

# Throughput Optimization of Wireless LANs by Surrogate Model Based Cognitive Decision Making

Mostafa Pakparvar\*, Krishnan Chemmangat<sup>†</sup>, Dirk Deschrijver\*, Michael Mehari\*, David Plets\*, Tom Dhaene\*, Jeroen Hoebeke\*, Ingrid Moerman\*, Luc Martens\* and Wout Joseph\*

\*Ghent University/iMinds, Gaston Crommenlaan 8, Bus 201,9050 Gent, Belgium

<sup>†</sup>Amrita Vishwa Vidyapeetham, School of Engineering, Bengaluru Campus, Bangalore, 560035, India

Email: Mostafa.Pakparvar@intec.ugent.be

**Abstract**—Large scale growth of wireless networks and the scarcity of the electromagnetic spectrum are imposing more interference to the wireless terminals which jeopardize the Quality of Service offered to the end users. In order to address this kind of performance degradation, this paper proposes a novel experimentally verified cognitive decision engine which aims at optimizing the throughput of IEEE 802.11 links in presence of homogeneous IEEE 802.11 interference. The decision engine is based on a surrogate model that takes the current state of the wireless network as input and makes a prediction of the throughput. The prediction enables the decision engine to find the optimal configuration of the controllable parameters of the network. The decision engine was applied in a realistic interference scenario where utilization of the cognitive decision engine outperformed the case where the decision engine was not deployed by a worst case improvement of more than 100%.

## I. INTRODUCTION

With the increasing demand of the scarce electromagnetic spectrum and rising interference effects, cognitive radios (CR) have become a promising solution to address spectrum over-utilization. To this end, cognitive solutions are of paramount importance in wireless networks to ensure that their performance is not degraded by the external interference, ensuring continuous connectivity and providing a good Quality of Service (QoS). A well-established component that targets optimal network communication is the Cognitive Decision Engine (CDE). The CDE is an intelligent module that aims to optimize network performance objectives (QoS) by making informed decisions upon changes in the wireless environment.

Three important aspects of a CDE comprise: (1) observation, (2) reconfiguration or adaptability and, (3) cognition which includes awareness, reasoning and learning [1]. In literature, these three characteristics have been addressed using various approaches, e.g. rule-based [2], case-based [3]–[5], search-based [6], knowledge-based [4] reasoning systems and many others. Such methods often require a certain set of representative cases or domain expert knowledge [7] to derive important analytical formulas or rules that steer the decision making process. The reader is referred to [1] for a detailed survey.

This paper focuses on the development of a model-based CDE that reconfigures the transmission parameters of a wireless LAN system in response to measured changes in the environment. This task is facilitated by an underlying surrogate model that characterizes and mimics the behavior of the

wireless system. It takes the current state of the wireless environment and a given configuration of the wireless system as input and makes a prediction of the resulting QoS. Rather than building a static set of models upfront, a self-learning modeling strategy is adopted that performs a limited set of experiments in a structured way to collect data. By using this data, a predictive surrogate model is built that can generalize this knowledge towards environmental conditions and configurations that are different from those that were observed during training by means of interpolation. This model will become an essential part of the decision engine. Once the model is sufficiently accurate, it can be deployed in an optimization procedure that maximizes the attainable QoS in real-time by optimizing the transmission parameters. If the model predictions deviate from those that are observed in practice, the model can be updated in an on-line fashion to improve its accuracy over time. The solution is experimentally demonstrated and verified using a realistic use case in a wireless testbed where the proposed algorithm is shown to optimize the throughput of a wireless link by changing the frequency channel using surrogate model predictions.

This paper is organized as follows: Section II gives a schematic overview of the different components that are present in a wireless network with a CDE architecture. In Section III, some notational conventions and details on the surrogate model for throughput prediction are explained. More explanations on the experimentation and model building is given in Section IV. The optimization algorithm that is used in the CDE to optimize the throughput of the network is briefly explained in Section V. Finally, in Section VI, the real-life example is provided wherein experiments are performed in a wireless testbed to show the applicability and performance of the proposed CDE. Section VII concludes this article.

