Towards Adaptive WLAN Frequency Management Using Intelligent Agents

Fiorenzo Gamba[1], Jean-Frédéric Wagen[1], and Daniel Rossier[2]

[1] University of Applied Sciences of Western Switzerland
Bd de Pérolles 80, 1705 Fribourg, Switzerland
{gamba,wagen}@eif.ch
[2] Swisscom Innovations AG, Ostermundigenstrasse 93, 3006 Bern, Switzerland
daniel.rossier@swisscom.com

Abstract. Private, corporate and public *Wireless Local Area Networks* (WLAN) Hot-Spots are emerging. In this rapidly evolving environment, the configuration of WLAN access points raise the classical problem of re-using limited radio resources. In this paper, the problem of dynamic frequency allocation of WLAN access point in a highly competitive multi-provider Hot-Spots environment is addressed. Our solution aims at working in locations where planned and ad-hoc deployments might be side by side. An on-line adaptive optimization process is proposed and relies on available information delivered only by the local access points in order to maintain the quality of service as high as possible. This optimization process is implemented on a scalable and highly flexible agent-based framework. The easy deployment of intelligent agents in a real WLAN network and their integration in a simulation context allows us to perform extensive tests for small and large-scale networks. The proposed approach has been tested on a limited but practical demonstrator that showed encouraging results.

Keywords: WLAN Network, Frequency Optimization, Software Agent.

1 Introduction

The evolution of telecommunications is characterivzed by an increase in bandwidth availability and an increase in user mobility.

Mobile networks are evolving to support flexible access services and offer increasingly higher bandwidths as well as attractive pricing. For example, several mobile operators are about to offer new complementary services named here "Hot-Spot" access [1–3]. Hot-Spot solutions provide broadband mobile public access to the Internet and to corporate intranets. The coverage of a WLAN Hot-Spot is typically poor compared to a 2G and 3G mobile cellular solution but this limitation can be seen as an advantage to provide end-users with high bandwidth capacity. Current Wireless Local Area Network (WLAN) technologies deliver services access around 100 meters. Several mobile operators currently present WLAN Hot-Spot access as a complement to their GPRS and future UMTS offerings. User applications will have to be able to roam from one local Hot-Spot to another. For example: a user can access to her company's Email from the hotel lobby and access it again at the airport gate.

Despite many shortcomings with respect to security and inter-network or inter-operator roaming, WLAN IEEE802.11b products are increasingly becoming very

S. Pierre, M. Barbeau, and E. Kranakis (Eds.): ADHOC-NOW 2003, LNCS 2865, pp. 116–127, 2003.

popular in many countries. Many reasons can be mentioned for such a success: low price, software support and a very successful interoperability under the WiFi logo.

WLAN products continue to raise new expectations since they can demonstrate efficient LAN access at rates above 2 Mbps (IEEE 802.11), at 11 Mbps for the IEEE 802.11b specifications, or even several tens of Mbps in the near future (IEEE 802.11a and HIPERLAN)[1].

In this paper, we propose a novel approach to address the dynamic frequency allocation of WLAN *Access Point* (AP) in a multi-provider Hot-Spots environment, in order to maintain high level of *Quality of Service* (QoS) taking into account users' traffic. In Section 2, a description of the WLAN channel allocation problem and management issues are outlined. Our intelligent agent based approach to tackle the channel allocation problem of the WLAN access points is described in Section 3. Section 4 briefly presents the fundaments of frequency optimization. Section 5 shows some preliminary results in our test environment, and Section 6 concludes this paper.

2 Channel Allocation in WLAN Networks

Corporate users and, as price decreased, campus communities [4], have been the first to exploit the benefits of WLAN based on the IEEE802.11*x* family of standards. Private users are also using the technology to avoid new wires in their homes and sometimes to share Internet access with other users. An increasing number of telecommunication operators and visionary companies have identified new business models for the deployment of public WLAN access at popular locations (Hot-Spots) or in their business premises [5]. As a single example close to the authors, the Swiss operator Swisscom Mobile has launched commercial services based on public WLAN Hot-Spot access based on GSM subscriptions or special value cards since the end of 2002[2].

