Jamming Resistant Architecture for WiMAX Mesh Network

Boris Makarevitch
Communications Laboratory
Helsinki University of Technology
Email: boris.makarevitch@tkk.fi

***Abstract*— The paper presents a jamming resistant architecture for military applications of WiMAX (802.16) mesh network. The main idea is to use multiple base stations (BS), access points to the fixed core network. This will facilitate network survivability both in case of a BS destruction and in case of jamming. When the nodes are affected by jamming they can reroute to another base station if possible. The multiple BS architecture requires distributed scheduling as there is no single point of control that can execute centralised scheduling. A distributed scheduling algorithm, performing well in both jammed and non-jammed environment, is a key issue in implementation of the architecture. Several scheduling algorithms are considered in the paper: a static algorithm using the number of traffic flows in the node, dynamic load-dependent algorithms using two- and three-hop neighbour information. Finally, an algorithm based on finite field initial slot assignment is proposed. This algorithm uses two-hop information and is also load-dependent. Performance of different algorithms is evaluated with and without jamming. The finite field algorithm shows the best performance in both cases.**

I. WIMAX MESH NETWORK

WiMAX (Worldwide Interoperability for Microwave Access) [1] is becoming a reasonable alternative for the military wireless access network. Its main strength is that due to a higher transmitted power it can provide coverage beyond the range of the wireless LAN [2]. WiMAX can form a wireless backbone for the WLAN cells or be connected directly to the users terminals if high user mobility is not required. An additional important consideration for military usage of WiMAX is its resistance to jamming.

WiMAX can be used with two network topologies, point-to-multipoint (PMP) and the mesh mode. In the former, an access node to the fixed network, called in the WiMAX terminology Base Station (BS), has a direct connection to the wireless nodes, called Subscriber's Stations (SS). In the latter, the SS's can serve as routers thus enabling multi-hop routes. While the PMP network's coverage is restricted to the range of one link, the mesh network can provide unlimited coverage, though with a price of decreased throughput, since then the links have to carry many traffic flows. The mesh is especially suitable for military applications as the military access networks usually should cover rather large areas.

Also for the area that can be covered by a PMP, the mesh provides a benefit in terms of jamming resistance. Consider the following example with three nodes: base station 0 and two subscriber stations 1, 2 and a jammer j.

Communication between BS 0 and SS 2 will be disrupted in the PMP mode if

$$\frac{P_{\max} g_{02}}{\eta + P_j g_{j2}} < \gamma_{req} \qquad or \qquad \frac{P_{\max} g_{20}}{\eta + P_j g_{j0}} < \gamma_{req}, \tag{1}$$

where $P_{\max}$ is the maximal transmit power, η is the noise power, P_j is the jammer power, γ_{req} is the required signal-to-noise ratio (SNR), and g_{ij} is the link gain for a link between i and j. Low SNR in one direction is sufficient for disruption, since WiMAX requires bidirectional links for operation. The mesh mode still gives a possibility to maintain the connection in this case if

$$\frac{P_{\max} g_{12}}{\eta + P_j g_{j2}} > \gamma_{req} \qquad and \qquad \frac{P_{\max} g_{20}}{\eta + P_j g_{j0}} > \gamma_{req}, \tag{2}$$

with similar conditions for the opposite direction.

Another advantage of the mesh topology is that it allows to reroute connections in case of jamming. Suppose, the jammer succeeded to jam out nodes B and C that were intermediate nodes in the route A-B-C-D. With the mesh topology another route A-E-F-D can be established if the jammer power was not enough to jam out the links A-E and D-F.

However, when a jammer is close to the BS, such that all BS links are jammed out, the whole mesh will be disconnected from the core network. An obvious solution to this problem is to distribute the core network's access, to provide several gateways between the mesh and the core network. Thus, we propose a jamming resistant mesh architecture, a mesh with multiple base stations.

