On the Selection of Working Channels in a Channel-Hopping Cognitive PAN

Jelena Mišić, Vojislav B. Mišić, and Mohammad Shahnoor Islam Khan
Ryerson University
Toronto, ON, Canada M5B 2K3
Email: {jmisic, vmisic}@ryerson.ca

***Abstract*—In a channel-hopping Cognitive Personal Area Network (CPAN), the main objective of the working channel selection mechanism is to avoid collisions with unpredictable primary user activity. As the result, the manner in which the working channel for the next hop is selected, is among the most important determinants of CPAN piconet performance. In this paper we investigate the performance of two working channel selection algorithms and compare them with the simple random selection approach. We show that a simple algorithm where the working channel is selected amongst those that have most recently turned idle offer better performance in the homogeneous case (i.e., where the primary user activity on all channels follows a similar pattern), while the case with heterogeneous primary user activity on different channels is better handled by the algorithm in which predicted probability distribution of channel idle times is used to select the next working channel.**

I. Introduction

Opportunistic or cognitive spectrum access often relies on channel hopping [1]. However, predefined hopping sequences, be they deterministic such as the one used in the so-called dynamic hopping communities [10] or pseudo-random such as the one used in Bluetooth [13], are incapable of providing uninterrupted piconet operation under unpredictable activity of primary users [7]. Instead, the hopping sequence must be dynamically determined by choosing the channels with best transmission parameters and free from interference by primary users.

In this paper we present a comparative analysis of two techniques for working channel selection, using blind random selection as the convenient reference point. We use the transmission tax-based protocol from [15], [16] as the environment in which to implement these techniques, on account of its availability of superframe-based structure with integrated sensing protocol and dynamic channel hopping. (More details on the operation of this MAC protocol can be found in Section III.) One of the techniques for working channel selection is based on selecting one among the channels that have most recently become free – preferably, but not necessarily, within the last superframe. The other technique uses slightly more complicated statistical approach of building and maintaining a histogram of channel idle times. Using the histogram and the information about the channels that are currently free, the channel with highest probability to remain idle for the duration of the next superframe is selected as the next working channel (i.e., the working channel for the next hop).

The rest of the paper is organized as follows. In Section II we present a brief overview of existing work on working channel selection. Section III outlines the operation of the transmission tax-based MAC protocol. Section IV presents the two selection protocols and discusses their relative advantages. Section V presents simulation-based performance evaluation of the protocols, and Section VI concludes the work.

II. Related work on spectrum decision

Selection of the working channel or, as it is often called, spectrum decision, has been identified as a crucial piece of functionality in the operation of cognitive networks [1]. While the spectrum decision problem has received some attention in the existing literature, its level of 'maturity' is well below that of the other processes such as spectrum sensing or spectrum access [12].

In cases where primary user activity patterns are known, spectrum decision process can be designed to make use of those patterns. In case of cognitive communications that use TV White Space [2], [3], [8], such information might be added to the extensive database of existing TV transmitters. (This is not defined in DARPA's documents – but neither it is forbidden by them.)

Unfortunately, the cases where primary user activity is unpredictable are much more frequent in practice, and decisions have to be made on the basis of some statistical model of spectrum usage [7]. A prerequisite for statistically meaningful decisions is a certain degree of stationarity of primary user activity [11], which may or may not hold in a given scenario.

A formal analysis of spectrum availability in the context of ISM band where primary users are 802.11 transmitters has been proposed in [5]. This model is then used to develop an access strategy in which a channel is sensed by a secondary user and, if idle, utilized with a specified probability [6]. This approach, however, requires that secondary users are well synchronized, which is not straightforward; moreover, the approach holds for two nodes only and may not be easy to extend to a piconet with several such nodes. A stochastic extension that avoids the need for strict synchronization has been described in [11].

Spectral estimation is the foundation of the approach described in [9], where distinction is made between white and gray zones in the scanned spectrum: the former contain noise

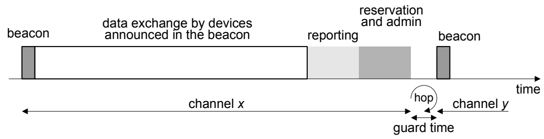


Fig. 1. Structure of the piconet superframe.

only, while the latter contain signals with distinct idle periods. Separating the two, however, is noted to be a difficult task.

