Max-Min Fair Throughput in Multi-Gateway Multi-Rate Mesh Networks

Dirk Staehle, Barbara Staehle, Rastin Pries
University of Würzburg, Chair of Communication Networks, Würzburg Germany,
{dstaehle, bstaehle, pries}@informatik.uni-wuerzburg.de

***Abstract*—The problem how to determine the capacity of and achieve fairness in mesh networks is one of the key topics in practical and theoretical research on mesh networks. Max-min fairness is one way to define fairness and several algorithms how to compute max-min fair rate allocations are already published. In this paper we make two major contributions to this area of research: First, we formulate an algorithm achieving max-min fairness among end-to-end flows based on the effective load of a collision domain. This allows us to determine max-min fair rate allocations in a multi-gateway, multi-channel mesh network with equal rates for all links. Second, we extend this algorithm for heterogeneous link rates.**

I. Introduction

Mesh networks or more general multi-hop networks are of great interest in the wireless research community for at least the last decade. One of the key questions is to determine the capacity of a wireless mesh networks under certain fairness constraints. This question is still not finally solved - even under the simplifying assumption of static topology, routing, channel and link rate allocations. A solution to this problem (1) gives more insights in the theoretical understanding of mesh networks, (2) may be directly applicable for assigning resources in centralized scheduling schemes, and (3) can be useful for planning mesh networks. An exact a priori evaluation of the actual performance of mesh networks is typically not feasible. The evaluation of the performance of an idealized mesh network might however give a first estimate what performance the deployed mesh network might achieve.

In this paper we focus on the throughput of a mesh network under the constraint of max-min fairness [3]. In literature, we already find a number of methods to determine the max-min throughput of mesh networks. Huang and Bensaou [5] propose an algorithm for determining the max-min fair throughput in an ad-hoc network. However, they do not consider the throughput of end-to-end flows but of links or single-hop flows.

Jun and Sichitiu [6] consider a mesh network with end-to-end flows. They introduce the term "nominal capacity" of a mesh network which relates to the nominal MAC layer capacity of a one hop infrastructure network. A mesh network's nominal capacity is this MAC layer capacity shared equally among all flows running through the bottleneck collision domain. A link's collision domain corresponds to all links on which a parallel transmission would lead to a collision. The load of a collision domain corresponds to the number of single hop transmissions on the collision domain's links including relayed transmissions. The bottleneck collision domain is the collision domain with the highest load. Jun and Sichitiu do not really determine the max-min fair throughput of a network but only the capacity of the main bottleneck and hence the minimum throughput.

However, Aoun and Boutaba [2] use the notation of a "collision domain" as defined by [6] in order to determine the max-min fair throughput. This is done by iteratively determining the bottleneck collision domain and fixing the rates of all flows traversing it. Additionally, Aoun and Boutaba introduce the definition of the "effective load" of a collision domain which is lower than the nominal load defined in [6]. The difference of effective load and nominal load is that the effective load considers the possibility of spatial reuse of components belonging to the same collisions domain. While Aoun and Boutaba introduce the notation of effective load and demonstrate it using examples, they do not specify an explicit algorithm how to compute it. The algorithm given in [5], however, is actually based on the effective load definition and can be modified to determine the max-min fair throughput in mesh networks with end-to-end flows.

Akhtar and Moessner [1] extended the work of [2] by considering multi-radio, multi-channel, and multi-gateway mesh networks. However, they went back to the concept of the nominal load of a collision domain instead of the effective load introduced by [2]. In this paper we make two major contributions to the research on max-min rate allocations in mesh networks. First, we provide a formal algorithm to compute max-min fair rate allocations according to the effective load definition of [2] and based on the algorithm of [5]. Second, we extend the algorithm to multi-rate networks. Altogether, we will present an algorithm to determine the max-min fair throughput in a multi-gateway, multi-radio, multi-rate mesh network with static routing, channel and rate allocation.

The rest of the paper is structured as follows. In Section II we formulate the problem and define variables. In Section IV we shortly describe existing algorithms. In Section IV-A we propose an algorithm to determine the effective load of a collision domain. In Section IV-B we extend the algorithm to a network with multiple rates. In Section V we investigate max-min fair rate allocations using the nominal load or effective load of a collision domain. Finally, we summarize the main contributions in Section VI.