In this rapidly evolving environment, the deployment of WLAN has to face typical issues regarding optimal utilization of radio resources that can be provided within the allocated radio spectrum. The complexity of the optimization problem is amplified by the fact that all WLAN stakeholders, from private to business entities, share the same spectrum allocation without any cooperation unless proprietary bi-lateral agreements could be arranged. For example, a public WLAN Hot-Spot operator might find difficulties to offer its service at the bus stop near a popular restaurant, which has installed its own WLAN coverage. Adding private WLAN users in the flats above the restaurant further illustrates the need for an autonomous management of the WLAN access points. A manual and static configuration of each access points, besides being tedious, could only provide an acceptable solution for the conditions found during the measurements survey. A better solution would be to centralize some information about all the access points potentially interfering. Based upon a centralized database, the best frequency allocation could be computed. However, this solution is not possible in practice because it does not scale to a very large number of access points. Furthermore, the conditions can be quite complex since the access points are deployed in planned or ad-hoc manners depending on operators and service providers.

[1] http://www.alcatel.com/atr

[2] http://www.swisscom-mobile.ch/sp/4EGAAAAA-fr.html, visited March, 2, 2003.

In current deployment of WLAN hot spots, the frequency channel is fixed manually [6] and will be changed only if consumers complain or if a new survey is undertaken by WLAN operators.

We developed an autonomous management system that will detect the new situation and adapt the radio resource allocation so that the quality of service is maximized.

2.1 Towards Agent-Based Management

This paper describes the architecture of an autonomous and adaptive management system dedicated to on-line optimization of access point channel assignment; the process mainly relies on environmental information issued from the surrounding access points and information exchange between enhanced access points.

The target architecture resorts to autonomous software agents which can run on physically distributed or centralized platforms. From a logical viewpoint, each software agent is delegated to a single access point. Technical information such as the currently used frequency channel, the number of users, the number of rejected packets, etc. are gathered by each software agent and constitute their internal knowledge, i.e. their internal representation of the *local* environment. The implementation of advanced mechanisms to exchange internal knowledge between agents will enable the enhanced access point to perform a *local* optimization.

One of the major characteristics of the proposed solution resides in its ability to deal with currently deployed WLAN networks in concordance with the established IEEE 802.11 standards [7][8] and related management systems. Thus, management techniques directly rely on standardized protocols and information models making a vendor-independent implementation possible. *Simple Network Management Protocol* (SNMP) has become the de facto standard for IP network management. SNMP is commonly use to manage IP based elements and also for wireless elements.

3 Intelligent Agent Approach

The management of future WLAN networks will have to cope with highly competitive environment where several operators will deploy their own infrastructure in the same geographical areas. Furthermore, the diversity of WLAN equipments will make such a network very heterogeneous. As a consequence, centralized network management systems will be not sufficient to control fine-grained resources allocation with respect to the available frequency spectrum. On the other hand, legacy management systems such as SNMP-based systems can not be ignored in the overall architecture. The elaboration of hybrid centralized and de-centralized network management systems therefore constitute a general trend, not only for WLAN networks but also for other mobile networking technology like ad-hoc networks [9].

In this context, *intelligent agents* can be considered as one of the most promising approaches addressing issues related to distributed applications in the rapidly expanding communication industry [10]. Intelligent agents can be seen as a software program that can perform specific tasks for a user and possesses a degree of intelligence that permits it to perform parts of its tasks autonomously and to interact with its environment in a useful manner [11]. An intelligent agent exhibits the following properties: *autonomy* - the agent is capable of following its goal autonomously that is, without interactions or commands from the environment - *reactivity* - the agent is capable of reacting appropriately to influences or information from its environment - *pro-activity* - under specific circumstances, the agent can take the initiative in performing appropriate actions - *social ability* - the agent is able to communicate with other agents and to interact with its environment in order to fulfill its tasks.

Intelligent agents have been considered for network management in numerous research projects[3]. These various projects have led to multi-layer agent architectures in which each layer implements different abstraction views; examples of such layers are the co-operation layer, the planning layer and the reactive layer. In this context, the *Belief-Desire-Intention* [12] probably constitutes one of the most popular agent architecture and has also been considered, under different forms, in agent-based network management systems.

The development of agent standards in telecommunication is obviously a *sine qua non* condition for the successful deployment of software agents in large-scale networks. The most popular agent standard at the moment is the *Foundation for Intelligent Physical Agents* (FIPA)[4].

The framework we propose in this paper consists of a simple architecture in which we mainly exploit the message and communication facilities provided by the agent platform on the one hand, and the capability of an agent to implement different parallel behaviors on the other hand. Details about agent behavior are given in Section 3.2.

