Enhancing Environment Perception for Cooperative Power Control: an Experimental Perspective

Panagiotis Spapis, George Katsikas, Konstantinos Chatzikokolakis, Roi Arapoglou, Makis Stamatelatos, and Nancy Alonistioti
Department of Informatics and Telecommunications
National and Kapodistrian University of Athens
Athens, Greece
e-mail:{pspapis, katsikas, kchatzi, k.arapoglou, makiss, nancy} @di.uoa.gr

Abstract - **Short range communications in dense residential environments enable anytime high data rate connectivity, however also pose new challenges regarding the efficient operation of network devices, related to their co-existence. These challenges mainly concern capacity requirements on the one hand and the interference effect that each device creates to its neighboring ones on the other. This paper presents a cooperative distributed algorithm for power control and interference mitigation based on ad-hoc communication of networking devices. The algorithm also incorporates learning capabilities for strengthening the situation perception of each network element. Both versions of the algorithm, the core cooperative power control, and the learning enhanced one, have been deployed in WiFi Access Points and tested in an office environment in order to showcase their applicability. The experimental results prove that the incorporation of the presented algorithms leads to significant gains both in the energy consumption and the interference mitigation at the same time.**

Keywords – co-existence; cooperative power control; interference mitigation; learning; data mining; fuzzy logic.

I. INTRODUCTION

The acute proliferation of wireless networking devices enables "anytime" and "anywhere" communications. This trend, coupled with large scale deployment of heterogeneous radio access networks in short range context, (APs, pico-cells, etc.) and in dense environments, (i.e., residential areas) imposes the need for developing mechanisms addressing issues related to co-existence in an efficient way. The capacity and energy efficiency requirements impose different constraints in the system, whereas the mentioned co-existence results in high interference levels.

In such communication environments, power control mechanisms aim at optimizing the network's capacity and coverage and at the same time at achieving interference mitigation, reducing power consumption and extending battery lifetime. The purpose is to have improved QoS for the users as well as having the optimum overall network's utility and reduced cost from the network operator's perspective. Given the two aforementioned objectives, the mechanisms should be developed following a cooperative and distributed paradigm in order to avoid selfish behaviors that lead to suboptimum solutions.

In this paper, a distributed and cooperative power control algorithm is presented and evaluated; the objective is, through power adjustment, to have an optimum tradeoff between the network elements' capacity and the interference caused to the rest of the network elements belonging in the scheme. The Cooperative Power Control (CPC) algorithm, initially described in our previous work in [1], is applicable to short-range wireless networking environments, where the network elements are able to exchange interference and power information. Moreover, the proposed solution deploys learning capabilities to the devices in order to facilitate the evaluation of the previous decisions and improve the interpretation of the environment conditions. This paper builds on the previous work and presents an extensive experiment for the validation of the proposed algorithm. More specifically, we have developed the CPC algorithm and incorporated it in WiFi APs; our solution has been used in a real life experiment, in an office network environment, which highlights the merits from its incorporation in both energy consumption and interference mitigation.

The rest of this paper is structured as follows: Section II presents proposed solutions available in the literature; Section III provides background information regarding fuzzy logic and k-Means; in Section IV, the baseline reference algorithm for cooperative power control is briefly described. Section V presents the learning-assisted algorithm, by describing the considered functionalities, the case study, and, the learning framework. Section VI describes the experimentation deployment and assumptions of the experimental analysis, whereas Section VII describes and analyses in details the experimental results. Finally, Section VIII concludes the paper.

II. RELATED WORK

The transmission power control adjustment has attracted the interest of researchers, given the benefits stemming from the introduction of power control schemes; thus several solutions have been proposed in the literature. In [2], Sun et al. propose to formulate the power control problem using a non-cooperative game; the solution converges once Nash equilibrium [3] is reached and is applicable to mobile ad-hoc networks. The strategy for the transmission power identification is related to the Shannon capacity [4] on the one hand and the energy waste due to the caused interference on the other. In [5], the authors introduce a competitive distributed and autonomous power control algorithm for cellular communication systems. This approach is mainly focused on the downlink communication but can be easily extended to take into account both downlink and uplink. The nodes set independently their Signal to Interference Ratio (SIR) targets and rely only on local information to proceed to power adjustment. The algorithm is proven to converge to the Pareto optimal solution when the system is feasible, but diverges otherwise [6]. In [7], a cooperative game-theoretic mechanism for optimizing power control is also proposed. In this solution, issues such as network efficiency and user fairness are taken into account in order to optimize a SINR-based utility function.

The afore-described solutions are generic and focus on the transmission power control problem in general. However, specific solutions have been proposed in the literature, trying to tackle the transmission power control problem in WiFi networks. In [8], Mhatre et al. propose a power control algorithm that tries to mitigate interference in 802.11 wireless network environments, by providing a starvation-free transmission scheme based on the assignment of higher transmission power to cells that are more heavily loaded (i.e., the cells that have higher number of clients or clients with poor quality channel). This solution can be implemented in a centralized or a distributed manner; in case of the former the authors use a sampler in order to compute the optimum power vector of the AP topology by avoiding extensive signaling. However, misbehavior may occur in case power-vectors of high probability exist, as the algorithm fails to search other possible vectors. In [9], a synchronous rate and power control system implemented in IEEE 802.11 AP is introduced. Such solution provides per-link power control without adaptations or modifications to the underlying 802.11 MAC protocol, following an approach with two synchronized phases. In the former, an initial power level is identified so as to achieve admirable link performance whereas in the latter, further enhancements in the data rate and the transmission power level, based on the packet delivery rate are considered for avoiding performance degradation. The main disadvantage of this solution is the use of greedy schemes for power level allocation that cannot provide a maximum network throughput. In [10], Kowalik et al. propose the introduction of ConTCP, a power adaptation scheme that takes into consideration the links' quality. Specifically, a reference node tries to calculate the approvable power level of each incoming wireless link, based on QoS level thresholds, and informs the AP for the selected power levels; the proposed scheme tends to perform well under specific network topologies, where simultaneous transmissions occur. In [11], the authors propose a power control method which discovers the required data-rate link within the transmission range through adjusting the transmission power to corresponding levels by recursively sensing the environment; this topology information is also used for the selection of the optimal route, in case of 802.11b WiFi mesh networks. In [12], ElBatt et al. propose a power management scheme for wireless ad-hoc networks with low mobility patterns; the classical shortest path routing algorithm coupled with the identification of the optimum transmission power level is used. This approach results to small clusters of ad-hoc nodes. However, even though the cluster-based interference is reduced, retransmission of packets and increase in the whole network interference is unavoidable.

In terms of this paper, we apply a solution described in [1], aiming at power control in WiFi networks in a distributed cooperative manner. Our solution is based on and extends a cooperative power control scheme for wireless sensors [13] [14]. In the proposed approach, the CPC algorithm is applied in 802.11 WiFi networks and is also enhanced by introducing a learning scheme to strengthen the situation perception capabilities of each network element. The solution is based on a hybrid model which exploits the merits of fuzzy logic and data clustering. Compared to the rest of the afore presented solutions, the proposed and implemented one aims at maximizing network utility, which is being captured by the Shannon capacity and the interference caused to the neighboring APs. Thus, the benefit is in the overall network utility which also benefits the SINR in every node. Furthermore, given the fact that we use an adaptation mechanism for enhancing the situation perception of each network element, we ensure that the APs' configuration will be the most suitable for the context where there are placed.

III. BACKGROUND

This section provides the background for the proposed solution. The baseline algorithm is based on an objective function which uses fuzzy logic for the calculation of the weights of the parts of the equation. The learning algorithm uses the k-Means data mining technique for the adaptation of the fuzzy logic controllers. The rest of this section presents the principles of fuzzy logic and k-Means so as to create a standalone document.