## II. SCHEMATIC OF COGNITIVE DECISION ENGINE

A schematic overview of the decision engine is shown in Figure 1. **The System under test** is the wireless system that needs to be optimized in a dynamically changing environment. **The Radio Environment Map (REM)** is a map that continuously aggregates all information that is monitored from the environment through spectrum sensing, network sniffing, and benchmarking tests [8]. Based on the REM map, a set of meters or measurable parameters (MP) can be extracted that allow the CDE to identify the current state of the environment. **The Cognitive decision engine (CDE)** is a self-learning decision engine that uses a surrogate model to optimize the

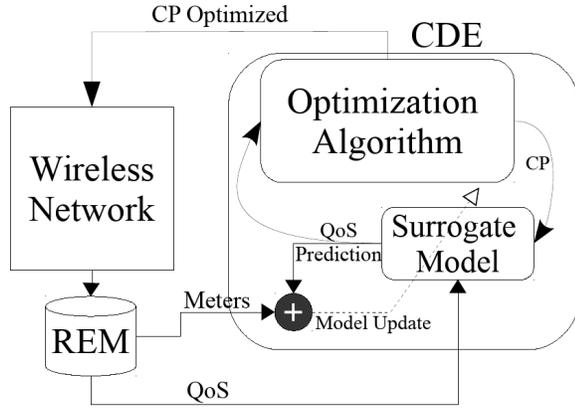


Fig. 1. A high-level schematic of the cognitive decision strategy

QoS by defining suitable cognitive actions in response to the MP parameters that are derived from REM. These actions are enforced by tuning of a set of knobs or control/transmission parameters (CP) that specify the configuration of the wireless system. By storing the outcome of each action, the algorithm collects data samples that quantify the QoS performance. *The surrogate model* is an analytical black-box model that characterizes and approximates this data. It is an input-output mathematical function that models the relation between the MP and CP parameters as inputs on one hand, and the resulting QoS performance of the wireless system that would be observed as an output on the other hand. *The optimization algorithm* exploits the surrogate model's predictions to optimize the QoS performance by tuning the CP parameters of the wireless system according to the MP parameters that are obtained from the REM. The outcome of the actions that were executed by the optimization process can be used in a feedback loop to update/improve the accuracy of surrogate model online during its operation. A more detailed explanation of these components will be provided in the later sections.

### III. SURROGATE MODELING OF QoS

#### A. Notational conventions

In order to model the QoS performance of the wireless system, a set of data samples must be collected by performing a limited set of  $k = 1, \dots, K$  experiments on a wireless testbed (w-iLab.t) as described in Section VI. Each experiment  $k$  is defined by a vector of  $i = 1, \dots, n_c$  knobs spanning the control space  $\mathbf{C}_k = \{C_k^i\}$ , and a vector of  $j = 1, \dots, n_m$  spanning the meter space  $\mathbf{M}_k = \{M_k^j\}$ . As mentioned in the previous section, the  $\mathbf{C}_k$  space comprises the control parameters that must be configured by the decision engine, whereas the  $\mathbf{M}_k$  space comprises a set of features that are extracted from the REM. The union of the control space  $\mathbf{C}$  and the meter space  $\mathbf{M}$  is called the parameter space. The outcome of each experiment  $k$  corresponds to a resulting  $QoS_k$  performance value. Depending on the ultimate usage of the wireless network, the target QoS parameter may be throughput, delay, jitter, or a combinatorial metric such as audio or video quality. Each experiment  $k$  is referred to as a data sample and will be represented by a tuple:  $\{\mathbf{C}_k, \mathbf{M}_k, QoS(\mathbf{C}_k, \mathbf{M}_k)\}$ .

