Self-Coordination Mechanisms for Wireless Community Networks

Frank A. Zdarsky, Ivan Martinovic, and Jens B. Schmitt
disco | Distributed Computer Systems Lab
University of Kaiserslautern, Germany

Technical Report No. 339/05

July 2005 (revised Jan. 2006)

Abstract

Co-channel interference and contention at shared medium access may significantly reduce the utilization of scarce radio frequency resources and degrade the performance of a CSMA/CA-based wireless LAN. While both phenomena may be controlled within a single administrative domain by choosing appropriate access point installation sites and assigning operating channels intelligently, there is usually 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* (based on wireless LAN technology), as each access point is operated by a different owner and can be viewed as a separate domain. Further, the situation is aggravated by the fact that wireless community networks are not pre-planned but grow wildly and evolutionary. Until recently, the problem of inter-domain contention in wireless LANs has received little attention. In this paper, we give a mathematical programming formulation of the minimum inter-domain contention problem and present two theoretical lower bounds on contention. A specifically-tuned genetic algorithm is then introduced that can be used to find near-optimal solutions in larger scenarios and serves us as benchmark for the main contribution of this paper: a *distributed algorithm and protocol for self-coordination of access points* from different domains based solely on knowledge about the immediate neighborhood.

1 Introduction

1.1 Background and Motivation

The emergence of wireless community networks as for example NYCwireless[2] is a remarkable and growing phenomenon and provides the major motivation for our work on self-coordination mechanisms in wireless networks (although it may be applicable to other scenarios as well). Wireless community 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. The growth of such networks is considerable, with e.g. about 150 access point in the NYCwireless. This success is fueled by the low cost of wireless LAN technology and by its relatively easy usage.

Unlike mobile telecommunication networks, wireless LANs (WLANs) are typically deployed on a small scale and in a rather ad-hoc manner. Public or private organizations intending to cover their premises with WLAN access often merely rely on expert knowledge and common best practices when planning their networks or use one of the commercially available WLAN planning tools that allow a user to determine access point (AP) installation sites with comparatively low effort for in-situ measurements. Most single AP installations in private homes (which are the basis for a wireless community network) are not planned at all.

Improperly located APs and a bad frequency assignment frequently result in co-channel interference and contention due to the CSMA/CA random access protocol of WLANs. Especially in areas with a high density of APs from different administrative domains, e.g. student dormitories, multi-tenant office buildings etc., the utilization of scarce frequency resources and the performance of the wireless LAN access network is usually low.

Interfering APs under different administrative control are hard to circumvent, especially since the number of available non-overlapping channels is low (typically 3 in 802.11b/g, 12 in 802.11a, depending on regulations). The problem might even become worse, if the spatial density of wireless LANs increases (as it does for wireless community networks), if more vendors adopt proprietary channel bonding techniques[23] to increase the throughput of their products, or if more products that are not 802.11-friendly use the license exempt ISM and U-NII frequency bands. Furthermore, although radio channels in 802.11a are non-overlapping, receivers of many cheaper wireless LAN adapters cannot cleanly filter out single channels. As a result they experience interference from adjacent channels as well.

A solution to all these problem is to introduce coordination mechanisms between APs of different administrative domains. While products such as wireless switches[4] and self-configuring APs[3] are a step towards radio resource management inside single administrative domains, the problem of inter-domain contention has only recently started to attract the attention of the scientific community[21]. In a wireless community network a further step towards the *self*-coordination of a very large number of domains needs to be taken.

Following up on some of our initial work on the problem of minimal inter-domain contention in wireless LANs[27], in this paper we present a distributed algorithm and protocol for the self-coordination of APs from a potentially large number of different administrative domains. We have taken a distributed, but local approach using only regional knowledge, as this lends more naturally to the problem of self-coordination between a large number of different domains. Our algorithm currently assumes cooperative behavior between APs participating in the protocol. It is therefore well suited for the use in wireless community networks, where this cooperation is a precondition anyway and in which the owners of APs often cannot know all other owners on a personal level but must rely on the self-organization of the wireless network.

1.2 Related Work

Substantial research contributions exist on the planning of mobile telecommunication networks. One aspect of planning in this context is the selection of installation sites from a set of available candidate sites (e.g. [16, 9]). This can be combined with the configuration of base stations, e.g. choosing antenna types and orientation, and transmission power [11]. Often, the placement problem has multiple, competing objectives, such as maximizing coverage, maximizing capacity, and minimizing installation cost. Channel assignment is another important planning aspect which has been studied both for fixed and dynamic assignments (e.g. [14, 15, 18]). Different heuristics such as simulated annealing [11], genetic algorithms [9], and tabu search [16] have been used for both of these aspects.

Partially, the work on cellular networks can be applied in the context of single-domain wireless LANs as well. For example, [12] formulated a coverage planning problem for outdoor wireless LANs, but did not consider any peculiarities of wireless LANs, such as carrier sensing or contention, so that their results can be applied in other radio networks as well.

In contrast to this, [5] investigated the WLAN planning problem accounting for the effect of contention introduced by the CSMA/CA mechanism. They give 0-1 hyperbolic formulations and quadratic formulations for the problem of maximizing overall capacity with and without covering constraints and for maximizing fairness with respect to capacity. They only consider a single-channel scenario, but claim that their proposed formulation can be easily generalized to multiple channels.

[19] formulated a channel assignment problem for CSMA/CA-based networks, considering the cumulative co-channel interference from neighboring APs leading to a busy carrier sense signal. Their objective is to minimize the maximum channel utilization experienced by an AP. The authors then proved this problem to be NP-complete and proposed a heuristic, which they applied to two scenarios with known optimal frequency assignments (hexagonally shaped lattice of cells) and uniform, fixed-power sectorized antennas.

[17] provided an integer linear programming formulation, which determines a placement of APs and a channel assignment that maximizes channel utilization in a single step. However in their formulation, APs within interference range have to always operate on different channels, which makes the problem unsolvable for scenarios with many nearby APs and only few available non-overlapping channels.