Some authors have also described game-theoretic approaches [4], but in the context of cooperation between primary and secondary users which is not readily applicable in most cases.

From this overview, two important observations can be made. First, in all cases, selection of the best available channel at any given time necessitates frequent and accurate spectrum sensing [14]. Sensing may be performed in centralized or distributed fashion – but the decision about the channel to use for the next hop must be made in a centralized manner, and announced this decision to all the nodes in the piconet in an appropriate manner, e.g., through a beacon frame that is periodically broadcast. Second, sensing data must be used to build a statistical model of primary user spectrum usage through some kind of learning process; this model can then be used to guide spectrum decisions. These observations form the foundation for our selection techniques; but let us first describe the MAC protocol that will be used.

III. Transmission tax-based MAC protocol

A number of nodes with cognitive capabilities form a cognitive piconet through a so-called rendezvous procedure [17], the details of which are beyond the scope of this paper. All nodes are assumed to be capable of half-duplex operation. Any of the nodes with sufficient computational capability can act as a coordinator, similar to Bluetooth [13]. The time in each superframe is divided into the following components: beacon frame, data exchange sub-frame, reporting sub-frame in which sensing results are sent back to the coordinator, and reservation sub-frame in which bandwidth requests and other administrative frames are sent. Beacon frames contain bandwidth allocation decisions, information about nodes joining the piconet or leaving it, network allocation vector outlining the boundaries of sub-frames in the next superframe, and the announcements about the next-hop channel – i.e., the channel which will be used for the next superframe. , a variable number of fixed-duration time slots. (Each superframe may take place on a different channel according to the hopping sequence dynamically selected by the coordinator.) The structure of the superframe is schematically shown in Fig. 1. Successive superframes are separates by a guard interval of sufficient duration which allows all nodes to hop to the next channel.

The label transmission tax stems from the fact that a node, once it request and receive bandwidth allocation to transmit a packet – and then successfully transmits it – is obliged to perform the sensing duty for a specified number of superframes. It reports the results in each of the superframes during which it performs the sensing duty. Thus transmission is effectively paid for by sensing; by adjusting the amount of transmission tax (i.e., the number of superframes for sensing per transmitted packet or packet burst), the coordinator is able to maintain a steady influx of accurate sensing information in a timely fashion [18]. Part of the sensing report is the information about the time of sensing, which may be recorded at the desired level of granularity. In the simplest case, the coordinator will just record the superframe in which it received the report; alternatively, the node may report (and the coordinator will record) the nearest time slot in which the sensing actually took place. The necessary granularity level will be dictated by the dynamics of primary user activity, but the algorithms described below can use any suitable level without a problem.

It is worth noting that, unlike transmission, reception is not actually taxed. Namely, nodes that are to receive packets can temporarily suspend the sensing during that superframe, and resume it in the next one. Thus sensing is preempted by packet reception. However, a node can request bandwidth for a new packet only upon completing its sensing duty related to the previous transmission.

IV. Channel selection algorithms

Let us now describe the working channel selection algorithms in more detail. We assume that the sensing nodes perform sensing for a number of channels in each of the superframes which they spend doing sensing. We also assume that sensing reports are truthful; while reporting may be blocked by noise and interference, we can easily account for this through reducing the number of sensing reports per each superframe. We assume that the coordinator records the results of the sensing and, thus maintains a map of busy and idle channels. For each channel, the coordinator also records the last time at which the channel has turned idle or busy, although this last piece of information is not actually used in the algorithms that follow.

A. Algorithm 1: Selecting the most recently idle channel

This is the simpler algorithm of the two; it actually consists of selecting an idle channel that has most recently turned idle. The assumption is that the channel that has most recently turned idle has the highest probability of remaining idle during the next superframe. If there are several such channels (a scenario which is more likely if the coordinator just records the superframe, but still possible even at finer time granularity), the coordinator will choose one of the channels at random.