Our agent-based framework is therefore composed of intelligent agents, called *AWM_agents*, which exchange information concerning their local environment in order to perform on-line optimization by (re-)configuring the access point frequency (or channel). The enhanced access point consequently exhibits an autonomous and adaptive behavior.

3.1 Jade and LEAP Agent Platforms

Jade[5] is a freely downloadable *Java* agent platform, which is fully compliant with the last revision of FIPA specifications; the intra-agent activity model defined in *Jade* is based upon a non-pre-emptive concurrency model. A *Jade* agent is implemented with a *Java* thread, which enables asynchronous inter-platform communication as specified by FIPA. The *Jade* agents can implement one or several *behaviors*: while intra-agent activities are synchronous, inter-agent communication relies on an asynchronous process. The behaviors are executed in a thread-per-agent concurrency model, in which there is no stack to be saved: they are managed by an internal scheduler implementing a round-robin non-pre-emptive policy among all the behaviors available in the ready queue of an agent [13]. The synchronous characteristic of intra-agent activity and related cooperative processes makes *Jade* an attractive agent platform for the study of the agent behavior in the context of telecommunication applications, so that our *AWM_agents* have been implemented into the *Jade* environment. Still, *Jade*

---

[3] An excellent overview of project activities concerning intelligent agents for network management can be found in [**13**].

[4] http://www.fipa.org

[5] http://sharon.cselt.it/projects/jade

is made up of numerous classes and can thus be difficult to implement into embedded systems.

The *Lightweight and Extensible Agent Platform* (LEAP)[6] is a project aiming at the realization of a FIPA platform that can be deployed seamlessly on any *Java*-enabled device endowed with sufficient resources and with a wired or wireless connection, such as PDAs and smart phones [15]. *LEAP* significantly contributes to providing network devices with an embedded agent platform.

LEAP appears to be particularly interesting because it can easily be deployed in a simple Java processor based platform, which can be connected to any vendor-independent access point. We are now investigating the deployment of our *AWM_agent* into a *LEAP*-based Java processor based platform.

3.2 Agent Behavior

The *Jade* agent platform [16] provides a novel approach towards task design with generic behaviors. Based on message exchanges between agents, several behavior schemes corresponding to various task types are defined in order to enable multiple interactions with other agents.

The behaviors are divided into two main categories, respectively *simple* and *composite* behaviors. A simple behavior consists in a task that is activated only once and cannot be blocked - *oneShotBehaviour* - or in a cyclically activated task. A composite behavior is made up of several behaviors according to a parent-child relationship; it may consist of a sequential behavior - *SequentialBehaviour* - which executes the sub-behaviors sequentially and terminates when all sub-behaviors have been executed; on the contrary, parallel behavior - *ParallelBehaviour* - allows the developer to implement sub-behaviors which can be executed in a non-deterministic order. Finally, a behavior can be described with a *finite state machine* (FSM); the parent behavior controls the transitions between the FSM states and activates the behaviors corresponding to the current state.

From the communication point of view, the agents can interact via *intra-platform* communication: all the agents participating in the interaction are managed by the same platform; they *reside* in the same environment. The agents can also be distributed over several platforms, in which case they interact via an *inter-platform* communication mechanism. In both cases, agents communicate via ACL messages. For example, if an agent platform is dedicated to one and only one access point, the hosted *AWM_agent* endowed with an SNMP manager can be logically perceived as the access point itself and only inter-platform communication will take place. On the contrary, if an agent platform hosts several *AWM_agents*, the agent platform is responsible for managing several access points, and intra-platform communication will take place between the *AWM_agents* residing in the platform.

3.3 AWM Agent Architecture

In our framework, an *AWM_agent* is dedicated to an access point and performs three basic tasks: at first, the agent continuously monitors its local environment by querying the SNMP agent of the access point; the retrieval of particular values from MIB vari-

[6] http://leap.crm-paris.com

ables will give information about the number of frames with errors (MIB:FCS), the number of frames delivered correctly (MIB:InUPkt), the number of associated stations (MIB:NAS) and the frequency channel (MIB:Channel), so that the agent can have an internal representation of the environment. Secondly, the agent handles incoming messages issued from other agents. The message contents are examined and processed accordingly. The exchange of information between neighboring agents improves the channel assignment. The agent finally has to perform local computation to determine the most appropriate channel.