II. Problem Formulation

The objective of this paper is to develop an algorithm for computing a max-min fair rate allocation for the flows in a mesh network. Shortly, max-min fairness means that the minimum throughput in a network is maximized. A solution is max-min fair if no rate can be increased without decreasing another rate [3]. We define a mesh network as a set of nodes $\mathcal{N}$ and a set of links $\mathcal{L}$ connecting the nodes. A subset $\mathcal{G} \subseteq \mathcal{N}$ contains the gateway nodes that are connected to the Internet. A link (i, j) exists between nodes i and j if a direct communication between these nodes is possible and $r_{i,j}$

is the data rate of link (i,j). All nodes except the gateway nodes are assumed to have a saturated best-effort data flow from the Internet to the node. For the sake of simplicity of notation we consider only downlink data flows. Including uplink data flows, however, would not increase the complexity of the model. Let us introduce the variable $\mathcal{F} = \mathcal{N} \setminus \mathcal{G}$ for the set of nodes having a data flow. Each of the nodes $i \in \mathcal{F}$ is connected to the Internet via a fixed gateway and the path to the gateway is also fixed. Node i's path is denoted as $\mathcal{P}_i$ and corresponds to a set of links, i.e. $\mathcal{P}_i \subseteq \mathcal{L}$. Furthermore, we consider a multi-channel, multi-radio mesh network, i.e. every link (i,j) is assigned a channel $q_{i,j}$ out of a set $\mathcal{Q}$ of non-overlapping channels. The goal is now to find data rates b_i for every $i \in \mathcal{F}$ that fulfill the definition of max-min fairness.

III. Previous Work

In this section we describe in detail the results of [5], [6], [2], [1] already discussed in the introduction. Let us start with the definition of a collision domain and its nominal load according to [6]. The collision domain $\mathcal{D}_{i,j}$ of link (i,j) corresponds to the set of all links (s,t) which can not be used in parallel to link (i,j). The term "can not be used in parallel" is intentionally chosen rather imprecisely since the exact definition depends on the underlying MAC technology. In general, a link (s,t) can not be used in parallel to a link (i,j) if the interference from a transmission on link (s,t) alone is strong enough to disturb a parallel transmission on link (i,j). Note that this definition does not include the case that small interferences from multiple parallel transmissions sum up and might together disturb a successful transmission. In [6] the symmetric and the asymmetric case are distinguished. In the asymmetric case, a transmission on link (s,t) prevents the successful transmission on link (i,j) if link (s,j) or (i,t) exist, i.e. all one-hop neighbors of j must not transmit and all one-hop neighbors of i must thus not receive. In the symmetric case, both ends of a transmission are protected, e.g. due to the RTS/CTS mechanism of IEEE 802.11 or the three way hand-shake in the IEEE 802.16 mesh mode. Consequently, a transmission on link (s,t) prevents the successful transmission on link (i,j) if at least one of the links $(i,s),(i,t),(j,s)$ or (j,t) exists. The entire two-hop neighborhood is blocked. In the following, we refer to the two cases as "asymmetric" and "symmetric" collision definitions.

The nominal load of a collision domain corresponds to the number of transmissions that take place in a collision domain. A transmission $t_{k,i,j}$ corresponds to the hop from node i to node j taken by the flow towards node k, i.e. $(i,j) \in \mathcal{P}_k$. The number of transmissions $n_{i,j}$ of link (i,j) corresponds to the number of end-to-end flows crossing it:

$$n_{i,j} = | \{k \mid (i,j) \in \mathcal{P}_k\} | \,. \tag{1}$$

Correspondingly, the number of transmissions in collision domain $\mathcal{D}_{i,j}$ is

$$m_{i,j} = \sum\nolimits_{(s,t)\in\mathcal{D}_{i,j}} n_{s,t}. \tag{2}$$

Assuming that all links have an equal rate r, the capacity of the whole collision domain is also r and the throughput of any flow traversing at least one link of collision domain $\mathcal{D}_{i,j}$ is $r/m_{i,j}$. The bottleneck collision domain is the collision domain with highest nominal load or lowest throughput.