#### B. Surrogate model type

All experiments that are performed constitute a dataset that will be used to build a surrogate model  $f$  such that  $f(\mathbf{C}_k, \mathbf{M}_k) = QoS_k$ . As surrogate models must be built and updated in real time, it is important to select an interpolation strategy that is simple and fast. It should 1) not require a lot of model tweaking; 2) be able to build models that are sufficiently accurate; 3) be robust towards noise and imperfections in the data; 4) have acceptable running times. In literature, many modeling algorithms are described, such as e.g. kriging [9], artificial neural networks [10], [11], radial basis function [12], etc. Most of the time, they require a trial-and-error approach for selecting a suitable set of hyperparameters (e.g., the number of neurons, the number of hidden layers, the model complexity, etc.). Also the model selection and cross-validation process can be very time consuming. With these considerations in mind, it is preferable to consider local interpolation schemes such as multi-linear interpolation [13], tessellation-based simplicial interpolation [14] and Shepard's interpolation [15], [16]. The Shepard interpolation algorithm was found to be the most adequate approach because of the simplicity of the underlying principle, and the speed in calculation. By using Shepard's interpolation, a model is built that exactly predicts the QoS performance (throughput) for experiments that were performed previously, and approximates the QoS for experiments that have not yet been performed (i.e. different values of  $\mathbf{C}$  and  $\mathbf{M}$ ), based on a distance-based similarity measure and an appropriate normalization of the parameter space.

### IV. MODEL BUILDING

In order to build the surrogate model, a representative set of data samples must be collected to build a model having sufficient accuracy to make reliable predictions of the QoS. As the QoS performance of the system under test depends on multiple parameters, it would be very consuming to collect data samples according to a grid-based structure. Instead, data samples are collected in such a way that the model accuracy can be maximized while minimizing the number of experiments needed. This happens in an online training phase that is executed prior to the deployment of the decision engine. In sequential steps, a well-chosen set of experiments are performed by making a balanced trade-off between two different criteria, namely exploration and exploitation.

#### A. Exploration phase

In the exploration phase, different settings of the knobs  $\mathbf{C}$  are explored in order to cover the parameter space as evenly as possible. The idea is that the most informative experiments are those that are as different as possible from those that were performed previously. Thereto, a distance-based exploration criterion is used to let the dataset grow over time. First, a large set of candidate settings is generated using a Monte Carlo method for the knobs in such a way that they cover the entire control space. For all the candidate settings of the knobs  $\mathbf{C}$ , the distance of each candidate point to all other data samples in the set is calculated and the farthest point is selected. So, if an  $n_c$ -dimensional control space with  $d = 1, \dots, D$  discrete candidate settings for the knobs is considered, then the proposed setting of the knobs is the one that maximizes the

minimum distance to the available data samples  $k = 1, \dots, K$ .

$$d^{\text{best}} = \arg \max_d \left[ \min_k h_k^d \right] \quad (1)$$

where  $h_k^d$  represents the Euclidean distance between data sample  $k$  in the dataset and candidate setting  $d$ . The values of the meters are set to those at current time  $\mathbf{M} = \mathbf{M}_{\text{current}}$ .

$$h_k^d = \sqrt{\sum_{i=1}^{n_c} (C_d^i - C_k^i)^2 + \sum_{j=1}^{n_m} (M_{\text{current}}^j - M_k^j)^2} \quad (2)$$

Once this exploratory setting  $d^{\text{best}}$  of the knobs is applied on the system under test, the resulting QoS is evaluated and the outcome is added as an additional data sample to the dataset. As the environment changes over time, the meters  $\mathbf{M}_{\text{current}}$  will change dynamically over time and sequentially data samples are added to the dataset. This leads to more accurate predictions of the surrogate model built from it.

### B. Exploitation phase

As the algorithm collects more data samples, it can also exploit data from previous experiments to make the model more accurate in those regions where the optimum configurations (i.e. those with higher QoS values) are located. To this end, the output of the intermediate surrogate models can be used to identify these settings. Just like in the exploration phase, the meters are monitored and a large set of candidate settings for the knobs are generated using Monte Carlo in such a way that they cover the entire control space. For all discrete candidate settings  $d$  of the knobs  $\mathbf{C}$ , the surrogate model is evaluated and the setting  $\mathbf{C}^{\text{opt}}$  is chosen for which the Shepard's model predicts the highest  $PQoS = f(\mathbf{C}_d, \mathbf{M}_{\text{current}})$  value.

$$\mathbf{C}^{\text{opt}} = \arg \max_d f(\mathbf{C}_d, \mathbf{M}_{\text{current}}) \quad (3)$$

