MAC-aware Routing in Wireless Mesh Networks - A survey

Alexander Ende
Freie Universität Berlin
Seminar: Telematik

***Abstract*—In the last years Wireless Mesh Networks (WMNs) became more and more important and due to that the number of clients and the amount of data transferred over those networks grew constantly. The main reason why WMNs earned more attention is because they are an easy and cheap alternative to connect large areas with the internet, that maybe could not be connected with wires before. Conservative routing algorithms, especially those that are used frequently in today's wired networks are not performing well in a wireless environment. They need to be optimized and configured regarding to the properties of WMN links. Further on, that means to specify routing metrics that are aware of all circumstances that may influence the performance of those networks. I will give an overview of current approaches that are trying to improve the performance of WMN links by using routing algorithms and metrics tailored especially for use within a wireless environment. More precisely, this essay will cover the approaches that try improve the throughput by incorporating the given mechanisms of the 802.11 MAC layer.**

I. Introduction

The importance of WMNs has been increasing constantly over the last years. The situations where such networks can improve network coverage or even make network access possible at all are manifold. Deploying wired infrastructure can be expensive and difficult, but radio nodes of wireless networks become more and more reasonable, being a main criteria for the growing interest they get. WMNs can be used at any place and in many different scenarios. A popular use is a community network like introduced in [1] allowing a certain region where wires are no option to gain internet access. WMNs will have to deal with a growing amount of users and an increasing amount of data that will be transmitted. Therefore new routing algorithms and especially new routing metrics need to be established that are able to efficiently handle and balance the network traffic.

The main difference between today's wired networks and their wireless counterparts is that the wireless network links all use a single shared medium to transmit their data. This leads to some serious differences how access to the medium has to be handled by the several participants of the network. For wired Ethernet networks the IEEE defined the 802.3 standards [2], where *Carrier Sense Multiple Access with Collision Detection* (CSMA/CD) is specified as access control algorithm. However, CSMA/CD is not designed for wireless networks, because a station is not able to detect a collision at the same time it already transmits one. The IEEE 802.11 standard [3] for wireless networks therefore defines the 802.11 DCF mechanism. DCF, being synonymous with *Distributed Coordination Function* uses *Carrier Sense Multiple Access with Collision Avoidance* (CSMA/CA) to control the medium access of the network. On the one hand collisions are avoided, but on the other hand delay times are increased and so the overall throughput is decreased in zones with many links. In this essay I am focusing on metrics for Wireless Mesh Networks with a multihop characteristic. This means we will look at a network consisting of several nodes, and there might be more than one path available between two transmitting stations. These nodes can either be single radio nodes and only be able to transmit or receive. Or they may be multi radio nodes and feature two or more radios to transmit on different channels at the same time. If several nodes compete for the shared medium, we are talking about two different kinds of interference. On the one hand we have *interflow* interference, where nodes transmitting different dataflows are interfering with each other, and on the other hand we have *intraflow* interference, where nodes are interfering while carrying the same dataflow.

Most routing algorithms need every link in the network to be weighted so they are able to calculate the optimal route. This link weight is determined by a *routing metric*. Well known metrics are for example link capacity, hop count or the link delay. In most cases these metrics are used together with a shortest-path algorithm to determine the best routing path. These commonly used routing metrics are not performing well for WMNs because here we have the situation that several links contend with each other in a so called *contention domain*. The more links in such a contention domain, the more difficult it is to determine if one path really performs best carrying the upcoming dataflow.

The most important aspect of routing in a wireless medium is that the weight of a link changes when another routing path is chosen. The routing itself has impact on the link quality and that is where MAC-aware routing steps in. It is not reliable to count the hops or measure the nominal bit-rate, we need more accurate and more precise routing metrics that cover the aspects of the radio transmission medium.

The remainder of this essay is structured as follows. In section II I will explain the 802.11 MAC layer basics and important vocabulary. In section III I will introduce the ETX metric. ETX itself is not MAC-aware but many MAC-aware metrics build upon ETX. Section IV will be the main part,

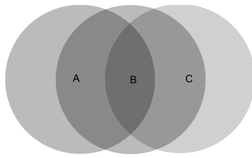

Fig. 1. Hidden station problem: Station A senses the medium and starts a transmission. It fails to register the already ongoing transmission from C to B and cripples all signals.

where I present several MAC-aware routing algorithms and metrics, which have been developed over the last few years. Finally, section V will be the conclusion.

