs in later superframes. These probabilities depend on the number of nodes that perform sensing duty and dynamics of the channel. For network configurations considered in this work, the probability that a channel transition is detected two superframes later is very small (below 0.01), which justifies the simplification.

Probability that the end of primary user activity was sensed and reported in the ongoing superframe, the immediately following superframe, or the second following superframe, respectively, can then be calculated as

$$Ps_0 = \sum_{k=1}^{s_f} c \sum_{n=0}^{k-1} R_n$$
$$Ps_i = \sum_{k=0}^{s_f} c \sum_{n=k+(i-1)s_f}^{k+is_f-1} R_n, k = 1, 2 \tag{3}$$

B. Probability distribution of the time of piconet access

Probability that access to an idle channel is not possible due to the sensing delay at the piconet and average period after the beginning of the idle channel time where access is not possible can be calculated as

$$P\theta_0 = \sum_{i=0}^{2} Ps_i \int_{x=0}^{s_f} c \int_{y=0}^{x+is_f} a(y)dydx$$
$$\overline{Noa} = \sum_{i=0}^{2} Ps_i \left(is_f + c \int_{x=0}^{s_f} xdx \right) \tag{4}$$

The piconet can access the channel after the superframe in which a channel is recorded as idle in the channel map. As the superframe duration is fixed at s_f, access can occur at any multiple of s_f slots (and, in fact, more than once) until the channel becomes busy again. Since sensing events are synchronized with the detection time of channel availability, we need to calculate mass probabilities $P\theta_k$ of access by the piconet in k-th superframe period ($k = 1, 2, \ldots$) after the channel is labeled as available. As channel availability is detected at a random time with respect to beginning of the idle channel period, we can obtain $P\theta_k$ by integrating probability density of idle channel elapsed time over superframe durations:

$$P\theta_k = \frac{1}{1-\theta_0} \sum_{i=0}^{k} Ps_i \int_{x=0}^{s_f} c \int_{y=x}^{x+s_f} a(y)dydx, k = 0, 1$$
$$P\theta_k = \frac{1}{1-\theta_0} \int_{x=0}^{s_f} c \int_{y=x+(k-1)s_f}^{x+k \cdot s_f} a(y)dydx, k \geq 2 \tag{5}$$

Probabilities θ_k need to be normalized to the probability that the idle channel is available to access, which is $1 - \theta_0$.

Note that access in the first superframe position is possible only if the change of channel status has occurred before the reporting subframe, but this problem may be overcome by considering that the superframe starts at the reporting subframe.

C. Probability of piconet access to the channel

The channel map is not perfectly accurate due to the insufficient number of sensing nodes and discrete nature of the sensing process. Let a_1 denote the probability that the channel map considers a channel where primary activity has ceased to be busy and therefore unusable, and let b_1 denote the probability that a busy channel is still considered to be idle in the channel map (exact values can be found in [20]). Therefore, the probability that the piconet will access a given channel is $pc = (1 - a_1)/(\overline{N_i}(1 + b_1))$. In general, $a_1 \neq b_1$, due to different durations of active and idle channel periods.

Then, the probability that the piconet will access a target channel at least once during its idle time (and not collide with onset of primary activity) can be calculated as

$$P_{acc} = \sum_{i=0}^{2} Ps_i \sum_{k=1}^{L} pc(1-pc)^{k-1}$$
$$\cdot \int_{x=0}^{s_f} cdx \int_{x+(k+i)s_f}^{\infty} t_i(y)dy - a_1 \tag{6}$$

Note that a_1 is subtracted since idle channels with obsolete information are considered busy, and therefore will not be chosen for piconet use.

In an analogous fashion, the probability that the piconet will not attempt to access an idle channel is

$$P_{nvis} = \sum_{i=0}^{2} Ps_i \sum_{k=1}^{L} (1-pc)^{k}$$
$$\cdot \int_{x=0}^{s_f} cdx \int_{(k+i)s_f-x}^{(k+i+1)s_f-x} t_i(y)dy + a_1 \tag{7}$$

D. Collision probability and recovery

Let us now calculate the probability that the piconet will collide with the primary user on a given channel. Collision probability has two components which correspond to two types of collisions. The first, b_1, is the probability that the next-hop channel has become active, but the coordinator does not know it because the information in the channel map is obsolete. Second component is the probability that the channel will become active during superframe time, which means that residual idle channel time is shorter than superframe duration:

$$P_c = \int_{x=0}^{\infty} (A(x + s_f) - A(x))a(x)dx \tag{8}$$

We note that $P_c + P_{acc} + P_{nvis} = 1$.

The total collision probability is, then,

$$P_{Col} = P_c + b_1 \tag{9}$$

Let the trailer contain a list with l candidate channels, which is normally updated in each superframe so as to reflect changing channel conditions. The first channel in the list is used as the next-hop channel. When a collision occurs, the recovery procedure attempts to re-establish the piconet on the remaining $l-1$ backup channels; if all of them fail, the nodes must undertake the rendezvous procedure.

We note that information about the state of backup channels becomes gradually more obsolete after each backup attempt, as no spectrum sensing is conducted after the collision, and the probability that a backup channel will not be idle when accessed increases. This increase can be alleviated to some extent by using smaller superframe size s_{fm}, as explained above. To model this fact, we will modify expressions (8) and (9) to obtain the probability that residual idle channel time is larger than i superframe times as

$$P_{c,i} = \int_{x=0}^{\infty} (A(x + s_f + (i-1)s_{fm}) - A(x))a(x)dx \tag{10}$$

$$P_{Col,i} = P_{c,i} + b_1 \tag{11}$$

The Laplace-Stieltjes Transform (LST) for the duration of recovery procedure, conditioned on collision on current superframe, becomes

$$T_{rec}^*(s) = \sum_{j=1}^{l-1} \left(\prod_{n=2}^{j} P_{Col,n} \right) (1 - P_{Col,j+1})e^{-sjs_{fm}}$$
$$+ \left(\prod_{n=2}^{l} P_{Col,n} \right) Ren^*(s) \tag{12}$$

where $\prod_{n=2}^{1} P_{Col,n} = 1$. The last term, $Ren^*(s)$, denotes the LST of the time needed for rendezvous in case all $l-1$ recovery attempts fail, the derivation of which is shown in the Appendix.

