On Self-coordination in Wireless Community Networks

Frank A. Zdarsky, Ivan Martinovic, and Jens B. Schmitt

disco | Distributed Computer Systems Lab
University of Kaiserslautern, 67655 Kaiserslautern, Germany
{zdarsky, martinovic, jschmitt}@informatik.uni-kl.de

Abstract. Co-channel interference and contention at shared medium access may significantly degrade the performance of a CSMA/CA-based wireless LAN. While this phenomenon may be controlled within a single administrative domain by choosing appropriate access point installation sites and assigning operating channels intelligently, there is little that can be done against interference by access points from other nearby administrative domains. This problem becomes paramount in so-called *wireless community networks*, as each access point is operated by a different owner and can be viewed as a separate domain. In this paper we propose a *distributed algorithm and protocol for self-coordination of access points* from different domains based solely on knowledge about the immediate neighborhood. We show that our distributed coordination algorithm may lower contention by around 19% compared to standard WLAN.

Keywords: Wireless LANs, contention, self-coordination.

1 Introduction and Motivation

The emergence of wireless community networks (e.g. NYCwireless[1]) is a remarkable and growing phenomenon that is fueled by the desire of ubiquitous, low-cost, and high-speed Internet access. These networks are based on access points which are independently run by volunteers with their own equipment. The common goal is to enable sharing of wireless Internet access with other members of the community, gradually growing the network to a large, city-wide scale.

Wireless community networks tend to be quite different from the typical wireless LAN deployment. A single public or private organisation is able to preplan access point locations, relying on expert knowledge or using commercially available WLAN planning tools. As a result, these networks may cover an area with comparatively few access points and little overlap between co-channel radio "cells". In contrast, wireless community networks usually grow in an unplanned, evolutionary process, and their access point locations are defined by the users willing to participate. Some areas covered by such networks may therefore have very high node densities. In fact, as observed in [2], areas with densities of more than 10 (and even up to 80!) overlapping access points from different networks are not uncommon in some major U.S. cities. As the number of available non-overlapping channels in IEEE 802.11 WLANs is very low, it is not surprising that the performance in such environments may be severely impaired.

P. Cuenca and L. Orozco-Barbosa (Eds.): PWC 2006, LNCS 4217, pp. 251–262, 2006.

A solution to this problem is to introduce coordination mechanisms between access points of different administrative domains. While products such as wireless switches[3] and self-configuring access points[4] are available for radio management inside single administrative domains, the problem of inter-domain contention has only recently started to attract the attention of the scientific community[5].

In previous work we have proposed a mathematical model of the minimum inter-domain contention problem and methods for finding near-optimal solutions based on global knowledge[6]. In this paper, we present a distributed algorithm and protocol for the self-coordination of access points that uses only regional knowledge and therefore lends more naturally to the problem of self-coordinating access points from a large number of different administrative domains, as is the case in wireless community networks. We show by simulation that our algorithm may reduce contention by 20% compared to standard WLAN. Furthermore, we show that in dense deployments with only few available channels, the intuitive and frequently proposed approach to load-balance between available access points may *not* be optimal and that it may sometimes be preferable to even *switch off* some access points.

2 Related Work

While the contributions on planning mobile telecommunication networks are numerous, they are only partly transferable to wireless LANs, which employ shared medium access schemes such as CSMA/CA. Comparatively few contributions consider the effects of contention that results from these access schemes.

Network planning problems specific to wireless LANs have been formulated for solving the access point placement problem[7] and the channel assignment problem[8]. Joint placement and channel assignment has been proposed, where co-channel overlapping may be allowed[9] or not[10]. In contrast to these contributions on the *planning* of wireless LANs, in [6] we proposed a model for the case where access point locations are already given and the problem is to determine the configuration of transmission power, channel assignment and associations of stations to access points that will minimize contention in the given network.

Previous work on the online reconfiguration of access points mainly focuses on transmit power control and load sharing in single administrative domains [11,12]. [5] suggest the use of a radio resource broker that controls contention between domains by assigning the channels and transmission powers that each domain may use. While being the most closely related work to ours, this proposal relies on a central component assuming a rather low number of different domains, i.e. it is not suited for a wireless community network.