II. MAC Layer basics for wireless LAN

The main difference between wireless and wired networking is that today all wired networks are fully switched and no collisions occur anymore. In a wireless network this differs completely. Collisions will occur if they are not avoided with a special mechanism. A wireless station that is sending its frames is usually not able to detect a collision at the same time. Further on it is very likely that not all stations in wireless networks can hear each other. That means it may be that a station that wants to send may sense the medium free but in fact it just can not hear another station currently transmitting its data to the same receiver. The MAC Layer of a wireless LAN needs to offer optimized methods to handle those difficulties. I will give a brief overview because it is important to know the basics to understand how routing may be improved by incorporating the MAC Layer.

A. 802.11 DCF

The main mechanism that controls medium access in a wireless LAN is the *Distributed Coordination Function*. The IEEE 802.11 standard requires network interfaces to implement the DCF which is based on *Carrier Sense Multiple Access* (CSMA). Using CSMA a station that wants to send a frame over the shared medium first has to sense that medium for ongoing transmissions. If the channel is idle, it immediately starts to transmit its frames. With pure CSMA it could happen that two stations start sending at nearly the same time and the frames collide. Therefore CSMA was extended with a collision avoidance mechanism. Now a station that senses the medium as free does not start to send its frames immediately, instead it waits a random time taken from an interval called *contention window*. It is unlikely to happen that two stations contending for the channel choose the same random timer and with that repeatedly occurring collisions are avoided. If it happens after all, both stations have to backoff again and the contention window is doubled. If several transmissions want to access the medium, DCF provides an equal amount of fairness to all contending links, a main reason why performance of wireless networks heavily depends on how many active links are together in each others contention domain.

B. RTS/CTS

Another problem occurring in wireless networks is that not all stations can hear each other. In a wireless LAN it may be that two stations A and C are not in each others sensing ranges. If station A now senses the medium because it wants to send data to station B it may happen that the medium is apparently free. But due to an ongoing transmission from station C to station B it was not and station A's transmission now cripples both transmissions without that station A or C even recognize the collision. This is called the *hidden station problem* and one of the major problems of wireless network traffic. Figure 1 shows an example.

A similar problem appears when two sending stations can hear each other but station A is not in range from station B's receiver and vice versa. Both senders sense the medium as occupied and the access control denies their sending requests. Nevertheless both transmissions could have been successful because the receivers are not contending with each other. This is what we call the problem of an *exposed station*.

To handle these situations 802.11 DCF offers the use of *request to send* and *clear to send* packets. A station that wants to transmit does a *clear channel assessment* and then first sends an RTS packet instead of immediately beginning the transmission. The RTS packet includes information about how long the transmission will be and all stations that receive it set their *network allocation vector* to that specific value. The NAV is part of a mechanism called *virtual carrier sense* and represents a timer that every station has. This timer only runs during the time the medium is really available. While the NAV is greater than zero a station will not try to access the medium, because it knows that there still is a transmission going on. After receiving a RTS packet the receiver answers with a CTS packet. The CTS enables all stations in the receiver's signaling range to be aware of that transmission as well and to set their NAV to the corresponding time.

C. Contention domain and interference

The research that focuses on MAC-aware routing protocols mainly deals with the problem how links are contending and interfering with each other. Links that are in each others signaling range are situated in the same *contention domain*. The problem is that the link quality changes due to transmissions happening in a certain contention domain. A new link means more contention and more interference in that particular zone of the network. Establishing new routes within a contention domain means the link quality of all contending links decreases. Quality in this aspect means overall throughput and delay of a link. A contention domain describes a neighbourhood of several links where a transmission of one link interferes with another. Important characteristic of wireless networks is, that signals become useless if they had to bypass a too long distance. The receiver will not be able to understand that signal, because it contains to many errors. Problem here is, that the receiver still can hear the signal. On the one hand we are speaking about the *transmission range* and on the other hand about the *interference range*.

The interference range is always bigger than the transmission range.