The mean and standard deviation of the recovery time are

$$\overline{T_{rec}} = -\frac{d}{ds}T_{rec}^*(s)\Big|_{s=0} \tag{13}$$

$$\sigma(Trec) = \sqrt{\frac{d^2}{ds^2}T_{rec}^*(s)\Big|_{s=0} - \overline{T_{rec}}^2} \tag{14}$$

IV. PERFORMANCE OF THE RECOVERY PROCESS

In order to evaluate the performance of the recovery process, we have solved the model presented above, together with sensing and transmission models taken from [24]. We have used Maple 13 software package by Maplesoft, Inc. [19]. To validate the analytical results, we have also built a simulator of the cognitive piconet, using the object-oriented Petri Net-based simulation engine Artifex by RSoft Design, Inc. [27].

Primary users, one for each of N channels, were assumed to operate independently, following exponential distribution of idle and active times with mean cycle time of T_{cyc} and duty cycle p_{on}. We have set the time unit to a single sensing slot, superframe duration was fixed to $s_f = 100$ or $s_f = 50$ units (unless variable), and the administrative portion of the superframe (including reporting, reservation and join requests, beacon, trailer, and guard intervals) to $\Delta = 20$ units. We have assumed that the piconet has $M = 16$ nodes, with each node having a buffer for a total of $K = 10$ packets.

Packet arrival process was set to Poisson with arrival rate of $\lambda = 0.002$ packets per slot per node, while packet duration was uniformly distributed between 8 and 12 time units with an average value of $k_d = 10$. Duration of the acknowledgment packet was set to one time unit. Packet destinations were uniformly distributed over all piconet nodes. Maximum number of packets from a single node that can be serviced in one superframe was set to 3. Transmission tax was set to $k_p = 4$ superframes per one transmission, regardless of the number of packets sent within a single superframe. Sensing of one channel, including channel switching, was assumed to take $d_s = 5$ slots.

A. Collision probability

First, we have calculated the probability of collision with primary user activity. Sum of the average idle and active time of the channel was set to $\overline{T_{cyc}} = 1500$ or 3000 time slots, as noted in the diagrams. In experiments where the primary user activity factor was constant, its value was set to $p_{on} = 1/3$; in experiments where the number of primary channels was constant, it was set to $N = 25$.

The results shown in Fig. 4 indicate that the collision probability increases with the primary user duty cycle (activity factor): as idle periods are shorter, more superframes will be affected by collisions. Collision probability increases with the number of channels due to errors in the channel map. Namely, more channels mean less frequent sensing of each channel and an increased probability that the channel has actually changed state since last sensing.

Longer primary activity cycle times mean that, at fixed superframe duration, the piconet will spend *proportionally* less time on any given channel; as the result, the collision probability decreases when the primary user cycle time increases, as shown in Figs. 4(a) and 4(b). Furthermore, shorter superframe duration means the piconet will spend less time on any given channel, which in turn reduces collision probability; this is confirmed by the reduction in collision probability in 4(c), where $s_f = 50$ unit slots, compared to the one in Fig. 4(b), obtained with $s_f = 100$. Simulation results, shown as squares,

(a) Variable primary user duty cycle and number of channels, primary user cycle time $\overline{T_{cyc}} = 1500$ unit slots, superframe duration $s_f = 100$ unit slots.

(b) Variable primary user duty cycle and number of channels, primary user cycle time $\overline{T_{cyc}} = 3000$ unit slots, superframe duration $s_f = 100$ unit slots.

(c) Variable primary user duty cycle and number of channels, primary user cycle time $\overline{T_{cyc}} = 3000$ unit slots, superframe duration $s_f = 50$ unit slots.
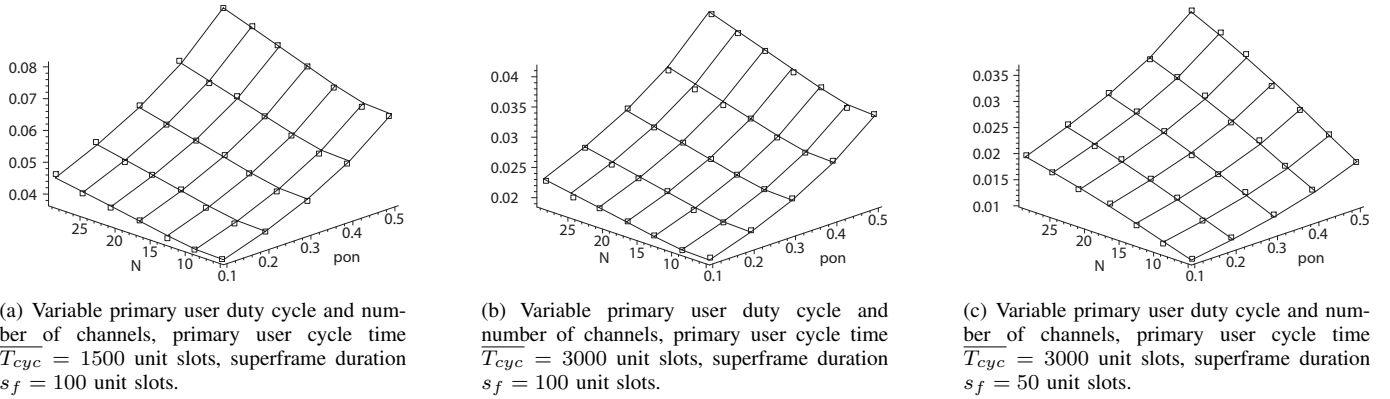
Fig. 4. Probability of collisions.

can be seen to be in good agreement with the analytical ones, shown with lines.

Overall, using shorter superframes and fewer channels (which piconet designers can do) and choosing frequency bands with less active primary users (which piconet designers can't really do) will lead to lower probability of collision. However, we note that the percentage of superframes destroyed by collisions is non-negligible – about 8% at $T_{cyc} = 1500$, $N = 30$, and $p_{on} = 0.5$ – and there is a definite need for a recovery mechanism.

B. The impact of collisions on primary users

Depending on the transmit power of primary user and the sensitivity of the corresponding receivers, cognitive network activity may or may not cause interference. But even if it does cause interference, its impact is rather limited, on account of the following.