Based on this definition of the collision domain throughput an algorithm for computing the max-min fair throughput is given in [2]. The principle of the algorithm is to iteratively determine the bottleneck collision domain and to allocate the rates of all flows traversing this collision domain. In the next iteration, only the remaining collision domains and flows are considered. The iteration converges when the rates of all flows are set.

Let $\mathcal{O}$ be the set of flows without rate allocation, $p_{i,j}$ be the percentage of unassigned nominal capacity of collision domain $\mathcal{D}_{i,j}$, and $\mathcal{L}^*$ be the set of links carrying flows with unassigned rates. Then, the algorithm shown in Fig. 1 yields the max-min fair rate allocation. Please note that we use a different notation though the principle of the algorithm is identical to the one in [2].

Initialization:
1 $\quad \mathcal{O} = \mathcal{F}$ .... all flows are unassigned
2 $\quad \mathcal{L}^* = \{(i,j) | n_{i,j} > 0\}$ .... all active links
3 $\quad p_{i,j} = 1,\ (i,j) \in \mathcal{L}^*$ .... all links have full capacity
Iteration:
1 $\quad$ for all links $(i,j) \in \mathcal{L}^*$:
$\qquad m_{i,j} = \sum_{k\in\mathcal{O}} |\mathcal{P}_k \cap \mathcal{D}_{i,j}|$ .... nominal load
$\qquad z_{i,j} = p_{i,j}/m_{i,j}$ .... throughput share per flow
2 $\quad (s,t) = \arg\min_{(i,j)\in L^*} z_{i,j}$ ... bottleneck collision domain
3 $\quad \mathcal{B} = \{k \in \mathcal{O} | \mathcal{P}_k \cap \mathcal{D}_{s,t} \neq \emptyset\}$ .... bottleneck flows
4 $\quad b_k = r \cdot z_{s,t}$ for all $k \in \mathcal{B}$ .... set bottleneck rates
5 $\quad \mathcal{O} = \mathcal{O} \setminus \mathcal{B}$ .... adapt unassigned flows
6 $\quad p_{i,j} = p_{i,j} - \sum_{k\in\mathcal{B}} |\mathcal{P}_k \cap \mathcal{D}_{i,j}| \cdot z_{s,t}$ ....
.... adapt free capacity of all collision domains
7 $\quad \mathcal{L}^* = \mathcal{L}^* \setminus \mathcal{D}_{s,t}$ .... adapt active links
Stop criterion: $\mathcal{O} = \emptyset$

Fig. 1. Collision domain based algorithm for end-to-end flows.

The iterative algorithm delivers max-min fair rates for the nominal load of collision domains. In [2] however, the effective load of a collision domain is introduced which is lower than the nominal load. Let us explain this by means of the simple example network shown in Fig. 2(a) assuming asymmetric collision domains. The network consists of 5 nodes on a chain with two gateways at the ends. Node 2 is routed over gateway 1 and nodes 3 and 4 are routed over gateway 5. The collision domain of link (4,3) comprises all active links such that $m_{i,j} = 4$, $z_{i,j} = 1/4$, and $b_k = r/4 = 13.5$ Mbps if we assume that all links offer a bandwidth of 54 Mbps for idealistic 802.11a/g. However, though the transmissions on links (1,2) and (5,4) can not take place simultaneous to the transmissions on link (4,3) they can take place in parallel to each other. Thus, another possibility to assign the rates would be $b_3 = b_4 = \frac{1}{3}r = 18$ Mbps and $b_2 = \frac{2}{3}r = 36$ Mbps. Then, transmissions from 4 to 3 take place in a third of the time, and for two thirds of the time transmissions from 1 to 2 and from 5 to 4 take place in parallel, the latter one consisting of one half with destination 4 and one half with destination 3. Obviously, this solution leads to a higher total throughput and a higher minimum throughput that is still max-min fair.

In [2] the effective load of a collision domain is determined by considering pairs of links within a collision domain that can transmit simultaneously. The load of the less loaded link is removed from the load of the collision domain. An algorithm how to proceed in more complex cases with many overlapping pairs of links with simultaneous transmissions is not specified.