Finally, [20] proposed a method for joint AP placement and channel assignment which permits co-channel overlapping and aims at maximizing throughput and fairness among stations.

As we are interested in analyzing already deployed wireless LANs from different domains, we do not consider AP placement and we also do not expect to be able to influence the hardware configuration of APs. Instead, we focus on the dynamically adjustable aspects which affect contention: transmission power, channel selection and assignments of STAs to APs. Our objective is to minimize contention experienced by APs *and* STAs by taking into account both direct contention via CSMA/CA's physical carrier sense as well as the virtual carrier sense of the RTS/CTS extension. Note that as a result of transmission power assignment an AP can be switched off, so that we also have some form of selection from candidate sites, but it is not an objective to keep the number of active sites small. Finally, we do not make any assumptions about the size, shape or overlap of co-channel radio cells, as we expect all kinds of heterogeneity to occur in our scenario under study and typically not the traditional hexagonal lattice.

There has been some work on on-line radio resource management for wireless LANs which is complementary to our work in that our global optimization algorithm can be used as a benchmark for determining the effectiveness of the proposed schemes in reducing contention inside a domain or between domains:

[10] described an architecture in which intelligent switches control APs within a single administrative domain to provide dynamic channel assignment, dynamic transmit power control and load sharing.

[24] proposed an agent-based radio resource management system in which the APs belonging to the same network cooperate with each other to provide full coverage for present STAs and perform load balancing between them.

[21] suggest the use of a radio resource broker that monitors traffic in the connected wireless LANs of different domains as well as the interference between these domains and then compensates networks with high traffic but much interference from other networks by assigning them more channels and transmission power, which it takes away from other domains. 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.

1.3 Contributions and Paper Structure

In our paper we pursue two questions: How much and to which level contention in an unplanned network of wireless APs can be reduced by introducing coordination between these APs and how close a distributed algorithm for self-coordination can get to this level. To this end we:

- propose mathematical programs for jointly determining the AP–STA associations as well as the transmission power and channel assignment parameters for all nodes of a CSMA/CA-based wireless LAN scenario that minimizes the amount of contention in the system both for operation with and without RTS/CTS extension (Sections 2.1 and 2.2, respectively),

- present two theoretical lower bounds for the minimal contention problem that exploit different levels of knowledge about the scenario (Section 3.1),
- introduce a genetic algorithm which is specifically tuned for finding near-optimal solutions also in larger scearios (Section 3.3),
- describe a distributed algorithm and protocol that allows APs to reconfigure the network within their neighborhood in order to reduce contention (Section 4), and
- present some of our experimental results (Section 5).

Finally, we conclude our paper with a short summary and an outlook.

2 Modeling the Minimal Contention Problem

In this section we provide mathematical programming formulations of the basic problem we are addressing: the minimization of contention in CSMA/CA-based wireless networks. We distinguish two cases: a simpler model under the assumption of low traffic load and a more sophisticated model under high traffic load which integrates the RTS/CTS mechanism.

2.1 Networks with Low Traffic Loads

Before a wireless station using CSMA/CA can start to transmit data, it needs to sense an idle channel for a specified amount of time (Distributed Inter Frame Spacing or DIFS in 802.11). Whether a channel is idle or not is determined by a Clear Channel Assessment (CCA) function of the physical layer. Depending on the implementation and the chosen operation mode, the CCA would for example indicate a busy channel when a certain energy detection threshold is exceeded (CCA Mode 1), when a valid signal from another station is detected (CCA Mode 4), or a combination of both (CCA Mode 5)[25]. In this paper we assume that physical carrier sense is solely based on detection of valid signals from other stations. The reason for this is that the default energy detection threshold is usually much higher than the signal level at which transmissions from a single stations can be detected. Only in the rare case that a station receives simultaneous transmissions from multiple co-channel stations (i.e. when these stations sense an idle channel both with physical and virtual carrier sense) would CCA Mode 1 detect a busy medium when CCA Mode 4 does not. Furthermore, the energy detection threshold is usually only adjustable in higher-priced equipment.

The signal strength above which a station is able to detect valid transmissions from other stations is typically much lower than the signal strength required for receiving transmissions at a desired data rate. Thus, a station which is farther away from a sending station than the intended receiver might still be restrained from sending to any other station, even though its transmission might be unproblematic.

As a first step, we will model a scenario with wireless access points and stations that use only simple CSMA/CA. Later we will extend the model for RTS/CTS operation.

Let i denote a wireless node with $i = 1, \ldots, I + K$, where I is the number of access points (APs) in the scenario and K the number of stations (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 $x_i \in \mathbb{R}$ between 0 and a node-specific maximum allowed power s_i. 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 a basic service set (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_{im}^{pc} 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. In summary, our first model takes as input

- s_i: the maximum transmission power of node i
  $s_i \in \mathbb{R},\ i = 1, \ldots, I + K$
- r_i: the minimum reception power requirement of node i
  $r_i \in \mathbb{R},\ i = 1, \ldots, I + K$
- l_i: the minimum signal power for node i to detect the channel as busy
  $l_i \in \mathbb{R},\ i = 1, \ldots, I + K$
- p_{im}: the signal propagation loss from node i to node m
  $p_i \in \mathbb{R},\ i = 1, \ldots, I + K,\ m = 1, \ldots, I + K$

and the following decision variables:

- x_i: the current transmission power of node i,
  $x_i \in \mathbb{R},\ i = 1, \ldots, I + K$
- $y_{ij} = \begin{cases} 1 & \text{iff node } i \text{ is set to channel } j \\ 0 & \text{otherwise} \end{cases}$
  $y_{ij} \in \{0, 1\},\ i = 1, \ldots, I + K,\ j = 1, \ldots, J$
- $f_{im} = \begin{cases} 1 & \text{iff AP } i \text{ is responsible for STA } m \\ 0 & \text{otherwise} \end{cases}$
  $f_{im} \in \{0, 1\},\ i = 1, \ldots, I,\ m = I + 1, \ldots, I + K$

[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.