B. Algorithm 2: Selecting the most likely idle channel

The second algorithm attempts to estimate the probability that the channels which are currently idle will remain idle throughout the next superframe. To this end, the coordinator builds a histogram of idle periods for each channel, using the recorded times when the channel turns idle or busy. These values are then used to fill in histogram bins. The histogram is constructed according to the following constraints.

- The width of each bin is determined as the shortest possible duration of the superframe.
- The number of bins in the histogram may be determined on the basis of the following two factors. First, the range of values for the duration of the superframe [18], since the channel must remain idle for the duration of the next superframe – which is known at the time the decision about the channel is to be made. Second, the statistics of the channel idle period, since the last ending of channel activity may have occurred some time ago.

In this manner, the values in those bins correspond to the probability that the channel will remain idle for the time interval that corresponds to the appropriate bin. They are still not probabilities, as their sum is not equal to one. To obtain actual probabilities, bin values must be normalized with the sum of all bins prior to algorithm execution.

Now, for each channel that is currently idle, say i, the coordinator calculates the index of the bin that corresponds to the time interval from the last time when the channel has turned idle until the end of the next superframe. Since different channels may have turned idle at different times, and thus the corresponding bin numbers will differ; let this bin be labeled as $j(i)$. The coordinator then finds the next channel as the one with the maximum probability in the bin $j(i)$. As before, in (the much less likely) case there are several such channels, the decision is made by random choice. Note that the channel selection will actually depend on the calculated duration of the next superframe, unlike in algorithm 1.

C. Reference algorithm

The yardstick against which we will measure the performance of both algorithms is the simple random selection algorithm. In this approach, the next working channel is selected through random choice from the set of channels that are currently assumed to be idle according to the information in the channel map. It is worth noting that the set of idle channels may be empty. This may be the actual state of the medium, or may be just the result of the inertia of the sensing process and the resulting errors [14]. In this case, the piconet may attempt recovery, or simply decide to repeat the piconet formation process; detailed analysis of both processes is beyond the scope of the present paper.

D. Performance indicators: collision probability

The main performance indicators are the probability of collision with the primary source. Two types of collisions are possible:

- Type 1 collision occurs when the coordinator selects a channel it thinks is idle, but is in fact busy. The probability of this event is, in fact, the function of the channel selection algorithm but also of the accuracy of the sensing process (or lack thereof).
- Type 2 collision occurs when the channel becomes busy during the superframe, in which case the superframe will be damaged and the piconet must attempt recovery, i.e., it has to switch to another channel in the hope it is free.

V. Experimental evaluation

To evaluate the performance impact of channel selection algorithms, we have built a simulator of the cognitive piconet using the object-oriented, Petri net-based simulation engine Artifex by RSoftDesign, Inc. [19]. The simulator implements the transmission tax-based MAC protocol with integrated sensing and the aforementioned channel selection algorithms. Unless otherwise specified, all experiments were performed with the following parameter values:

- the piconet has 16 nodes and the coordinator;
- the working set of channels contains 11 channels;
- each channel contains an independent primary source with random activity with Erlang-distributed ON (busy) and OFF (idle) intervals; the mean period of primary user activity is 1000 time units and the mean ratio of active period vs. total period (i.e., activity factor or duty cycle) is 0.5;
- each node has a traffic generator which generates Poisson-distributed single packet traffic at a rate of 0.002 packets per node per time unit;
- transmission tax was set to 4 (expressed as the number of superframes of sensing duty per each packet or group of packets sent);
- superframe duration was fixed at 100 time units, 15 of which were set aside for administrative superframes.

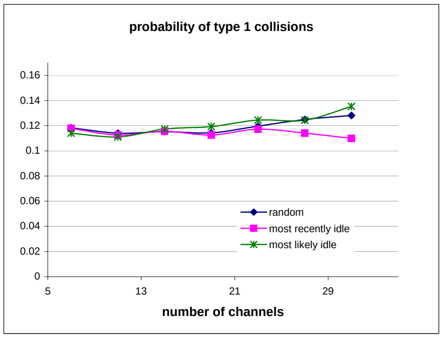
Our main performance indicators are the probability of type 1 and type 2 collisions, as defined above.