(a) Network topology and routing (b) Contention graph

Fig. 2. Simple example network

In [5] such an algorithm is given considering max-min fairness however among links instead of end-to-end flows. The algorithm is based on a so called contention graph that is separated into cliques. In the contention graph, the active links, i.e. the links actually carrying traffic become nodes and two nodes in the contention graph are interconnected by an edge if parallel transmissions are not possible. Fig. 2(b) shows the contention graph of our example network. We have three nodes and two edges, one between nodes (1,2) and (4,3), and one between nodes (4,3) and (5,4). The algorithm for determining the max-min fair rate allocation is based on the maximum cliques of the contention graph. A subset of nodes is called a clique if an edge exists between any distinct pair of nodes. We define $\Omega_{\mathcal{C}}$ as the clique corpus, i.e. the set of all cliques $\mathcal{C}$ in the contention graph that are not subset of another larger clique. In the example network, we find two cliques, one with links (1,2) and (4,3) and one with links (4,3) and (5,4). In the context of mesh networks, a clique in the contention graph corresponds to a set of links no pair thereof can transmit in parallel. Now, the load $m_{\mathcal{C}}$ of a clique $\mathcal{C}$ corresponds to the degree of the clique, i.e. the number of links forming the clique. The clique based algorithm computes the max-min fair rate allocation analogously to the collision domain based algorithm.

Please note again that the algorithm yields a max-min fair rate allocation for one hop flows only. In the next section we will make a simple modification in order to consider multi-hop end-to-end flows, as well.

Let us now finally discuss the extension of [1] for the multi-channel case. The algorithm is analogous to the one formulated with collision domains, except that now we have collision domains for every channel. Note that $q_{i,j}$ is the channel assigned to link (i,j). Then, the multi-channel collision domain $\mathcal{D}^+_{i,j}$ is the subset of those links in the single channel collision domain also using channel $q_{i,j}$:

$$\mathcal{D}^+_{i,j} = \{(s,t) \in \mathcal{D}_{i,j} \mid q_{i,j} = q_{s,t}\}. \tag{3}$$

IV. Extensions

In this section we present two extensions to the existing max-min fair rate allocation algorithms. First, we modify the clique based algorithm for end-to-end multi-hop flows. Second, we consider the case of different but static link rates.

A. End-to-end Flow Extension of the Clique Algorithm

In the original algorithm proposed in [5] the load of a clique is equal to the number of links in the clique independent on the number of flows a link carries. Support of end-to-end flows is achieved by changing the load of a clique $\mathcal{C}$ to the number of single hop transmissions over the links of the clique. This leads to the following definition of the clique load:

$$m_{\mathcal{C}} = \sum_{(i,j)\in\mathcal{C}} n_{i,j} \tag{4}$$

The resulting algorithm is shown in Fig. 3. The key difference is in lines 6 and 7 where the algorithm adapts the capacity and load of all cliques used by any flow also using the bottleneck clique. In the algorithm of [5] only the cliques sharing a common link with the bottleneck clique required an adaptation. This algorithm determines the max-min fair rate allocation according to the notion of effective collision domain load introduced by [1]. We refer to this algorithm as effective load based algorithm (ELBA) in contrast to the nominal load based algorithm (NLBA) of Fig. 1.

B. Extension for Different Link Rates

To the best of our knowledge, all previously published theoretical max-min fair share algorithms for mesh networks work with equal data rates on all links only. In this section we propose an algorithm for different but static link rates. The described algorithm is an extension to ELBA but the multi rate extension can be applied to NLBA analogously. Again, the key modification is in the definition of the clique load. Let us first have a look at the capacity of a clique. In the single-rate algorithm this capacity is equal to the nominal link bandwidth. In the multi-rate case, we interpret the clique capacity as a basic unit of time that is shared among the flows running through a clique. If a flow with rate b_k is transmitted over link (i,j) with rate $r_{i,j}$ it requires a percentage of $b_k/r_{i,j}$ of the link's bandwidth and the link is active for $b_k/r_{i,j}$ of the time. Let $\mathcal{K}_{i,j}$ be the set of flows using link (i,j). Then, the link is active for

$$\theta_{i,j} = \sum_{k\in\mathcal{K}_{i,j}} \frac{b_k}{r_{i,j}} \tag{5}$$

percent of the time. Extending this to a clique $\mathcal{C}$, we obtain an activity percentage of

$$\theta_{\mathcal{C}} = \sum_{(i,j)\in\mathcal{C}} \sum_{k\in\mathcal{K}_{i,j}} \frac{b_k}{r_{i,j}}. \tag{6}$$