A. *Fuzzy logic*

Fuzzy logic is an ideal tool for dealing with complex multi-variable problems; the nature of the decision making mechanism makes it very suitable for problems with often contradictive inputs. A fuzzy reasoner (Fig. 1) consists of three parts, namely:

- The fuzzifier, which undertakes to transform the input values (crisp values) to a degree that these inputs belong to a specific state (e.g., low, medium, high, etc.) using the input membership functions.
- The inference part, which correlates the inputs and the outputs using simple "IF...THEN..." rules. Each rule results to a specific degree of certainty for each output; these degrees then are being aggregated.

- The defuzzifier, where the outcome of the abovementioned aggregation is being mapped to the degree of a specific state that the decision maker belongs to. Several defuzzification methods exist; the most popular is the centroid one, which returns the center of gravity of the degrees of the outputs, taking into account all the rules, and is calculated using the following mathematical formula:

$$u_{COG} = \frac{\int u_i \mu_F(u_i)du}{\int \mu_F(u_i)du} \quad (1)$$

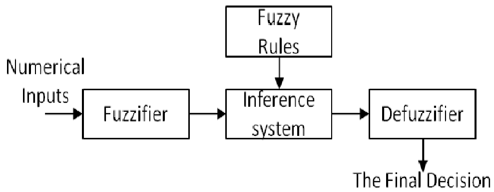


Figure 1: High level view of a fuzzy inference system

B. k-Means

k-Means is a well known data-mining clustering technique. The core idea of data clustering is to partition a set of N, d-dimensional, observations into such groups that intra-group observations exhibit minimum distances from each other (Fig. 2), while inter-group distances are maximized. k-Means [15] is based on the following objective function:

$$J = \sum_{i=1}^{c} J_i = \sum_{i=1}^{c}\left(\sum_{k, x_k \in G_i} \left\| x_k - c_i \right\|\right) \quad (2)$$

Where

- c: the number of clusters,
- G_i: the i^{th} group,
- x_k: the k^{th} vector in group J_i and represents the Euclidean distance between x_k and the cluster center c_i.

The partitioned groups are defined by using a membership matrix described by the variable U. Each element U_{ij} of this matrix equals to 1 if the specific j^{th} data point x_j belongs to cluster i, and 0 otherwise. The element U_{ij} is analyzed as follows:

$$U_{ij} = \begin{cases} 1, if \left\| x_j - c_j \right\|^2 \le \left\| x_j - c_k \right\|^2, \forall k \ne i \\ 0, otherwise \end{cases} \quad (3)$$

This means that x_j belongs to group i, if c_i is the closest of all centers.

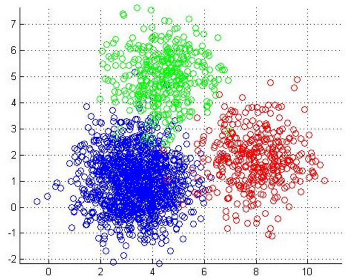


Figure 2: Visualization of k-Means clustering for three clusters

IV. COOPERATIVE POWER CONTROL- BASELINE ALGORITHM

The proposed CPC algorithm is based on [13] and [14]; both approaches propose a scheme for distributed interference compensation in Cognitive Radio that operates in license exempt spectrum bands, using transmission power adjustment methodologies. The solution concerns ad-hoc networks and is based on an information exchange scheme for the identification of the appropriate transmission power levels. Each independent node of the topology sets its power by considering individual information, as well as information related to the neighboring nodes. More specifically, a node sets its power level by considering its Signal to Interference plus Noise Ratio (SINR) and the interference caused to its neighbors. The main idea of this approach is to prevent users to operate in the maximum transmission power levels.

The authors assume a set of node pairs L that operate in the same frequency. The SINR for the i^{th} pair is given below [13]:

$$\gamma_i(p_i^k) = \frac{p_i^k \cdot h_{ii}}{n_o + \sum_{j \neq i} p_j^k \cdot h_{ji}} \quad (4)$$

Where

- p^k_i: transmission power for user i on channel k
- h_{ii}: link gain between i^{th} receiver and i^{th} transmitter
- n_o : noise level (equals to 10^{-2})
- p^k_j: transmission power for all other users on channel k, assuming that $j \in \{1,2,\ldots,L\}$ and $j \neq i$
- h_{ji}: link gain between i^{th} receiver and j^{th} transmitter

It is also assumed that the channel is flat-faded without shadowing effects. Since the channel is static, the only identified attenuation is the path loss h (channel attenuation or channel gain). Given that indoor urban environments are considered, the channel gain is $h_{ji} = d_{ji}^{-3}$, where d is the distance between the j^{th} transmitter and the i^{th} receiver.

The decision for the transmission power levels takes into account the negative impact (i.e., interference) of a node to its neighboring nodes. This is formalized using (5), which captures the notion of interference price; such price reflects the interference a user causes to other users within its transmission range and is given by:

$$\pi_i^k = \frac{\partial u_i(\gamma_i(p_i^k))}{\partial(\sum_{j \neq i} p_j^k \cdot h_{ji})} \tag{5}$$

Where

- $u_i(\gamma_i(p^k_i)=\theta_i log(\gamma_i(p^k_i))$: logarithmic utility function,
- θ_i: user dependent parameter.

Both of the algorithms presented in [13] and [14] are based on a tradeoff between the capacity of a user and the interference caused to the corresponding neighborhood. This balance is being captured by the following objective function:

$$u_i(\gamma_i(p_i^k)) - \alpha \cdot p_i^k \sum_{j \neq i} \pi_j^k \cdot h_{ji} \tag{6}$$

The first part indicates a relation to the Shannon capacity for the corresponding user, while the second part captures the negative impact in terms of interference prices that a user causes to its neighborhood. The *a* factor is introduced so as to capture uncertainties in the network; these uncertainties reflect the precision of the received and compiled information of each network element regarding the interference price which should have been available by the node's neighbors. This is related to the fact that once a network element adjusts its transmission power, it informs its neighbors in an ad-hoc manner. This implies that even though a network element has collected information from all of its neighbors in order to adjust its transmission, the gathered data could be obsolete and, as a consequence, they will not capture neighborhood's current state. The obsolescence of the interference prices is related to the update interval (i.e., the periodic update) of each network element. In [13], α is set in a static manner as 25%. In [14], a fuzzy reasoner is introduced in order to identify, in a more dynamic way, uncertainties in the network based on the network's status; the inputs (number of users, mobility, update interval) of the fuzzy reasoner capture the volatile nature of the ad-hoc network, whereas the output of the fuzzy reasoner is the *Interference Weight*. The *a* factor is defined as 1/β *Interference Weight* + 1 (β has the maximum value of the *Interference Weight*).

The algorithm consists of three steps, namely, the initialization, the power update and the interference price update. The former is related to the assignment of initial valid transmission power and interference price values. The second part concerns the transmission power update based on the interference prices each node receives from its neighbors. Finally, the interference price update captures the communication of its interference prices to the neighborhood, by every network node. The second and the third steps are asynchronously repeated until the algorithm reaches a steady state (i.e., a state where every network element has the same transmission power for two consecutive time iterations).

The main deficiency of the afore-described scheme is related to the static perception of the environment (i.e., *a* factor that captures the network's dynamics). Even in the case where the fuzzy reasoner is used for capturing the uncertainties in the network, the environment interpretation model (i.e., membership functions of the fuzzy reasoner) is static. More specifically, in the latter case, the environment interpretation is based on expert's knowledge and is induced to the network elements by its input membership functions. This implies that all network elements with the same configuration have the same situation perception as well. Moreover, it would be a major benefit for the network administrators to enable network elements to evolve the way they interpret their environment; this could be achieved by changing the shape of the input membership functions. In order to tackle the static definition of the situation perception, we propose a feedback-based learning scheme that evaluates how the network performed after a transmission power adjustment, in terms of the interference prices.

V. Learning Enhanced Cooperative Power Control Framework

In our previous work in [1], we have proposed the application of the algorithms introduced in [13] and [14] in a completely new application area, that of WiFi Access points; the cooperative power control among the network elements is the objective of this algorithm in order to maximize the network's utility. More specifically, we suggest that the WiFi APs should cooperate in order to minimize the caused interference, by adjusting their transmission power and at the same time having the optimum transmission power based on the Shannon capacity.

