A Novel Simulation Framework for Wireless Cognitive Networks: Application to Cooperative Spectrum Sensing

Dimitri Tassetto, Sergio Bovelli
EADS Innovation Works (Sensors, Electronics & Systems Integration)
{dimitri.tassetto, sergio.bovelli}@eads.net

Auon Akhtar, Oliver Holland
Centre for Telecommunications Research, King's College London
{auon.akhtar, oliver.holland}@kcl.ac.uk

Luca De Nardis, Maria-Gabriella Di Benedetto
DIET Department, Sapienza University of Rome
{lucadn, dibenedetto} @newyork.ing.uniroma1.it

ABSTRACT

This paper proposes a novel simulation framework designed to assess the performance of wireless cognitive networks in realistic scenarios. The simulation framework is based on the combination of the OMNeT++ discrete event simulator with Matlab functionalities. The popular INET simulation package for OMNeT++ is extended with the introduction of cognitive functionalities and integrated with a more accurate wireless channel model developed in Matlab, which is employed to simulate propagation effects of any radio activity and serves as an input for spectrum sensing algorithms. The utilization of a general purpose network simulator makes it possible to evaluate the algorithm's efficiency from a system-level point of view; moreover, it allows us to take into account many factors which influence network behavior in real scenarios (for example, node mobility), that would usually be neglected in analytical analyses.

Keywords

Cognitive Networking, Cooperative Spectrum Sensing, IEEE 1900.6

1. INTRODUCTION

Cognitive radio has lately become one of the most popular research topics in the field of wireless communications. Since this is a comprehensive subject that can potentially consider operations at any layer of the ISO/OSI stack among an arbitrarily high number of nodes, the utilization of a discrete event simulator like OMNeT++[23] is an efficient way to allow for performance assessment even in complex scenarios. Given the observation that OMNeT++ relies on a quite simplistic abstraction of the physical layer, in this paper an extension of OMNeT++ based on the introduction of Matlab's advanced modeling capabilities is proposed. The creation of the proposed simulation framework, derived from a general purpose network simulator, facilitates in taking into account many factors that influence network behavior in real scenarios (for example, node mobility), but are usually neglected in performance evaluation studies based on analytical approaches. Furthermore, it allows to evaluate the algorithms' efficiency from a system level point of view, beyond what is usually achieved within physical layer simulators, by taking into account the cross-layer interactions between physical layer aspects (channel effect, sensing accuracy) and higher layer protocols and solutions related for example to network organization, medium access control, and routing.

The paper is organized as follows. Section 2 describes the simulation framework and introduces the definition of the sensing node, which has been implemented taking into account the primitives for interaction within different cognitive enabled entities as defined in the IEEE 1900.6 standard [1]. The utilization of the described simulation framework is then demonstrated by applying it to the specific case study of spectrum sensing. The traditional sensing techniques are described in Section 3, that also covers the case of cooperative sensing techniques based on the combination of measurements retrieved by multiple devices. Next, early simulation results highlighting the correctness of the proposed implementation are presented and discussed in Section 4. Finally, Section 5 discusses the interaction between the sensing function and higher layers, while Section 6 draws conclusions.

2. OVERVIEW ON SIMULATION FRAMEWORK STRUCTURE

2.1 State of the art

The use of a network simulator is a complementary approach to analytical calculation towards evaluation and comparison of cognitive capabilities in realistic scenarios. However, the efficiency of such methodology is strongly affected by the accuracy of the simulation, especially at physical layer. OMNeT++ is a discrete event simulation environment primarily used to implement higher layer protocols, e.g., routing protocols at the network layer. As a matter of fact most of the network simulators, including OMNeT++, rely on a quite simplistic abstraction of the physical layer. More specifically, the INET framework [2] first computes the received power level and corresponding Signal to Noise Plus Interference Ratio (SNIR) value using the popular attenuation law defined by Rappaport in [18], that models the shadowing effects by introducing a lognormal random term. Afterwards,

the packet error rate after channel decoding is estimated by simply extracting a random variable and comparing it with the Bit Error Rate (BER) curve of the modulation being used. The described code is built into the framework software as a default model. As such, the user cannot easily design custom physical layer models using OMNeT++.