The values of meters  $\mathbf{M}_{\text{current}}$  in (3) are again set to those at the current time. Once the optimal setting  $\mathbf{C}^{\text{opt}}$  of the knobs is applied on the system, the resulting  $QoS(\mathbf{C}^{\text{opt}}, \mathbf{M}_{\text{current}})$  value is evaluated and the outcome is compared to the predicted  $PQoS = f(\mathbf{C}^{\text{opt}}, \mathbf{M}_{\text{current}})$  of the surrogate model.

$$|f(\mathbf{C}^{\text{opt}}, \mathbf{M}_{\text{current}}) - QoS(\mathbf{C}^{\text{opt}}, \mathbf{M}_{\text{current}})| < \tau \quad (4)$$

If the discrepancy of the outcome is larger than a predefined threshold  $\tau$ , then the prediction of the model was not sufficiently accurate and a corresponding data sample  $\{\mathbf{C}^{\text{opt}}, \mathbf{M}_{\text{current}}, QoS(\mathbf{C}^{\text{opt}}, \mathbf{M}_{\text{current}})\}$  is added to the dataset. If the surrogate model prediction was sufficiently accurate, then the experiment can be discarded. Since the QoS metric chosen in this paper is throughput, a value of  $\tau = 3[\text{Mbps}]$  is proposed. This step improves the accuracy of the surrogate model in regions where optima are located.

## V. MODEL OPTIMIZATION

Once a surrogate model is considered to be sufficiently accurate, it can be deployed in the cognitive decision framework for real-time optimization. The optimization algorithm continuously monitors the QoS performance and the values of  $\mathbf{M}$  that are changing in a dynamical way over time. If network degradation is detected, a genetic algorithm can be used to solve (3) and to determine the optimum value  $\mathbf{C}^{\text{opt}}$

of the knobs. Note that the optimization is quasi real time because the surrogate model is very fast to evaluate. In cases where the result of the optimization step was unsuccessful in predicting the QoS, the outcome of the experiment can be added as additional data to improve the model predictions.

## VI. EXPERIMENTAL EXAMPLE: PRACTICAL CASE STUDY

### A. Setup configuration

The practical case study is conducted in a pseudo-shielded testbed environment w-iLab.t [17] in Ghent, Belgium. The nodes in the testbed are mounted in an open room (66 m x 20.5 m) in a grid configuration with an x-separation of 6 m and a y-separation of 3.6 m. Figure 2 shows a partial ground plan of the test lab with an indication of the location of the nodes. Each node has two Wi-Fi interfaces. Furthermore, a ZigBee sensor node and a USB 2.0 Bluetooth interface are incorporated into each node.

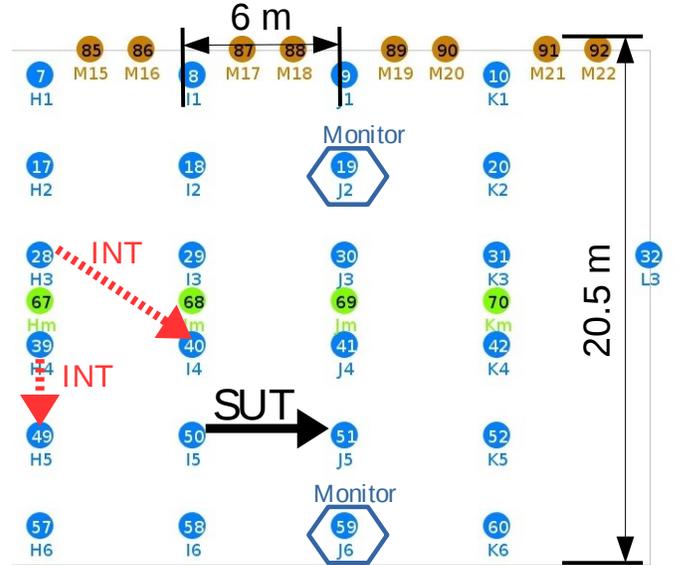


Fig. 2. Partial layout of the W-iLab.t living lab test environment (66m x 20.5m) with indication of the nodesRSSI.