- $e_{im}^{pc} = \begin{cases} 1 & \text{iff node } i \text{ is potential contender of node } m \\ 0 & \text{otherwise} \end{cases}$
  $e_{im}^{pc} \in \{0, 1\}\,,\ i = 1, \ldots, I + K,\ m = 1, \ldots, I + K$

A valid solution of our optimization problem needs to satisfy several constraints, which we will discuss in detail.

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:

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

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_{im}^{pc} 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):

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

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

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

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:

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

Our objective is then to minimize the amount of contention experienced by the nodes in the system:

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

This optimization problem requires $I^2 + K^2 + 3IK + (J+1)(I+K)$ decision variables and $I^2 + K^2 + (J+4)IK + 2I + 3K$ constraints and is unfortunately of multiplicative form, which makes it still difficult to solve. In section 3.2 we will show how to make this problem solvable by transforming it into an equivalent linear problem.

2.2 Networks with High Traffic Loads

When traffic in the wireless network increases, so does the number of collisions of transmission attempts. In wireless networks with high traffic loads, a mechanism called RTS/
CTS, first proposed as part of the MACA protocol [13], is usually employed to increase utilization.

In CSMA/CA with RTS/CTS, when a node i wants to transmit data to a node m, it first sends a small Request To Send (RTS) frame containing the receiver address and the duration of the transmission including the final ACK. Upon receiving the RTS frame, m reponds with a Clear To Send (CTS) frame, which contains the remaining transmission duration as well. All other nodes (APs and STAs) which can hear either the RTS or the CTS store the time during which the medium is expected to be busy in their local network allocation vector (NAV) timer and then defer access until the transmission between i and m is over. Since the specified procedure of deferring access is similar to the physical carrier sense described in the previous section, this mechanism is called virtual carrier sense.

Activating RTS/CTS has the advantage, that collisions can in general only occur on RTS transmissions. As RTS frames are comparatively small, the collision probability is significantly reduced. Furthermore RTS/CTS solves the hidden terminal problem, where two stations that cannot hear each other try to send data to the same access point simultaneously. As a drawback, more stations experience contention indirectly, as they are within carrier sense distance of a node receiving a transmission.

We are now going to extend the previous model for the case of CSMA/CA networks using RTS/CTS. This is simple as the previous model already accounts for calculating the number of direct contenders for a given node m. There, a direct contender was defined as a node which, when it transmits, causes m to defer transmissions due to a positive physical carrier sense indication, which is equivalent to the effect of the virtual carrier sense after reception of a RTS frame. All we have to do further is to take into account those contenders i, which interfere with m's transmissions by being able to send RTS frames to at least one node k whose CTS answers m can hear. We call i an indirect contender of m, if it is not a direct contender at the same time, so that the sets of direct and indirect contenders for a given node are disjoint. To indicate that a node is *not* potential contender of another node, we need to define a new helper decision variable e_{im}^{npc}:

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

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

$$e_{ii}^{npc} = 0, \quad i = 1, \ldots, I+K \tag{12}$$

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 $\sigma(x)$. Our objective function thus becomes:

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

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

$$\sigma(x) = \begin{cases} 1 & x > 0 \\ 0 & x \leq 0 \end{cases}$$

This model extension adds $(I+K)^2$ decision variables and $(I+K)^2$ constraints. Note that e_{ik}^{pc} and e_{ki}^{pc} always have the same value as f_{ik}, since a STA and the AP it is associated to need to be able to hear each other. We can therefore simply omit these variables in the objective function.

3 Lower Bounds for the Minimal Contention Problem

3.1 Theoretical Lower Bounds

In this section, we derive theoretical lower bounds on the minimum contention. The bounds are based on optimistic assumptions about possible contention between APs and STAs, i.e. a best-case analysis for a given number of STAs and APs is performed. In the following two bounds are presented: one independent of the radio range of APs and STAs and then a (better) one which takes into account the radio range of APs and STAs.

3.1.1 Radio-Range Independent Lower Bound

As above, let I denote the number of APs and K the number of STAs. We make the following two optimistic assumptions:

1. APs (and their associated STAs) do not contend with APs (and STAs) of other basic service sets.

2. STAs assigned to a given AP do not contend with each other.

The first assumption requires APs (and their associated STAs) to either be spaced far away enough from each other or to use different channels. The second assumption is optimistic in the spacing between STAs that are associated to the same AP.

Let n_i denote the number of STAs associated to AP i. Under these assumptions the overall contention can be calculated as follows:

$$C = \sum_{i=1}^{I}(2n_i + n_i(n_i - 1)) = \sum_{i=1}^{I} n_i^2 + n_i$$

This is due to the fact that an AP is in direct contention with each of its associated STAs (and the other way around) and that each STA is in indirect contention with each other STA associated to the same AP. This contention is minimal if the STAs are as uniformly distributed over the APs as possible:

Proposition 1: C is minimal if $\forall i, j\ n_i + 1 \geq n_j$.

Proof: Assume C is minimal for a given assignment of STAs to APs but $\exists i_0, j_0$ with $n_{i_0} + 1 < n_{j_0}$. That means $\exists k \geq 2$ with $n_{i_0} + k = n_{j_0}$. Hence, with $A = \sum_{i=1, i\neq i_0, i\neq j_0}^{I} n_i^2 + n_i$

$$\begin{aligned} C &= A + n_{i_0}^2 + (n_{i_0} + k)^2 + n_{i_0} + n_{i_0} + k \\ &= A + 2n_{i_0}^2 + 2n_{i_0}k + k^2 + 2n_{i_0} + k \\ &> A + 2n_{i_0}^2 + 2n_{i_0}k + k^2 - 2(k-1) + 2n_{i_0} + k \\ &= A + (n_{i_0} + 1)^2 + (n_{j_0} - 1)^2 + (n_{i_0} + 1) + (n_{j_0} - 1) \end{aligned}$$

This contradicts the assumption that C is minimal for this distribution, as it can be improved by reassigning STAs to APs and thus the proposition must be correct. ■

We therefore make the further optimistic assumption, that the APs achieve a perfect load balancing with respect to their assigned STAs (modulo 1) to find a lower bound on contention for a given number of APs and STAs.