In terms of this paper the learning enhanced CPC is being presented and evaluated in a real life experiment. In this section the functionalities that should be incorporated in the CPC enabled network elements (i.e., in this case WiFi APs) are described. Then, the case under investigation, where the modified CPC is applied is being presented along with the learning algorithm used for the adaptation of the situation perception of the CPC.

A. Functional Architecture

In order to deploy the CPC in the considered environment, network elements should be enhanced with a set of software modules namely "Power Control", the "Learning", the "Memory", the "CPC communication", the "Control Engine" and the "Monitoring". Fig. 3 presents the functional architecture of the software implementation of the CPC.

Each software module provides a set of functionalities in order to enable the instantiation of the CPC in WiFi APs; more specifically:

- The "Power Control" incorporates the functionalities for the calculation of the metrics (interference prices) and the objective function that each network element has to maximize. Furthermore, this part of the mechanism implements the fuzzy logic reasoner for the calculation of the *Interference Weight* and the α factor,

- The "Learning" part incorporates the learning mechanism for enhancing the network element's situation perception.
- The "Memory" contains all the information required for the CPC; this information may be local and related to the AP under consideration (ex. TxPower, SINR, local IPs and MACs, etc.), or related to neighboring network APs (physical topology information – distances from neighbors, network information – neighbors' IPs and MACs, algorithm information – neighbors' interference prices and TxPowers).
- The "CPC communication" software module consists of two parts, the client and the server. As mentioned afore, the basis of the CPC scheme is related to the asynchronous information exchange among the network elements. This implies that each network element operates as a server, where the neighboring WiFi APs are being associated and also as a client in order to associate to the neighboring APs.
- The "Control Engine" is responsible for the enforcement of the re-configuration action, which in the considered case is the TxPower adjustment.
- The "Monitoring" software module is responsible for the two types of monitoring tasks, the local and the neighborhood/cluster. The former is related to monitoring of local metrics and measurements (e.g., identification of local TxPower, associated users, sensed APs, etc.) whereas the latter is related to cluster information (e.g. MACs and IPs of neighboring APs, physical topology graph, etc.).

The afore-described software comprises the CPC application that has been deployed in every CPC-enabled network element.

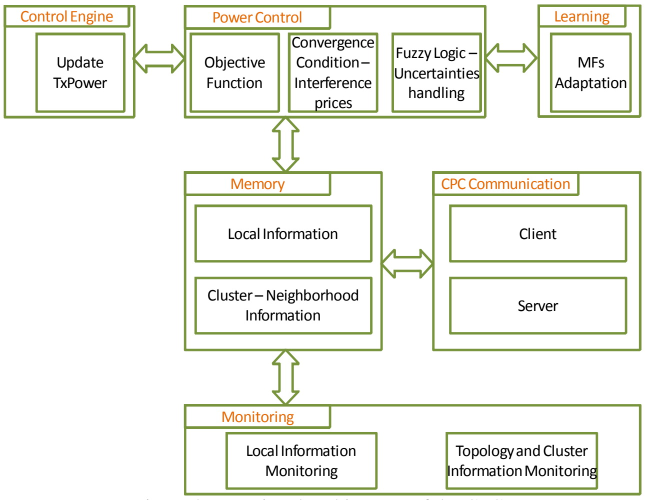


Figure 3: Functional architecture of the CPC

B. Case Study

In the case study under investigation, we assume the presence of several WiFi APs located in the considered area. These APs communicate via wireless links in order to exchange their interference values. Based on these values each network element adjusts its transmission power (Fig. 4).

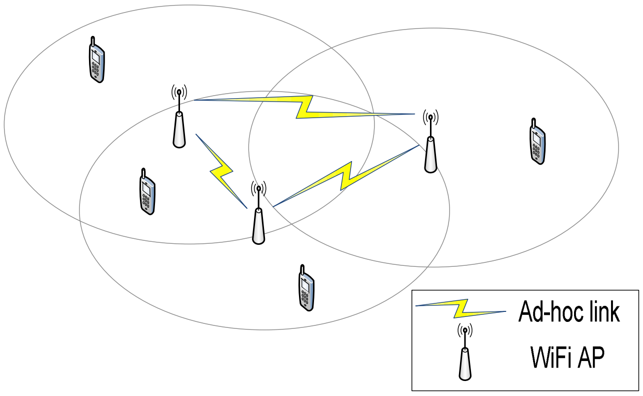


Figure 4: Envisaged network topology

Given the assumption that the APs communicate asynchronously and each one might have its locally-set update period, it is possible that the APs are unaware of the current network's status (from the messages exchange). Such problem becomes even more acute if we consider that the network elements might lose some messages during the messages exchange procedure due to the nature of the applied information fusion scheme and the sensitivity of the wireless medium. This implies that the use of the fuzzy reasoner is imperative for capturing the uncertainties [14]. The WiFi application area though, poses the need for modification of the inputs and the inference engine of the fuzzy logic controller. Thus, the number of the WiFi APs in the vicinity, the number of users in the vicinity (associated to WiFi APs) and the update interval are used as inputs of the fuzzy reasoner. In case of completely new application areas, new/modified fuzzy reasoners could be incorporated so as to be more suitable to the use case under discussion. The way a network element perceives its environment is based on the input and output membership functions. As in [14], the inputs' membership functions initially are set to have triangular shape, mainly in order to capture the strict nature of the inputs.

Table I provides the rules of the inference engine of the fuzzy reasoner. The most crucial input for the decision making process is the update interval. This input depicts the frequency of the information updates about the interference price of a network element to its neighbors thus capturing how recent is the view of a network element, based on the inputs from its neighbors. These inputs will be used for the calculation of the TxPower (Section IV).

TABLE I. RULES OF THE FUZZY REASONER

Rule Number	Num of WiFi Aps	Num of Users	Update Interval	Interference price
1	Low	Low	Low	Low
2	Low	Low	Medium	Low
3	Low	Low	High	Medium
4	Low	Medium	Low	Low
5	Low	Medium	Medium	Medium
6	Low	Medium	High	Medium
7	Low	High	Low	Medium
8	Low	High	Medium	Medium
9	Low	High	High	High
10	Medium	Low	Low	Low
11	Medium	Low	Medium	Medium
12	Medium	Low	High	High
13	Medium	Medium	Low	Medium
14	Medium	Medium	Medium	Medium
15	Medium	Medium	High	High
16	Medium	High	Low	Medium
17	Medium	High	Medium	Medium
18	Medium	High	High	High
19	High	Low	Low	Medium
20	High	Low	Medium	Medium
21	High	Low	High	High
22	High	Medium	Low	Medium
23	High	Medium	Medium	Medium
24	High	Medium	High	High
25	High	High	Low	Medium
26	High	High	Medium	High
27	High	High	High	High

As briefly described in Section IV, the CPC consists of two separate iterative procedures, the power update and the interference price update. In the former, consider a network element i, which updates its transmission power using a time interval $t_{ai} \in T_{ai}$, where T_{ai} is a set of positive time instances in which the AP i will update its transmission power level and $t_{a1} \neq t_{a2} \neq \cdots \neq t_{ai}$. Similarly, each WiFi AP *i* has an interference price update interval $t_{bi} \in T_{bi}$, where it updates its interference price and announces the updated interference price π_i^k to the rest of the WiFi APs belonging in the scheme. Fig. 5 provides the messages exchange and the operations' sequence on a scheme with two WiFi APs; this could be generalized for more APs as well.