The setup that was considered is a typical scenario of two IEEE 802.11 standard compliant nodes, operating on infrastructure mode with 802.11g standard and generating traffic on up-link (node 50 to 51, see Figure 2). We refer to these two nodes as the System Under Test (SUT). The experiments are performed in a quasi-ideal shielded environment, where the effect of external interference or environmental conditions is not dominantly present. For each experiment that is configured, the sender node will directly transmit iperf [18] UDP (User Datagram Protocol) data to a receiver node over a period of 30 seconds. At the same time, two external collocated pair of IEEE802.11 compliant interference links start generating iperf traffic. We refer to these links as the interference generating group, abbreviated as INT. The effect of these external interference sources on the QoS performance of the wireless system is highly unpredictable in nature, so this experiment can be seen as an analogy to real-life operation of the system. Setting 30 seconds as the duration of time steps is motivated by the

TABLE I. OVERVIEW OF PARAMETER LIST CONSIDERED FOR THE CASE STUDY.

Parameter type	Name	Values	Unit
Meter	COD of channel 1	[0-100]	%
Meter	COD of channel 6	[0-100]	%
Knob	Channel of SUT	{1,6}	-

fact that based on our experience, obtaining reliable throughput values in presence of a fixed interference setup requires letting the iperf produce its stream for so long that the competition for spectrum utilization between the interference link and the SUT converge to equilibrium.

In this scenario we use the proposed CDE to optimize the throughput of the SUT which is an important QoS parameter for many wireless networks. We assume that INT and SUT operate on the same or non-overlapping IEEE 802.11g channels [19]. In this condition, the dominant external parameters influencing the throughput of the SUT is the traffic characterization of the homogeneous collocated interference links. As authors in [7] suggest, the most crucial parameters of the interference, namely channel occupancy degree (COD) and transmission rate of the interference could be practically measured by means of packet sniffers and looking up the radio tap header information of sniffed packets. Two dedicated monitoring agents are incorporated that are basically Wi-Fi interfaces operating on IEEE 802.11 monitor mode [19] on every of the channels in the vicinity of the SUT (see Fig. 2).

For simplicity, we only consider two non-overlapping IEEE 802.11g channels. Thus, the meters measured in this scenario are the COD values of IEEE 802.11g channels 1 and 6, the knob is the SUT channel of operation (1 or 6). The INT links (node 28 to 40 and node 39 to 49 in Figure 2) operate on channel 1 and 6 with interference CODs according to each experiment configuration. In order to create a dynamic environment, the COD of the INT links is varied after a certain number of time steps during learning. As such, this creates an interference pattern that is changing over time and it forces the CDE to react and switch between “optimal” operating points. The layout of the testbed for this scenario is visualized in Fig. 2, and Table I lists the parameters that are considered during the cognitive decision making. In addition to the knobs, also some meters must be derived to assess the current state of the environment. To this end, CODs of the channel of INT can be derived from the REM. This meter provides additional information to the CDE about the interference pattern. Hence, the decision engine should monitor the current value of the meters and optimizes the channel of SUT, in such a way that the SUT throughput is maximized. As the meter changes over time, the CDE adjusts the optimal knob (SUT channel) instantaneously. Since a real-time response of the CDE is desired, the engine is subjected to the model building phase where it learns the behavior of the wireless system.

### B. Building the surrogate model

In the model building phase, a limited set of experiments are performed to learn the resulting throughput over the SUT link as a function of 3 parameters (interference COD on channels 1 and 6, SUT operating channel). The meters change in an uncontrollable way. However, the values of the knob are chosen by the CDE such that the parameter space is well covered,

i.e. a balanced trade-off is made between exploitation (trying configurations that are most likely to give good throughput performance) and exploration (trying configurations that are different from what that the algorithm has tried before) of the parameter space. The settings of the 87 experiments that were performed during this learning phase of the algorithm are shown in Figure 3 where experiments were carried out to cover parameter combinations on both channels and for different pairs of interference CODs on channel 1 and 6.