First, a collision occurs only on the channel on which the cognitive piconet operates and thus impacts the primary user operating on that channel. The other $N - 1$ channels and their respective primary users are unaffected by it. Assuming that primary users are homogeneous, i.e., that their activity follows the same probability distribution with identical mean active and idle times, we may assume that any given primary user will experience collisions with the probability of $P_{Col}^{(p)} = P_{Col}/N$, where N is the number of channels.

Second, the maximum duration of a collision on a given channel is close to the duration of the cognitive piconet superframe s_f (in fact, the time interval between the beginning of the leading beacon and the end of the trailing beacon, which is less than s_f), but its mean duration will be closer to $s_f/2$. Namely, interference to the primary user will end as soon as all the nodes that might be transmitting in the affected superframe actually detect the collision and stop transmitting; this will happen at different times during the superframe, perhaps even at its very beginning due to the leading beacon and the phantom ACK period that follows it.

C. Do we actually need recovery, or rendezvous would suffice?

A question worth asking is, whether the rendezvous mechanism such as the one described in Section II-D would suffice

to ensure speedy piconet recovery in case of a collision, or a dedicated recovery mechanism is still needed. To investigate this, we have solved the analytical model presented above – actually, the part of it that describes the performance of the rendezvous mechanism – to obtain mean value and coefficient of variation of rendezvous times.

As rendezvous involves random hopping through available channels, the most important choice to be made relates to the two timeout parameters, T_{wb} and T_{wi}. Their values are set according to the following argument [25].

First, if the channel is found to be busy, it can't be used by the cognitive piconet and there is no point staying there for long. Note that the channel will not be used by the piconet immediately upon turning idle – it must first be sensed as idle and reported to the coordinator before it may actually be used as the next-hop channel. Given the temporal granularity of sensing, this will require at least one superframe. Hence, T_{wb} is set to $T_{wb} = 10$ unit slots, about twice the time required for actually sensing the channel.

Second, if the channel is found to be idle, longer sojourn of the node increases the probability of overlap with the cognitive piconet. in an ideal scenario in which channels are visited sequentially and there is no primary user activity, the time required to ensure an overlap between the node and the piconet is equal to the product of the number of channels and the maximum duration of the superframe. While this simple calculation does not hold under random hopping through the channels, some of which may be busy due to primary user activity, the probability of overlap still depends on those two variables. To reduce the confounding effect of the number of channels, the idle timeout was set to $T_{wi} = nTOI \cdot N$, with the normalized timeout $nTOI$ set to 140 unit slots [25].

The resulting diagrams of mean time to rendezvous obtained under different values for mean cycle time of primary user activity are shown in Fig. 5. As before, the diagrams show both analytical results (lines) and simulation results (squares). Mean TTR is seen to be a nearly linear function of the primary user duty cycle, with higher values of the duty cycle resulting in lower rendezvous times. This is due to the shortening of channel idle times, which increases the probability that the nodes will meet. At the same time, increasing the number

(a) Mean primary user cycle time $T_{cyc} =$ 1500 unit slots.

(b) Mean primary user cycle time $T_{cyc} =$ 3000 unit slots.

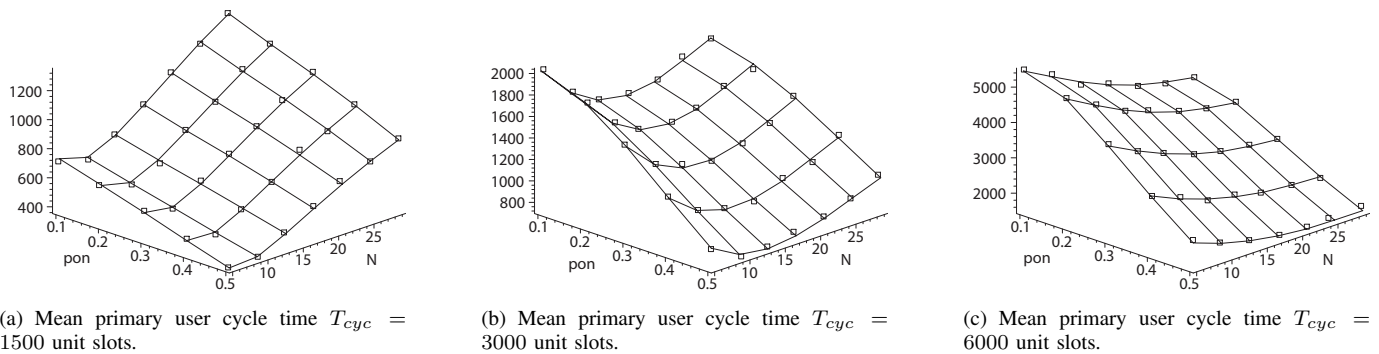(c) Mean primary user cycle time $T_{cyc} =$ 6000 unit slots.

Fig. 5. Mean rendezvous time under superframe duration $s_{fm} = 50$ unit slots.

of channels leads to an increase in mean TTR, since more channels reduce the probability of nodes meeting on any given channel.

Furthermore, a distinct minimum may be observed which shifts towards higher number of channels as the mean cycle time of primary source increases. This behaviour is actually a consequence of the increase of busy channel time. Namely, choosing a relatively small number of channels for the rendezvous algorithm when the channel busy time is increasing will increase the duration of the time when all channels are busy and the rendezvous algorithm is just probing busy channels. Probability of this event is p_{off}^N, where N is number of channels, and its duration is a random variable which is inversely proportional to N. Therefore, when the duration of channel busy time is increasing, the number of channels involved in rendezvous should also increase.

As can be seen, the shortest value of rendezvous time, about 400 unit slots in the front corner of the surface in Fig. 5(a), is about *eight times* longer than the superframe duration, and it occurs at about five to ten channels, which is quite low. When the piconet uses more channels, mean rendezvous time is well over about ten superframe durations. Given this range, and the fact that collision probability has non-negligible values, we can safely conclude that the performance of rendezvous protocol by itself is not satisfactory. While rendezvous must be used as the fallback in case recovery fails, it is too slow to be the sole remedy in case of collisions, and a dedicated recovery protocol is a necessity.