That means the lower bound is given by

$$C = K + m(n+1) + (I-m)n + mn(n+1) + (I-m)n(n-1) \tag{14}$$

where $n = K\,div\,I$ is the number of STAs per AP (possibly plus one) and $m = k\,mod\,I$ is the number of APs with one STA more than others.

Note that this bound makes very optimistic assumptions on the spatial distribution of nodes and assumes enough channels to prevent contention between basic service sets. Hence, in some actual scenarios it can be a very loose lower bound.

3.1.2 Radio-Range Dependent Lower Bound

For this second bound, some more information besides the number of STAs and APs is required. In particular, now also the feasible associations of STAs to APs due to radio reachability is taken into account. If we have this information we can capture situations where a perfect load balancing is not possible and increase the lower bound accordingly.

Let us define a *radio range constrained load balanced* assignment of STA to APs as follows:

Definition: An assignment of STAs to APs is called *radio range constrained load balanced* iff for all AP i_0 and i_1 with $n_{i_0} > n_{i_1} + 1$ it applies that if a STA $j \in C_{i_0}^{STA} \wedge j \in C_{i_1}^{STA}$ then j must be assigned to AP i_1.

Equipped with this definition the following proposition parallels Proposition 1 from the preceding section:

Proposition 2: The minimal contention under optimistic assumptions is achieved by the radio range constrained load balanced assignment of STAs to APs.

Proof: The proof is a straightforward extension of the one for Proposition 1 taking into account the further restrictions due to radio range constraints. ■

An algorithm to compute the radio range constrained load balanced assignment of STAs to APs, which hence achieves the lower bound on contention (under the optimistic assumptions of no inter-BSS contention and no intra-BSS contention between STAs), is given as Algorithm 1.

3.2 Exact Lower Bounds

The problem presented in Section 2.2 has a polynominal structure, as the terms of the objective function are products of three and more variables. The binary nature of variables allows us to adopt the technique from [7] to derive an equivalent linear model at the cost of additional decision variables and constraints.

Algorithm 1

Computation of Radio Range Constrained Load Balancing

Parameters

I: number of APs

K: number of STAs

C_i^{STA}: set of STAs which can be covered by AP i at its maximum signal strength, i.e.

$C_i^{STA} = \{m \in \{I+1, ..., I+K\} : s_i + p_{im} \geq r_m\}$

C_m^{AP}: set of APs of which each covers STA m at its maximum signal strength, i.e.

$C_m^{AP} = \{i \in \{1, ..., I\} : s_i + p_{im} \geq r_m\}$

Variables

M^{STA}: set of STAs

M^{AP}, M^{EXCL}: sets of APs

n_i: number of STAs associated to AP i

Algorithm

$M^{STA} = \{I+1, ..., I+K\}$;
$M^{EXCL} = \emptyset$;
FOR $i = 1$ TO I: $n_i = 0$;
WHILE $M^{STA} \neq \emptyset$ DO
    $M^{AP} = \{1, ..., I\} - M^{EXCL}$;
    WHILE $M^{STA} \neq \emptyset \wedge M^{AP} \neq \emptyset$ DO
        $i_0 = \arg\min_{i \in M^{AP}} \left|C_i^{STA}\right|$;
        IF $\left|C_{i_0}^{STA}\right| \neq 0$ THEN
            $m_0 = \arg\min_{m \in C_{i_0}^{STA}} \left|C_m^{AP}\right|$;
            $n_{i_0} = n_{i_0} + 1$;
            $M^{STA} = M^{STA} \setminus \{m_0\}$;
            FOR $i = 1$ TO I: $C_i^{STA} = C_i^{STA} \setminus \{m_0\}$
        ELSE
            $M^{EXCL} = M^{EXCL} \cup \{i_0\}$;
        $M^{AP} = M^{AP} \setminus \{i_0\}$;

For every product of binary variables we introduce a new variable and substitute it with a product which is then transformed to a new constraint.

We substitute $e_{im}^{dc} := e_{im}^{pc} y_{ij} y_{mj}$, $e_{im}^{icAS} := f_{ik} e_{km}^{pc} e_{im}^{npc} y_{ij} y_{kj} y_{mj}$, and $e_{im}^{icSA} := f_{ik} e_{im}^{pc} e_{km}^{npc} y_{ij} y_{kj} y_{mj}$ by adding the following variables:

- $e_{im}^{dc} = \begin{cases} 1 & \text{iff node } i \text{ is direct contender of node } m \\ 0 & \text{otherwise} \end{cases}$
  $e_{im}^{dc} \in \{0,1\}, \; i = 1, \ldots, I+K, \; m = 1, \ldots, I+K$
- $e_{im}^{icAS} = \begin{cases} 1 & \text{iff AP } i \text{ is indirect contender of node } m \\ 0 & \text{otherwise} \end{cases}$
  $e_{im}^{icAS} \in \{0,1\}, \; i = 1, \ldots, I, \; m = 1, \ldots, I+K$
- $e_{im}^{icSA} = \begin{cases} 1 & \text{iff STA } i \text{ is indirect contender of node } m \\ 0 & \text{otherwise} \end{cases}$
  $e_{im}^{icSA} \in \{0,1\}, \; i = I+1, \ldots, I+K, \; m = 1, \ldots, I+K$

The products are then added as new constraints:

Force e_{im}^{dc} to be 1 if node i is potential contender of m and both use the same channel

$$e_{im}^{pc} + y_{ij} + y_{mj} - e_{im}^{dc} \leq 2, \tag{15}$$
$$i = 1, \ldots, I+K, \; m = 1, \ldots, I+K, \; j = 1, \ldots, J$$

Force e_{im}^{icAS} to be 1 if AP i sends an RTS to its associated STA k and node m can hear k's CTS, but not the original RTS

$$f_{ik} + e_{km}^{pc} + e_{im}^{npc} + y_{ij} + y_{kj} + y_{mj} - e_{im}^{icAS} \leq 5, \tag{16}$$
$$i = 1, \ldots, I, \; k = I+1, \ldots, I+K,$$
$$m = 1, \ldots, I+K, \; j = 1, \ldots, J$$