3 Modeling of the Minimal Contention Problem

In this section we provide a very brief overview of our mathematical programming formulation of the minimal contention problem for CSMA/CA-based wireless networks. The optimization model takes as parameters the locations and radio configurations of a set of access points (APs) and stations (STAs), and a matrix of propagation losses between each pair of wireless nodes. The model is agnostic of the radio propagation model, so any analytical or empirical model may be used to instantiate the loss matrix. The model allows to determine the configuration of transmission power, channel assignment, and station association that will minimize the amount of contention in a scenario both for basic CSMA/CA and RTS/CTS modes. Due to space restrictions, we have to refer the reader to [6] for details on the model and some of its useful further extensions.

Let i denote a wireless node with $i = 1, \ldots, I + K$, where I is the number of APs in the scenario and K the number of STAs. Nodes shall be ordered such that $i = 1, \ldots, I$ for APs and $i = I + 1, \ldots, I + K$ for STAs. Each node i can transmit with a transmission power $x_i \in [0, ..., s_i]$, where s_i is the maximum allowed power of node i and $x_i \in \mathbb{R}$. On the way from a sender i to a receiver m, a signal experiences a path loss given by p_{im}[1]. A receiving node requires a minimum signal strength r_m to be able to decode a frame transmitted at the desired data rate correctly. If a node i receives a signal from another node with a power above or equal to l_i, its CCA will report the channel as busy.

APs and their associated STAs form basic service sets (BSS). A BSS can operate on one of J different non-overlapping radio channels, $j = 1, \ldots, J$. y_{ij} is a binary variable indicating whether node i currently uses channel j or not. We further define a binary variable f_{im} indicating whether a node i (which must be a STA) is currently associated to node m (an AP) and a helper variable e^{pc}_{im} which indicates whether node i is a potential contender of node m. With potential contender we mean that node m is close enough to i that it can detect i's carrier if both are operating on the same channel.

A valid solution of our optimization problem needs to satisfy several constraints. First of all, each node's transmission power must be between zero and the node-specific maximum:

$$0 \leq x_i \leq s_i, \quad i = 1, \ldots, I + K \tag{1}$$

All STAs have to receive their minimum power requirement from the AP they are associated to:

$$x_i + p_{im} \geq f_{im} r_m, \quad i = 1, \ldots, I, \; m = I + 1, \ldots, I + K \tag{2}$$

Likewise, all APs have to receive their minimum power requirement from the STAs in their BSS:

$$x_m + p_{mi} \geq f_{im} r_i, \quad i = 1, \ldots, I, \; m = I + 1, \ldots, I + K \tag{3}$$

---

[1] Note that we assume dBm as the unit of signal strength. Due to its logarithmic scale, losses (negative values) in dB are actually added to the transmission power to calculate the received signal strength.

All STAs are associated to exactly one AP:

$$\sum_{i=1}^{I} f_{im} = 1, \quad m = I+1, \ldots, I+K \tag{4}$$

Each AP and STA uses exactly one channel:

$$\sum_{j=1}^{J} y_{ij} = 1, \quad i = 1, \ldots, I+K \tag{5}$$

All STAs use the channel of the AP which they are associated to:

$$y_{ij} - y_{mj} - (1 - f_{im}) \leq 0, \tag{6}$$

$$i = 1, \ldots, I, \ m = I+1, \ldots, I+K, \ J = 1, \ldots, J$$

Finally, we force e^{pc}_{im} to be 1 if nodes i and m are so close to each other, that m detects the channel busy if i currently transmits on the same channel (for $i \neq m$, of course, since nodes cannot contend for access with themselves):

$$x_i + p_{im} \leq l_m + e^{pc}_{im} M_{im}, \quad M_{im} = s_i + p_{im} - l_m \tag{7}$$

$$i = 1, \ldots I+K, \ m = 1, \ldots, I+K \ \wedge \ i \neq m$$

$$e^{pc}_{ii} = 0, \quad i = 1, \ldots, I+K \tag{8}$$

Considering that a node can only contend for access with another node when both are on the same channel, we are able to calculate a_m, the number of nodes contending for access with node m:

$$a_m = \sum_{i=1}^{I+K} e^{pc}_{im} \left(\sum_{j=1}^{J} y_{ij} y_{mj} \right) \tag{9}$$