C. *Learning Algorithm*

The proposed learning algorithm consists of three parts, namely, the monitoring/labeling, the classification and the adaptation of the fuzzy reasoner. Each network element that is part of the network monitors its own environment. Every time that the network elements collaboratively proceed in transmission power adjustment, their interference prices are being compared to the previous ones and the interference factor calculations are being labeled as:

- Beneficiaries: for the decisions that led to reduction of the interference value caused to the neighboring network elements,
- Neutral: for the decisions that led to similar interference values. In such cases the decision could not be characterized either as correct or wrong,
- Non Beneficiaries: the decision led to an increase of the interference value caused to the neighboring network elements.

More specifically, periodically, the network elements cooperatively identify the optimum transmission power using the methodology described in Section IV; the iterative procedure requires finite number of steps (i.e., maximum 30 iterations). Before every periodic transmission power adjustment, the interference value is being compared to the value before the last transmission power adjustment (Fig. 6).

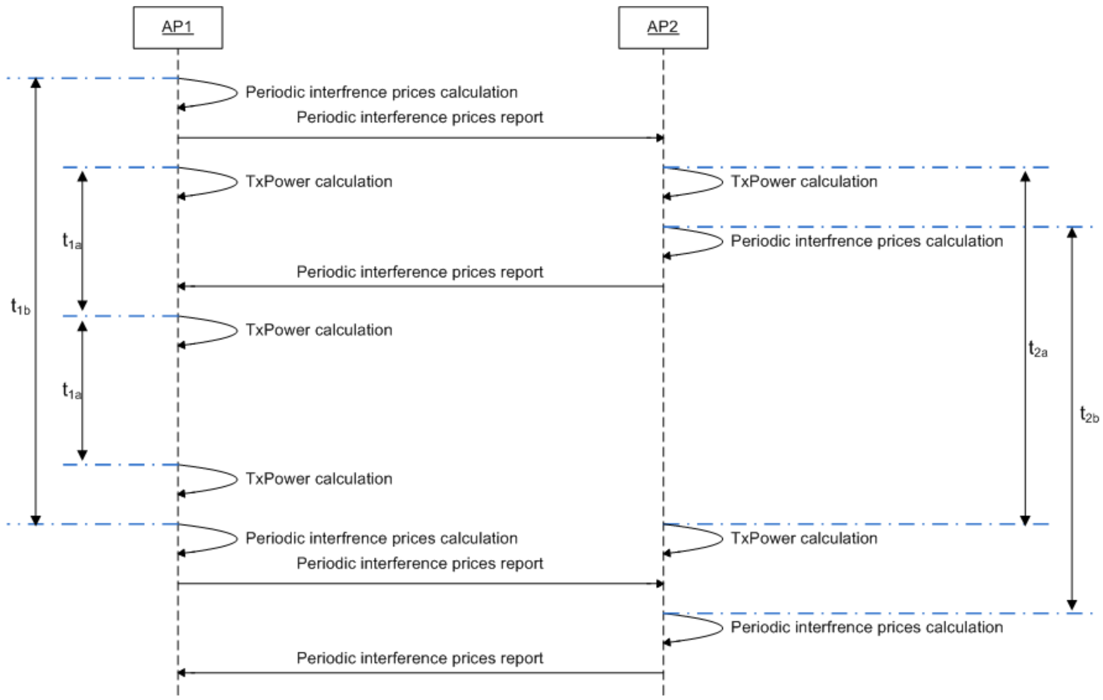


Figure 5: Message sequence chart for two WiFi APs

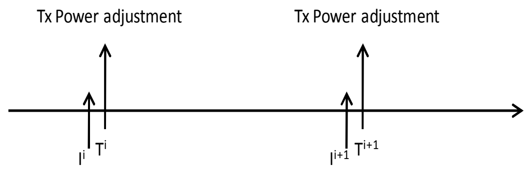


Figure 6: Timeline for Interference calculation and transmission per adjustment

The input vector $\vec{Z}_i$ (i.e., num of WiFi APs, num of users, update interval) of each network element is being evaluated against a predefined fuzzy inference system and results to an a value which, in conjunction to the interference prices, is used for the calculation of the optimum transmission power. Comparing the interference prices just before the initiation of the i^{th} transmission power adjustment and the $(i+1)^{th}$ we label the decision accordingly (i.e., Y_i is beneficiary, neutral or non beneficiary). The comparison is done using the Euclidian distance metric. This procedure results to a set (S) of labeled decisions which have been correctly labeled (at a great level of certainty) through the afore-described phase. Table II presents the key points of monitoring/labeling part of the developed algorithm.

TABLE II. MONITORING/LABELING ALGORITHM

Input:	**Approximation Parameter ε, Sample Size N**
Output:	**Set of observations S**
1.	$S \leftarrow O$
2.	$i=0$
3.	while true
4.1	i++
4.2	Retrieve vector $\vec{Z}_i$ and $\vec{IP}_i$
4.3	$\alpha_i \leftarrow$ fuzzy logic ({# WiFi APs, # Users, Update Interval})
4.4	Calculate TxPower
4.5	Wait for $\vec{Z}_{i+1}$ and $\vec{IP}_{i+1}$
4.6	Calculate I^{factor}_{i+1}
4.7	If ($\lvert I^{factor}_{i} - I^{factor}_{i+1}\rvert < \varepsilon) \rightarrow Y_i = Neutral$ Else ($\lvert I^{factor}_{i} - I^{factor}_{i+1}\rvert > \varepsilon$) and ($I^{factor}_{i} - I^{factor}_{i+1} > 0) \rightarrow Y_i = Beneficiary$ Else ($\lvert I^{factor}_{i} - I^{factor}_{i+1}\rvert > \varepsilon$) and ($I^{factor}_{i} - I^{factor}_{i+1} < 0) \rightarrow Y_i = Non\ Beneficiary$
4.8	$S \leftarrow S \cup \{\vec{Z}_{i+1}, \vec{IP}_{i+1}, Y_i\}$
5.	return S

On sequence, we formulate three clusters using the labeled data in order to exclude the misclassfied data from the previous step; the clustering is performed using k-Means Thus, each network element maintains a set of three clusters, one for classifying every decision type. By representing each cluster to a 3D grid we map each cluster to a geometrical object (i.e., sphere S_i). Each sphere is centered at $C_j=\Sigma_{i=1}^{|Ci|}S_i/|C_i|$ and has radius $R_j=max_{i=1}^{|Ci|}||CE_i-S_i||$.

For each couple of clusters i, j, the cluster centers C_i, C_j define a line ε that interconnects the two points. This line can be described by the following equation:

$$p_m = x_m + u \cdot (y_m - x_m),\ m = 1 \ldots d \tag{7}$$

Line ε intersects with spheres S_i and S_j in four points which can be retrieved by substituting the p_m values into the following hypersphere equations:

$$D_i \rightarrow \sum_{m=1}^{d} (p_m - x_m)^2 = R_i^2 \tag{8}$$

$$D_j \rightarrow \sum_{m=1}^{d} (p_m - y_m)^2 = R_j^2 \tag{9}$$

A simple way of identifying the bounds would be to extract the intersection points which belong to different hyperspheres and exhibit minimum distance from each other [16]. Then, as shown in Fig. 7, we map the identified bounds to the input membership functions of the fuzzy reasoner; this results to the modification of the environment perception of each network element.

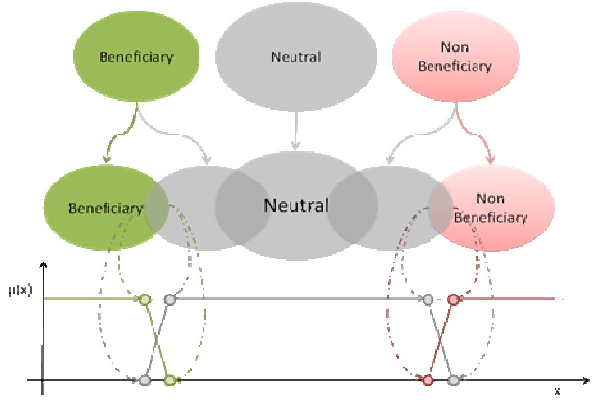


Figure 7: Clustering and bounds extraction mechanisms

VI. DEPLOYMENT

In order to experiment with the developed solution, we have proceeded in a series of real life experimentations in our premises. For this purpose we have used the proof of concept that we have implemented, which instantiates the algorithm described in Section V.