Force e_{im}^{icSA} to be 1 if STA i sends RTS to its AP k and node m can hear k's CTS, but not the original RTS:

$$f_{ki} + e_{km}^{pc} + e_{im}^{npc} + y_{ij} + y_{kj} + y_{mj} - e_{im}^{icSA} \leq 5, \tag{17}$$
$$i = I+1, \ldots, I+K, \; k = 1, \ldots, I,$$
$$m = 1, \ldots, I+K, \; j = 1, \ldots, J$$

Finally we obtain our new linear objective function:

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

$$a_m = \sum_{i=1}^{I+K} e_{im}^{dc} + \sum_{i=1}^{I} e_{im}^{icAS} + \sum_{i=I+1}^{I+K} e_{im}^{icSA}$$

This new formulation can now be solved with any mixed integer program solver. For our evaluations, we have used the open-source software `lp_solve`[1]. During our initial testing we found out that we could vastly improve the time that `lp_solve` takes to find the optimal solution, by giving it a hint to use all available channels. We did this by adding the following additional constraints:

$$\sum_{i=1}^{I} y_{ij} \geq 1, \quad j = 1, \ldots, J \tag{18}$$

Note that this hint helped `lp_solve` to more quickly reduce the search space by enabling a better branching, although it might not have the same effect with other solvers that follow a different branch and bound strategy.

3.3 Heuristics for Lower Bounds

As we have only been able to solve small problem instances exactly with `lp_solve`, we decided to implement a genetic algorithm (GA) that is specially tailored to our optimization models and allows us to study large problem instances as well. Our GA repeats the following steps iteratively until the population has converged:

1. Generate a new generation of individuals by recombining randomly chosen pairs of parent individuals.

2. Mutate each gene of an individual with a probability of $p_{mutation}$. Transmission powers $x_i \in \mathbb{R}$ are mutated by adding a random value drawn from a Gaussian distribution with mean 0 and a standard deviation of σ_i to it, where σ_i is adapted during evolution. Radio channels and AP associations are mutated by randomly choosing a new value from the respective set of allowed values.

3. Finally, we use a tournament selection strategy, where randomly chosen pairs of individuals taken from both parent and child generation compete with each other and the fitter individual of each pair (i.e. one with the lower contention) survives until the next round.

Up to now, the algorithm is pretty much standard. However, we have had good experience with equipping our GA with a special crossover operator and a healing strategy.

According to the building-block hypothesis [8], one should arrange the genes on an individual's chromosome in such a way that those genes that are correlated in their influence on an individual's fitness should be placed close to each other, so that it is less likely that the cross-over operator would tear them apart during recombination. We have therefore arranged genes representing a node's transmission power, channel selection and AP assignment on a 2-dimensional plane instead of the traditional 1-dimensional string, and we have done so in such a

		Available Channels			
		1	2	3	4
ILB		12	12	12	12
DLB	all	12	12	12	12
OPT	1	42	20	14	12
	2	28	12	12	12
	3	17	12	12	12
	4	34	14	12	12
	5	19	12	12	12
	6	33	15	12	12
GA	1	42.0	20.0	14.0	12.0
	2	28.4	12.0	12.0	12.0
	3	17.0	12.4	12.0	12.0
	4	34.8	14.0	12.0	12.0
	5	19.0	13.6	12.4	12.0
	6	33.0	15.0	12.4	12.0
MC	1	58.8	40.0	24.2	27.6
	2	44.8	28.6	23.2	26.8
	3	67.0	46.4	24.2	23.6
	4	62.4	29.6	35.8	24.2
	5	62.2	30.4	28.0	23.4
	6	45.4	29.0	23.4	22.2

Table 1: Minimum contention for varying number of available channels in 6 different scenarios of 4 APs and 5 STAs each.

way that the distance relationships between nodes are preserved on the chromosome. Our crossover operator then chooses a random straight cut through the chromosome plane, recombining the cut-off chromosome fragments of the chosen pair of individuals.

Furthermore, in order to improve the chances of obtaining a large amount of valid solutions within our population, we apply a healing strategy after each iteration. The healing process involves two phases. First, it searches for nodes whose minimum signal strength requirements are not met and adapts the sender's transmitting power to the required value, if it does not exceed the maximum allowed power. If this is not successful, the healing process tries to find a better AP to associate to for all STAs in turn.

In preparatory experiments we have verified that the genetic algorithm does indeed find near-optimal or even the optimal solutions. The experiments were performed on 6 different scenarios of 4 APs and 5 STAs each. APs have been placed in locations drawn from a bivariate normal distribution around the center of a 1km x 1km simulation area, with the constraints that they are placed not closer than 20m and not farther than 150m apart from the next AP. For the remaining parameters, please refer to the later section on our main experiments and results (Section 5.1).

In Table 1 we have listed the minimum contention for all 6 scenarios, as calculated by the genetic algorithm (GA) for 1 to 4 available channels, averaged over 5 independent simulation runs each. The table also shows the minimum contention values as calculated by the solver (OPT), the values for the radio range independent and radio range dependent lower bounds (ILB and DLB, respectively) and finally the average results from 5 runs of a single, randomly generated solution (Monte Carlo (MC)), with one application of the healing process to generate valid solutions.

As the results of our experiments show, the theoretical lower bound can be reached in all 6 scenarios if there are 4 available channels. The fact that the lower bound has been reached means that all but one AP have one STA assigned, the other has 2 STAs. Note that this well-balanced case can usually not be reached in larger scenarios. As the number of available channels decreases, it is not possible to avoid contention between basic service sets anymore in some of the scenarios. Note that in most cases, the GA was able to find the optimal solution.

4 Distributed 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 chosing the extent of the horizon, one has to make the typical tradeoff 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. Adhering to the notation from previous sections, the horizon H_i of AP i can be mathematically formulated as:

$$
\begin{aligned}
H_i \quad = \quad & \{m \in \{1,...,I\} \mid s_i * p_{im} \geq l_m \\
& \vee (\exists k \in \{I+1,...,I+K\} : \\
& \; s_i * p_{ik} \geq l_k \wedge s_m * p_{ik} \geq r_k)\}
\end{aligned}
$$

4.1 Data Dissemination

The objective of the first building block of our algorithm is to keep APs updated about other APs within their horizon and all STAs whose minimum signal strength requirement can be met by at least one of the APs within the horizon.