II. JAMMING RESISTANT MESH ARCHITECTURE

Having several BSs, access points to the core network, also increases the network survivability since when a BS is destroyed the nodes can reroute to another BS. Another significant benefit of multi-BS mesh is increase of the network capacity as it is not anymore restricted by the throughput of one BS. Fig. 1 shows a mesh with 9 nodes and 2 BS that are marked by bigger circles. The nodes are placed in irregular grid, and an appropriate mean distance between the nodes is chosen to achieve full coverage for the given link gain, the maximal transmission power and required SNR. For routing an energy-efficient routing suggested in [1] was used. It is based on energy/bit factor as a metric, which is the ratio between the transmitted power and the channel rate P_t/R. Since for the mesh nodes the energy is usually not a scarce resource, the main contribution of the energy-efficient routing is in increase of link physical rate by selecting links with less attenuation comparing to the minimum-hop routing.

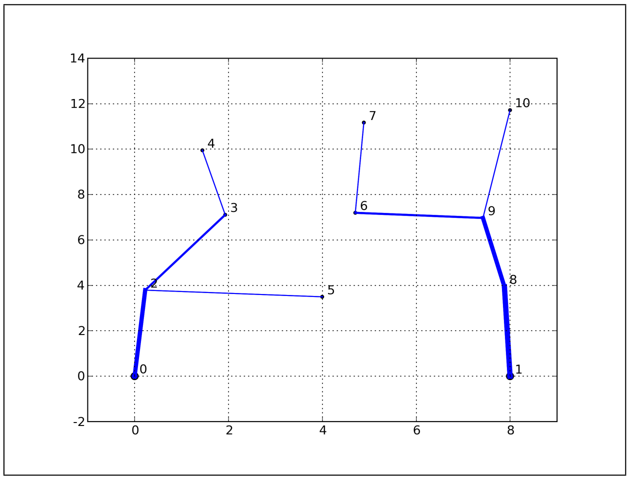

Fig. 1. Mesh with 2 base stations

Figure 2 shows the same network but with a jammer, shown as rectangle, and located at coordinates $\{1{,}1\}$ km, close to BS 0. We can see that only one node 2 is disconnected, and nodes 3,4 and 5 rerouted to BS 1. The same network with only one BS, depicted in fig. 3 would be completely disconnected by the same jammer.

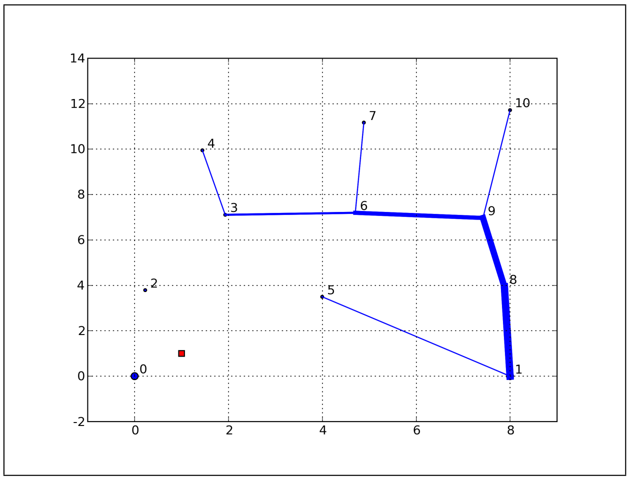

Fig. 2. Mesh with 2 base stations under jamming

The benefits of the multi-BS mesh come with the price of loosing a single point of control as in case of one BS. Therefore the multi-BS mesh requires distributed scheduling. The WiMAX standard provides signalling messages for both centralised and distributed scheduling but leaves the scheduling algorithms open for the vendor's implementation. The centralised scheduling is simpler to implement and it can obtain information about the whole network. The distributed scheduling is more complex and it has to cope with discrepancies in the network view of different nodes. A distributed scheduling algorithm ensuring good performance with and without jamming is crucial for the proposed jamming resistant architecture. In the following we consider several scheduling algorithms and study their performance.