The overall agent architecture and processes are depicted in Figure 1.

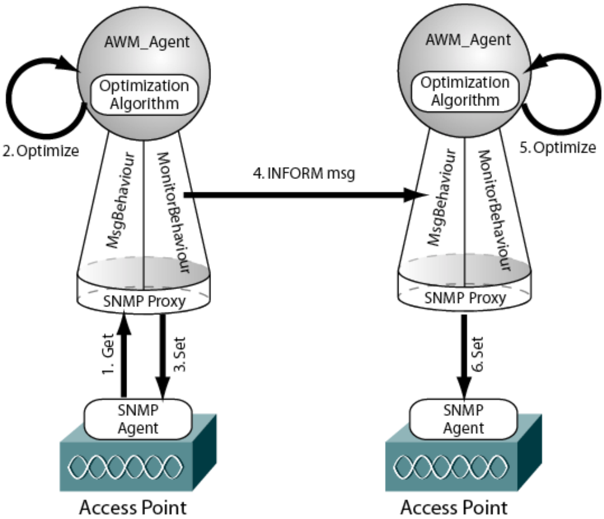

Fig. 1. AWM Agent Architecture

We now briefly introduce the general scenario involving our *AWM_agents* and related access points. The optimization algorithm we have implemented is currently being patented and is therefore not detailed in this paper. It is however important to mention that the overall architecture perfectly suits a wide range of distributed optimization algorithms.

The monitoring task and the message processing are implemented by means of two *Jade* cyclic behaviors. The monitoring process is implemented into the *MonitorBehaviour*, while message handling is implemented into the *MsgBehaviour*.

The *AWM_agent* queries the SNMP agent in order to retrieve the information from the access point, which is controlled by the agent (1). In case of interference, the agent activates the optimization algorithm (2) and computes the new channel to be assigned to the access point. The new channel is set via a SNMP request (3). The agent then sends the new channel to the neighboring agents (4). The receiving agent reads the message contents, makes sure that it fits the AWM ontology and in turn activates the optimization algorithm if necessary (5) to compute the new channel based upon the updated information. Finally, the new channel is assigned by means of a SNMP request (6).

4 Basic Principles for Channel Assignment Optimization

Classical frequency optimization in cellular networks is based on simple rules regarding frequency channel allocation. Usually, the same frequency and even an adjacent frequency cannot be repeated at the same location or neighbored locations.

The particular definition [8] of overlapping frequency channels in WLAN IEEE802.11b with DSSS (*Direct Sequence Spread Spectrum*) and the CSMA/CA (*Carrier Sense Multiple Access/Collision Avoidance*) technique lead that a more complex rules regarding frequency channel allocation. Indeed, measurements as depicted on Figure 2 shows that a better user throughput is obtained when there is either a total overlapping or, as expected, no overlapping of the interfering channels allocated to different access points.