Note that a more detailed account of rendezvous performance can be found in [22].

D. Performance of the recovery mechanism

We have solved the complete analytical model from Section III to obtain the mean value and coefficient of variation of recovery time. As explained above, when a collision is detected, the nodes will attempt to switch to a backup channel in order to continue operating as a piconet. If recovery attempts on all existing $l - 1$ backup channels eventually fail, the nodes must revert to a rendezvous procedure to re-create the piconet. During both recovery and, if necessary, rendezvous, the superframe duration is set to a value of $s_{fm} = 50$ unit slots which is shorter than the one used in normal operation, for reasons explained in Section II-B.

Performance of recovery procedure with $l = 3$ backup channels, for different mean cycle times of primary users, is presented in Fig. 6. As the diagrams indicate, smaller channel cycle times result in larger mean recovery times; values of the coefficient of variation increase as well. This is a consequence of larger probability of collision during attempts to re-establish communication on one of the backup channels. Although mean recovery time is close to minimum superframe duration of $s_{fm} = 50$ unit slots, and always below two minimum superframe durations, the coefficient of variation exceeds one in all cases, except for very small number of channels at short primary user cycle times, as in Fig. 6(e). Namely, short cycle times may limit the number of backup channels to a value smaller than l (remember that backup channels must be idle at the time the list is updated). Small number of backup channels translates into a small number of attempts, the success of which, due to exponentially distributed channel idle and busy times, is uncertain; as the result, recovery time will vary in a wide range. At longer cycle times, more backup channels are available, and the mean value of recovery time decreases, but its coefficient of variation decreases even more.

To investigate the impact of the length of the backup channel list, we have calculated mean value and coefficient of variation of recovery time under $l = 4$, 5, and 6 backup channels, as illustrated in Fig. 7; these diagrams should be compared to those in Figs. 6(b) and 6(e) as these were obtained under the same mean primary cycle time of $T_{cyc} = 3000$ unit slots. The agreement between analytical and simulation results, shown by lines and results, respectively, is quite good.

As can be seen, larger number of backup channels leads to a slight decrease of mean recovery time but also to a large decrease of the coefficient of variation. In fact, the difference between the mean values for $l = 5$ and $l = 6$ backup channels is almost unnoticeable, and a similar observation holds for the coefficient of variation. This is mainly due to the lower collision probability which essentially leads to a flattening of both mean and coefficient of variation of recovery time.

We stress that these results were obtained under exponential distribution of primary user active and idle times; they may differ under other distributions. In particular, distributions which are more symmetric and concentrated around the mean value may lead to collision probabilities that might grow abruptly for each subsequent recovery attempt. The last obser-
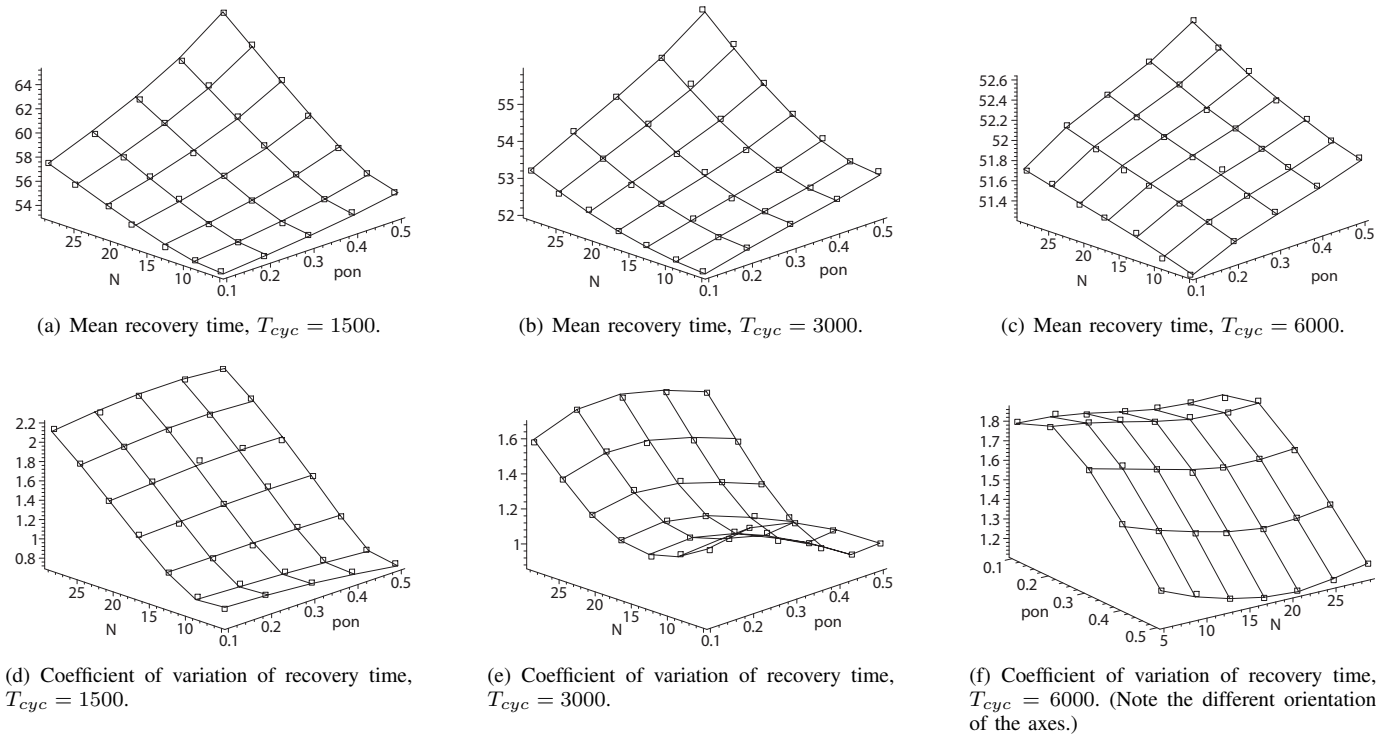
(a) Mean recovery time, $T_{cyc} = 1500$.

(b) Mean recovery time, $T_{cyc} = 3000$.

(c) Mean recovery time, $T_{cyc} = 6000$.