A. Homogeneous Poisson-distributed primary users

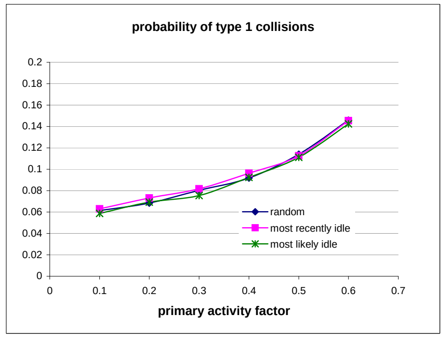
Our first experiments consisted of running the three algorithms with exponentially distributed busy and idle periods of primary sources, which corresponds to the shape parameter of Erlang distribution of $k = 1$. All primary sources use the same probability distribution, therefore we refer to this scenario as homogeneous. The results are shown in Fig. 2, with the number of channels, primary user activity factor, and primary user period as independent variables, respectively. The three curves overlap to the point of being indistinguishable in Figs. 2(c) or Fig. 2(f), which is due to the memoryless distribution that makes all channels exhibit equal probability of going busy, hence prediction does not bring any benefit whatsoever. We also observe that the longer idle periods lead to lower collision probability, regardless of whether they are cause by longer total activity period or lower values of the activity factor. However, collision probability is virtually independent of the number of channels.

B. The impact of shape parameter k

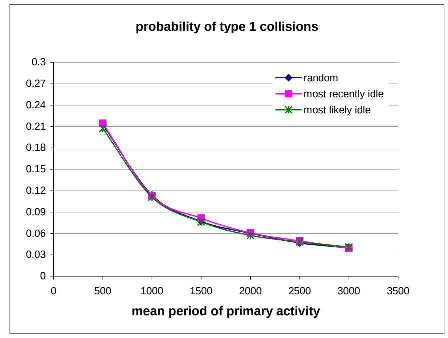
In our second experiment, we have kept the number of channels, the primary user activity period and activity factor as constants, and varied the shape parameter k of the probability distribution of primary user busy and idle periods. The resulting collision probabilities are shown in Fig. 3. As can be expected, the performance of the random selection algorithm is unaffected by the change in k. However, the other two algorithms behave rather differently: the most recently idle



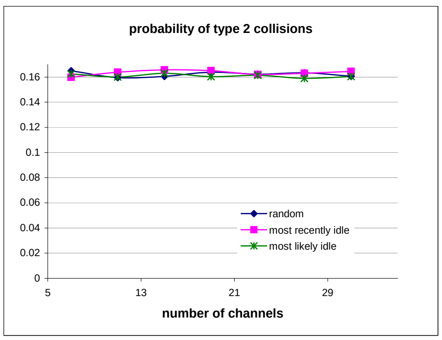
(a) Probability of type 1 collisions, variable number of primary channels.



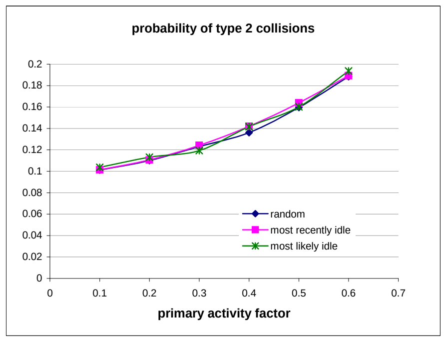
(b) Probability of type 1 collisions, variable primary user activity factor.



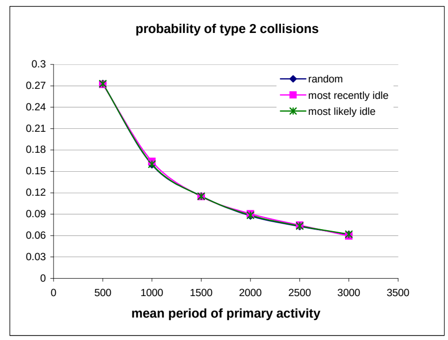
(c) Probability of type 1 collisions, variable primary user activity period.