To overcome the above limitations, this work introduces a detailed channel model developed in Matlab in the simulation of physical layer during OMNeT++ system simulations. Matlab is a software for numerical calculations and provides the user with a variety of tools that can be used to implement user defined channel models (for example, the well accepted models computing received signals as affected by Nakagami [16], Weibull [19], Rayleigh [21] or Ricean [13] fading). However, implementing higher layer functionalities in Matlab requires extensive programming, making it difficult to simulate the complex and dynamic behavior of a wireless network. By integrating Matlab with OMNeT++, one can reuse already existing code for physical layer models, for whom many research teams have usually preferred to develop Matlab software. Moreover, Matlab native support of matrix operations can be exploited when considering specific issues such as asymmetrical communications, network nodes equipped with directional antennas, as well as when an accurate description of the modulation schemes is required. Finally, the simulation of propagation effects may be enhanced taking into account the impact of spatial correlation among sensing observation: in fact it represents a key factor affecting the performance of collaborative spectrum sensing techniques that would be neglected using the original framework, since in this context any receiver runs independently from the others and regards solely its position relative to the transmitter's one.

A second point of interaction among the two programming languages resides in the implementation of spectrum sensing algorithms, that are coded in Matlab taking advantage of the built-in functions to compute the Fast Fourier Transform (FFT) of the received signals. The integration of the resulting code in the network simulator enables the users to test their spectrum sensing algorithms in a specific environment of their choice and to assess their performance and effects on the upper layers of the communication system.

2.2 Sensing node structure

The implementation of network nodes with spectrum sensing capabilities has been accomplished in accordance with the structure in the IEEE 1900.6 standard [1]. This is the most appropriate solution to guarantee the development of legacy-compliant devices and facilitate interoperability among terminals from different manufacturers. The standard identifies three main entities involved in the spectrum sensing activity:

- **Cognitive Engine (CE)**: the entity responsible for the cognitive control mechanisms, the collection of sensing measurements and the assessment of spectrum availability;

- **Sensor**: the entity that takes sensing measurements according to the configurations and commands requested by the CE;

- **Data Archive (DA)**: a storage unit for sensing-related information.
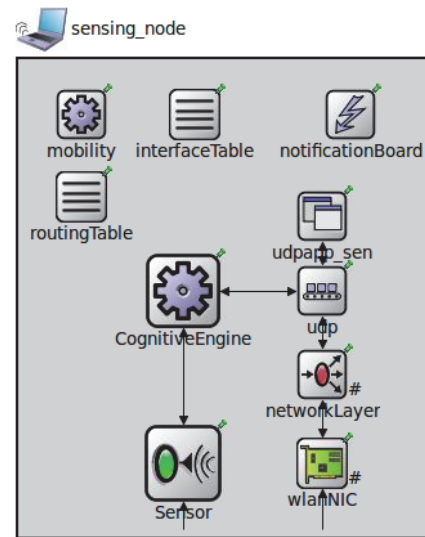

Figure 1: Design structure of a sensing node.

The interactions among these entities are intended to take place using the interfaces provided by three Service Access Points (SAPs): the Measurement-SAP (M-SAP), exploited to access the sensing-related information; the Application-SAP (A-SAP), used by the control application to benefit from the sensing services; and the Communication-SAP (C-SAP), which is at users' and clients' disposal to provide communication between the aforementioned entities. The simplifying assumption has been made in this work that a node with spectrum sensing capabilities, later on referred as "sensing node", is always embodying a CE (which is optionally equipped with a DA) and at least one Sensor entity, as illustrated in Figure 1. As a consequence, some of the requirements on the definitions of primitives to be used for the communication among the entities involved in spectrum sensing may be relaxed. Specifically, the C-SAP in this case is solely interconnecting the CE entities of peer sensing nodes; hence, sensor entities belonging to different sensing nodes communicate by means of their integrated CEs. Since the standard does not constrain the specific communication protocol to be used, two different methods were developed for the interactions among entities in the same sensing node and the communication between peer CEs. In the first case the most efficient solution is to exploit the utility called Notification Board: this module, already included in the standard INET framework, collects any notification of parameters, variables and module state changes occurring within the compound module modeling a network node at any time during the simulation period. A module interested in a particular event can subscribe to it, so to receive an immediate notification as soon as a change in the corresponding parameter/variable occurs.