A. Environment Description

For the experimentation a set of Soekris devices has been used; such devices are low-power, low-cost, Linux-based communication computers (500MHz AMD Geode LX, 512MByte DDR-SDRAM) that act as re-programmable WiFi APs by using IEEE 802.11b/g radio access technology [17]. In all Soekris devices we consider two wireless interfaces, one is the actual AP interface and the other one is used for monitoring; the former is the AR5413 mini-PCI [18] Card whereas the latter is the WUSB54GC USB card [19]. The APs deploy their own network and route the information to the internet through NAT. APs are connected through the backbone network and communicate with a standalone machine which aggregates information and provided triggers for the initiation of algorithms. The CPC implementation is based on Java programming language using several external libraries. The most important of them are the jFuzzyLogic [20] for the "Power Control" module and Apache MINA [21] for the "CPC communication" module. For the "Monitoring" module the Linux kernel utilities are exploited.

Four Soekris devices have been placed in our premises, which suggest a typical small office environment consisting of three rooms, with 15 researchers (Fig. 8(a)). The researchers used these APs for 3 consecutive days for 10 hours each day (from 10:00 CET until 20:00 CET on July 9th 2012, where our algorithms are not installed and the measurements are used for extracting the control data, and on July 10th and 11th 2012 where our algorithms operate for the transmission power control) in order to access the internet and perform all normal, working-day, activities. Overall traffic throughout the day ranged from 1 to 10 Mbps while APs were configured to operate at 5.5Mbps throughput. The network layout is depicted in Fig. 8(b).

In all three days of our experiment, the one for the control data generation and the two where the CPC was embedded in the Soekris devices, we have attempted to procedure almost identical experimental conditions. The bandwidth requirements were reproduced – however user's mobility could not be identically reproduced.

B. Assumptions

As mentioned afore, the CPC scheme is based on the assumption that it will operate on an urban area. Thus, the generic assumptions of the algorithm should be also adapted accordingly.

The WiFi APs are placed in an indoor environment and communicate via specific communication interfaces. This implies that the distance among the network elements needs to be defined. In the proposed approach, the methodology of [22] and [23] is being followed.

The propagation obeys to certain models, from which the log-distance model is one of the most simple; the following equation describes the behaviour of such model:

$$\log d = \frac{1}{10 \cdot n}(P_{TX} - P_{RX} + G_{TX} + G_{RX} - X_a + 20\log\lambda - 20\log(4\pi)) \qquad (10)$$

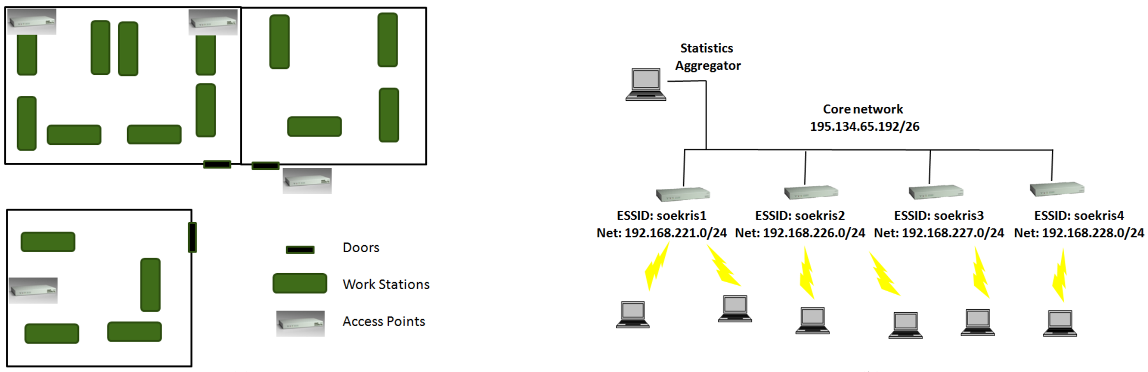
Where

- d (m): the estimated distance between the transmitter and the receiver,
- P_{TX} (dBm): the transmitted power level,
- P_{RX} (dBm) is the power level measured by the receiver,
- G_{TX} (dBi): the antenna gain of the transmitter,
- G_{RX} (dBi): the antenna gain of the receiver,
- n: measure of the influence of obstacles like partitions and ranges from 2-5 (2 for free space, 4-5 in case obstacles are considered),
- X_α: normal random variable with standard deviation of α. This variable captures the variance of the fading phenomena in an indoor environment,
- λ (m): the wavelength of the signal (for WiFi can be considered 0.12).

In the proposed experimentation, and for a typical office environment, n has been set to 5 and X_α to 20. Regarding the transmission power, which is the actual parameter of our implementation, it is related to the equipment's capabilities. Specifically, TxPower, is limited by the WiFi card's capabilities; 10dBm is the lowest price whereas 27dBm is the highest.

VII. Experimentation Analysis

For the evaluation of the CPC algorithm, we have followed an extensive experimentation scenario, in a real office environment in order to validate the applicability of the proposed solution and also to highlight the energy and network benefits from the incorporation of our algorithms. The experimentation analysis moves towards two directions, on the testing of the CPC and its applicability in the use case under consideration (i.e., a realistic WiFi office environment)



(a) (b)

Figure 8: (a) Physical topology of the experimentation environment, (b) network topology of the experimentation environment.

and on the evaluation of the learning/adaptation capabilities.

Fig. 9 and 10 capture our experimentation results for the first day, where the CPC algorithm operates without the learning part; this implies that the first day of experimentations the algorithm operates in order to gather data which will be used for the adaptation of the situation perception (i.e., first day of the experimentations). The experiment has started 10:00 CET and has finished 20:00 CET. The four Soekris APs have been placed in our testbed and we have been measuring for this period the transmission power of their WiFi cards; the transmission power ranges from 10 to 27 dBm. Fig. 9 presents the transmission power for the 10 hours of the experiment. In order to evaluate the operation of the network for several topologies, initially we have all four Soekris operating, whereas as the experiment proceeds we turn them off one by one and we leave only one operational. For each of the Soekris devices (and considering that the 10dBm is the basis of the TxPower for each AP) we see the actual gain compared to setting the transmission power to the maximum TxPower (i.e., 27 dBm). The energy gain at each of APs 1, 2, 3, and 4 is 12.51%, 10.75%, 33.33% and 21.23% respectively. Also, it is obvious that the more the APs, the more energy gains we have, due to the collaborative nature of the algorithm. Also, what should be noticed is the fact that the APs change very often their TxPower levels. This is related to the highly volatile office environment, with moving users and the many interference sources (i.e., moving users, cell phones, Bluetooth devices, etc.), in relation to the fact that the APs identify the network topology considering indoor path loss models. Such models, if we assume static environments, without moving users operate with accuracy, however in the case under discussion, the network elements need to calculate the topology on a constant basis, in every CPC loop.

Fig.10 provides the 6th degree polynomial function of the SINR measurements during the experimentation. At any case, the SINR is better compared to the case where maximum TxPower has been set to the APs. For the AP 3 and 4 the experiment stops at the time that these APs are being turned off (13.20 and 14:20 respectively) and we see that when all four Soekris operate, the SINR to all of them is low. When we start turning off AP we observe that the SINR to all the operating ones starts increasing; this is related to the fact that the interference that is caused reduces as well. Finally, only one, AP 2, remains operational and we have a huge increase in the SINR, which has started when we turned off AP3 and AP4; however we should take under consideration that the overall capacity reduces.

Fig.11 presents the number of iterations every time the CPC is being triggered. We consider that the CPC is being triggered periodically, every 5 minutes. The Soekris APs exchange messages asynchronously; everyone using its own intervals. We observe that the scheme converges in small number of iterations most of the times (mean value of iterations 3.876).