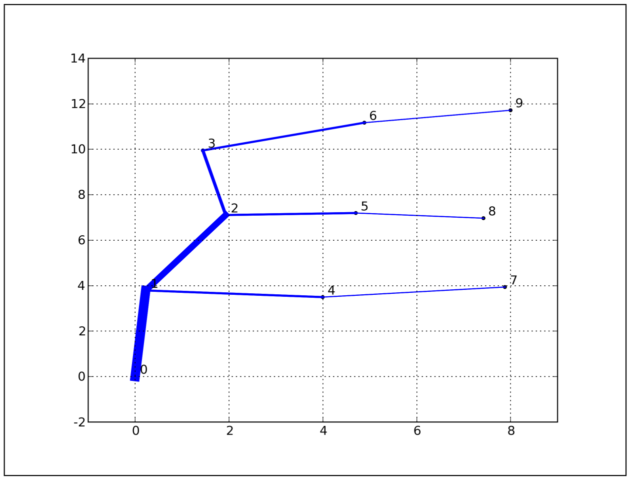

Fig. 3. Mesh with one base station

III. DISTRIBUTED SCHEDULING ALGORITHMS

The Internet traffic is characterised by high variability and burstiness, and therefore the network performance is improved if scheduling is capable to adapt to traffic variations by giving the transmission slots to the most needed nodes. The single BS mesh can use centralised scheduling where the nodes supply their slot allocation requests in a centralised scheduling message MSCH-CSCH. The BS can divide the available slot capacity proportionally to the requests. A simple scheduling algorithm without spatial reuse assigns to node i the number of slots:

$$N_i^s = \left\lfloor \frac{Q_i N_{fr}^s}{\sum_{k=1}^{M} Q_k} \right\rfloor, \tag{3}$$

where Q_i is the queue size of i-th node, and M is the number of nodes in the network.

The multi-BS mesh with distributed scheduling lacks a central entity where traffic requests can be processed. The simplest implementation of the distributed scheduling is to use the fixed allocation according to the number of nodes in the network, where $N_i^s = \lfloor N_{fr}^s / N_{node} \rfloor$. We assume that when a new node joins the network, all the nodes are notified about this event in a network configuration message MSH-NCFG.

An obvious improvement over the fixed scheduling is to make slot allocation dependent on the number of data flows going through a node. A data flow is all data frames having the same origin and destination. Apparently, the base stations carry more data flows than the terminating nodes and require a bigger slot capacity share. With this scheme, which we call flow proportional scheduling, the number of slots allocated to the node is:

$$N_i^s = \left\lfloor \frac{N_i^{fl} N_{fr}^s}{\sum_{k=1}^{M} N_k^{fl}} \right\rfloor, \tag{4}$$

where N_i^{fl} is the number of flows for i-th node.

The information on number of flows can be supplied upon the node entry in the network and propagated to the other nodes in the net configuration message.

A. Traffic dependent scheduling

Though the flow proportional scheduling adapts better for the network topology than the fixed allocation it can not account for the traffic load variations. Adaptivity to traffic is important for the bursty data services as web browsing, or real-time streaming with a variable bit rate. The WiMAX signalling gives possibility to convey the traffic load information also for the distributed scheduling. The standard [1] specifies a three-way handshake for distributed scheduling. The message MSH-DSCH is used for all three information flows. This message contains slot requests, slot availabilities, and slot grants. First, the node broadcasts the slot requests for their links giving in the same message information on the available slots. Since the available slot space may not be contiguous but can include many windows of several slots, the size of this information element might be significant, as each availability occupies 32 bits. After that, each node must assign the available slots to the requests and issue the slot grants, which are sent in the reply MSH-DSCH message. The original requesters should confirm the grants by copying them to the third confirmation message. The node neighbours will infer the schedule by listening to all MSH-DSCH messages in their neighbourhood. For this signalling scheme, the actual job of the scheduling algorithm is to distribute the available slots between the requests. The nodes must align the available slots positions between them to avoid collisions.