The utilization of this module is the most straight-forward method to communicate within a compound module and allows distributing notifications to a set of subscribers avoiding the definition of specific communication protocols or the transmission of duplicated messages over multiple ports. On the other hand, peer sensing nodes are logically connected through their combined CEs at the application level of a

Table 1: Messages and events to exchange sensing information

Name	Parameters	Type(Direction)
Get_Spectrum_ Measurement_ Description	Status, MeasuRange, Sensing-Mode, FrequencyResolution, LocationTimeCapability	Event (CE <–> SENS)
Set_Sensor_ Measurement_Obj	StartTime, EndTime, Bandwidth	Event (CE –> SENS)
Set_Sensor_ Measurement_Profile	SensingMode, ChOrder, ChList, ReportRate, Scan.Lower_ Threshold, ReportingMode	Event (CE –> SENS)
Channel_ Measurement_Value_ Resp	Status, Bandwidth, Noise-Power, SignalLevel	Event (SENS –> CE)
Sensing_ Related_ Information_Send	InfoSource, InfoDestination, ReportMode, ReportRate, SensingRelatedInformation	Message (CE –> CE)
Sensing_ Related_ Information_Receive _Req	InfoSource, InfoDestination, ReportMode, ReportRate	Message (CE –> CE)

standard ISO/OSI protocol stack, and their interaction is regulated by a dedicated messaging protocol. A subset of the primitives listed in the standard has been implemented to accomplish the tasks of configuring the measurements and retrieving the sensing output, listed in Table 1 (a detailed description of the parameters defined in the standard can be found in [1]).

More specifically, the CE can acquire a description of the sensor capabilities (frequency band to be scanned, sensing techniques to be used etc.) and configure the sensor measurements by means of instructions on the temporal duration of the sensing campaign, the channel to be scanned and their ordering, the report rate and the report mode (that is if the CE expects to receive hard, soft or quantized reports). Once the measurement settings are retrieved each sensor begins to collect received power samples until the sensing period is over. The processing of the samples is then performed differently for network nodes operating in soft measurement and hard measurements mode.

In the first case the list of collected samples is transferred to the Cognitive Engine, where the measurements obtained from the cooperating sensing nodes are collected and a Matlab function implementing the spectrum sensing algorithm requested during the configuration phase is called.

On the other hand, in case of hard measurement mode the list of samples is immediately examined in order to obtain a local decision on the status of the frequency band of interest in the current period. Then, peer Cognitive Engines communicate their results using the defined messages to request and send sensing-related information and finally determine a global decision applying a fusion rule (e.g. AND, OR, majority rule).

3. SPECTRUM SENSING TECHNIQUES

Spectrum sensing belongs to the main capabilities of a cognitive device, since it enables it to acquire awareness on the surrounding environment and detect spectrum availabilities [3]. In the following subsections the sensing measurements output and the sensing techniques selected for the simulations are described.