MAC-aware routing protocols need to investigate link loss rates and interference levels to be able to establish efficient paths. A major difficulty is the interference between links. If a routing algorithm sets a new routing path all the paths that are in its contention domain need to be recalculated. This process is not trivial and has a high combinatorial part. A routing algorithm that does not take the MAC interaction into account just reads the link weights as metric and uses for example *shortest path routing* to determine the best route. A MAC-aware algorithm on the other hand changes the quality of a link by deploying a new route. This means link quality changes when new routes are established and the link weights are no fix values that can be easily read out like hop count or nominal bit rate. The protocol has to check how a route will change the link quality of all other links before it deploys the new route because it may be that by adding the new route the situation changes and needs to be reviewed.

To give an example of the problems MAC-aware routing protocols have to deal with, think about a network where station A wants to send data to station C. It may send the packets over a direct connection to station C that got a high nominal bitrate and as said is only one hop but it is part of a zone with high contention between links. The second possible route has two hops with station B in between and a noticable lower nominal bitrate but nearly no contention with other links. A routing algorithm using minimal hop count now takes the direct link. A routing algorithm that takes the nominal bitrate as metric also would take the direct link because it seems to be faster. What both algorithms are not aware of is that the direct link resides in a contention domain with much more contending links than the indirect slow link. It is quite possible that the slower indirect route would be the better one after looking at link interference.

D. Interflow vs. intraflow interference

Two major types of interferences are occurring in a mesh network. On the one hand we have *interflow* interference *between* the different flows in a network. Each flow uses a different route with several links. If one or more links are in another flows contention domain these flows suffer interflow interference. It is difficult to calculate and exactly predict interflow interference because more than one path is involved and the level of interference changes constantly.

To handle a routing path as isolated when applying a metric is much more manageable and economic performance wise. Interference between the links of one single path is called *intraflow* interference. Intraflow interference occurs between the nodes of one path. Those nodes that are far enough from each other do not suffer intraflow interference because they are not in each others signaling range.

E. Congestion and load balance

If in a particular region many active links try to access the medium, it may happen that due to interference and bandwidth sharing nearly no traffic comes through. If that happens, we are talking about *congestion*. An efficient algorithm tries to avoid using links that tend to be congested by establishing routes around that zone and with that try to balance the load of the network. Bottlenecks are a major problem of wireless networks and routing algorithms have to avoid them. We refer to that as *load balancing*. The most important aspect of load balancing is about balancing between gateways of mesh networks. Gateways are most of the time the nodes with the highest load because they are connecting the whole network to the internet. That means all traffic from the network designated to the internet, has to pass through a gateway.

III. Two basic MAC-agnostic representatives

The two routing metrics presented in the following section are not interference aware or taking into account channel utilization and link contention. Briefly said, they are not MAC-aware like those metrics we will discuss later on, but they offer a good overview about the problematics of routing metrics in wireless lans and are as well used as starting point for MAC-aware metrics.

A. ETX with DSR and DSV

The first metric we take a look at is called the *Expected Transmission Count* introduced in [4]. Its value reflects the amount of transmissions that will be needed for sending a packet over a link and mainly covers the link loss rate. Let us assume the link is perfect and neither packet loss occurs nor the link suffers any interference. The ETX value would be one for a one-hop path and for example five for a perfect five-hop route. A route's ETX is the ETX of each link summed up. ETX calculates these values for both directions of a link and takes into account the possibility that link loss rates are asymmetric. The probability that a packet from a sending station is received without errors is called forward delivery ratio and identified by d_f. Because of the mentioned asymmetry the value of the link direction from receiver to sender is also calculated. This corresponds to the ACK packet that will be sent acknowledging the original packet and is called reverse delivery ratio d_r. The term $d_f \cdot d_r$ reflects the probability that a packet is sent correctly and that its ACK packet is received without failure.

The dependence on delivery ratios in both directions directly implies a relationship with throughput of a certain link. The fewer loss a link has the lesser its overall throughput is. Important is that the link is not seen as one link but as two. In a real network it is likely to happen that two stations do not have the same transmission range and a link might have more packet loss in one direction than in the other and with that the throughput differs as well. In other words ETX is able to deal with link asymmetry often occurring in wireless networks. Further on, if we compare short routes with long routes, ETX prefers the shorter ones. Long routes mean more devices are interfering with each other. A node receiving a transmission can not send at the same time and that means lower throughput compared to a link with fewer hops. This is true for routes using up to three hops. A route using four or even more hops maybe is not suffering the same interferences and even may be faster than a shorter route. However, it may be that ETX chooses the slower shorter route. Not a main feature but a nice side effect is that ETX lowers energy consumption. That is quite understandable because it minimizes the total transmission count and with that the number of retransmission needed.