The messages are oriented on link scheduling performed by the receivers. The amount of signalling data can be reduced if the requests are supplied for the total slot demand of a node and scheduling is performed by the transmitters. Further, there is no need to broadcast the schedule if the nodes allocate slots according to common rules. A rule allowing for traffic adaptivity is to allocate more slots to the nodes with the highest current traffic demand. The nodes broadcast their request sizes in the first MSH-DSCH message. Thus, each node receives information on request sizes of all its direct neighbours, which are nodes that can decode the broadcasted message.This happens if the signal-to-noise ratio (SNR) Γ_{ij} for a link between nodes i and j is higher than the required SNR. The network control and scheduling messages are sent with the lowest and most robust data rate which has the lowest required SNR γ_1 and the fixed signalling power P_{sig}. The signalling SNR is:

$$\Gamma_{ij}^{sig} = \frac{P_{sig} g_{ij}}{\eta + I}, \tag{5}$$

where the link gain is $g_{ij} \propto r_{ij}^{-m}$ with the distance between nodes r_{ij} , η is noise and I is interference, which can be omitted if no other signalling messages are allowed to be sent in the same slot. The set of direct neighbours $\mathcal{N}_{j1}$ of a node j in the network graph $\mathcal{G}$ is defined as:

$$\mathcal{N}_{j1} \triangleq \{n \in \mathcal{G} : \Gamma_{nj}^{sig} > \gamma_1\} \tag{6}$$

Then the set of n-hop neighbours of a node j $\mathcal{N}_{jn}$ is:

$$\mathcal{N}_{jn} = \mathcal{N}_{j,n-1} \bigcup_{k \in \mathcal{N}_{j,n-1}} \mathcal{N}_{j,n-1} \cap \mathcal{N}_{k,n-1} \tag{7}$$

By propagating the slot requests of direct neighbours in subsequent MSCH-DSCH messages the nodes can get data on slot requests of their 2- and 3-hop neighbours depending on the number of messages. This data can be used for slot allocation by sequential request comparison and slot assignment to the node with maximal request. If a slot should be assigned to the node itself, the receiver for the node's flow with maximal data queue is chosen for transmission in the slot. The resulting schedule vector for a node j having as entries receiver $Rl = n \in \mathcal{N}_{j1}$ if j is scheduled to transmit in slot l and zero otherwise is filled according to:

```
Algorithm 1
init: $R_l \leftarrow 0 \ \forall l = 1 \ldots N_{fr}^s$
l ← 1
while $S_j > 0$ or $l < N_{fr}^s$
    $n_t \leftarrow \arg\max_{n \in \mathcal{N}_{jc}} S_n$
    $S_{n_t} \leftarrow S_{n_t} - 1$
    if $n_t$ = j
        $k_t \leftarrow \arg\max_{k \in \mathcal{N}_{jf}} S_k$
        $R_l \leftarrow R_{k_t}$
        $S_{k_t} \leftarrow S_{k_t} - 1$
    end
    l ← l + 1
end
```

Here, $\mathcal{N}_{jc}$ is the set of nodes competing with j, and $\mathcal{N}_{jf}$ is the set of flows for node j, and S is the slot demand for a node and a flow

If the nodes have distinct competing sets, which happens when $\mathcal{N}_{jc} \subset \mathcal{G}$, they may assign different transmitters to the same slot which can result in collisions. On the other hand, having the same competing set for the entire net, when $\mathcal{N}_{nc} \subseteq \mathcal{G}, \forall n$ assures only one transmitter in each slot, and thus the traffic dependent scheduling results in orthogonal time division without any spatial reuse and risk for collisions. Presence of jammer decreases the number of neighbours for given $P_{sig} = P_{\max}^t$ by increasing I in eq. 5 and causes more collisions comparing to the case with no jamming. Thus, to achieve collision-free transmissions under jamming information on more hop neighbours is needed. The price to pay for more information is a higher signalling overhead, as we need more MSH-DSCH messages. Since the traffic dependent algorithm needs to collect information on slot requests for n-hop neighbours, it must wait at least n frames before making decisions on slot assignment. Hence, algorithm is run once in a scheduling period with duration of at least n frames. To reduce the signalling overhead it is better to prolong the scheduling period with several frames without the signalling messages.