The objective function that minimizes contention in a CSMA/CA network in basic mode (i.e. without RTS/CTS) can then be stated as:

$$\min \sum_{m=1}^{I+K} a_m = \min \sum_{m=1}^{I+K} \sum_{i=1}^{I+K} e^{pc}_{im} \left(\sum_{j=1}^{J} y_{ij} y_{mj} \right) \tag{10}$$

For a CSMA/CA network in RTS/CTS mode, we furthermore have to take into consideration the indirect contention between wireless nodes. We call a node i an indirect contender of m, if there exists at least one node k that can hear i's RTS frames and whose CTS replies m can hear. In order not to count a node twice, we further require an indirect contender not to be a direct contender at the same time. To indicate that a node is *not* potential direct contender of another node, we need to define a new helper decision variable e^{npc}_{im}:

$$x_i + p_{im} \geq l_m - e^{npc}_{im} M_{im}, \quad M_{im} = l_m - p_{im} \tag{11}$$

$$i = 1, \ldots I+K, \ m = 1, \ldots, I+K \ \wedge \ i \neq m$$

$$e_{ii}^{npc} = 1, \quad i = 1, \dots, I + K \tag{12}$$

We can now extend $a_m$with the number of indirect contenders, but have to take into consideration that APs only send to STAs but not to other APs and vice versa. Furthermore, an AP that does not have STAs assigned should not be counted as an indirect contender. On the other hand, if it has STAs, it should be counted exactly once, no matter how many STAs are assigned to it. This is why we introduce the step function $\sigma(x)$ with $\sigma(x) = 1$ if $x > 0$ and 0 otherwise. Our objective function thus becomes:

$$\min \sum_{m=1}^{I+K} a_m,$$

$$\begin{aligned} a_m = {} & \sum_{i=1}^{I+K} e_{im}^{pc} \left(\sum_{j=1}^{J} y_{ij} y_{mj} \right) + \sum_{k=I+1}^{I+K} \sum_{i=1}^{I} f_{ik} e_{ki}^{pc} e_{im}^{pc} e_{km}^{npc} \left(\sum_{j=1}^{J} y_{ij} y_{kj} y_{mj} \right) \\ & + \sum_{i=1}^{I} \sigma \left(\sum_{k=I+1}^{I+K} f_{ik} e_{ik}^{pc} e_{km}^{pc} e_{im}^{npc} \left(\sum_{j=1}^{J} y_{ij} y_{kj} y_{mj} \right) \right) \end{aligned} \tag{13}$$

The polynomial structure of the presented optimization model make this problem difficult to solve exactly. We have, however, been able to transform this problem into an equivalent linear formulation, which allows us to solve small problem instances with any mixed integer program solver. Due to space restrictions we again refer the reader to [6] for further details. In the same paper we also describe a genetic algorithm heuristic which due to its custom tailored design allows to find near-optimal solutions for comparatively large scenarios (s.a. 200 APs and 400 STAs).

4 Distributed Coordination Algorithm

In this section we describe our distributed algorithm for reducing the contention in a wireless access network. It consists of five building blocks:

- Data dissemination, in which each AP gains knowledge about other APs within its horizon as well as the STAs which these APs are aware of and are able to cover at the required signal strength.
- Local negotiation, in which an AP suggests a local reconfiguration of the network to all APs within its horizon, waits for their feedback on how this reconfiguration would affect network performance in their vicinity and then decides either to commit or abandon this reconfiguration.
- A fitness function with which to evaluate the current state of the network within an APs horizon and the effect of a proposed reconfiguration.
- An algorithm used to find local reconfigurations.

– A mechanism to determine, which APs are allowed to propose local reconfigurations and when.

An AP's *horizon* defines which other APs and STAs in its geographical vicinity it knows and considers in finding improvements. When choosing the extent of the horizon, one has to make the typical trade-off between the chances for finding the globally optimal configuration and the computational effort and signaling overhead. In our experiments we have defined the horizon of an AP i as the set of all APs that are either within contention range of AP i themselves or are able to serve a STA that is in contention range of i.