(d) Probability of type 2 collisions, variable number of primary channels.



(e) Probability of type 2 collisions, variable primary user activity factor.



(f) Probability of type 2 collisions, variable primary user activity period.

Fig. 2. Performance of working channel selection algorithms under homogeneous primary user activity. Primary user period is 1000, primary user activity factor is 0.5, and the number of channels is 11, unless explicitly designated as variable.

algorithm (algorithm 1) exhibits a noticeable reduction in both types of collision probability, which is to be expected as the Erlang probability distribution for $k \geq 2$ is not memoryless any more. As a result, the basic assumption of the algorithm, namely, that the channel that has most recently turned idle has the highest probability of remaining idle during the next superframe, actually holds. At the same time, the most likely idle algorithm (algorithm 2) exhibits a slight increase of both types of collision probability, which is not expected, but may be attributed to the shape of the Erlang probability distribution. Namely, the presence of the peak may 'trick' algorithm 2 into choosing a channel which is already idle for a longer (and sometimes much longer) time, so its basic premise may not hold in this case. Yet the increase is not too big – it's about half the amount of the decrease exhibited by algorithm 1.

C. Heterogeneous primary users – case where only the activity factor is variable

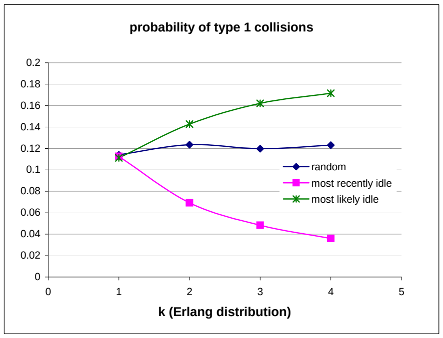
We have also investigated the case in which primary users have heterogeneous parameters, namely their activity factors are randomly chosen in the range from 0.1 to 0.9 while the mean activity period and the number of channels were independent variables. The shape factor of Erlang distribution was kept constant at $k = 3$. The resulting diagrams are shown in Fig. 4. In this case, algorithm 2 generally performs better than algorithm 1, although the collision probabilities appear to converge, esp. in case of variable mean activity period. This may be explained as follows: due to the heterogeneity of primary user activity factors, algorithm 2 performs much better than algorithm 1 at low values of mean activity period, where idle periods are harder to find. However, its relative advantage is offset by the 'peakiness' of the Erlang distribution with the increase in mean activity period or the number of channels.

D. Heterogeneous primary users – case where both the period and activity factor are variable

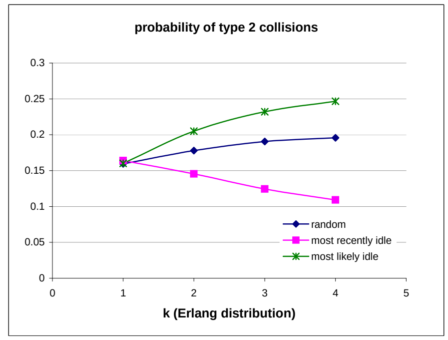
In our fourth and final experiment, both the mean period and the activity factor of primary user activity were selected as random values in the range 500 to 3000 time units for the former, and from 0.1 to 0.9 for the latter. The results are shown in Fig. 5; they clearly demonstrate the advantage of algorithm 2 over algorithm 1.

VI. Conclusions and Future Work

Overall, the results se results confirm our hypothesis that some knowledge of primary user activity patterns can reduce the probability that a cognitive piconet will experience collisions with primary user activity. Apparently algorithm 2, which relies on selection of the channel that is most likely to remain idle on the basis of estimated probability distribution of idle periods on each channel, has an advantage over

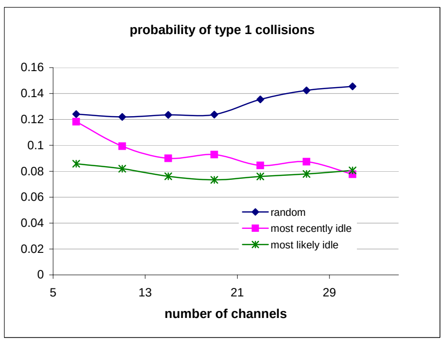


(a) Probability of type 1 collisions.