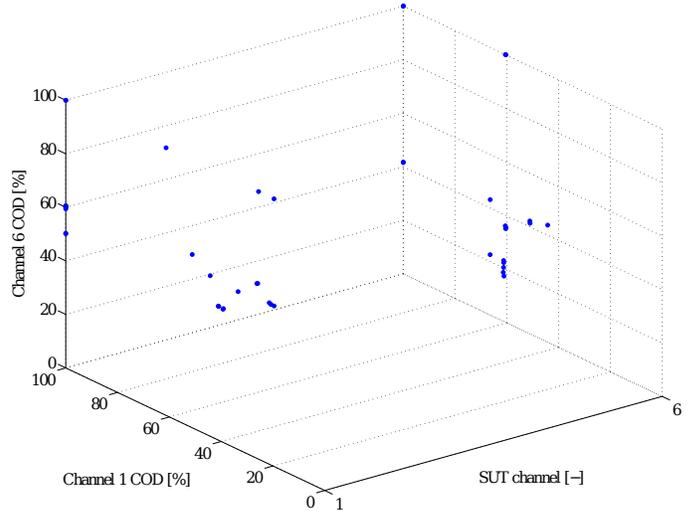


Fig. 3. Exploration space used for the learning phase of the algorithm in practical example.

Using these selected samples, an interpolation model is generated by the CDE so as to find a surrogate model for the throughput over the entire design space. The model is expected to become more accurate as additional experiments are performed. However the number of experiments that can be performed is limited by the amount of time that is available for the experimentation. In order to better quantify the accuracy of the surrogate model, the predicted throughput is compared to the actual throughput that is obtained after executing 15 validation experiments, shown in Figure 4. In the ideal case, all data samples (marked by a red dot) are located close to the linear solid line which indicates a perfect agreement. Taking into account stochastic variability of the testbed, it is found that most experiments lie with a 3 Mbps boundary (marked by 2 linear dashed lines), that corresponds closely to the model accuracy threshold  $\tau$  as chosen in Equation 4. As seen in Figure 4, only a couple of sample points are out of the accuracy bound.

### C. Model deployment

Finally the performance of the CDE was assessed by comparing the obtained throughput in 15 experiments with different configurations. Once with an static operation of the SUT and second time with utilization of the CDE to steer the SUT channel. The results are shown in Figure 5 where for each time step, its corresponding meter and knob values are also indicated. From time step 1 to 6, channel 6 was the optimum channel of operation for the SUT since as seen in Figure 5 staying on channel 1 caused a huge performance degradation for the case where the CDE was not deployed. The CDE

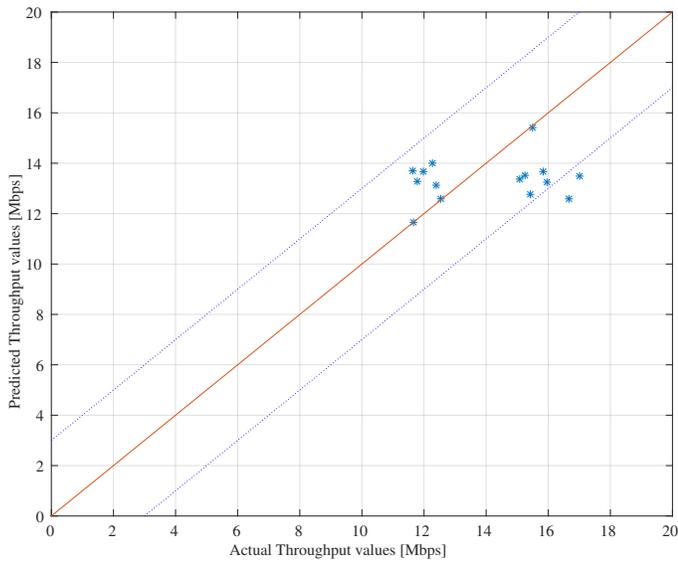


Fig. 4. Actual QoS versus predicted QoS for 15 experiments.

switched the SUT channel at time steps 6 when according to its internal surrogate model, a better throughput was predicted and this decision works out also in reality as evidenced also by observing the COD values of the channels. Figure 5 also reveals the performance improvement of the CDE at different time steps when different interference configurations are in place, e.g. at the worst case condition (time step 1) more than 100% performance improvement was achieved.