(d) Coefficient of variation of recovery time, $T_{cyc} = 1500$.

(e) Coefficient of variation of recovery time, $T_{cyc} = 3000$.

(f) Coefficient of variation of recovery time, $T_{cyc} = 6000$. (Note the different orientation of the axes.)

Fig. 6. Performance of recovery mechanism with $l = 3$ backup channels. Top row: mean value; bottom row: coefficient of variation.

(a) Mean recovery time, $l = 4$ backup channels.

(b) Mean recovery time, $l = 5$ backup channels.

(c) Mean recovery time, $l = 6$ backup channels.

(d) Coefficient of variation of recovery time, $l = 4$ backup channels.

(e) Coefficient of variation of recovery time, $l = 5$ backup channels.

(f) Coefficient of variation of recovery time, $l = 6$ backup channels.
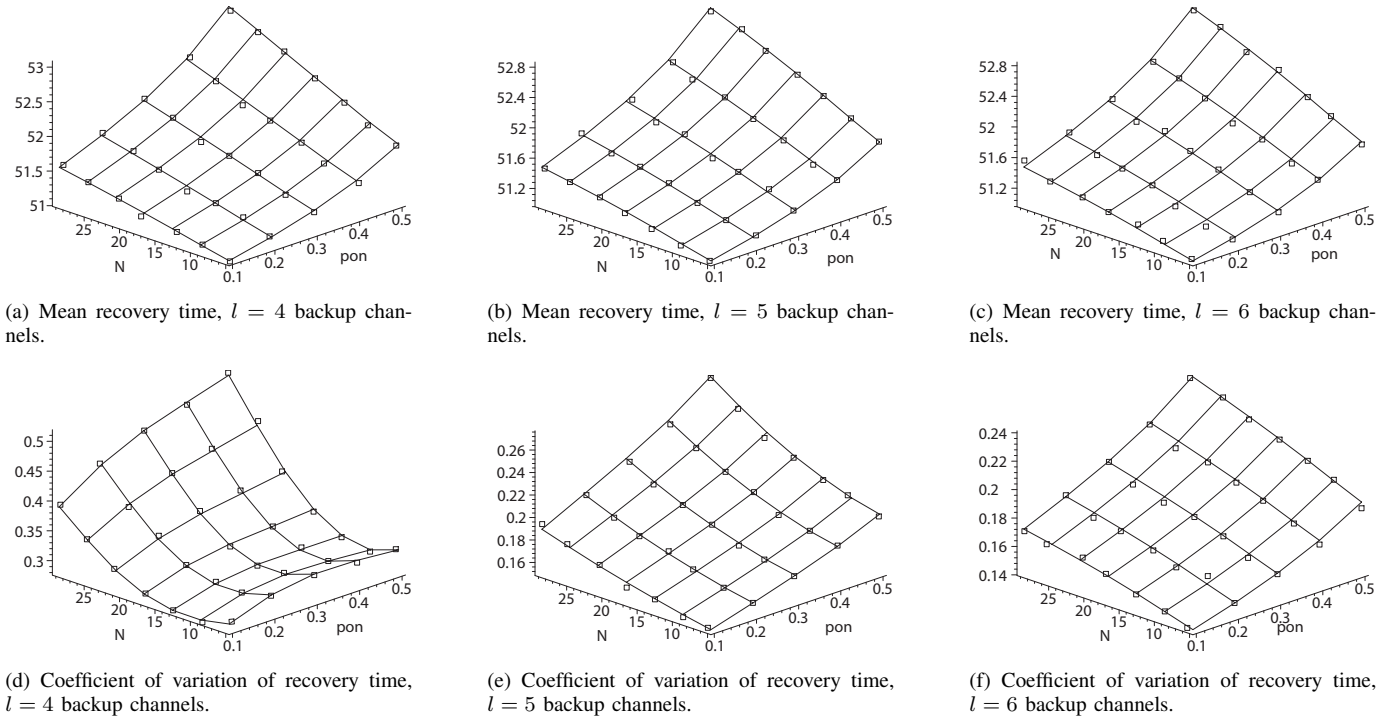
Fig. 7. Recovery time vs. number of backup channels, primary user cycle time $\overline{T_{cyc}} = 3000$ unit slots. Top row: mean value; bottom row: coefficient of variation.

vation is corroborated by the diagram in Fig. 8 which shows collision probability (both analytical and simulation results) as a function of primary user duty cycle and the index of recovery attempt. As can be seen, collision probability is a nearly linear function of the primary user duty cycle – longer duty cycle results in shorter idle times which lead to an increase in collision probability. Collision probability exhibits similar dependency on the index of successive recovery attempt,
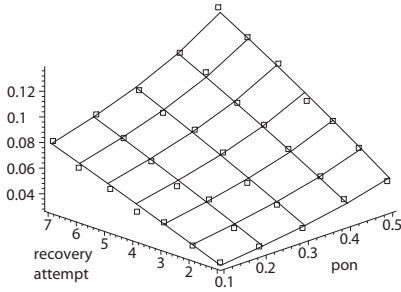
Fig. 8. Collision probability vs. recovery attempt and primary user duty cycle p_{on}.

which is due to the fact that, with every subsequent attempt, the information about the status of the backup channel becomes older, and the probability that it will turn busy increases.

To further investigate the dependency of recovery time on the accuracy of sensing information, we have repeated our experiments in a piconet with $M = 10$ nodes and penalty coefficient of $k_p = 3$, and $M = 5$ and $k_p = 2$; the results (collision probability under normal operation, mean recovery time, and coefficient of variation of recovery time) are shown in the top and bottom row, respectively, of Fig. 9. In both cases, the primary user cycle time was $T_{cyc} = 3000$ unit slots, the packet arrival rate was kept at $\lambda = 0.002$ packets/node/unit time, and the list of backup channels had $l = 3$ items.