To determine the value of the delivery ratios d_f and d_r ETX needs dedicated link probe packets to be send. These packets have a fixed size and are send via broadcast every τ seconds. A node that receives such a probe packet has to remember it for a timespan w. So every node is able to calculate the delivery rate of the sender because it knows how many probe packets it should have received and how many it actually did receive. Even if no probe packets arrive at a certain station it is still able to calculate the delivery ratios because it knows exactly when there should have been packets. Further on each probe packet includes the number of probing packets the sender received from its neighbours during the time period w. This enables all nodes that received that packet to calculate the forward delivery ratio to the sender.

ETX has been implemented for testing with the DSDV [5] and DSR [6] routing algorithms in a testbed using 29 802.11b network cards. The nodes themselves are linux based PCs that are positioned in a common office building and located on different floors.

Too measure performance gains ETX was directly compared with DSDV and DSR using minimum hop count as metric. Testcases were made with packetsizes of 134 byte and 1386 byte both sent with either 1 mW transmit power and 30 mW transmit power. Small packetsizes resulted in a greater advantage for ETX using DSDV while using bigger packages closed the gap between ETX and minimum hop a bit. This is related to the circumstance that ETX probing packets have a size of 134 byte and ETX underestimates the expected transmissions of certain links when big packages are used while it overestimates the estimated transmissions with packets smaller then 134 byte, for example ACK packets which have a size of 38 byte. When sending with a higher transmission power the network nodes gain more range and with that the number of hops decreases. The possibility that minimum hop count chooses a good route raises and the overall possibility that ETX can choose from is lower than before, with the consequence that the performance gain is not as high as with more hops available.

The DSR algorithm gains a mentionable performance boost when determining the initial routes. Minimum hop count has to randomly choose one of the shortest routes while ETX has far more choices. DSR itself uses a method called *Link Layer Transmission Feedback* to avoid bad routes. The performance boost was measured with link-layer feedback set to off. When link-layer feedback was enabled the performance boost was not as high and DSR nearly performed as well as ETX. Regarding packetsizes DSR showed the same characteristics as DSDV.

B. Expected Transmission Time

Now, ETT 10 is a slightly modified version of the original ETX metric and can be represented as follows:

$$ETT = ETX * \frac{S}{B} \quad (1)$$

To be able to evaluate that calculation the forward and reverse loss rates and the bandwith of that corresponding link need to be known. Again the loss rates are determined by using the link probe broadcast mechanism already introduced with the ETX metric. Measuring the bandwidth of a link appears to be more complicated. Modern wireless radio interfaces are using an autorate mechanism that is able to set the bandwidth for each single packet. The network interfaces used in the testbed where ETT was applied did not offer the functionality to read out current bandwidth values so they chose to use a technique called packet pairs [7] to measure it. Packet pairs means that two packets, a small and a large one are send repeatedly over a link and back, and with that the average link bandwidth is calculated.

Overall, the authors call ETT the ‘bandwidth adjusted ETX’. Typical characteristics of MAC-aware routing metrics are only adressed indirectly. If several nodes are contending for a link, the link loss is increasing due to interference and collisions. Furtheron the available bandwidth is decreasing with more nodes in a contention domain. ETT treats packet loss as an independent parameter that is not related to the network topology, and it uses the total capacity of a link instead of the currently available bandwidth. However by probing the links frequently, changes to the packet loss ratios are taken into account afterall. Better mechanisms to measure the bandwidth and incorporating them into ETT are future topics of the authors’ work. As already said, only small changes were made. Later on I will introduce the path metric WCETT [8], that further improves ETT and adds important features upgrading it into the category of MAC-aware routing metrics.

IV. MAC-aware routing metrics and algorithms

In the following main section of this essay I will present routing metrics and algorithms that have been developed over the last few years. Their main goal is to improve overall Wireless Mesh Network performance by overcoming the problems that standard routing metrics used with common routing algorithms have regarding to the wireless medium. Research is focusing on intraflow and interflow interference and possible performance increasing solutions using cross layer approaches incorporating the MAC layer.