Concerning the limitation of the proposed approach, firstly, the algorithm was applied in a pseudo-shielded all line of sight single hop environment where the measurements were performed. The hidden node problem is not taken into consideration since it is not an issue in this environment, i.e. the surrogate model does not consider interference of nodes that are not in the radio range of the monitors. The monitors are assumed to capture all present traffic without dropping any packets. If for any reason the interference is not perceived by the sniffers, an event detector as proposed in [7] could assist by detecting the unpredicted throughput degradation in the network. Secondly, deploying both the CDE and the REM imposes data and processing overhead on the network. Spectrum monitoring data overhead could be formulated in two cases depending on the mobility of spectrum sensing devices. Excluding mobile devices from spectrum monitoring and only depending on the fixed Wi-Fi sniffers does not introduce any data overhead to the wireless network.

## VII. CONCLUSION

A novel approach for decision making is presented in the context of cognitive ISM band WLANs where all users have equal regulatory status in terms of spectrum utilization. The method incorporates surrogate models for predicting the performance of the network by monitoring environmental parameters and transmission parameters. It improves the models by following the accuracy of its predictions in real time. The generic design of the algorithm is beneficial for different scenarios with different set of parameters. As proof of concept, a use case was experimentally investigated to verify

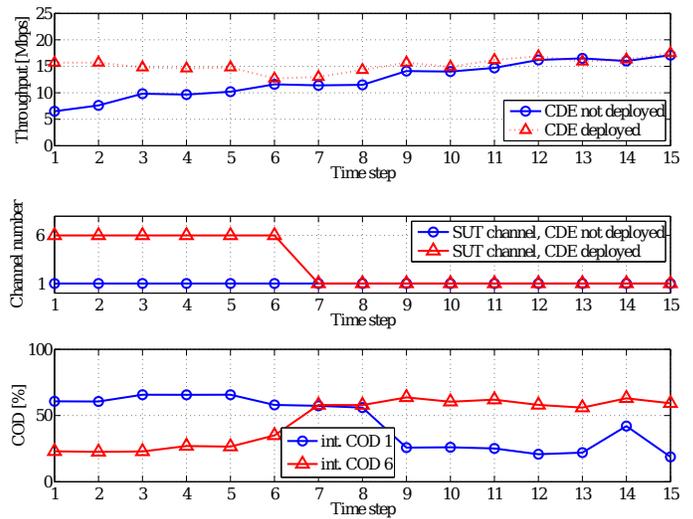


Fig. 5. (a) Throughput performance of cognitive vs. static configuration, (b) Channel of INT. Each time step is 30s.

the efficiency of this method in optimizing the performance of wireless networks. This was shown by comparing the throughput of the link when its channel of operation was selected by the CDE with a case when the channel was fixed. A worst case improvement of more than 100% was observed.

Large scale implementation of the algorithm in different environments such as home, office, industry, etc. would be the topic of future research. Finding the most appropriate set of meters and knobs in different scenarios on the one hand, and, coupling the model-based predictions with heuristic decision making methods on the other hand, would lead to useful endeavors aiming at improving the current algorithm.

## ACKNOWLEDGMENTS

The research activities that have been described in this paper were funded by Ghent University, iMinds, the Fund for Scientific Research in Flanders (FWO-V) and the Interuniversity Attraction Poles Programme BESTCOM initiated by the Belgian Science Policy Office. This paper is also the result of research carried out as part of the QoCON project funded by iMinds. QoCON is being carried out by a consortium of the industrial partners: Televic, Option and Barco in cooperation with iMinds research groups: IBCN (UGent), WiCa (UGent), SMIT (VUB), PATS (UA) and IMEC. D. Deschrijver is a post-doctoral research fellow of FWO-V.