As can be seen, fewer nodes and lower penalty lead to higher collision probability (compare the diagrams in Figs. 9(a) and 9(d) with the one in Fig. 4(c)) and longer mean recovery times with higher coefficient of variation. This effect can be explained as follows: the backup channels should be chosen among those that are currently known (or, rather, thought) to be idle and most likely to remain idle throughout the recovery process. Both criteria are critically dependent on the accuracy of the channel map. In all fairness, they also depend on the algorithm for creating the list of backup channels, but only if the probability distribution of idle channel times is not memoryless, as shown in [23]. When the information in the channel map is less accurate, the probability increases that a chosen backup channel will turn to be busy which might prolong the recovery process. The accuracy of the channel map depends on the amount of sensing information obtained from the sensing nodes. This amount is determined by the number of nodes in the piconet, the volume of traffic (i.e., packet arrival rate), and the value of the penalty coefficient. An increase in any of these parameters leads to an increase in the number of sensing reports which should lead, eventually, to more efficient recovery process, as witnessed by the diagrams in Fig. 9.

V. CONCLUSION

In this paper we have considered a simple recovery mechanism for channel hopping cognitive networks that uses a list of backup channels which, upon detecting a collision, are sequentially probed. This solution was found to offer short recovery times which makes it well suited for a number of cognitive network applications, including emergency response,

disaster management, and military communications. Recovery time was found to be dependent on the length of the list of backup channels, channel parameters (which are, unfortunately, beyond control of the network designer), and on the accuracy of spectrum sensing; this offers important guidelines for the design of MAC protocols and related spectrum sensing algorithms.

Our future work will focus on practical implementations and tuning of these algorithms. Furthermore, we plan to investigate issues related to coexistence of several channel-hopping piconets in the same physical space.

APPENDIX

If the recovery process is not successful, the nodes must undertake a full rendezvous, which incurs additional delay. *Direct overlap:* Let the piconet arrive to the channel and begin a new superframe at the time θ, defined in (5), while the new node arrives at τ and begins to listen for up to T_{wi}. Rendezvous will be successful if the node sojourn on the channel overlaps with the superframe, i.e., $\theta \leq \tau + T_{wi} \leq \theta + s_f$. Probability of the overlap, conditioned on the duration of the piconet superframe and the waiting time of the node, is

$$P_{ov} = \sum_{k=1}^{L} P\theta_k \int_{x=0}^{s_f} c\,dx \int_{\max(x+(k-1)s_f - T_{wi},0)}^{x+ks_f - \Delta} a(u)du \quad (15)$$

from which we can find the LST of the time spent to achieve direct overlap as

$$T_{ov}^*(s) = \frac{1}{T_{ov}^*(0)} \sum_{k=1}^{L} P\theta_k \int_{x=0}^{s_f} c\,dx$$
$$\cdot \int_{t=0}^{T_{wi}} e^{-st} a(\max(x+(k-1)s_f - t,0))dt \quad (16)$$

Actual calculation of the last expression requires it to be broken into separate cases that correspond to $T_{wi} \leq \tau < \theta$, $\tau < \theta < T_{wi}$, and $\theta < \tau < \theta + s_f - \Delta$, respectively.

As noted above, the node may arrive to the channel after the reservation subframe but in time to hear the trailer and obtain the information about the next-hop channel. The node will then follow the piconet to the next superframe and submit a join request in the corresponding reservation sub-frame. The corresponding probability may be calculated as

$$P_{late} = \sum_{k=1}^{L} P\theta_k \int_{x=0}^{s_f} c\,dx \int_{x+ks_f - \Delta}^{x+ks_f} a(u)du \quad (17)$$

The time spent in waiting is $T_{late} = \Delta + s_f$ slots. *Rendezvous destroyed by a collision:* A pending rendezvous can be destroyed by the onset of primary user activity on the channel. This event differs from a collision of piconet with primary source on account of the presence of a new node that wants to join the piconet. Probability that the collision occurs during the rendezvous process and the LST for the time spent in waiting for a rendezvous that will ultimately fail can be
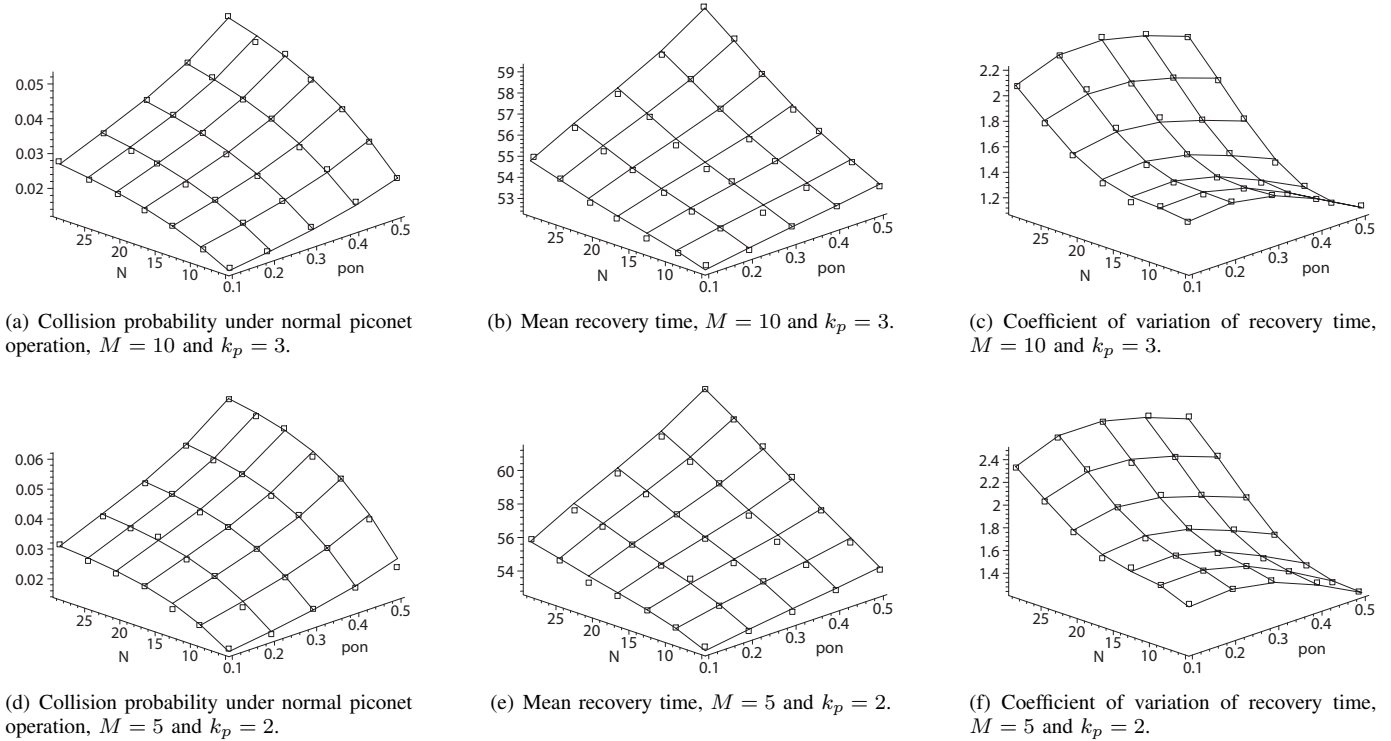
(a) Collision probability under normal piconet operation, $M = 10$ and $k_p = 3$.