The initial configuration of the network elements is a

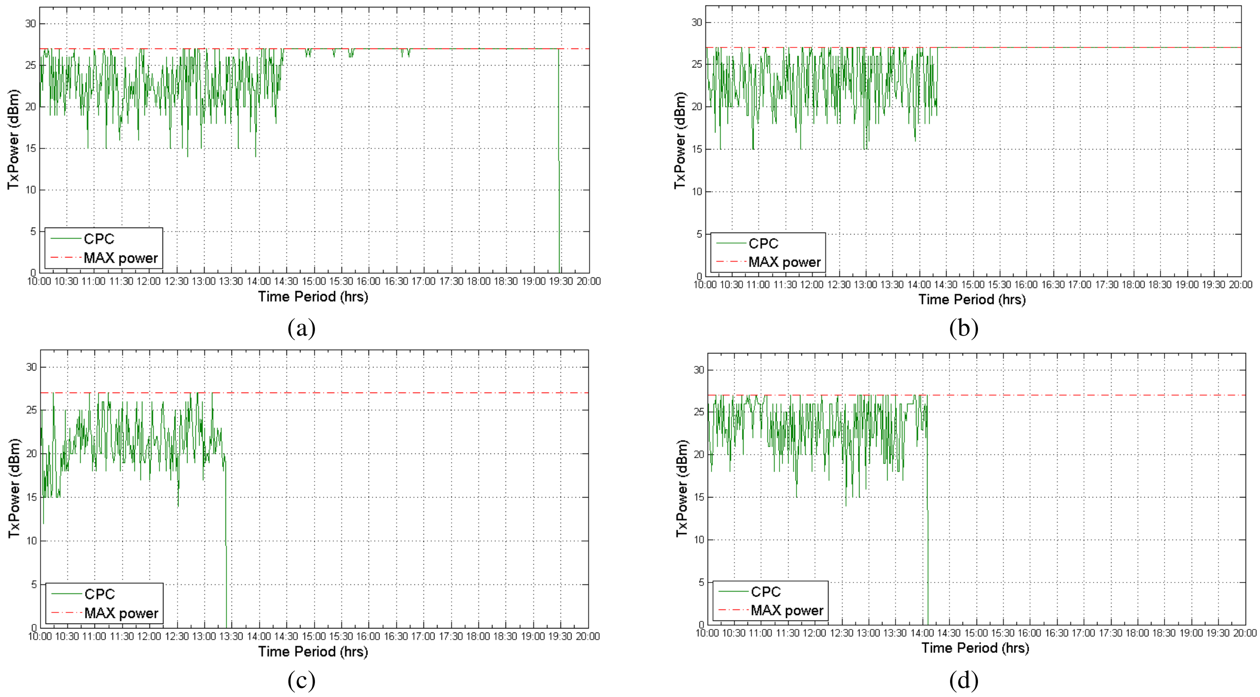


Figure 9: Transmission power adjustments in the four Soekris APs using the Cooperative Power Control scheme in (a) Soekris AP1, (b) Soekris AP2, (c) Soekris AP3, (d) Soekris AP4

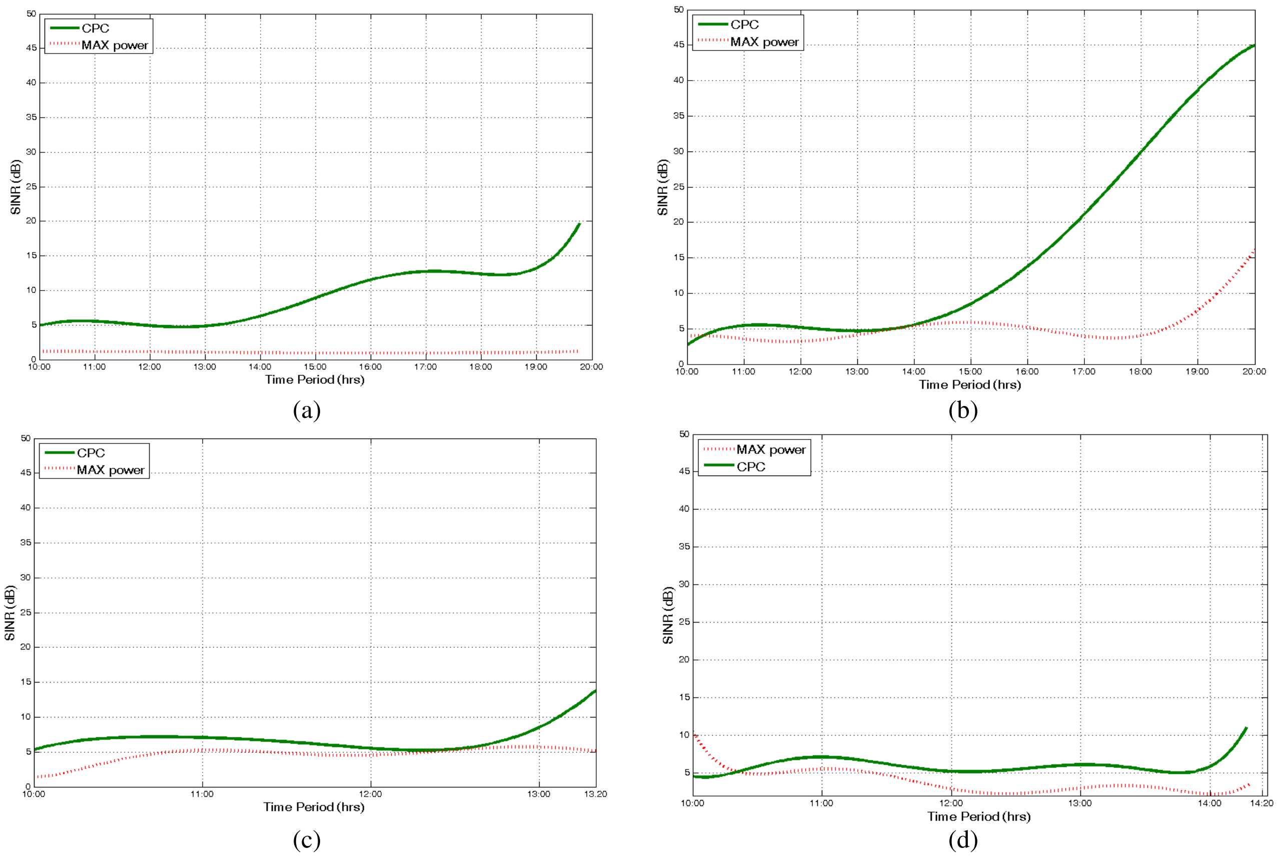


Figure 10: SINR evolution during the experimentation period for (a) Soekris AP1, (b)Soekris AP2, (c)Soekris AP3, (d)Soekris AP4

generic one and captures a great variety of environments. However, for both the physical and network topology which has been used for experimentation, this configuration is not

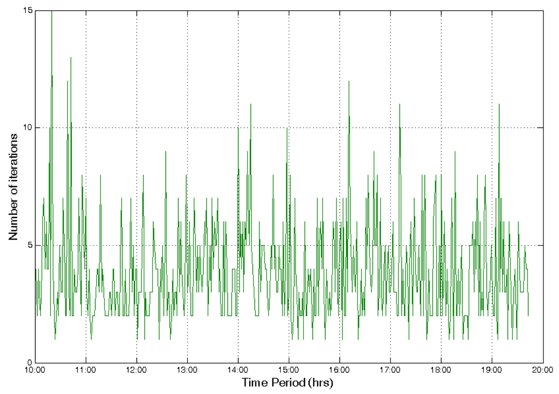


Figure 11: Number of iterations every time the CPC is being triggered

the most suitable one. Thus, the adaptation scheme that has been presented in Section V.C has been incorporated.