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[26] in its broadcasted beacons. As legacy stations ignore unknown information elements, this solution is backward compatible. Another solution would be to use the Candidate Access Router Discovery (CARD) protocol[6], which is an experimental protocol defined by the IETF Seamoby working group.

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 when 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. As in the previous section we use the amount of contention present within the AP's horizon as the function to minimize, but different fitness function may be used as well. If an AP finds a configuration that will improve contention within its own horizon by a certain positive delta, 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 would have 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 delta in contention that would result from actually committing the configuration change. When the AP that initiated negotiations has received replies from all its neighbors, it calculates the sum of all delta values including its own. 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. In the current version of our algorithm, an AP sends a message to each of its associated STAs to instruct it to change in channels according to the intended new configuration. Alternatively, all associated STAs could be informed by letting APs include a Channel Switch Announcement element (defined in the 802.11h standard) in their management frames.

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 our problem-specific genetic algorithm and a new greedy heuristic which we termed "balance and separate". It works in two phases:

1. In the balancing phase, the heuristic tries to distribute the number of associated STAs to an AP as evenly as possible using the algorithm described in Section 3.1.

2. As mentioned in the discussion of the balancing algorithm, it optimistically assumes that there is no contention between BSSes, either because

they are spaced sufficiently far apart from each other or operate on different channels. In the separation phase the heuristic therefore tries to assign channels in such a way that the two BSSes with the highest amount of inter-BSS contention operate on different channels and that the remaining BSSes are assigned channels in the order of decreasing inter-BSS contention. If it is unavoidable to choose an already assigned channel, the heuristic chooses the one that will add the least amount of contention to the network.

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. However, we have not considered the case of token loss and replacement.

The rationale behind experimenting with different reconfiguration coordination approaches is that one can expect the global level of contention in the system to increase more rapidly when a high number of access points concurrently try to find and propose reconfigurations, as it 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. The token passing approach increases the probability that two subsequent reconfigurations take place on neighboring or at least nearby access points. Finally, to start with a high number of tokens that gradually decreases, might be a compromise between the two former approaches.

5 Experiments and Results

5.1 Scenario Generation and Simulation Setup

Unless otherwise noted, 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

experiment sets	preparatory	evaluation
repetitions	6	10
simulation area	1km^2	1km^2
# of APs / STAs	4 / 5	50 / 100
channels	3	3
s_i	20 dBm	20 dBm
r_i	-82 dBm	-82 dBm
l_i	-84 dBm	-84 dBm
algorithms	OPT, GA	Local GA, B&S
tokens	n/a	0, 1, N

Table 2: Parameters for Simulation Experiments.

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 [22]. 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.

Initially, all nodes in the network are inactive. When a simulation run is started, nodes are activated as a Poisson process with rate 1/30 seconds. An activated node is an AP in 60% of the cases, otherwise an STA. Activated APs immediately start to contact APs in their vicinity. Upon detecting a new neighbor, an AP will provide it with updates on its state every 10 seconds via the backbone. When an STA is activated, it immediately starts scanning for beacon frames which APs broadcast every 50 seconds. After 5 seconds it checks whether it has already received beacons and then either associates with the nearest AP or continues scanning. Simulations run for a duration of 10 hours 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 1/100 seconds. If one or more tokens are present, the holding time of a token is exponentially distributed with mean 100s.

Table 2 once more summarizes the most important parameters used in our experiments in a compact form. For illustrative purposes, we provide a snapshot of an ongoing simulation in Figure 1.

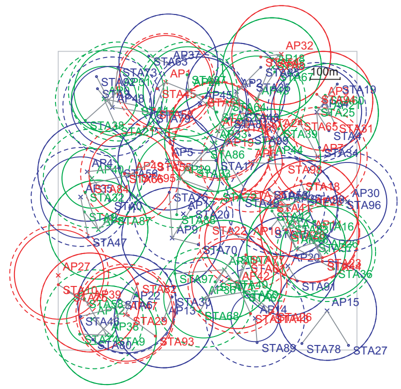

Figure 1: Snapshot of a Simulation Run

	Local GA			Local B&S		
initial tokens	0	1	N	0	1	N
mean	776.8	828.1	779.2	828.5	868.9	835.1
std. dev.	121.8	94.1	107.1	112.5	118.5	133.8
GA mean	551.0					
GA std. dev.	61.9					
MC mean	1237.9					
MC std. dev.	141.2					

Table 3: Comparison of Contention Levels Achieved by the Distributed Algorithm Using GA and B&S as Local Reconfiguration Algorithms.

5.2 Comparison of Reconfiguration and Coordination Algorithms

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-and-separate (B&S) 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 10,000 iterations of our genetic algorithm and single shots of a Monte Carlo optimizer to the whole network to serve as estimates for the best and worst case behaviour. The resulting average contention values and their standard deviation are shown in Table 3 . Figure 2 additionally shows the development of the amount of contention over time

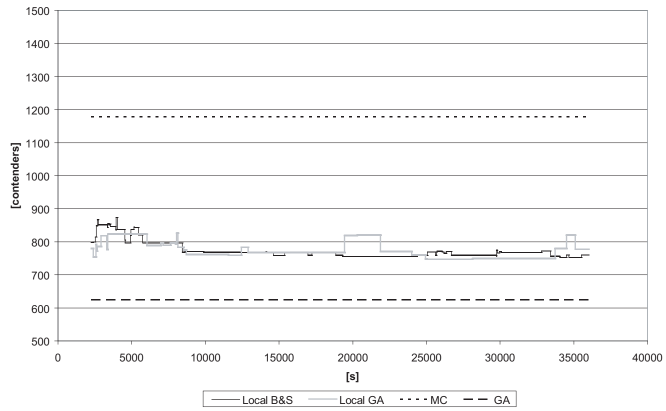

Figure 2: Performance of B&S and GA as Local Reconfiguration Algorithms compared with Global Minimum and Random Configuration

for one of the simulated scenarios starting from the moment that all nodes are activated.