A. Expected Throughput ETP

The *Expected Throughput* metric proposed in [9] focuses on improving the network throughput by taking into account the bandwidth sharing mechanism of *802.11 DCF*. DCF tries to offer fair channel access for all contending links but it does not care about nominal bit rates. A fast and a slow link both are treated the same. If a fast and a slow link contend with each other, the DCF mechanism provides an equal amount of channel access possibilities to both of them. That means that the slow link will have the same number of transmissions like the fast one and it is obvious that the slow link will occupy the channel for a longer time then the fast one. The slow link hinders the faster one what often leads to notably decreases in throughput, that is where ETP tries to step in.

$$t = \frac{mL}{r_1} + \frac{mL}{r_2} \Rightarrow \frac{mL}{b} = \frac{mL}{r_1} + \frac{mL}{r_2} \quad (2)$$
$$\Rightarrow b = b_1 = b_2 = \frac{1}{(\frac{1}{r_1} + \frac{1}{r_2})}$$

To point out the problem of bandwidth sharing the following calculations are presented by the ETP developers. Looking at a timespan of t seconds assume that m is the number of transmissions each link gets. Both links have the same packetsize L but different nominal bitrates of $r1 = 54Mbps$ and $r2 = 1Mbps$. Equation 3 shows how that leads to an overall throughput of only $0.98Mbps$. It is understandable now that the nominal bitrate of contending links has a huge impact on link quality.

$$b_k = \frac{1}{\sum_{j \in S_k \cap P} \frac{1}{r_j}} \quad (3)$$

To find the best route between two nodes in a mesh net ETP uses the following approach. At first ETP determines each possible path P between those nodes called candidate path. A link that is part of such a path P is identified as k. Now a set is defined that includes all links of the whole network that contend with that particular link k. In other words the contention domain of k is calculated. All links in k's contention domain without the links that already are in the current candidate path are links interfering with k. Let r_k be the nominal bitrate of link k we can calculate the expected bandwidth that k has to expect by using formula 3. That means the denominator of the formula containing all links from k's contention domain now consists of the sum of all those links contending with k. The interference maintained by that ETP formula covers the intraflow interference coming from the current route itself. It is not measuring the overall interflow interference that occurs in the whole network caused by other routes.

$$ETP(k) = \frac{p_k^f \cdot p_k^r}{\sum_{j \in S_k \cap P} \frac{1}{r_j}} \quad (4)$$

However, equation 3 represents the expected bandwidth that link k will receive but usually in a wireless network the actual throughput is essentially lower due to packet loss rates

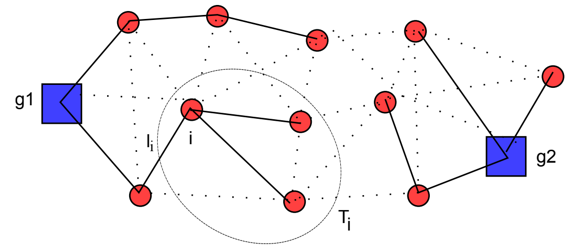

Fig. 2. MaLB routing trees

on that link. ETP accounts to packetloss by introducing the forward and reverse packetloss probabilities of link k into the formula. Let p_k^f and p_k^r be those packetloss probabilities we get formula 4. Now the metric of the whole path and not just the single link k is the minimum ETP value of all links from that path. A routing algorithm has to take the path with the highest possible value to pick the best route.

ETP can be used with any of the well known routing algorithms. All nodes just need to know about the nominal bitrate of their associated links and the ones contending with them. That can be implemented by sending probe packages or using beacon messages.