4.1 Data Dissemination

APs initially find out about their neighbors by scanning for periodic beacon signals on all available channels. Upon receiving a beacon from a previously unknown neighbor, the AP sends out a WELCOME message to its new neighbor, both on the wireless link and on the wired backbone network. This assumes that the IP address of the new neighbor is known. The most simple solution is to let each AP include its IP address as an additional Management Frame Information Element in its broadcasted beacons.

Both the WELCOME message and the reply to it (WELCOME_ACK) contain information about the sending AP and about all STAs which the sender is currently aware of and whose minimum signal strength requirements it can meet. By sending these messages over both the wireless link and the backbone, we can further gain information about whether the wireless link is asymmetric or not, i.e. if one access point is able to hear the other but not vice versa.

Furthermore, all active APs periodically send UPDATE messages to all APs within their horizon containing their current STA information list. This information has an explicit expiration time, so if an AP does not receive UPDATE messages from a neighbor for a certain length of time, it will assume it has deactivated without signing off. UPDATE messages are always sent via the wired backbone, so that this soft-state approach does not consume valuable wireless resources.

We also consider the case that two APs that cannot hear each other directly nevertheless produce contention in each other's BSS. This may happen when an STA is located in between the AP it is associated to and another AP that is within contention range. The STA may then notify its own AP of the contending AP's presence so that both APs may contact each other using the mechanism described above.

4.2 Local Negotiation

Based on its knowledge about APs and STAs within its horizon, an AP may run a local optimization algorithm to search for better configurations for itself and its neighboring APs. If an AP finds a configuration that will improve contention within its own horizon, it suggests the new configuration to its neighbors by sending them an OFFER message with the new configuration.

Upon receiving an OFFER, every neighbor determines the effect of the configuration change on their part of the network. Note that the sets of nodes within the horizons of the APs sending the OFFER and receiving the offer will usually not be identical, although the intersection set should usually be large. All receivers of an OFFER then answer with an OFFER_REPLY message containing the change in contention that would result from actually committing the configuration change. If the net effect of the reconfiguration proposal is positive, the initiating AP sends a COMMIT message to all neighbors, who then update the local knowledge about their neighborhood and possibly change the radio channel they operate on or instruct individual STAs to reassociate with a different AP.

There are three cases in which the initiating AP will send a WITHDRAW message to its neighbors in order to cancel a reconfiguration attempt. The first case is that the initiator calculates a negative or zero net effect of the reconfiguration proposal. Secondly, it may happen that one of the receivers of an OFFER message is already processing a reconfiguration proposal by a different AP which has not been committed or rejected yet. It then refuses the new OFFER by answering with a BUSY message. Finally, if at least one of the neighbors does not respond to the OFFER within a certain time interval, the initiator will assume the message was lost or the receiver has deactivated.

4.3 Reconfiguration Algorithms

In order to find a reconfiguration that will yield a lower amount of contention, an AP applies an optimization algorithm to the set of APs and STAs within its horizon, including itself. We have experimented both with a problem-specific genetic algorithm (please again refer to [6]) and a greedy heuristic which we termed “balance or conquer”. An AP using this heuristic will choose one of the four following actions, depending on which action will have the most positive effect on contention within its horizon:

1. Try to transfer STAs to (from) other APs such that the number of STAs per channel (not per AP!) is roughly the same within the horizon (= balance). Change your own channel, if necessary.
2. Find another AP whose stations you can cover completely and take them all (= conquer), effectively switching the other AP off.
3. Try transferring all stations to other APs, balancing the number of STAs per channel, effectively switching yourself off.
4. If currently switched off, try to take over STAs (starting with the nearest one) from other APs, as long as this does not increase contention. Change your channel, if necessary.

In our experiments the APs used their collected information to instantiate the model in Section 3 and compute the best action.

4.4 Coordination of Reconfigurations

The last building block of our algorithm is concerned with the question *when* APs attempt to find and propose an improved configuration. We have used both an uncoordinated approach, in which each AP performs reconfiguration attempts as a Poisson process. Furthermore, we have used two token-passing algorithms, where an AP currently holding a token waits for a random time interval before attempting to propose a reconfiguration and passing the token on to a randomly chosen neighboring AP. The two token-based approaches differ in that the first approach starts with a single token that circulates the network, while in the second all APs initially hold a token. When an AP receives a new token from a neighbor while already holding one, the new token is destroyed, so that eventually only one token remains in the network. Lost or destroyed tokens could be replaced by letting each AP generate a new token at a very small rate, which could vary with the amount of contention—and therefore the necessity for a new token—within an AP's horizon.