During the first experimentation day of the CPC in the Soekris devices, the inputs of the fuzzy reasoner are being collected. Then, the tuples are being clustered and the overlapping areas are being mapped to the uncertainty bounds in the input membership functions. Fig. 12 provides the transmission power throughout the second experimentation day for all Soekris devices, with the adapted input membership functions (learning-based CPC scheme).

As it is obvious, the CPC scheme is more sensitive to the environment, compared to the previous day of experimentations. Given the fact that they operate in the same environment, the APs proceed even more often in transmission power adjustments. Also, when only two APs remain operational, as the experimentation proceeds, we observe that they proceed in transmission power adjustments, according to the environment stimuli, contrary to the first day, where the transmission power adjustment mainly occurred when all the APs were operational.

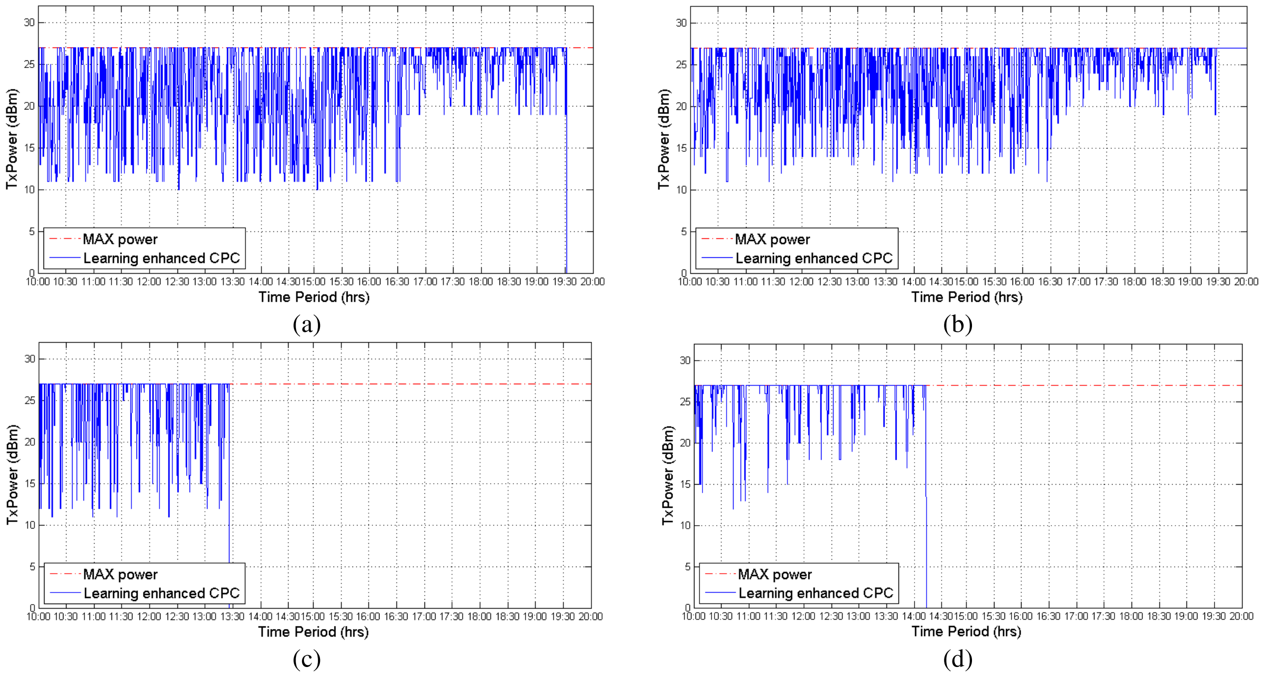


Figure 12: Transmission power adjustments in the four Soekris APs using the Learning enhanced Cooperative Power Control scheme in (a) Soekris AP1, (b)Soekris AP2, (c)Soekris AP3, (d)Soekris AP4

Furthermore, we observe significant energy gains, in relation to the case without learning capabilities. More specifically, AP 1 has 24.73% less power consumption compared to the maximum transmission power, whereas AP 2 consumes 18.01% less power, AP 3 14.69% and AP 4 5.65%. Given the fact that AP1 and 2 are the APs that remain operational almost throughout the experiment, we conclude that the energy gains are even more significant. Regarding the SINR, it remains in the same levels as in the case of the core CPC algorithm (Fig. 13), due to the fact that the objective function to be optimized is the same. The APs proceed in power adjustments in lower transmission power

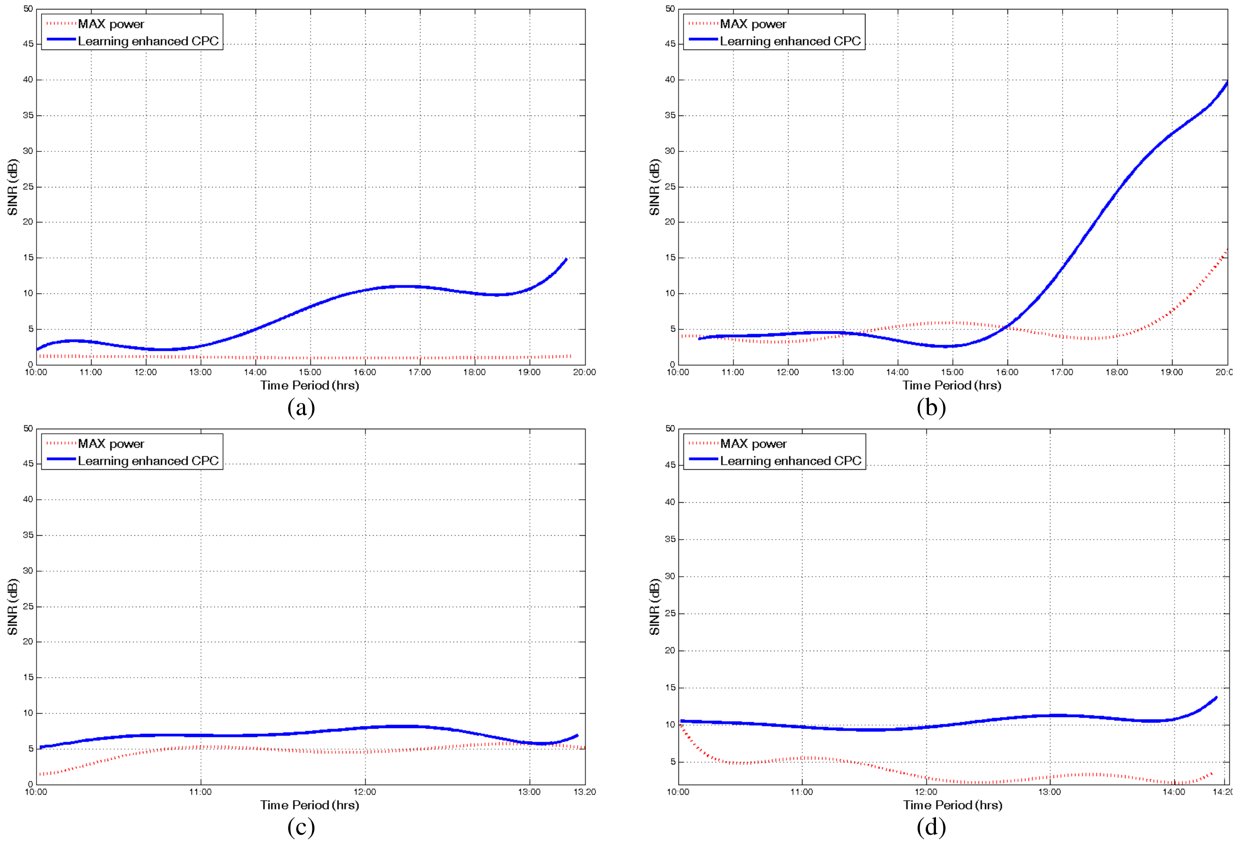


Figure 13: SINR evolution during the experimentation period for (a) Soekris AP1, (b)Soekris AP2, (c)Soekris AP3, (d)Soekris AP4

levels resulting in less interference as well; however the SINR remains at the same levels, due to the decrease in both metrics (i.e., TxPower and interference). Fig. 14 presents the number of iterations every time the CPC is being triggered after the learning procedure. Similarly to the core CPC, we observe that the scheme converges in small number of iterations most of the times; furthermore, we observe a slight decrease in the overall mean value of iterations (3.47) which also highlights that the system has become more suitable to its environment. Finally, considering that the algorithm is being triggered periodically, every 5 minutes for 10 hours, we observe that the adaptation algorithm enhances the situation perception scheme using relatively small amount of measurements (4 AP * 120 measurements/AP = 480 measurements).