The results show that variations of our distributed algorithm manage to exploit between 53% and 67% of the potential for reducing network contention compared to what could be achieved with global knowledge. This fact does not mean that all variations are equally suitable for real-world application, though. 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&S, while only achieving slightly better results. Furthermore, the stability of the contention levels is not the same between the different variations as can be seen in Figure 3 which is a close-up of the previous figure.

To evaluate the stability of the different variations of the distributed algorithm more thoroughly, we removed the initial transient phase by discarding the first 10,000 seconds of global contention samples. Afterwards we computed the standard deviation of the increment process for each of the contention time series as a measure of stability, the lower the standard deviation of the increment process the more stable the algorithm performs in steady state. Table 4 shows the standard deviation for 10 different scenarios for the two different reconfiguration algorithms GA and B&S, both of them using the N token coordination scheme (as it is the most stable of the coordination schemes and thus provides a lower bound on the difference in stability for the reconfiguration algorithms). Using this data to perform a sign test, we can show that with a probability of 99% the B&S algorithm achieves a 50% lower standard deviation than the GA, which of course means that B&S performs considerably more stable than the GA.

Another important aspect in judging the different variants of our distributed algorithm is the frequency in which stations are reassigned to different access

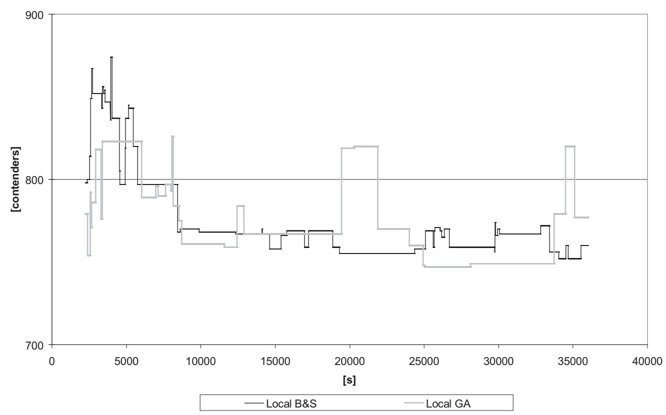

Figure 3: Stability of B&S and GA as Local Reconfiguration Algorithms Over Time

Scenario #	GA	B&S
1	0.374	0.126
2	0.523	1.443
3	0.785	0.141
4	0.460	0.241
5	0.755	0.152
6	0.515	0.308
7	0.799	0.200
8	1.130	0.000
9	0.573	0.212
10	0.354	0.000

Table 4: Stability of Local GA and B&S – Standard deviations of increment process in steady state

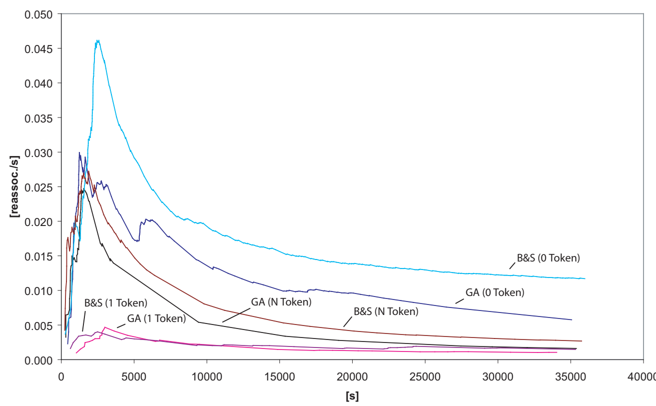

Figure 4: Rate of Station Reassociations over Time for Different Variants of the Distributed Algorithm

points. A high reassociation rate does not only result in high signalling traffic on the wireless link but also leads to serious degradation of the perceived link quality, as running sessions are interrupted while the reassociation is in progress. Figure 4 shows how the reassociation rate behaves over time for all six distributed algorithms. As expected, uncoordinated reconfiguration leads to the highest reassociation rates for both GA and B&S heuristics, as all APs try to reconfigure their network neighborhood concurrently. On the other hand, this concurrency of reconfiguration results in the fact that both algorithm are able to very quickly reach low levels of contention. In contrast, the single token approach with its sequential reconfigurations displays the lowest reassociation rates, but takes longer to converge to low contention levels. Note that our previous results have shown that both extremes eventually produce configurations of comparable quality. The N token case, finally, displays both behaviours: It has high reassociation rates at the beginning when many tokens exist and quickly reaches low contention levels. As more and more tokens are destroyed, its characteristics become more similar to the 1-token approach.

It is further interesting to note that the B&S heuristic behaves worse than the genetic algorithm with many tokens, but that the differences vanish as the number of tokens decreases.

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—is hard to circumvent without introducing 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.

In particular we have contributed mathematical optimization models that can be used to jointly determine the optimal transmission power settings and channel assignments for access points *and* stations, as well as the optimal assignment of stations to access points which will result in the least amount of contention in the network. The proposed models cover the case of low traffic intensity, in which only physical carrier sense is used, as well as the case of high traffic intensity, considering the additional contention caused by RTS/CTS frames. In addition, we have presented two theoretical lower bounds on contention, we have provided a transformation of our model to make it solvable with linear optimizers for small instances, and we have presented a genetic algorithm which is specially tailored to solve our contention minimization problem, but is likely to be useful in other wireless network optimization problems as well.

Our main contribution in this paper is 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 between 53% and 67% of the potential for reducing network contention compared to what could be achieved with perfect knowledge. Furthermore, we have presented different self-coordination schemes, enabling tradeoffs between fast convergence on low contention levels on the one hand and low reassociation rates respectively low signaling overhead on the other hand. These tradeoffs correspond to the degree of concurrency that is controlled by our different token schemes.