Another approach is to start with initial preallocation of slots, and then resolve possible conflicts by using the slot demand information. Then, performance can be improved by reducing signalling overhead and achieving spatial reuse with less collisions. Below, we propose such an algorithm based on the finite field preallocation.

B. Scheduling algorithm based on finite fields

The problem of distributed scheduling was firstly encountered in research on packet radio networks. Initially only scheduling of broadcasts was considered. In broadcast scheduling a time slot is allocated to a sending node which transmission must be received collision free by all of its neighbours [3]. This is different from the link scheduling, where transmission is intended for a particular node. A distributed scheduling algorithm using one-hop neighbour information and token passing is presented in [4]. In [5] an algorithm using two-hop information is described. These algorithms require a significant amount of information to be exchanged between the nodes and are not efficient in mobile environment. They can be defined as topology dependent, since changes in the network topology require calculation of a new schedule.

To remove the topology dependency, [6] proposed a topology independent algorithm based on the properties of finite fields. The requirement of collision-free transmission in every slot was relaxed to one collision-free slot in the frame for each node. Then the slot allocation is performed in advance and is immune to topology changes.

Availability of a collision-free slot is guaranteed by the following property of polynomials over a Galois field GF(q), where q is either a prime number or a power of a prime number p, $q = p^i$. A polynomial $f(x) = \sum_{i=0}^{k} a_i x^i$ of degree k will have at most k distinct roots. Then the ranges of two polynomials cannot have more than k common values that correspond to the roots of the difference of these polynomials, which is a polynomial of degree $\leq k$, and therefore cannot have more than k roots. Now, a mapping of polynomials into the set of time slots will produce subsets of cardinality q having no more than k common slots between them. Each node is assigned a unique coefficient vector $a_0, \ldots, a_k \in$ GF(q), which results in unique polynomial for each node.

The number of common slots between a node and its neighbours is bound by kD, where D is the maximum network degree or the maximum number of neighbours. Then since a node is assigned totally q slots, a link between any two nodes will always have a free slot, if the condition $q \geq kD + 1$ is satisfied. The algorithm's implementation chooses k and q to satisfy conditions:

$$k = \lfloor (q-1)/D \rfloor, k \geq 1, q^{k+1} \geq M, \tag{8}$$

where M is the number of nodes in the network. The values of M and D not need to be exact, but can be chosen large enough that their actual values do not exceed the parameter values. This reduces the amount of information needed by the algorithm. The frame length for the algorithms must be $L = q^2$, so that it consists of q subframes, each having one slot allocated for each node.

The algorithm guarantees a minimum throughput for a node, it does not need to recalculate the schedule due to mobility, and thus is the most efficient for mobile networks with constant traffic load. When the traffic load varies, the algorithm will have poor utilisation, since the nodes with low traffic will have abundant slots, while heavily loaded nodes will not have enough slots for transmission. This possibility was envisaged in [6], and redistribution of slots between the nodes was suggested, however no algorithm for this redistribution was given.

The WiMAX nodes are not movable, and therefore the schedule need not be recalculated because of mobility changes. The traffic variability requires however constant rescheduling to adapt to the traffic variations. The main strength of the finite field algorithm in WiMAX context is restricting the amount of possible conflicts by limiting the set of competitors for a slot to the preallocated one.

In order to apply the algorithm its frame structure must be aligned to the WiMAX mesh frame format. The WiMAX mesh frame consists of control and data subframes. The control subframe contains network entry and configuration messages followed by scheduling messages, MSH-DSCH for distributed scheduling. The data subframe is divided into slots, or minislots in WiMAX terminology with possibility to combine several minislots to one scheduling unit or slot. Here, we assume one slot equal to a minislot, and use the term slot throughout the paper. A slot begins with one or two preamble symbols N_{pre} and ends with the guard symbol. The number of OFDM symbols in the frame $N_{sym} = T_{fr}/T_{sym}$, where T_{fr} and T_{sym} are the frame and symbol times. The symbol time depends on the allocated bandwidth W, number of subcarriers N_c, sampling rate F_s, and the guard time fraction T_g:

$$T_{sym} = \frac{(1+T_g)N_c}{F_s W} \tag{9}$$

The number of data slots then can be calculated as:

$$N_{fr}^{s} = \left\lfloor \frac{N_{sym} - 7(N_{ctrl} + M)}{N_{data} + N_{pre} + 1} \right\rfloor \tag{10}$$

where N_{ctrl} is the number of network control messages and N_{data} is the number of data symbols per slot. The number of scheduling messages we assume to be equal to the number of nodes to completely avoid collisions and guarantee their reception. The length of both type of messages is 7 symbols, including 4 data symbols and the long preamble. Since for the proposed algorithms, the DSCH messages carry only slot demand per node, this length is more than enough for their data. The order of the field q is chosen to match L to N_{fr}^{s} and to satisfy 8 for chosen k. Thus for $W = 50$ MHz, $T_{fr} = 5$ ms, $N_c = 256$ we will have $N_{sym} = 911$, which for $N_{ctrl} = 3$ and $N_{data} = 4$ with short preamble of one symbol will give 137 slots, matching to $L = 121$ and $q = 11$. For the considered network of 11 nodes, $k = 2$ will satisfy 8. Further, we assume a scheduling period of 5 frames, so that for 2-hop neighbour information 3 frames do not include the signalling messages and have 148 slots.

For successful operation of the finite field initial preallocation, each node must be assigned a unique set of polynomial coefficients a_i. Since the multi BS mesh does not have a central authority, the coefficients should be set by the nodes themselves. A possible way is to generate them randomly with a seed bound to unique number, e.g. MAC address. However, since the number of possible combinations is restricted and equal to q^{k+1}, there is the probability $1/q^{k+1}$ of generating the same set for neighbour nodes. To avoid this to happen, the coefficients must be notified in the network entry message, MSCH-NENT, and propagated throughout the network in the configuration message MSCH-NCFG. A node must generate a new set of a_i until it gets a set unique for the network.

The algorithm consists of three steps. First, upon the network entry node j calculates an initial schedule, matrix I_{nl} for its set of competing nodes $\mathcal{N}_{jc}$:

```
Algorithm 2
init: $I_{nl} \leftarrow 0,\ \forall n \in \mathcal{N}_{jc}\ \forall l = 1 \ldots L$
for n = 1 : $\overline{\mathcal{N}_{jc}}$
    for m = 1 : q
        l = $q(m-1) + \sum_{i=0}^{k} a_{ni}(m-1)^i \bmod q$
        $I_{nl} = 1$
    end
end
```

From this initial schedule the node derives a set of competing nodes for each slot:

$$\mathcal{N}_{jcl} \triangleq \{n \in \mathcal{N}_{jc} : I_{nl} = 1\} \tag{11}$$

For the second step, in each scheduling period, an algorithm similar to the Algorithm 1 is executed with difference that now $n_t = \arg\max_{n \in \mathcal{N}_{jcl}} S_n$.

After initial preallocation there will be many slots left with no competitors, also there are remaining slots since $L < N^s_{fr}$. For example, for a particular network configuration the amount of free slots ranges from 54 to 69. In the third step, the Algorithm 1 is executed for the free slots if a slot demand is still remaining for the node after the second step.

IV. PERFORMANCE EVALUATION

Performance of the above described algorithms was evaluated by computer simulations. As performance metric the average data flow delay was used. For simulation of the underlying WLAN traffic we use a multi-state Markov source which mean bit rate is the load parameter.