## REFERENCES

- [1] A. He, K. K. Bae, T. R. Newman, J. Gaeddert, K. Kim, R. Menon, L. Morales-Tirado, J. J. Neel, Y. Zhao, J. H. Reed, *et al.*, "A survey of artificial intelligence for cognitive radios," *Vehicular Technology, IEEE Transactions on*, vol. 59, no. 4, pp. 1578–1592, 2010.
- [2] B. G. Buchanan, E. H. Shortliffe, *et al.*, *Rule-based expert systems*, vol. 3. Addison-Wesley Reading, MA, 1984.
- [3] D. B. Leake, "Case-based reasoning," *The knowledge engineering review*, vol. 9, no. 01, pp. 61–64, 1994.

- [4] Y. Zhao, J. Gaeddert, L. Morales, K. Bae, J.-S. Um, and J. Reed, "Development of radio environment map enabled case- and knowledge-based learning algorithms for ieee 802.22 wran cognitive engines," in *Cognitive Radio Oriented Wireless Networks and Communications, 2007. CrownCom 2007. 2nd International Conference on*, pp. 44–49, Aug 2007.
- [5] A. He, J. Gaeddert, K. K. Bae, T. R. Newman, J. H. Reed, L. Morales, and C.-H. Park, "Development of a case-based reasoning cognitive engine for ieee 802.22 wran applications," *SIGMOBILE Mob. Comput. Commun. Rev.*, vol. 13, pp. 37–48, Sept. 2009.
- [6] J. D. Ser, M. Matinmikko, S. Gil-Lpez, and M. Mustonen, "Centralized and distributed spectrum channel assignment in cognitive wireless networks: A harmony search approach," *Applied Soft Computing*, vol. 12, no. 2, pp. 921 – 930, 2012.
- [7] M. Pakparvar, D. Plets, E. Tanghe, D. Deschrijver, W. Liu, K. Chemmangat, I. Moerman, T. Dhaene, L. Martens, and W. Joseph, "A cognitive qos management framework for wlangs," *EURASIP Journal on Wireless Communications and Networking*, 2014.
- [8] Y. Zhao, *Enabling cognitive radios through radio environment maps*. PhD thesis, Virginia Polytechnic Institute and State University, 2007.
- [9] J. Sacks, W. J. Welch, T. J. Mitchell, and H. P. Wynn, "Design and analysis of computer experiments," *Statistical science*, vol. 4, no. 4, pp. 409–423, 1989.
- [10] S. S. Haykin, *Neural networks: a comprehensive foundation*. Prentice Hall Englewood Cliffs, NJ, 2007.
- [11] M. Hasegawa, H. N. Tran, G. Miyamoto, Y. Murata, and S. Kato, "Distributed optimization based on neurodynamics for cognitive wireless clouds," in *IEEE 18th International Symposium on Personal, Indoor and Mobile Radio Communications, 2007. PIMRC 2007.*, pp. 1–5, IEEE, 2007.
- [12] M. D. Buhmann, *Radial basis functions: theory and implementations*, vol. 12. Cambridge university press, 2003.
- [13] E. W. Cheney, *Multivariate approximation theory: Selected topics*. SIAM, 1986.
- [14] C. B. Barber, D. P. Dobkin, and H. Huhdanpaa, "The quickhull algorithm for convex hulls," *ACM Transactions on Mathematical Software (TOMS)*, vol. 22, no. 4, pp. 469–483, 1996.
- [15] D. Shepard, "A two-dimensional interpolation function for irregularly-spaced data," in *Proceedings of the 1968 23rd ACM national conference*, pp. 517–524, ACM, 1968.
- [16] G. Allasia, "Simultaneous interpolation and approximation by a class of multivariate positive operators," *Numerical Algorithms*, vol. 34, no. 2-4, pp. 147–158, 2003.
- [17] S. B. et al, "Federating wired and wireless test facilities through emulab and omf: the ilab.t use case" in the proceedings of tridentcom 2012," in *Proceedings of TridentCom*, 2012.
- [18] "iperf - the TCP/UDP bandwidth measurement tool." <http://iperf.fr/>. [Online; accessed 2013-07-19].
- [19] "IEEE standard for information technologyTelecommunications and information exchange between systems local and metropolitan area networksSpecific requirements part 11," *IEEE Std 802.11-2012 (Revision of IEEE Std 802.11-2007)*, pp. 1 –2793, 2012.