Now, let $\mathcal{K}^a_{i,j}$ be the subset of flows with and $\mathcal{K}^u_{i,j}$ be the subset without rate allocations. From the obvious condition that a clique can be active to at most hundred percent, i.e. $\theta_{\mathcal{C}} \leq 1$, we obtain

$$b_{\mathcal{C}} = \frac{1 - \sum\limits_{(i,j)\in\mathcal{C}} \sum\limits_{k\in\mathcal{K}^a_{i,j}} \frac{b_k}{r_{i,j}}}{\sum\limits_{(i,j)\in\mathcal{C}} \sum\limits_{k\in\mathcal{K}^u_{i,j}} \frac{b_k}{r_{i,j}}} \tag{7}$$

Initialization:
1 $\mathcal{O} = \mathcal{F}$ .... all flows are unassigned
2 $\Omega^*_C = \Omega_C$ ....clique corpus
3 $m_{\mathcal{C}} = \sum_{(i,j)\in\mathcal{C}} n_{i,j}$, for all $\mathcal{C} \in \Omega_C$ ....load of the cliques
4 $p_{\mathcal{C}} = 1$, for all $\mathcal{C} \in \Omega_C$ ....all cliques have full capacity
Iteration:
1 $z_{\mathcal{C}} = p_{\mathcal{C}}/m_{\mathcal{C}}$ .... rate share per flow through clique $\mathcal{C}$
2 $\mathcal{C}^\dagger = \arg\min_{\mathcal{C}\in\Omega^*_C} z_{\mathcal{C}}$ ....bottleneck clique
3 $\mathcal{B} = \{k \in \mathcal{O} | \mathcal{P}_k \cap \mathcal{C}^\dagger \neq \emptyset\}$ ....bottleneck flows
4 $b_k = r \cdot z_{\mathcal{C}^\dagger}$ for all $k \in B$ ....set bottleneck rates
5 $\mathcal{O} = \mathcal{O} \setminus \mathcal{B}$ .... adapt unassigned flows
6 $p_{\mathcal{C}} = p_{\mathcal{C}} - z_{\mathcal{C}^\dagger} \cdot \sum_{k\in\mathcal{B}} |\mathcal{P}_k \cap \mathcal{C}|$ ....adapt free capacity
7 $m_{\mathcal{C}} = m_{\mathcal{C}} - \sum_{k\in\mathcal{B}} |\mathcal{P}_k \cap \mathcal{C}|$ ..... adapt load of the cliques
8 $\Omega^*_C = \{\mathcal{C} \in \Omega^*_C | m_{\mathcal{C}} > 0\}$ ...adapt remaining clique corpus
Stop criterion: $\Omega^*_C = \emptyset$

Fig. 3. Effective load based algorithm (ELBA) for end-to-end flows.

as the maximum rate available for unassigned flows. Consequently, we define the load of a clique as

$$m_{\mathcal{C}} = \sum_{k \in \mathcal{F}} \sum_{(i,j) \in \mathcal{P}_k \cap \mathcal{C}} \frac{1}{r_{i,j}}. \tag{8}$$

The extension of ELBA for multi-rates is shown in Fig. 4.

Let us shortly discuss lines 6 and 7 of the algorithm. Line 6 updates the fraction of time that is still unassigned for a clique. A flow k that traverses the bottleneck clique $\mathcal{C}^\dagger$ has a data rate of $b_{\mathcal{C}}^\dagger$. Let us consider a link (i,j) that transports flow k and can be either in the bottleneck clique or not. Then, flow k occupies a fraction $b_{\mathcal{C}}^\dagger / r_{i,j}$ of link (i,j)'s time. Consequently, the fraction of unassigned time of all cliques $\mathcal{C}$ including link (i,j) is reduced by $b_{\mathcal{C}}^\dagger / r_{i,j}$. In line 7, the load of a clique is consequently reduced by $1/r_{i,j}$ for every link (i,j) and bottleneck flow using this link. The extension to the case of multiple channels is straightforward. In the contention flow graph edges between links on different channels are simply removed. Otherwise, the algorithm remains exactly the same.