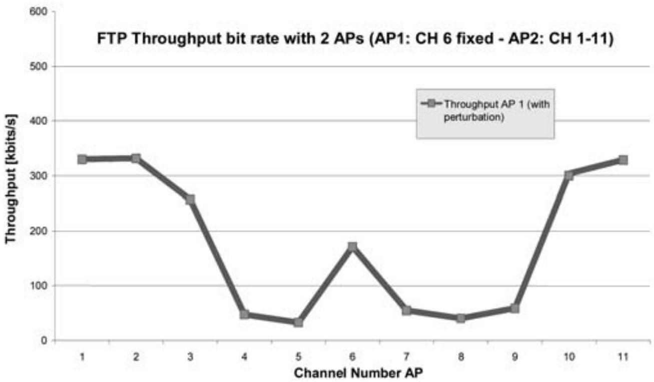

Fig. 2. Throughput measurements versus channel separation

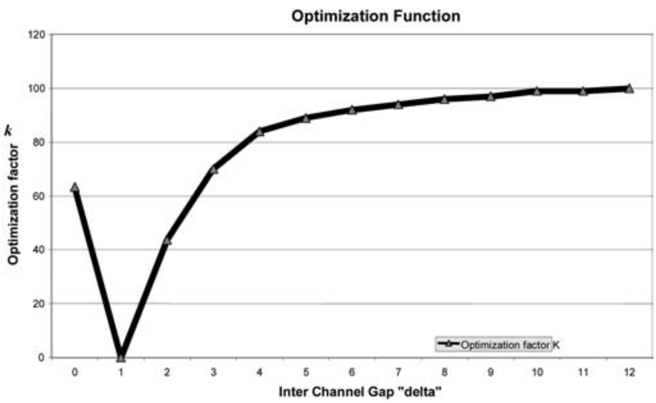

Fig. 3. Optimization function

Figure 2 reports FTP total throughputs measured in the following conditions: two users, each linked to its own access point: one with frequency 6 (AP1) and the other access point (AP2) with the frequency channel x (value on the x-axis). The result shows that partial overlapping is worst than a complete overlapping of the frequency channels. This can be explained from the effectiveness of the *collision avoidance* (CA) when the two channels are equal. Error rate measurements not reported here also shows that partial overlapping of frequency channels lead to a larger number of errors, while total overlapping or non-overlapping channels lead to negligible numbers of errors.

The results presented here have been taken into account in the design of our optimization function represented on Figure 3: a value between 0 and 100 is assigned to the difference between 2 channels.
The optimization function is not monotone. Partially overlapping channels lead to low numbers. However, re-using the same channel on competing access points is better that choosing partially overlapping frequency channels. As expected, choosing non-overlapping channel leads to the highest score.

The optimization function provided in Figure 3 can be adapted if necessary to take into account other functions if desired.

5 Experiments and Results

5.1 Test Bed Environment

Our test bed environment is based on four access points (AP) representing two *Wireless Internet Service Providers* (WISPs). An *Autonomous WLAN Management* (AWM) agent is connected to each AP and each AP is configured with a *Service Set Identifier* (SSID) that characterizes the WISP. Since The AWM agent must communicate between WISPs, then it is assumed that WISPs have to be inter-connected. At least, WISPs must allow their agents to exchange messages. It is recalled that the software agent platform chosen in this work simplify greatly this exchange of messages. Figure 4 shows the test environment, its architecture and the exchange of messages. Each of the 4 access points has its own PC acting as a proxy for the access point. The proxy runs the software agent platform and the software agents that have been designed to implement the AWM system.

Access Points are Cisco Aironet 350 products. The AWM Agent platform is based on Jade platform and runs on Pentium-III PCs. Wireless LAN clients are laptops with PCMCIA WLAN cards. To test traffic congestion, we have implemented a client emulator in the AWM agent. Thus associated terminals can be emulated by this feature on each AP.

This practical test environment has a limited size and can be used to demonstrate the feasibility of our approach and determine the user experience under different the frequency adaptation algorithm. Simulation environment has also deployed using the *Generic Network Management Tool* (GNMT) [17] described in the next sub-section. In this case, larger network with several tens of access points have been simulated. Comparisons with the practical test environment can also be performed.

5.2 Preliminary Results

In this section, we briefly present the first results we obtain with our experimental environment. Figure 5 presents the four access points with virtual interference links. It is recalled that a *Virtual Interference Link* (VIL) is defined as a communication channel between two access points which are subject to interfere each with other. Currently, VIL topology is determined and configured manually by editing a property file for each *AWM_agent*. Automatic VIL discovery mechanism is currently being investigated.

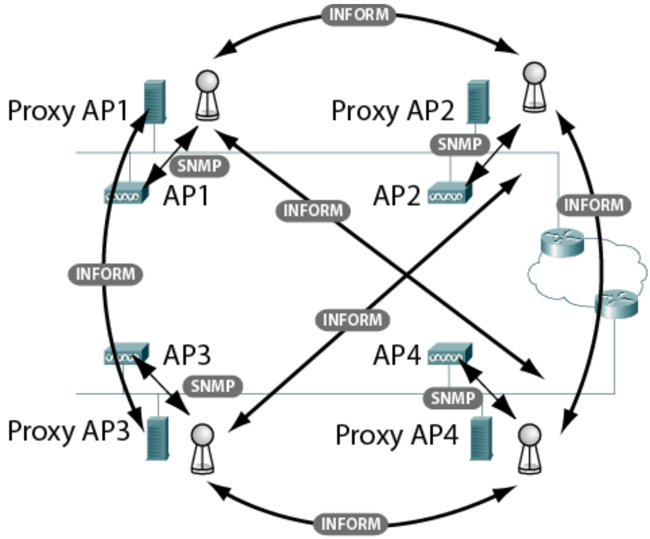

Fig. 4. Test environment architecture

The number appearing on each VIL corresponds to the difference of frequency channel between APs. For example, AP1 is configured on channel 13 and AP2 is also on channel 13, therefore the number 0 (=|freq(AP1)-freq(AP2)|=|13-13|) is circled on the VIL (AP1, AP2).