3.1 Output of spectrum sensing measurements

It is usually accepted to consider as the final output of the spectrum sensing activities a verdict on the presence or absence of primary users in the three dimensions of frequency, time and space. More exactly, given a particular frequency channel, a spectrum hole can be characterized as either a spatial or a temporal opportunity. In the former case it represents a geographical region where a secondary transmitter can send at a pre-defined maximum power without causing harmful interference to primary receivers in the same region. In the latter one the spectrum hole corresponds to a temporal interval during which there is no radio activity by any primary user. It is also possible, as proposed in [9], to adopt a joint spatial-temporal spectrum sensing technique, where the final decision on the ON-OFF state of the primary transmitter is taken considering the individual decisions of a subset of sensors selected according to the information retrieved through the spatial spectrum sensing, i.e. transmitted power and position of the primary user. However, raw measurements collected by sensors could also be exploited to get more elaborated analysis on the spectrum neighborhood if they are examined in the light of higher layers (e.g. MAC) characteristics of the technology generating the intercepted signals. First of all, it is necessary to identify the MAC protocols utilized by the networks which are active in the region of interest. A possible approach is described in [5], where the features extracted from the observed traffic serve as the input for linear classifiers capable of determining the technology being used. Once the MAC protocol has been successfully recognized it would be possible to predict some aspects of the network behavior, like the periodic transmission of beacons, and use them in addition to the sensing information to determine future transmission opportunities.

3.2 Spectrum sensing algorithms

As regards the spectrum sensing algorithms used, three broad approaches are reviewed in the following. The first approach is simple energy detection, whereby the metric used by the sensor is the received energy in a certain time duration, to be compared with a threshold value to determine whether or not a signal is present. The second approach is autocorrelation detection, where the autocorrelation function of a signal is studied and the resulting characteristics of the peaks of

that function at lags equivalent to plus or minus the symbol duration are studied to determine if a signal is present. The third approach is cyclostationary detection, whereby peaks resulting from autocorrelation shifts corresponding to symbol durations are considered, as well as Fourier coefficients of the autocorrelation function of the signal, which is cyclic in such cases. For the autocorrelation and cyclostationary detection cases, this work considers scenarios where one of the cyclic coefficients is used as the metric, and where multiple peaks are used. For the case of cooperative sensing it is assumed that each sensor conveys a hard statistic ("signal present", or "signal not present") to the CE. The CE, or a collaboration of CEs in the cases were the sensor and CE are combined into a node, takes then the decision on whether a primary signal is present or not in the vicinity, by simply assuming that if one or more of the sensors returns a positive "signal present", then the signal is present ("OR" rule).

3.3   Cooperative spectrum sensing

As already explained in the previous section, several techniques have been developed to perform spectrum sensing resulting in different outcomes in terms of complexity, required a priori information on primary signal characteristics and accuracy. However, any technique if applied on a single node exhibits poor performance in presence of noise uncertainty, hidden primary transmitters or due to the degradation of the SNR value caused by small-scale fading and shadowing effects. The use of cooperative activities usually allows to alleviate these difficulties and to achieve a decrease in term of probability of false alarm and missed detection, although the optimal results are only obtained if the cooperating nodes observe independent channel conditions.

The work in [17] analyses indeed the impact of correlation between measurements and sensing decisions on cooperative sensing performance: the authors show that as the number of sensors increases, the correlation between the measurements increases as well, reducing and eventually nullifying the positive effect of introducing new sensors in the sensing procedure. For a correlation index as low as 0.2, the net effect of increasing the number of sensors is actually to reduce the sensing performance as a result of an increased probability of missed detection. Such a result suggests that effective cooperative spectrum sensing requires in most cases the selection of a subset of sensors on the basis of a set of selection criteria aiming at the maximization of sensing performance. Several schemes have been proposed for introducing cooperation in sensing. A simple solution is proposed in [12], where a network of n devices sensing the environment by means of an energy-detection receiver is considered. The algorithm proposed in [12] combines the individual decisions of the devices according to an OR rule, leading the whole network to decide that a primary is present if any of the devices decides so. Such approach leads to an improvement in the probability of detection, at the price of an increase in the probability of false alarm.