V. NUMERICAL RESULTS

This section is intended to illustrate the results of the different algorithms using some randomly generated mesh networks. We first describe the process how to generate a network and then show the difference of ELBA and NLBA with both symmetric and asymmetric collision definitions. We refer to the four scenarios as S-ELBA, A-ELBA, S-NLBA, and A-NLBA.

A. Generating Test Networks

A number of nodes and gateways are randomly placed on a rectangular grid with grid length d. Using a grid ensures a certain minimum distance between two nodes. Assuming adaptive modulation and coding, the data rate from i to j depends on the signal to noise ratio (SNR)

$$\gamma_{i,j} = \tfrac{T_x \cdot g_{i,j}}{N_0 \cdot W}, \tag{9}$$

at the receiving node j. Here, T_x is the transmit power (100 mW), N_0 is the thermal noise spectral density (-174 dBm/Hz), W is the system bandwidth (20 MHz), and $g_{i,j}$ is the path gain from i to j. The used path loss model [4] in decibel scale for a reference distance of 10 m and a path loss exponent of 4 is

Initialization:
1 $\mathcal{O} = \mathcal{F}$ .... all flows are unassigned
2 $\Omega_C^* = \Omega_C$ ....clique corpus
3 $m_{\mathcal{C}} = \sum_{(i,j) \in \mathcal{C}} \frac{n_{i,j}}{r_{i,j}}, \mathcal{C} \in \Omega_C$ .... load of the cliques
4 $p_{\mathcal{C}} = 1, \mathcal{C} \in \Omega_C$ .... all cliques have full capacity
Iteration:
1 $b_{\mathcal{C}} = p_{\mathcal{C}} / m_{\mathcal{C}}$ ....throughput per flow through clique $\mathcal{C}$
2 $\mathcal{C}^\dagger = \arg\min_{\mathcal{C} \in \Omega_C^*} b_{\mathcal{C}}$ ....bottleneck clique
3 $\mathcal{B} = \{k \in \mathcal{O} | \mathcal{P}_k \cap \mathcal{C}^\dagger \neq \emptyset\}$ ....bottleneck flows
4 $b_k = b_{\mathcal{C}^\dagger}$ for all $k \in \mathcal{B}$ .... set bottleneck rates
5 $\mathcal{O} = \mathcal{O} \setminus \mathcal{B}$ .... adapt unassigned flows
6 $p_{\mathcal{C}} = p_{\mathcal{C}} - \sum_{k \in \mathcal{B}} \sum_{(i,j) \in \mathcal{P}_k \cap C} \frac{b_{\mathcal{C}^\dagger}}{r_{i,j}}$ .... adapt free capacity
7 $m_{\mathcal{C}} = m_{\mathcal{C}} - \sum_{k \in \mathcal{B}} \sum_{(i,j) \in \mathcal{P}_k \cap \mathcal{C}} \frac{1}{r_{i,j}}$ .... adapt load
8 $\Omega_C^* = \{\mathcal{C} \in \Omega_C^* | m_{\mathcal{C}} > 0\}$ ... adapt remaining clique corpus
Stop criterion: $\Omega_C^* = \emptyset$

Fig. 4. Extension of ELBA for multiple link rates.

$$g_{i,j} = -140.046 - 40 \cdot \log_{10}(d_{i,j}), \tag{10}$$

where $d_{i,j}$ is the distance of nodes i and j in kilometers.

Adaptive modulation and coding now selects the modulation and coding scheme (MCS) k with an SNR requirement γ_k^* that is just smaller than the link's SNR $\gamma_{i,j}$. Correspondingly, the maximum data rate $r_{i,j}$ corresponds to the data rate r_k of the MCS k that has the largest SNR requirement below $\gamma_{i,j}$. The SNR requirement of a MCS is typically derived from SNR-to-FER (frame error rate) curves that are obtained by link level simulations.