The performance evaluation results for the network with 9 nodes and 2 BS without jamming are presented on fig. 4. The BS positions were the same as in fig. 1 but the nodes positions were varied for 5 network configurations. The delays are shown for the fixed scheduling **fix**, scheduling with proportional to number of flows allocation **pr**, load dependent scheduling with 2- and 3-hop neighbours information **ld2** and **ld3**, and the algorithm using finite field preallocation with 2-hop information **ff**. The delay of the fixed scheduling grows steeply after 250 kbps. The traffic dependent scheduling with 2-hop information has higher delays than the proportional scheduling, and even with the 3-hop information it performs worse than the proportional scheduling for the moderate load due to the signalling overhead. The 3-hop information results in the common request knowledge among the nodes for the considered network size and achieves the full TDMA without spatial reuse and collisions. The best performance is shown by the finite field algorithm. It achieves better performance because of reduced signalling overhead, as it uses only 2-hop information and requires one less signalling message per scheduling period, and due to spatial reuse as it facilitates multiple transmissions per slot.

Performance of the scheduling algorithms was also simulated for the same network configurations and the jammer located at coordinates {1,1} km. The results are shown on fig. 5. Jamming worsens performance for all algorithms but the severity of its impact depends on the algorithm. It causes excessive delays already under low load for fixed and load dependent scheduling with 2-hop neighbour information. Scheduling with 3-hop information has now much better performance than the proportional scheduling, but the finite field algorithm has again the lowest delay.

V. CONCLUSION

The paper proposed a jamming resistant architecture for the WiMAX mesh network. The essential feature of the architecture is usage of multiple base stations. The largest gain in jamming resistance comparing to the single BS the architecture shows when the jammer is located near a base station. Besides better network survivability under jamming, the multiple-BS architecture also increases the network throughput. The multi-BS mesh introduces additional complexity as it requires a distributed scheduling algorithm. Several algorithms were considered in the paper and their performance evaluated by the network simulation for both jamming and non-jamming cases. The best performance in both cases shows an algorithm with initial slot allocation based on the finite field properties.

REFERENCES

[1] IEEE, *Std. 802.16: Air Interface for Fixed Broadband Wireless Access Systems*, 2004.

[2] IEEE, *Std. 802.11: Wireless LAN Medium Access Control (MAC) and Physical layer (PHY) Specifications*, 1999.

[3] S. Ramanathan and E. L. Lloyd, "Scheduling algorithms for multihop radio networks," *IEEE/ACM Transactions on Networking*, vol. 1, no. 2, pp. 166–177, 1993.

[4] R. Ramaswami and K. K. Parhi, "Distributed scheduling of broadcasts in a radio network," in *IEEE Infocom*, 1989.

[5] A. Ephremides and T. Truong, "Scheduling broadcasts in multihop radio networks," *IEEE Transactions on Communications*, vol. 38, no. 4, pp. 456–460, 1990.

[6] I. Chlamtac and A. Farago, "Making transmission schedules immune to topology changes in multi-hop packet radio networks," *IEEE/ACM Transactions on Networking*, vol. 2, no. 1, pp. 23–29, 1994.