Other authors introduce cooperation in a more advanced form: this is the case for the solution proposed in [10], where two sensing devices cooperate by alternating listening and transmitting phases according to a Time Division Multiple Access scheme, each phase being formed of two slots. The peculiarity of the solution proposed in [10] resides in the decision to use the first slot in each transmitting phase to relay the signal received in the second slot of the listening phase,

actually implementing an Amplify and Forward scheme. Results in [10] show that each device achieves a higher probability of detecting a primary user by subtracting its own (known) signal transmitted in the transmitting phase from the received signal before performing sensing. The solution is extended in [11] to the case of multiple users, although the proposed solution poses scalability issues due to the required large control traffic exchanges.

Other solutions rely on a clustered network architecture, where a hierarchy is introduced between sensing devices in order to reduce control traffic and spare part of the devices from the energy consumption related to sensing. This is the case of [22], where however the criterion for grouping devices in the same cluster is not defined, and [4], where a sensing and energy aware clustering strategy is proposed.

An additional relevant distinction can be made between fusion strategies applied to raw measurements (soft decision) or to binary outcomes on spectrum vacancy (hard decision). Soft decision schemes [14] [20] offer the highest flexibility and can achieve better accuracy. However, their requirements in terms of bandwidth may become prohibitive, especially when the number of cooperative nodes increase and no dedicated channel has been reserved for measurements reports. For this reason the intermediate solution of taking a quantized reported measurement from all the participating nodes is frequently considered . The authors in [15] proved that the utilization of a 2-bit quantized value computed after comparing the measurements with three energy thresholds is sufficient to guarantee a substantial improvement with respect to the hard scheme, whereas the quantization loss causes a decrease in the probability of detection limited to few percentage units.

When moving from the sensing node structure defined in Section 2.2, the realization of a cooperative sensing technique in the simulated network is straight-forward. Provided services are in fact already intended to offer send and receive abilities to CE entities. These abilities are exploited to collect sensing related data, namely local decision or soft measurements, generated by different sensing nodes. Depending on the selected sensing protocol, these information is grouped in a single sensing node (centralized cooperative sensing), in a subset of the nodes (cluster-based sensing) or in all sensing nodes (distributed sensing), where the CE finally applies the fusion algorithm and retrieves the output of spectrum sensing operation.

4.   SIMULATION RESULTS

The considered simulation scenario foresees one primary transmitter and a varying number of cooperative sensors, situated within a square area of 1km by 1km. The simulations are all performed over 10,000 iterations, whereby for each iteration there is an independently-sampled random placement of the primary transmitter and the sensors. The primary transmitter transmits with a power of 40mW, and the channel is assumed to be multipath fading with lognormal shadowing.

Given the above scenario, results in [24] reflecting single-sensor performance in a DVB-T detection case are extended to the multi-sensor cooperative sensing case. This is done by assessing one minus the joint probability of all sensors failing to detect the primary signal, which is of course the probability of one or more of the sensors successfully detecting the signal (i.e., the probability one is interested in when adopting an OR decision rule). The calculation is done
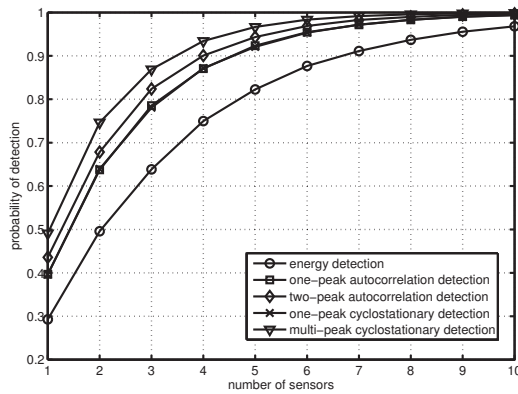
Figure 2: Probability of successful detection as a function of the number of sensor nodes.

for each iteration, hence random placement of the primary transmitter and sensors, whereby the average performance is then taken among all simulation iterations.