Fig. 5 shows the SNR-to-FER curves for the transmission of an IP packet with 1500 Bytes payload over an AWGN channel for the modulation and coding schemes (MCS) available for the different bandwidths supported by IEEE 802.11a/g. The SNR requirements γ_k^* are chosen to meet a FER of 1%. Two nodes that do not fulfill the SNR requirement of the most robust MCS BPSK 1/2 are assumed to be disconnected and also not to interfere with each other, i.e. there is no link between them.

If the assignment of link rates leads to separated network components without gateway node, one random node is selected as gateway node for this component. A simple routing algorithm is used for establishing a routing tree rooted at the gateway node. This algorithm leads to a not completely arbitrary but also not very sophisticated routing. The algorithm is not further specified here since the objective of this paper is not to optimize the routing but to evaluate the performance of a network with existing routing.

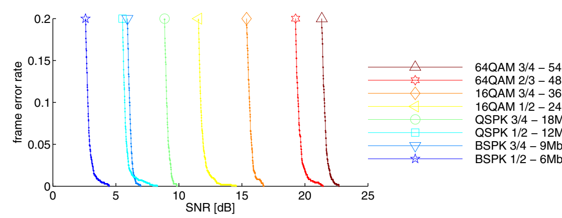

Fig. 5. FER for a 1500 Byte payload packet over IEEE 802.11a/g

B. Max-Min Fair Rate Assignments

Let us consider the example network in Fig. 6. Gateways are marked by squares, normal nodes by circles. The color and the width of the lines indicate the rate of the links. Solid lines are used for active links belonging to the routing tree and dotted lines are used for inactive links to neighboring nodes that are not used but may cause collisions. Fig. 7 shows the resulting rates per node for the four scenarios and gives some statistics on the obtained rates allowing a better comparison of the performance. Let us focus on S-ELBA. We observe that initially nodes 11, 12, 13, 16, and 20 obtain the lowest rates of about 1.9 Mbps. Their bottleneck clique comprises links (13,12), (18,13), (6,11) and (12,16). In the next iteration the bottleneck clique contains links (6,11), (10,5), (10,8), (10,9), (13,12) and (18,13), of which (6,11) and (13,12) are already bottlenecked. The flows running through this bottleneck clique belong to nodes 5, 6, 7, 8, 9, 11, 12, 13, 16 and 20. The still open flows 5, 6, 7, 8 and 9 obtain a rate of about 2.4 Mbps. The next iterations give a rate of 3.1 Mbps to flows 15, 17, 19, and 21 in the upper right corner of the network, and in the

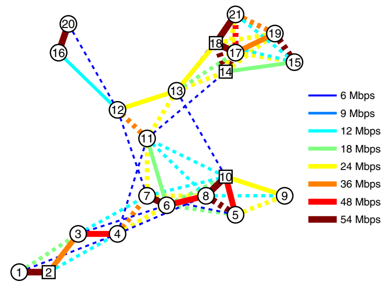

Fig. 6. Example network

lower left corner flows 3 and 4 obtain 9.2 Mbps and finally flow 1 obtains 16 Mbps.

Let us compare the results of the four different algorithms. Intuitively, one would expect that in general ELBA produces higher rates than NLBA, simply due to the tighter spatial packing of transmissions. And, one would expect that an asymmetric collision definition yields higher rates than a symmetric collision definition, since the links blocked by a transmission in the asymmetric case are a subset of those in the symmetric case. However, we observe that not all nodes profit as expected. Instead, considering effective instead of nominal load or asymmetric instead of symmetric collision definitions mainly increases the minimum throughput. With respect to the maximum throughput, we observe exactly the opposite: asymmetric collision definition and nominal load lead to a higher maximum throughput. The explanation is that the higher the bottleneck rates are in the first iteration, the more resources are occupied and the less resources remain for the flows in the final iterations. Regarding the mean or aggregate throughput, the results are rather similar, only clique-symmetric is discernibly worse. Consequently, the standard deviation increases for the symmetric and collision domain case which means that the rate allocations are less max-min fair.