MIB Parameters have been introduced in Section 3.3. In the beginning of our experiment, we have configured each access point on the channel 1. We have then associated a certain number of stations (mobile users) to the access points according to the following scheme: two stations (users) are associated to AP1, three to AP3, five to AP2 and no station is associated to AP4. The numbers depicted on the figure shows the final (and stable) configuration we obtain after less than 10 minutes. It is important to see that the optimization algorithm takes into account the possibility to have two access points configured with the same channel (AP1 and AP2). As explained in Section 4, having two neighboring access points on the same channels may be considered as a better solution than having a small difference of frequency.

Figure 6 shows the adaptive process over time and, hence, the evolution of the assigned frequency channel for each access point. A different line profile (size and style) is given to each AP.

The optimization algorithm must obviously ensure that the process will converge and avoids cycles. The algorithm we implemented becomes stable after a few minutes. This algorithm is being patented, thus no more details are provided. During the optimization process, the APs may change several time their frequency channel. A user associated with a particular access point loose a few packets during the 1 to 3 seconds break occurring at each change of frequency channel. Practically, the end users do not feel these losses especially if the newer operating systems are used.

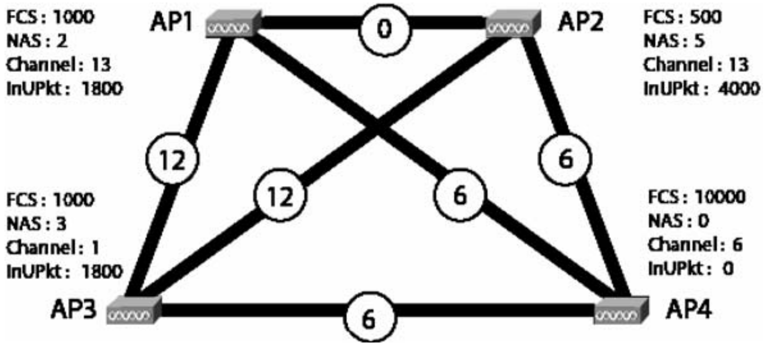

Fig. 5. Final values when the adaptive process becomes stable (circle number is on VIL)

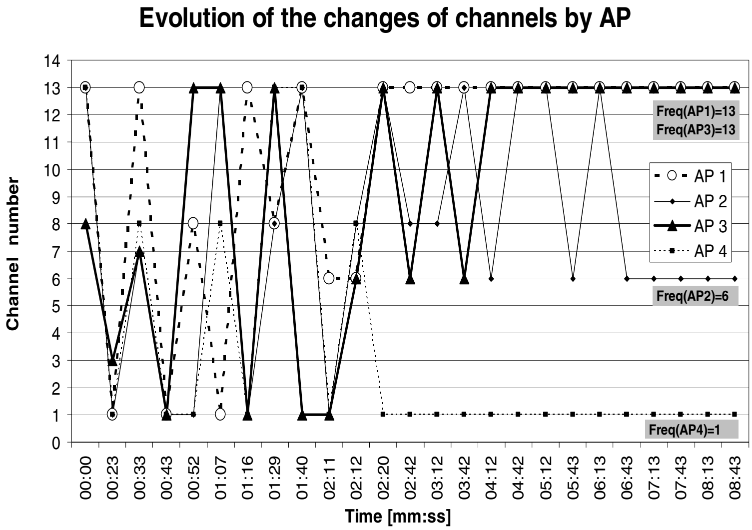

Fig. 6. Evolution of the adaptive process over time

6 Conclusions

An approach to address the problem of dynamic frequency allocation for WLAN access point in a multi-provider Hot-Spots environment has been presented. The operating frequency channel at each access point is modified in order to increase the quality of service measured at the access points. This channel (re-)configuration is performed via SNMP. Several parameters are combined in a metric defined to objectively measure the performance at each access point and compare the results to its neighbors. In our demonstrator, four parameters obtained from the MIB access points have been used: the number of frames with errors (FCS), the number of frames delivered correctly (InUPkt), the number of associated stations (NAS) and the frequency channel.