Figure 2 plots the probability of successful detection as a function of the number of cooperating sensors for five spectrum sensing techniques: energy detection, one-peak autocorrelation detection, two-peak autocorrelation detection, one-peak cyclostationary detection, and multi-peak cyclostationary detection. This figure shows that the probability of successful primary detection increases as the number of cooperating sensor nodes is increased. Furthermore, multi-peak cyclostationary detection gives the best performance among the five techniques considered, achieving a viable detection performance (i.e., 99% probability of detection) for some 7 or more cooperating sensors being in the scenario. It is noted that if 9 or more sensors are cooperating in the scenario all investigated sensing schemes give a viable performance, with the exception of energy detection.

5. INTERACTIONS WITH HIGHER LAYERS

The introduction of cooperation in spectrum sensing can provide a significant performance increase in sensing reliability, while distributing the resource and energy consumption connected to sensing between several devices. In order to translate this performance increase for sensing in an actual increase in network performance, two key aspects related to the interaction between the sensing function and higher layers function should be taken into account:

- Transfer and integration of sensing information - sensing information should be transferred to higher layers and integrated to the maximum extent in cost and utility functions adopted at such layers. Routing and admission control, just to mention two end-to-end functions, should include sensing information in their rules of operation; a similar approach has been followed in the past by taking into account the role of sensing in higher layer protocols such as transport protocols [6], [8] and even up to the application layer [7].

- Combination of sensing and communications requirements in the determination of network organization - It was pointed out in Section 3.3 that achieving optimal sensing performance may require the selection of a subset of network devices and their organization in a hierarchical fashion. Requirements imposed by communications-related aspects should not however be neglected, as the final goal of a cognitive network is in most cases to transfer data traffic. This means for example that in a clustered network architecture the determination of the composition of clusters should be the result of a trade-off between sensing efficiency and communications requirements: the resulting organization should thus be the one that provides the best overall performance, rather than the one the maximizes sensing performance.

6. CONCLUSIONS

This paper has described the novel simulation framework for cognitive radio scenarios being developed at EADS Innovation Works, and contributed to by Sapienza University of Rome and King's College London, and the application of that framework to spectrum sensing scenarios. It has introduced the background to such a framework, including state-of-the-art developments, and has then detailed the framework itself by addressing aspects such as the implementation along the lines of the 1900.6 standard and the entities and service access points therein, and the simulation mechanics such as the need to combine the core tool, OMNeT++, with Matlab capabilities.

The paper moved then to the discussion of the procedure and results of simulations for the application to spectrum sensing. It has shown the performance of the investigated range of spectrum sensing algorithms in such a cooperative spectrum sensing scenario, for different numbers of sensors cooperating. The simulation framework may also represent an effective tool to assess the performance of other algorithms executing typical operations of a cognitive radio device. The authors plan to contact the INET framework developers' community to inform it about the availability of the framework extension for cognitive network simulations described in this paper.

7. ACKNOWLEDGMENTS

8. REFERENCES

[1] IEEE Std 1900.6^{TM}-2011 IEEE Standard for Spectrum Sensing Interfaces and Data Structures for Dynamic Spectrum Access and Other Advanced Radio Communication Systems, accessed February 2012.

[2] The INET simulation package, http://inet.omnetpp.org/index.php, accessed February 2012.

[3] I. Akyildiz, W.-Y. Lee, and K. Chowdhury. Spectrum management in cognitive radio ad hoc networks. 23(4):6–12, 2009.

[4] L. De Nardis, D. Domenicali, and M.-G. Di Benedetto. Clustered hybrid energy-aware cooperative spectrum

sensing (chess). In *Proc. 4th Int. Conf. Cognitive Radio Oriented Wireless Networks and Communications CROWNCOM '09*, pages 1–6, 2009.

[5] M. Di Benedetto, S. Boldrini, C. J. M. Martin, and J. R. Diaz. Automatic network recognition by feature extraction: A case study in the ism band. In *Proc. Fifth Int Cognitive Radio Oriented Wireless Networks & Communications (CROWNCOM) Conf*, pages 1–5, 2010.