Let us now shift from the example network to a more general case. For this purpose, we generate 100 networks according to the algorithm described in Section V-A with on average 100 nodes and 10 gateways placed on a 100×50 grid with a grid length of 10 m. Max-min fair rate allocations are computed for all networks. We compare the four algorithms by means of the cumulative distribution functions (CDFs) of the minimum, mean, and maximum rate per generated network. The CDFs are shown in Fig. 8. Let us first consider the minimum throughput. NLBA leads to clearly lower minimal rates than ELBA. The difference of symmetric and asymmetric collision definitions is less significant. Looking at the mean rates, we can see that A-ELBA clearly produces the highest rates and S-NLBA the by far lowest ones. S-ELBA and A-NLBA are almost identical, i.e., the two effects seem to

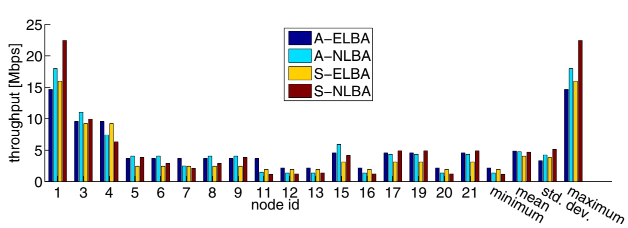

Fig. 7. Max-min fair throughput

compensate for each other with respect to the total throughput. Again, for the maximum rates we obtain a picture just opposite to that for the minimum rates: ELBA leads to smaller maximum rates than NLBA independent of the collision definition. The impact of the symmetric or asymmetric collision definition is again less significant.

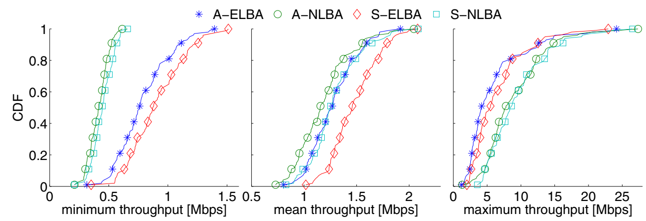

Fig. 8. Minimum, mean, and maximum throughput for generic networks

VI. Conclusion and Outlook

In this paper we focused on the problem to determine max-min fair rates in a mesh network with previously known static routing and channel assignment. We summarized the current status of research on max-min fair rate allocations in general and developed two extensions. First, we extended the algorithm of Huang and Bensaou [5] from fairness among links to fairness among end-to-end flows which essentially means that we provided a formal algorithm, ELBA, following the idea of the effective load of a collision domain. This idea was already introduced by Aoun and Boutaba [2] though no detailed algorithm was given formulated. ELBA determines max-min fair rate allocations in a multi-gateway, multi-radio mesh network with equal rates for all links. Our second contribution is the extension of effective and nominal load based algorithms to support heterogeneous link rates. The numerical results mainly compare the rate allocations achieved for a nominal and effective load definition. Additionally, we investigated the impact of symmetric or asymmetric collision definitions. We could show that the major impact of using the effective or nominal load definition is on the minimum and maximum rate while the main impact of a symmetric or asymmetric collision definition is on the mean or total throughput.

One critical point is how to determine collision domains and how to select modulation and coding schemes. In this paper we used a rather simplistic scheme, only. We plan to extend this work for more realistic collision domains and to examine the impact of different methods for selecting modulation and coding schemes.

References

[1] N. Akhtar and K. Moessner. On the nominal capacity of multi-radio multi-channel wireless mesh networks. *Computer Communications*, 31:1475–1483, 2008.

[2] B. Aoun and R. Boutaba. Max-min fair capacity of wireless mesh networks. In *MASS*, 2006.

[3] D. Bertsekas and R. Gallager. *Data networks*. Prentice-Hall, 1987.

[4] A. Goldsmith. *Wireless Communications*. Cambridge University Press, 2005.

[5] X. Huang and B. Bensaou. On max-min fairness and scheduling in wireless ad-hoc networks: Analytical framework and implementation. In *MobiHoc*, 2001.

[6] J. Jun and M. L. Sichitiu. The nominal capacity of wireless mesh networks. *IEEE Wireless Communications*, 10(5):8–14, October 2003.