(b) Mean recovery time, $M = 10$ and $k_p = 3$.

(c) Coefficient of variation of recovery time, $M = 10$ and $k_p = 3$.

(d) Collision probability under normal piconet operation, $M = 5$ and $k_p = 2$.

(e) Mean recovery time, $M = 5$ and $k_p = 2$.

(f) Coefficient of variation of recovery time, $M = 5$ and $k_p = 2$.

Fig. 9. The impact of accuracy of sensing information on recovery time, with number of nodes M and penalty coefficient k_p as independent variables. Top row: piconet with $M = 10$ nodes and penalty coefficient $k_p = 3$; bottom row: piconet with $M = 5$ nodes and penalty coefficient $k_p = 2$.

found as

$$P_{ColR} = \int_{x=0}^{s_f} \left(\int_{y=0}^{x+T_{wi}} a(y)dy \right) a(x)dx \qquad (18)$$

$$T_4^*(s) = \frac{1}{T_4^*(0)} \int_{x=0}^{s_f} a(x)dx \int_{t=0}^{T_{wi}} e^{-st} a(x+t)dt \qquad (19)$$

Missed rendezvous: Rendezvous does not occur on some channels on account of the following.

The node may visit a busy channel; it takes $T_1 = T_{wb}$ time slots to realize that a primary source is active and decide to switch to another channel.

The node may visit an idle channel but the piconet does not access that same channel during the channel idle time. The LST of this time is

$$T_2^* = e^{-sT_{wi}} \int_{x=T_{wi}}^{\infty} a(x)dx + e^{-sT_{wb}} \int_{x=0}^{T_{wi}} e^{-sx} a(x)dx \qquad (20)$$

The node may visit an idle channel but leaves after an idle timeout, before the piconet accesses that channel in the same idle channel period. Probability of this event is

$$P_3^- = \sum_{k=\min(w,L)}^{L} P\theta_k \int_{x=0}^{s_f} cdx \int_0^{\max(0, x+(k-1)s_f - T_{wi})} a(y)dy \qquad (21)$$

and the LST of the corresponding waiting time is

$$T_3^{*-}(s) = (1 - P_3^-) + P_3^- e^{-sT_{wi}} \qquad (22)$$

The node may visit an idle channel after the piconet has left. The probability of this scenario and corresponding LST,

obtained using the approach outlined in (16), and assuming that y denotes the time of node arrival, are

$$P_3^+ = \sum_{k=1}^{L} P\theta_k \int_{x=0}^{s_f} cdx \int_{y=x+ks_f}^{\infty} a(y)dy \qquad (23)$$

$$T_3^{*+}(s) = \frac{1}{T_3^{*+}(0)} \sum_{k=1}^{L} P\theta_k \int_{x=0}^{s_f} cdx \int_{t=0}^{T_{wi}} e^{-st} a(x + ks_f + t)dt \qquad (24)$$

Distribution of unsuccessful waiting time on an idle channel: We can now derive the LST for the waiting time on the channel where rendezvous was missed as

$$M^*(s) = p_{on} e^{-sT_{wb}} + p_{off} P_{nvis} T_2^*(s) + p_{off} P_{Col} T_4^*(s) \\ + p_{off} P_{acc}(P_{ov} + P_{late} + P_3^+ T_3^{*+}(s) + P_3^- T_3^{*-}(s)) \qquad (25)$$

The probability of successful rendezvous is conditioned by the need for the piconet and the node to access the same idle channel and the constraint on the overlap between their respective residence times:

$$P_{rv} = p_{off}(1 - a_1)(1 - P_{ColR})(P_{ov} + P_{late}) \qquad (26)$$

With all the components in place, we can describe the probability distribution of the time needed for a successful rendezvous with the LST of

$$Ren^*(s) = \frac{P_{rv}}{P_{ov} + P_{late}} \cdot \frac{T_{ov}^*(s)P_{ov} + e^{-s(s_f + \Delta)} P_{late}}{1 - (1 - P_{rv})M^*(s)} \qquad (27)$$

with mean of $\overline{Ren} = -\frac{d}{ds}Ren^*(0)$ and coefficient of variation of $c_v(Ren) = \frac{1}{\overline{Ren}}\sqrt{\frac{d^2}{ds^2}Ren^*(0) - \overline{Ren}^2}$. The value of $Ren^*(s)$ can, then, be substituted in (12) to obtain the complete LST of the recovery time. We note that (27) allows us to derive all higher moments of this probability distribution.

REFERENCES

[1] I. F. Akyildiz, W.-Y. Lee, and K. Chowdhury. CRAHNs: Cognitive radio ad hoc networks. *Ad Hoc Networks*, 7(5):810–836, 2009.

[2] E. Anderson and R. Weber. The rendezvous problem on discrete locations. *J. Applied Probability and Statistics*, 28:839–851, 1990.

[3] ANSI/IEEE. Standard for part 22: Cognitive wireless RAN medium access control (MAC) and physical layer (PHY) specifications: Policies and procedures for operation in the TV bands. IEEE Std 802.22-11, IEEE, New York, NY, 2011.

[4] C. J. L. Arachchige, S. Venkatesan, and N. Mittal. An asynchronous neighbor discovery algorithm for cognitive radio networks. In *Proc. DySPAN 2008*, Chicago, IL, Oct. 2008.