The rationale behind experimenting with different reconfiguration coordination approaches is that one can expect the global level of contention in the system to decrease more rapidly when a high number of access points concurrently try to find and propose reconfigurations, as is the case with the uncoordinated approach. On the other hand, when reconfigurations are made at different locations of the network at the same time, there is a chance that the effect of one reconfiguration will be counterproductive with respect to another reconfiguration in the long run.

5 Experiments and Results

All experiments were conducted in scenarios with 50 APs and 100 STAs within a 1km by 1km simulation area. In a first step, 16 of the APs were placed to regularly cover the simulation area. Afterwards, the remaining APs were placed uniformly over the simulation area. The location of each STA was chosen by picking an AP randomly and then placing the STA within a distance of 10% to 90% of the radio range of the AP, drawn from a uniform distribution.

We then calculated the path losses between each pair of nodes based on the empirical indoor propagation loss model recommended in ITU-R P.1238-2 [13]. The maximum transmission power s_i for each node was set to 20dBm (or 100mW), which is the maximum power allowed for IEEE 802.11b wireless LANs in Europe. We have set l_i, the minimum signal strength to detect a busy medium, and r_i, the minimum signal strength requirement of a node to -84dBm and -82dBm, respectively, as these are typical values for an Orinoco Gold IEEE 802.11b adapter.

At the start of a simulation run, all APs choose an unused channel or pick one randomly if all are already occupied. All STAs associate with the AP offering the strongest radio signal, as typical for wireless LANs.

Simulations run for a duration of 1 hour of simulation time, each and every simulation run is repeated 10 times with different scenarios.

If no tokens are passed in the network, the generation of reconfiguration attempts per AP is a Poisson process with rate 1. If one or more tokens are present, the holding time of a token is exponentially distributed with mean 1s.

The objective of our first experiment has been to find out how well our distributed algorithm manages to reduce the contention in the network under study. We have therefore run our algorithm on 10 different wireless network scenarios with both the genetic algorithm (GA) as heuristic for finding local reconfiguration potential as well as the balance-or-conquer (B|C) heuristic. In order to study the effect of concurrent reconfigurations versus sequential reconfigurations, we also combined each of our three different reconfiguration coordination approaches with both algorithms: Uncoordinated reconfiguration (0 tokens), token-passing with 1 token, and *N* initial tokens for each of *N* access points. Additionally, we have applied a run over 100,000 iterations of our genetic algorithm to serve as an estimate for the best-case behavior and we use standard WLAN as a reference. The resulting average contention values (both absolute and relative decrease compared to WLAN) and their standard deviation are shown in Table 1.

Table 1. Comparison of contention levels achieved by the distributed algorithm using GA and B|C

	WLAN	GA	Local GA			Local B|C		
initial tokens			0	1	*N*	0	1	*N*
mean	512.5 (0.0%)	374.5 (-26.4%)	409.8 (-19.8%)	411.5 (-19.3%)	413.5 (-18.9%)	454.0 (-11.1%)	416.8 (-18.2%)	425.3 (-16.6%)
std. error	18.8	7.2	13.6	11.9	10.9	15.9	10.6	11.6

Figure 1 additionally shows the development of the amount of contention over time for one of the simulated scenarios. As the global GA does not necessarily find the global minimum and the true minimum cannot be determined, we have additionally included a (loose) theoretical lower bound (TLB), which we derived in [14].