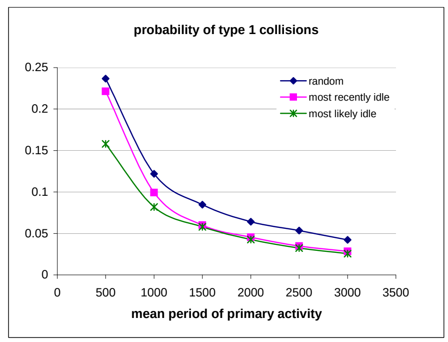


(b) Probability of type 2 collisions.

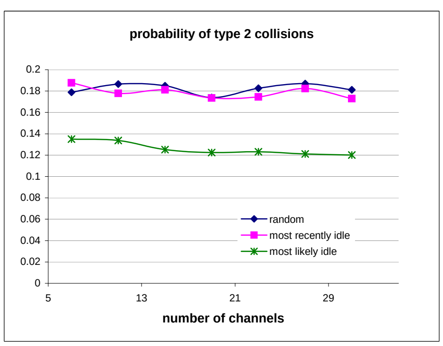
Fig. 3. Performance of working channel selection algorithms under variable shape parameter k. Primary user period is 1000, primary user activity factor is 0.5, and the number of channels is 11.



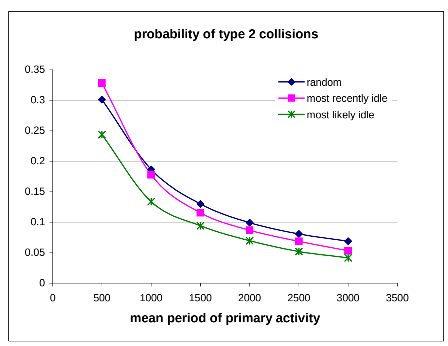
(a) Probability of type 1 collisions, variable number of primary channels.



(b) Probability of type 1 collisions, variable primary user activity period.



(c) Probability of type 2 collisions, variable number of primary channels.

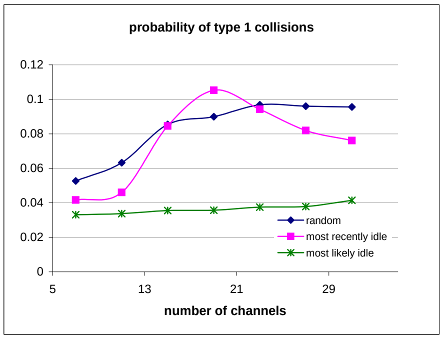


(d) Probability of type 2 collisions, variable primary user activity period.

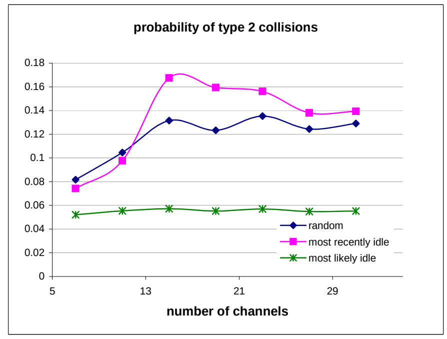
Fig. 4. Performance of working channel selection algorithms (heterogeneous primary users). The activity factor is randomly selected between 0.1 and 0.9, while the number of primary channels is 11. The shape factor of Erlang distribution is $k = 3$.

algorithm 1 where the working channel is selected amongst those that have most recently turned idle. However, the case of homogeneous primary sources shows that the advantage is not unqualified, and there may be cases where algorithm 1 performs better. Obviously the relative range in which one algorithm is better than the other and vice-versa should be examined in more detail.