[6] M. Di Felice, K. Chowdhury, and L. Bononi. Modeling and performance evaluation of transmission control protocol over cognitive radio ad hoc networks. In *Proceedings of the 12-th ACM International Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems (MSWiM 2009)*, 2009.

[7] M. Di Felice, K. Chowdhury, and L. Bononi. Cooperative spectrum management in cognitive vehicular ad hoc networks. In *Proceedings of the 3rd IEEE Vehicular Networking Conference 2011 (IEEE VNC 2011)*, 2011.

[8] M. Di Felice, K. Chowdhury, W. Kim, A. Kassler, and L. Bononi. End-to-end protocols for cognitive radio ad hoc networks: An evaluation study. *Elsevier Performance Evaluation Journal*, 68(9):859–875, 2011.

[9] T. Do and B. L. Mark. Joint spatial-temporal spectrum sensing for cognitive radio networks. In *Proc. 43rd Annual Conf. Information Sciences and Systems CISS 2009*, pages 124–129, 2009.

[10] G. Ganesan and Y. Li. Cooperative spectrum sensing in cognitive radio, part i: Two user networks. *IEEE Transactions on Wireless Communications*, 6(6):2204–2213, 2007.

[11] G. Ganesan and L. Ye. Cooperative spectrum sensing in cognitive radio, part ii: Multiuser networks. *IEEE Transactions on Wireless Communications*, 6(6):2214–2222, 2007.

[12] A. Ghasemi and E. S. Sousa. Collaborative spectrum sensing for opportunistic access in fading environments. In *Proc. First IEEE Int. Symp. New Frontiers in Dynamic Spectrum Access Networks DySPAN 2005*, pages 131–136, 2005.

[13] W. Lindsey. Error probabilities for rician fading multichannel reception of binary and n-ary signals. 10(4):339–350, 1964.

[14] J. Ma, G. Zhao, and Y. Li. Soft combination and detection for cooperative spectrum sensing in cognitive radio networks. *IEEE Transactions on Wireless Communications*, 7(11):4502–4507, 2008.

[15] M. Mustonen, M. Matinmikko, and A. Mammela. Cooperative spectrum sensing using quantized soft decision combining. In *Proc. 4th Int. Conf. Cognitive Radio Oriented Wireless Networks and Communications CROWNCOM '09*, pages 1–5, 2009.

[16] N. Nakagami. The m-distribution, a general formula for intensity distribution of rapid fading. In W. G. Hoffman, editor, *Statistical Methods in Radio Wave Propagation*. Oxford, England: Pergamon, 1960.

[17] N. Pratas, N. Marchetti, A. Rodrigues, and R. Prasad. Capacity limits introduced by data fusion on cooperative spectrum sensing under correlated environments. In *Proc. 8th Int Communications (COMM) Conf*, pages 497–502, 2010.

[18] T. S. Rappaport. *Wireless Communications - Principles and Practice*. Prentice Hall, 2002.

[19] N. C. Sagias and G. K. Karagiannidis. Gaussian class multivariate weibull distributions: theory and applications in fading channels. 51(10):3608–3619, 2005.

[20] B. Shen and K. S. Kwak. Soft combination schemes for cooperative spectrum sensing in cognitive radio networks. *ETRI Journal*, 31(3), 2009.

[21] B. Sklar. Rayleigh fading channels in mobile digital communication systems. i. characterization. 35(9):136–146, 1997.

[22] C. Sun, W. Zhang, and K. Ben. Cluster-based cooperative spectrum sensing in cognitive radio systems. In *Proc. IEEE Int. Conf. Communications ICC '07*, pages 2511–2515, 2007.

[23] A. Varga. Omnet++. *IEEE Network Interactive*, 16(4), 2002.

[24] J. Wang, A. Pervez, and M. Nekovee. A review and verification of detection algorithms for dvb-t signals. In *Proc. 7th Int Communication Systems Networks and Digital Signal Processing (CSNDSP) Symp*, pages 469–473, 2010.