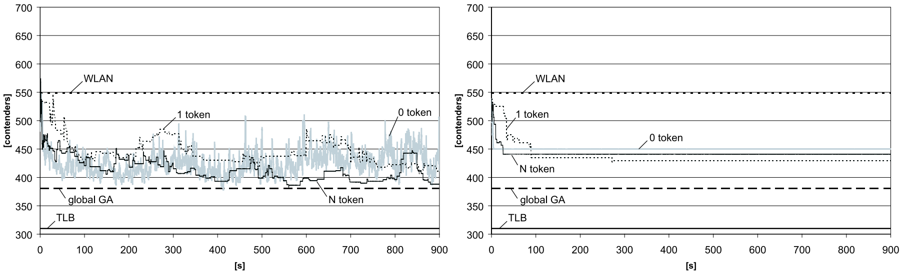

Fig. 1. Performance of GA (left) and B|C (right) as local reconfiguration algorithms compared to global minimum and WLAN

In our simulations, the GA version of our distributed algorithm managed to realize on the average 65.5% of the improvement potential compared to WLAN, the B|C version 61.8%, both for the 1 token case. This corresponds to a decrease in network-wide contention by 19.3% and 18.2%, respectively. Both versions switched off a significant number of APs to achieve this result (12.2% and 13.6%, respectively). Interestingly, this fact seems to contradict previous findings (e.g. [15]) that load balancing between APs leads to optimal allocations. Note, however, that this is only true if there is no interference between BSSes, i.e. when they are separated spatially or by operating on different channels[14], which is uncommon in highly dense scenarios such as wireless community networks.

Although both versions achieve comparable results, this does not mean that both versions are equally suitable for real-world application. The computational effort per search for a better local reconfiguration is on the order of two magnitudes higher for the genetic algorithm than for B|C, while only achieving slightly better results. Furthermore, the stability of the contention levels is not the same between the two versions as can be directly seen from Fig.1 as well.

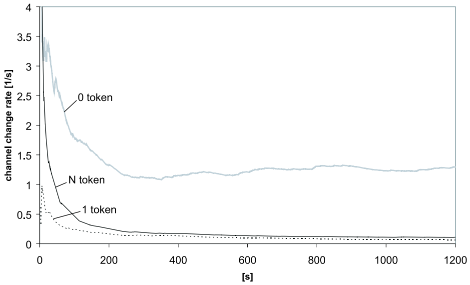

Fig. 2. Channel change rate of GA as local reconfiguration algorithm

We have also observed that the choice of the reconfiguration coordination mechanism has a strong effect on the speed of the improvements in contention, but also on the quality of the attained contention level. Using no coordination between reconfiguration attempts of different APs leads to very quick improvements compared to the 1 token approach. Interestingly, though, in almost all cases the B|C heuristic was able to converge to lower contention levels the slower the rate of reconfigurations was. The *N* token case was usually somewhere in between, reacting as the uncoordinated case when a large number of tokens was still present, and with time converging to the behavior of the 1 token case as more and more tokens are destroyed. Figure 2 shows the channel changes per second (as a total over the whole network) for the local GA algorithm and the 0, 1, and N token cases, which again supports the aforementioned observations.

Finally, we wanted to find out how important the local negotiation part is for our distributed algorithm. We therefore performed a set of experiments in which we removed the negotiation process, so that an AP finding a better configuration immediately commits the necessary changes instead of sending offers to all other APs within its horizon asking for feedback. The results are shown

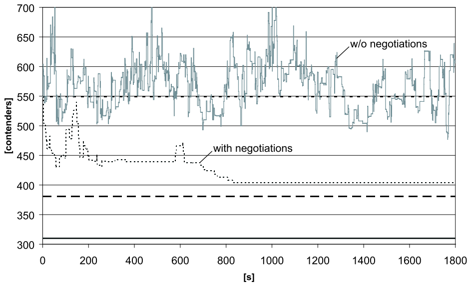

Fig. 3. Comparison of algorithm performance with and without negotiations

in Fig.3 for a single scenario. Indeed, when an AP does not ask its neighbors for possible negative effects of a configuration change, it frequently happens that an AP reconfigures to gain a small improvement, but that this reconfiguration has strong negative effects on the network just outside its horizon. As a consequence, the contention levels heavily fluctuate and may on the average even be higher than with plain WLAN.

6 Conclusions and Outlook

The problem of contention between wireless LANs consisting of a large number of different administrative domains—a common situation in wireless community networks—necessitates some form of self-coordination. In this paper we have taken a first step at tackling the problem of minimizing contention in decentralized wireless community networks, an issue which until now has not received much attention in the literature, but poses a real practical problem to the deployment of emerging large-scale WLANs.