Besides these encouraging performance results, we also want to stress the modular framework we devised for the self-coordination in large-scale WLANs under different domains. It should allow for an easy extension of the candidate building blocks we devised with building blocks from other researchers, e.g. for the reconfiguration algorithm where we perceive that there is still room for improvement. In particular, it was also possible for us to integrate results from previous research into our framework and show its effectiveness.

For future work, we perceive the development of even more effective reconfiguration and/or coordination schemes as a short-term goal. However, our attention should now also be brought to the cooperation assumption and see if we can relax this towards non-cooperative environments where, of course, we would require the right incentive structures. In the same direction we are already actively thinking on how to make the protocols secure especially with regard to resilience against denial of service attacks. Currently, we are implementing the presented framework on a set of 4G Access Cubes manufactured by 4G Systeme Ltd. in order to be able to investigate its feasibility and scalability in a real-world environment.

References

[1] lp_solve. http://groups.yahoo.com/group/lp_solve/.

[2] NYCwireless. http://www.nycwireless.net.

[3] AutoCell—The Self-Organizing WLAN. White paper, Propagate Networks, 2003. http://www.propagatenet.com/resources/docs/whitepaper_autocell.pdf (last access: 2005-07-05).

[4] WS5100 Wireless Switch Reviewer's Guide. Product brochure, symbol, February 2005. ftp://symstore.longisland.com/Symstore/pdf/wireless/WS5100ReviewersGuide.pdf (last access: 2005-07-05).

[5] E. Amaldi, A. Capone, M. Cesana, and F. Malucelli. Optimizing WLAN Radio Coverage. In *IEEE International Conference on Communications (ICC 2004)*, pages 180–184, Paris, France, June 2004.

[6] Marco Liebsch et al. Candidate Access Router Discovery. Internet draft, IETF Seamoby Working Group, September 2004. http://www.ietf.org/internet-drafts/draft-ietf-seamoby-card-protocol-08.txt (last access: 2005-07-05).

[7] F. Glover and R.E. Woolsey. Note on converting the 0-1 polynomial programming problems to zero-one linear programming problems. *Operations Research*, 22:180–181, 1974.

[8] D.A. Goldberg. *Genetic Algorithms in Search, Optimization, and Machine Learning.* Addison-Wesley, 1989.

[9] J.K. Han, B.S. Park, Y.S. Choi, and H.K. Park. Genetic Approach with a New Representation for Base Station Placement in Mobile Communications. In *54th IEEE Vehicular Technology Conference (VTC 2001 Fall)*, pages 2703–2707, October 2001.

[10] A. Hills and B. Friday. Radio Resource Management in Wireless LANs. *IEEE Communications Magazine*, 42(10):9–14, December 2004.

[11] S. Hurley. Planning Effective Cellular Mobile Radio Networks. *IEEE Transactions on Vehicular Technology*, 51(2):243–253, March 2002.

[12] M. Kamenetsky and M. Unbehaun. Coverage Planning for Outdoor Wireless LAN Systems. In *2002 International Zurich Seminar on Broadband Communications Access, Transmission, Networking*, pages 491–496, March 2002.

[13] P. Karn. MACA – A New Channel Access Method for Packet Radio. In *Proceedings of the ARRL 9th Computer Networking Conference*, pages 134–140, London, Ontario Canada, September 1990.

[14] I. Katzela and M. Naghshineh. Channel Assignment Schemes for Cellular Mobile Telecommunication Systems: A Comprehensive Survey. *IEEE Personal Communications*, 3(3):10–31, June 1996.

[15] S.O. Krumke, M.V. Marathe, and S.S. Ravi. Models and Approximation Algorithms for Channel Assignment in Radio Networks. *Wireless Networks*, 7(6):575–583, November 2001.

[16] C.Y. Lee and H.G. Kang. Cell planning with Capacity Expansion in Mobile Communications: A Tabu Search Approach. *IEEE Transactions on Vehicular Technology*, 49(5):1678–1691, March 2000.

[17] Y. Lee, K. Kim, and Y. Choi. Optimization of AP Placement and Channel Assignment in Wireless LANs. In *IEEE Conference on Local Computer Networks (LCN 2002)*, 2002.

[18] R.A. Leese. A Unified Approach to the Assignment of Radio Channels on a Regular Hexagonal Grid. *Transactions on Vehicular Technology*, 46(4):968–980, November 1997.

[19] K.K. Leung and B.-J. Kim. Frequency Assignment for IEEE 802.11 Wireless Networks. In *58th IEEE Vehicular Technology Conference (VTC 2003 Fall)*, pages 1422–1426. IEEE, October 2003.

[20] X. Ling and K.L. Yeung. Joint Access Point Placement and Channel Assignment for 802.11 Wireless LANs. In *IEEE Wireless Communications and Networking Conference (WCNC 2005)*, 2005.

[21] Y. Matsunaga and R.H. Katz. Inter-Domain Radio Resource Management for Wireless LANs. In *IEEE Wireless Communications and Networking Conference (WCNC 2004)*, pages 2183–2188, Atlanta, Georgia, USA, March 2004.

[22] 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.

[23] P. Struhsake. WLAN Channel Bonding: Causing Greater Problems Than It Solves. White Paper SPLY003, Texas Instruments, September 2003.

[24] Y. Wang, L. Cuthbert, and J. Bigham. Intelligent Radio Resource Management for IEEE 802.11 WLAN. In *IEEE Wireless Communications and Networking Conference (WCNC 2004)*, pages 1365–1370, Atlanta, Gergia USA, March 2004.

[25] IEEE 802.11 WG. Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications. IEEE Standard, August 1999.

[26] IEEE 802.11 WG. Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications, Amendment 5: Spectrum and Transmit Power Management Extensions in the 5 GHz band in Europe. IEEE Standard, October 2003.

[27] Frank A. Zdarsky, Ivan Martinovic, and Jens B. Schmitt. On Lower Bounds for MAC Layer Contention in CSMA/CA-Based Wireless Networks. In *Third ACM/SIGMOBILE Workshop on Foundations of Mobile Computing (DIALM/POMC)*, Cologne, Germany, August 2005. To be published.