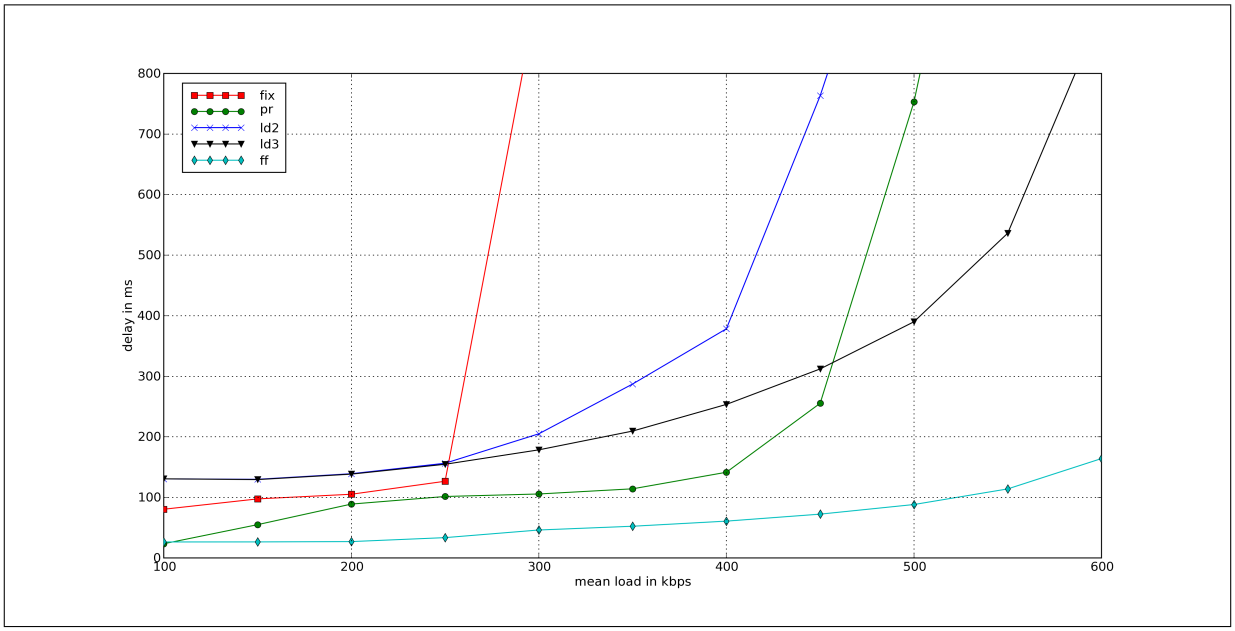

Fig. 4. Scheduling performance without jamming

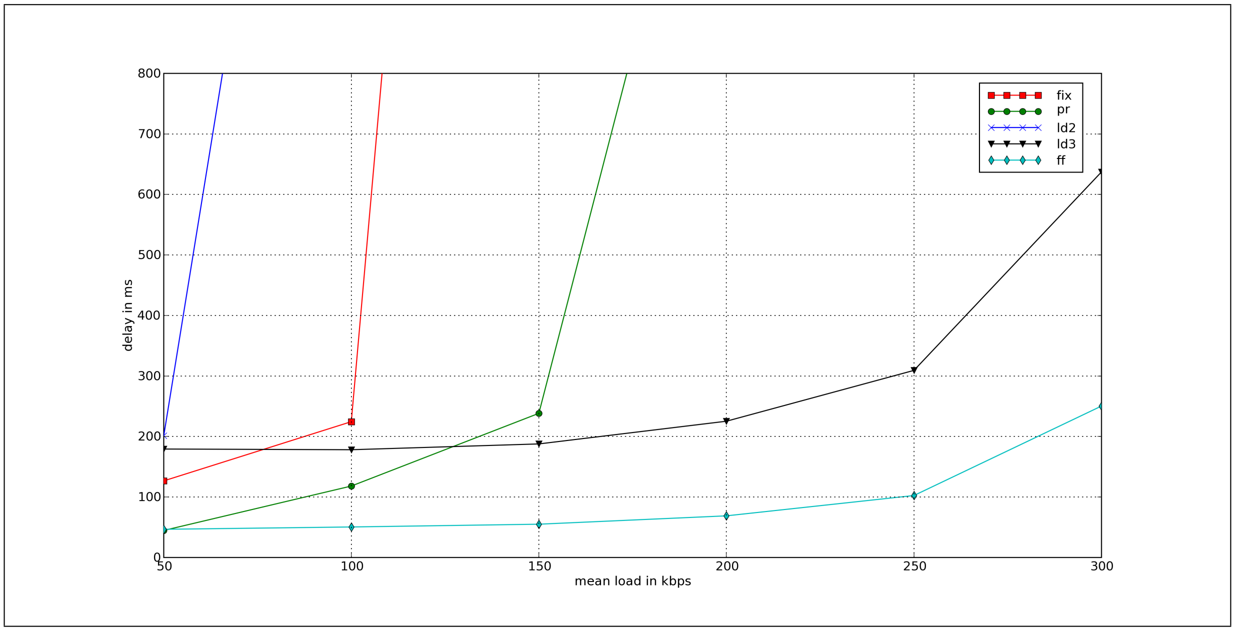

Fig. 5. Scheduling performance under jamming