We have proposed a distributed algorithm and protocol for self-coordination of wireless access points from different administrative domains based solely on knowledge about the immediate neighborhood. Experimental results have shown that our distributed algorithm is capable of exploiting 61.8% of the potential for reducing network contention over WLAN, compared to what could be achieved with perfect knowledge. We have also shown that in dense deployments with only few available channels it may be necessary to switch off some APs to reduce contention, rather than performing load-balancing between them. Furthermore, we have found that performing local reconfigurations without feedback from neighboring access points may lead to heavily fluctuating levels of contention which may even be higher than in plain WLAN.

For future work, we perceive the development of even more effective reconfiguration and/or coordination schemes as a short-term goal. We would also like to relax the implicit assumption of cooperative access points towards non-cooperative environments. Currently, we are implementing the presented framework on a set of 4G Access Cubes manufactured by 4G Systems Ltd. in order to be able to investigate its feasibility and scalability in a real-world environment.

References

1. NYCwireless (2006) `http://www.nycwireless.net`.
2. Akella, A., Judd, G., Seshan, S., Steenkiste, P.: Self-Management in Chaotic Wireless Deployments. In: 11th International Conference on Mobile Computing and Networking (MOBICOM '05), Cologne, Germany (2005)
3. WS5100 Wireless Switch Reviewer's Guide. Product brochure, Symbol Technologies (2005) `ftp://symstore.longisland.com/Symstore/pdf/wireless/WS5100ReviewersGuide.pdf` (last access: 2006-01-01).
4. AutoCell—The Self-Organizing WLAN. White paper, Propagate Networks (2003) `http://www.propagatenet.com/resources/docs/whitepaper_autocell.pdf` (last access: 2006-01-01).
5. Matsunaga, Y., Katz, R.: Inter-Domain Radio Resource Management for Wireless LANs. In: IEEE Wireless Communications and Networking Conference (WCNC 2004), Atlanta, Georgia, USA (2004) 2183–2188
6. Zdarsky, F.A., Martinovic, I., Schmitt, J.B.: On Lower Bounds for MAC Layer Contention in CSMA/CA-Based Wireless Networks. In: 3rd ACM/SIGMOBILE International Workshop on Foundations of Mobile Computing (DIALM-POMC'05), Cologne, Germany (2005) 8–16
7. Amaldi, E., Capone, A., Cesana, M., Malucelli, F.: Optimizing WLAN Radio Coverage. In: IEEE International Conference on Communications (ICC 2004), Paris, France (2004) 180–184
8. Leung, K., Kim, B.J.: Frequency Assignment for IEEE 802.11 Wireless Networks. In: 58th IEEE Vehicular Technology Conference (VTC 2003 Fall), IEEE (2003) 1422–1426
9. Ling, X., Yeung, K.: Joint Access Point Placement and Channel Assignment for 802.11 Wireless LANs. In: IEEE Wireless Communications and Networking Conference (WCNC 2005). (2005)
10. Lee, Y., Kim, K., Choi, Y.: Optimization of AP Placement and Channel Assignment in Wireless LANs. In: IEEE Conference on Local Computer Networks (LCN 2002). (2002)
11. Hills, A., Friday, B.: Radio Resource Management in Wireless LANs. IEEE Communications Magazine **42**(10) (2004) 9–14
12. Wang, Y., Cuthbert, L., Bigham, J.: Intelligent Radio Resource Management for IEEE 802.11 WLAN. In: IEEE Wireless Communications and Networking Conference (WCNC 2004), Atlanta, Gergia USA (2004) 1365–1370
13. ITU-R P.1238-2: Propagation data and prediction methods for the planning of radio communication systems and radio local area networks in the frequency range of 900 MHz to 100 GHz (2001)
14. Zdarsky, F.A., Martinovic, I., Schmitt, J.B.: Self-Coordination Mechanisms for Wireless Community Networks. Technical Report 339/05, University of Kaiserslautern, Germany (2005)
15. Kumar, A., Kumar, V.: Optimal Association of Stations and APs in an IEEE 802.11 WLAN. In: Proceedings of the National Conference on Communications (NCC), Kharagpur, India (2005) 1–5