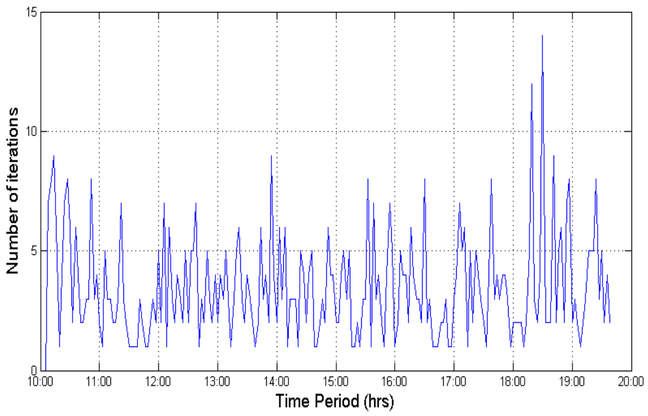


Figure 14: Number of iterations every time the CPC is being triggered

VIII. Conclusion And FutureWork

This paper proposes an algorithm for power control and interference mitigation. The solution also incorporates learning capabilities in order to enable the network elements to adapt to their situation perception according to the environment stimuli. The learning procedure captures the positive or the negative impact of an action (i.e., transmission power set value) in the interference that a network element causes to its neighbors.

The novelty of our contribution is the combination of the merits of fuzzy logic and data clustering for the optimal interpretation of the network uncertainties and its incorporation to the CPC algorithm. The network uncertainties have been identified using the cluster overlaps; the latter are then being translated in the environment perception of the fuzzy reasoners (i.e., input membership functions).

The proposed solution has been tested in a realistic office environment in a real life experiment. The algorithm has been deployed in WiFi APs and used for their transmission power control. The experimental analysis proved the applicability of the CPC and the benefits from its incorporation. More specifically, the network elements have significant energy gains by incorporating the CPC; the addition of the learning capabilities in the APs makes them more sensitive in the environment stimuli. The experimental analysis proved that also the WiFi APs SINR benefits from the incorporation of the CPC; in every case the SINR is improved compared to an environment where all APs set their TxPower to maximum levels. Thus, the network elements achieve higher SINR levels and also have energy gains by setting their TxPower to the most suitable level for them and for the overall network.

Our future work includes the incorporation of more sophisticated data mining techniques (e.g., Support Vector Machines, C-Means, Subtractive clustering, etc.) in order to have better adaptation to the environment. Furthermore, the incorporation of outlier detection techniques will be investigated in order to ensure that only valid measurements will be used for the learning procedure.

Acknowledgment

The research leading to these results has received funding from the European Community's Seventh Framework Programme (FP7/2007-2013) under grant agreement CONSERN n° 257542.

The authors would like to thank Dr. Panagis Magdalinos for his help in order to improve the paper and the SCAN Lab members for participating in the experimentation procedure.

References

[1] P. Spapis, G. Katsikas, M. Stamatelatos, K. Chatzikokolakis, R. Arapoglou, and N. Alonistioti “Learning Enhanced Environment Perception for Cooperative Power Control”, Fifth International Conference on Mobile Ubiquitous Computing, Systems, Services and Technologies (UBICOMM 2011), 2011.

[2] Q. Sun, X. Zeng, N. Chen, Z. Ke, and R. Ur Rasool, “A Non-cooperative Power Control Algorithm for Wireless Ad Hoc and Sensor Networks”, Second International Conference on Genetic and Evolutionary Computing (WGEC), 2008.

[3] J.F. Nash, "Equilibrium points in n-person games", Proceedings of the National Academy of Sciences 36(1): 48-49, 1950.

[4] P. C. E. Shannon, “Communication in the presence of noise,” Proceedings of the Institute of Radio Engineers, vol. 37, pp. 10–21, 1949.

[5] G. J. Foschini and Z. Miljanic, “A simple distributed autonomous power control algorithm and its convergence,” IEEE Trans. Veh. Technol., vol. 42, pp. 641–646, 1993.

[6] D. Mitra, “An asynchronous distributed algorithm for power control in cellular radio systems,” in Proc. 4th Winlab Workshop Third Generation Wireless Information Network, pp. 249–257, 1993.

[7] Chun-Gang Yang, Jian-Dong Li, and Zhi Tian, “Optimal Power Control for Cognitive Radio Networks Under Coupled Interference Constraints: A Cooperative Game-Theoretic Perspective”, IEEE transactions on vehicular technology, vol. 59, no. 4, pp. 1696-1706, 2010.

[8] V.P. Mhatre, K. Papagiannaki, F. Baccelli,"Interference Mitigation Through Power Control in High Density 802.11 WLANs", 26th IEEE International Conference on Computer Communications. (IEEE INFOCOM 2007), pp.535-543, 2007.

[9] K. Ramachandran, R. Kokku, Honghai Zhang, and M. Gruteser, "Symphony: Synchronous Two-Phase Rate and Power Control in

802.11 WLANs," IEEE/ACM Transactions on Networking, vol.18, no.4, pp.1289-1302, Aug. 2010.

[10] K. Kowalik, M. Bykowski, B. Keegan, and M. Davis, "An evaluation of a conservative transmit power control mechanism on an indoor 802.11 wireless mesh testbed," International Conference on Wireless Information Networks and Systems, (WINSYS'08), 2008.

[11] Y. Wei, M. Song, and J. Song, "An AODV-improved routing based on power control in WiFi mesh networks," Canadian Conference Electrical and Computer Engineering 2008, (CCECE 2008), pp.001349-001352, 2008

[12] T.A. ElBatt, S.V. Krishnamurthy, D. Connors, S. Dao, "Power management for throughput enhancement in wireless ad-hoc networks," International Conference on Communications, (IEEE ICC 2000), pp.1506-1513, 2000.

[13] J. Huang, R. Berry, and M. Honig,"Spectrum sharing with distributed interference compensation", First IEEE International Symposium New Frontiers in Dynamic Spectrum Access Networks (DySPAN), 2005.

[14] A. Merentitis and D. Triantafyllopoulou,"Transmission Power Regulation in Cooperative Cognitive Radio Systems Under Uncertainties", IEEE International Symposium on Wireless Pervasive Computing (ISWPC), 2010.

[15] J. Han. and M. Kamber, "Data Mining: Concepts and Techniques", The Morgan Kaufmann Series in Data Management System. ISBN-13: 978-0123814791

[16] P. Magdalinos, A. Kousaridas, P. Spapis, G. Katsikas, and N. Alonistioti, "Feedback-based Learning for Self-Managed Network Elements", 12th IEEE International Symposium on Integrated Network Management, (IM2011), 2011.

[17] http://soekris.com/products/net5501.html, Dec. 2012.

[18] http://homesupport.cisco.com/en-eu/support/adapters/WUSB54GC, Dec. 2012.

[19] http://www.pcengines.ch/pdf/wlm54ag23.pdf, Dec. 2012.

[20] http://jfuzzylogic.sourceforge.net/html/index.html, Dec. 2012.

[21] http://mina.apache.org/, Dec. 2012.

[22] A. Bose and F. H. Chuan, "A practical path loss model for indoor WiFi positioning enhancement," 6th International Conference on Information, Communications & Signal Processing, 2007, pp.1-5, 2007.

[23] J. S. Seybold, "Introduction to RF propagation", Wiley, ISBN-13 978-0-471-65596-1.