Overall, the work reported here is but a start, and improved channel selection algorithms should be sought. In particular, the selection algorithm should look at the entire shape of the histogram that serves as a proxy for the probability distribution, rather than just at specific values. Also, the mean activity period of primary users should be taken into account and possible non-stationarity of primary user activity should be accounted for, perhaps through windowing or some other way to obtain a moving average of histogram values.



(a) Probability of type 1 collisions.



(b) Probability of type 2 collisions.

Fig. 5. Performance of working channel selection algorithms (heterogeneous primary users). The activity period is randomly selected between 500 and 3000 time slots, activity factor is randomly selected between 0.1 and 0.9, and the number of channels is 11. The shape factor of Erlang distribution is $k = 3$.

REFERENCES

[1] I. F. Akyildiz, W.-Y. Lee, M. C. Vuran, and S. Mohanty. NeXt generation/dynamic spectrum access/cognitive radio wireless networks: A survey. *Computer Networks*, 50:2127–2159, 2006.

[2] DARPA. The XG architectural framework. Request for comments, July 2003.

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

[4] A. O. Ercan, J. Lee, S. Pollin, and J. Rabaey. A revenue enhancing Stackelberg game for owners in opportunistic spectrum access. In *3rd IEEE Int'l Symp. on New Frontiers in Dynamic Spectrum Access Networks DySPAN 2008*, Chicago, IL, Oct. 2008.

[5] S. Geirhofer, L. Tong, and B. Sadler. A measurement-based model for dynamic spectrum access in WLAN channels. In *IEEE Military Communications Conference (MILCOM 2006)*, Oct. 2006.

[6] S. Geirhofer, L. Tong, and B. Sadler. Cognitive medium access: Constraining interference based on experimental models. *IEEE Journal on Selected Areas in Communications – Wireless Series*, 26(1):95–105, Jan. 2008.

[7] S. Geirhofer, L. Tong, and B. M. Sadler. Dynamic spectrum access in WLAN channels: empirical model and its stochastic analysis. In *Proc. first int. workshop on Technology and policy for accessing spectrum (TAPAS '06)*, 2006.

[8] S. Geirhofer, L. Tong, and B. M. Sadler. Dynamic spectrum access in the time domain: Modeling and exploiting white space. *IEEE Communications Magazine*, 45(5):66–72, May 2007.

[9] F. J. Harris. On detecting white space spectra for spectral scavenging in cognitive radios. *Wireless Personal Communications*, 45(3):325–342, May 2008.

[10] 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.

[11] A. P. Jr., M. Taghizadeh, and S. Biswas. Bandwidth scavenging for device coexistence in pervasive computing systems. *Pervasive and Mobile Computing*, 2012.

[12] J. Marinho and E. Monteiro. Cognitive radio: survey on communication protocols, spectrum decision issues, and future research directions. *Wireless Networks Journal (WINET)*, 18(2):147–164, Feb. 2012.

[13] J. Mišić and V. B. Mišić. *Performance Modeling and Analysis of Bluetooth Networks: Network Formation, Polling, Scheduling, and Traffic Control*. CRC Press, Boca Raton, FL, July 2005.

[14] 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.

[15] J. Mišić and V. B. Mišić. Simple and efficient MAC for cognitive wireless personal area networks. In *Proc. Global Telecommunications Conference GLOBECOM'09*, Honolulu, HI, Nov. 2009.

[16] V. B. Mišić and J. Mišić. Performance of a cognitive MAC protocol with transmission tax. In *IEEE INFOCOM-2010 Workshop on Cognitive Wireless Communications and Networking*, San Diego, CA, Mar. 2010.

[17] J. Mišić, H. Khojasteh, N. Khan, and V. B. Mišić. Towards an efficient rendezvous protocol for a cognitive PAN. In *IEEE International Conference on Communications (ICC'2012)*, Ottawa, ON, June 2012.

[18] V. B. Mišić and J. Mišić. Cognitive MAC protocol with transmission tax: Dynamically adjusting sensing and data performance. In *Proc. Global Telecommunications Conference GLOBECOM'10*, Miami, FL, Dec. 2010.

[19] RSoft Design. *Artifex v.4.4.2*. RSoft Design Group, Inc., San Jose, CA, 2003.