The exchange of information between access points regarding performance measures and newly assigned channel plays a central role in the solution presented here. Advanced communication mechanisms rely on the *Jade* agent-platform. Our agent-based framework is composed of intelligent agents, called *AWM_agents*, which closely interact in order to keep an up-to-date internal representation of their local environment and therefore to perform on-line optimization by (re-)configuring the frequency channel of the access point.

The architecture and functionalities of our solution has been explained. Each access point is controlled by an *AWM_agent*. The *AWM_agent* queries a *SNMP* agent in order to retrieve the information from the access point. In case of interference, the agent activates an optimization algorithm and computes the new channel to be assigned to the access point.

Preliminary results have been presented to illustrate the feasibility of our approach. The stability of the optimization process has been tested on the demonstrator and the simulator. Measurements demonstrated that frequency channels can be modified with little perturbation to the users associated to a given access point.

Future work will focus on extending the demonstrators to a large number of access points in a campus environment. Further tests and measurements will also be performed on the simulator. Furthermore, improvements of the optimization algorithm will be investigated.

References

1. Levillain, P., Wireless LAN for Entreprise. Alcatel Telecommunications Review, Q2002
2. Juha Ala-Laurila, Jouni Mikkonen, Jyri Rinnemaa, "Wireless LAN Access Network Architecture for Mobile Operators", IEEE Commun. Mag,. November 2001, pp 82-89.
3. Shidong Zhou, Ming Zhao, Xibin Xu, Jing Wang, Yan Yao, "Distributed Wireless Communication System: A New Architecture for Futur Public Wireless Access", ", IEEE Commun. Mag, .March 2003, pp 108-113.
4. David Kotz, Kobby Essein, "Analysis of a Campus-wide Wireless Network", MOBICOM'02, September 23-26, 2002 Atlanta, Georgia, USA
5. Heinz Luediger, Sven Zeisberg, "User and Business Perspectives on an Open Mobile Access Standard", IEEE Commun. Mag,. September 2000, pp 160-163.
6. Alex Hills, "Large-Scale Wireless LAN Design", IEEE Commun. Mag,. November 2001, pp 98-104.
7. IEEE Std 802.11, Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications, including ieee802.11-mib, edition 1999
8. IEEE Std 802.11b, Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications: Higher-Speed Physical Layer Extension in the 2.4 GHz Band, ISBN 0-7381-1812-5.
9. Chien-Chung Shen, Chavalit Srisathapornphat, and Chaiporn Jaikaeo, "An Adaptive Management Architecture for Ad Hoc Networks", IEEE Commun. Mag, February 2003
10. Nicholas R. Jennings, "An Agent-based Approach for Building Complex Software Systems", Communications of the ACM, Vol.44, No.4 (April 2001).
11. Walter Brenner, Rüdiger Zarnekow, Hartmut Wittig, "Intelligent Software Agents", Springer-Verlag (Berlin Heidelberg, 1998).

12. Wooldridge and N.R. Jennings, "Agent Theories, Architectures, and Languages: a Survey", In M. Wooldridge and N.R. Jennings, editors, Intelligent Agents, number 890 in LNCS, pages 1-39 (Springer Verlag, 1995).
13. Alex L. G. Hayzelden, Rachel A. Bourne, "Agent Technology for Communication Infrastructures", Wiley & Sons Ltd, 2001.
14. Fabio Bellifemine, Agostino Poggi, Giovanni Rimassa, "Jade - a FIPA-compliant agent framework", in Proc. of the 4th International Conference on the Practical Application of Artificial Intelligence and Multi-agent Technology (PAAM'99), pp.97-108, (London, UK, 1999).
15. Federico Bergenti and Agostino Poggi, "LEAP: A FIPA Platform for Handheld and Mobile Devices", in Proc. of 8th Intl. Workshop on Agent Theories, Architecture and Languages (ATAL'2001) (Seattle, USA, August 2001).
16. Fabio Bellifemine, Giovanni Caire, Tiziana Trucco, Giovanni Rimassa, "Jade Programmer's Guide", available at http://sharon.cselt.it/projects/jade, 2002.
17. Daniel Rossier, “A Description of the Generic Network Management Tool”, Technical Report, Department of Informatics, University of Fribourg (Switzerland, August 2002).