[5] K. Bian and J.-M. Park. Asynchronous channel hopping for establishing rendezvous in cognitive radio networks. In *Proc. INFOCOM 2011*, pages 236–240, Apr. 2011.

[6] K. Challapali, C. Cordeiro, and D. Birru. Evolution of spectrum-agile cognitive radios: first wireless internet standard and beyond. In *Proc. 2nd Ann. Int. Workshop on Wireless Internet*, Boston, MA, 2006.

[7] C. Cordeiro, K. Challapali, and M. Ghosh. Cognitive PHY and MAC layers for dynamic spectrum access and sharing of TV bands. In *Proc. Int'l Workshop on Technology And Policy for Accessing Spectrum (TAPAS)*, Boston, MA, 2006.

[8] C. Cormio and K. R. Chowdhury. A survey on MAC protocols for cognitive radio networks. *Ad Hoc Networks*, 7(7):1315–1329, 2009.

[9] DARPA. The XG vision. Request for comments, Jan. 2004.

[10] L. DaSilva and I. Guerreiro. Sequence-based rendezvous for dynamic spectrum access. In *Proc. DySPAN 2008*, pages 1–7, Oct. 2008.

[11] D. P. Heyman and M. J. Sobel. *Stochastic Models in Operations Research, Volume I: Stochastic Processes and Operating Characteristics*. McGraw-Hill, New York, 1982.

[12] R. Hincapié, L. Zhang, J. Tang, G. Xue, R. S. Wolff, and R. Bustamante. Efficient recovery algorithms for wireless mesh networks with cognitive radios. In *Proc. ICC 2009*, Dresden, Germany, June 2009.

[13] W. Hu, D. Willkomm, M. Abusubaih, J. Gross, G. Vlantis, M. Gerla, and A. Wolisz. Dynamic frequency hopping communities for efficient IEEE 802.22 operation. *IEEE Communications Magazine*, 45(5):80–87, May 2007.

[14] IEEE. Wireless medium access control (MAC) and physical layer (PHY) specifications for wireless personal area networks (WPAN). IEEE Std 802.15.1, IEEE, New York, NY, 2005.

[15] H. Kim and K. G. Shin. Fast discovery of spectrum opportunities in cognitive radio networks. In *Proc. DySPAN 2008*, Chicago, IL, Oct. 2008.

[16] B. Lee and S. H. Rhee. Adaptive MAC protocol for throughput enhancement in cognitive radio networks. In *Int. Conf. Information Networking (ICOIN 2008)*, Busan, Korea, Jan. 2008.

[17] P. K. Lee. Joint frequency hopping and adaptive spectrum exploitation. In *Proc. MILCOM 2001*, volume 1, pages 566–570, Washington, DC, Oct. 2001.

[18] B. F. Lo, I. F. Akyildiz, and A. M. Al-Dhelaan. Efficient recovery control channel design in cognitive radio ad hoc networks. *IEEE Trans. on Vehicular Technology*, 59(9):4513–4526, 2010.

[19] Maplesoft, Inc. *Maple 13*. Waterloo, ON, Canada, 2009.

[20] J. Mišić and V. B. Mišić. Performance of cooperative sensing at the MAC level: Error minimization through differential sensing. *IEEE Transactions on Vehicular Technology* 58(5):2457–2470, June 2009.

[21] J. Mišić and V. B. Mišić. Dynamic parameter adjustment in a cognitive MAC protocol with transmission tax. *Int. J. Autonomous and Adaptive Communications Systems*, 4(4):398–411, 2011.

[22] J. Mišić and V. B. Mišić. Probabilistic vs. sequence-based rendezvous in channel-hopping cognitive networks. *IEEE Transactions on Parallel and Distributed Systems*, 2013 (to appear).

[23] J. Mišić, V. B. Mišić, and M. S. I. Khan. On the selection of working channels in a channel-hopping cognitive PAN. In *Proc. IWCMC 2013*, Cagliari, Sardinia, Italy, July 2013.

[24] J. Mišić. Cooperative sensing at the MAC level in simple cognitive personal area networks. *IEEE J. on Selected Areas in Communications – Wireless Series*, 30(9):1711–1718, 2012.

[25] J. Mišić, H. Khojasteh, N. Khan, and V. B. Mišić. Towards an efficient rendezvous protocol for a cognitive PAN. In *Proc. ICC 2012*, Ottawa, ON, June 2012.

[26] V. B. Mišić and J. Mišić. Cognitive MAC protocol with transmission tax: Dynamically adjusting sensing and data performance. In *Proc. GLOBECOM 2010*, Miami, FL, Dec. 2010.

[27] RSoft Design. *Artifex v.4.4.2*. RSoft Design Group, Inc., Ossining, NY, 2003.

[28] A. K.-L. Yau, P. Komisarczuk, and P. D. Teal. A context-aware and intelligent dynamic channel selection scheme for cognitive radio networks. In *Proc. CrownCOM 2009*, Hannover, Germany, June 2009.

[29] K.-L. A. Yau, P. Komisarczuk, and P. D. Teal. On multi-channel MAC protocols in cognitive radio networks. In *Australasian Telecomm. Networks Appl. Conf.*, pp. 300–305, Adelaide, Australia, Dec. 2008.

Jelena Mišić (M91, SM08) Professor of Computer Science at Ryerson University in Toronto, Ontario, Canada. She has published over 90 papers in archival journals and more than 120 papers at international conferences in the areas of wireless networks, in particular wireless personal area network and wireless sensor network protocols, performance evaluation, and security. She serves on editorial boards of *IEEE Transactions on Vehicular Technology*, *Computer Networks*, *Ad hoc Networks*, *Security and Communication Networks*, *Ad Hoc & Sensor Wireless Networks*, *Int. Journal of Sensor Networks*, and *Int. Journal of Telemedicine and Applications*. She is a Senior Member of IEEE and Member of ACM.

Vojislav B. Mišić is Professor of Computer Science at Ryerson University in Toronto, Ontario, Canada. He received his PhD in Computer Science from University of Belgrade, Serbia, in 1993. His research interests include performance evaluation of wireless networks and systems and software engineering. He has authored or co-authored six books, 20 book chapters, and over 200 papers in archival journals and at prestigious international conferences. He is a Senior Member of IEEE, and member of ACM and AIS.