B. MaLB using ETP

The in [10] proposed routing algorithm is called *MaLB*, the MAC-aware and Load Balancing routing algorithm. Its aim is to improve network throughput by avoiding congested links and especially to offer a fair distribution of traffic between the different gateways of the network. It uses the already introduced ETP as metric. Some improvements were made to ETP that enable it to calculate interflow *and* intraflow interference. ETP assumes that all links in a network can hear each other but in most networks this is not the case and some links may not hear others. In formula 5 all active links are taken into consideration and not only those links on the current calculated path, so the bandwidth received by link l_i is approximated.

$$\frac{1}{ETP_{l_i}} = \frac{1}{P_{l_i}} \cdot \sum_{j \in M} \frac{l_{\{l_i \Delta l_j\}}}{R_{l_j}} \quad (5)$$

The analytical model splits the network into two subsets, on the one hand the set G containing all gateways and on the other hand M that consists of all other mesh nodes. Each mesh node $i \in M$ connects to only one single gateway. This means the whole network is partitioned into several subnets all with their own gateway. These subnets are viewed as disjoint trees T_{g_i} with the gateway g_i as root. The union of all these trees T is one possible routing that connects all nodes. A subtree that is rooted at a mesh node and not at a gateway node is called T_i. In figure 2 you can see solid and dashed lines where the dashed lines are links not used in the current routing trees and the solid lines are links active in the routing scheme. The link between node i and its parent node is called l_i and the probability that a packet transmission on link l_i will be successful is P_{l_i}. At last we need to know the nominal bit rate of each link, and identify that as R_{l_i}.

The modified ETP formula needs a link state variable that is set to 1 if two links l and j contend with each other and 0 if not. Figure 5 shows the whole equation. The sum of all the inverse data rates reflects the DCF bandwidth sharing while the term including P_{l_i} obtains the useful throughput of a link. It is important to mention that this equation models the situation where all contending links are able to hear each other. This is a simplification of real networks because it may happen that not all contending links are in each others transmission range. However this model is an approximation of the expected throughput that takes into account intraflow interference as well as interflow interference. The equation covers the impact of routing on link quality.

Further on a cost function $D(T)$ is introduced that reflects the transmission delay for all nodes of T sending a bit to its associated gateway. It assumes that every node in a subtree T_i is treated equally and receives the same bandwidth and therefore the expected throughput of a node j over a certain link i is $\frac{ETP_{l_i}}{|T_i|}$. $|T_i|$ is the number of nodes in each subtree with root i. The overall cost function is shown in equation 6. The routing algorithm now has to find a routing forest that minimizes the value of D.

$$D(T) = \sum_i \frac{|T_i|^2}{ETP_{l_i}} \tag{6}$$

The proposed algorithm MaLB needs to calculate the delay value for each possible routing forest. Because the link weights are not fixed it is not possible to use a shortest path approach. At start the network needs to already be in a forest topology. MaLB then starts to reconfigure this topology step by step. Therefore each node has a timer that counts down after the network bootstrap has completed. After the counter reaches zero node i searches for the best parent node to which it can migrate with its whole subtree. The process of migration has an impact on the ETP value and the subtreesize of all nodes in i's contention domain. Therefore MaLB first determines all nodes S_i that may be i's new parent node. After that it finds all nodes H_{ik} that would be affected if node i was migrating to node k. Then it calculates how the cost function would change when node i migrated to k and determines the parent node that results to the highest reduction of it. If a parent is found that leads to a better value the subtree migrates to this new node.

Every time a node migrates it has to inform its parent node of its subtree size. Further on every time the subtree size of a node changes all nodes of that subtree have to be informed. Each node has to put the bit rate and the packet success probability of each link that connects to itself in its beacon message.

The performance was tested in a simulated 100-node testbed running 802.11g network interfaces that use single channel radios. The used simulator was ns-2 [11]. MaLB was compared

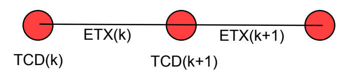

Fig. 3. Transmission contention degree

to ETX using a shortest path algorithm and to LB using ETP. LB is the MAC-agnostic version of MaLB. Testing showed that MaLB outperformed the others in a scenario where a gateway failed and the load had to be rebalanced and especially when the topology of the network was highly unbalanced. MaLB was able to adapt to gateway failures and achieve nearly 60% better performance.

C. Expected Data Rate (EDR)

EDR is a metric proposed in [12] that also builds upon ETX. It tries to increase the throughput of a wireless mesh net by introducing a new transmission interference model. Therefore it uses a new function named *transmission contention degree* which directly relies on wireless link loss and the medium access backoff mechanism. It also considers concurrent packet transmission on different hops unlike ETX that penalizes long routes by summing up the ETX value. Overall EDR determines a transmission interference degree for each link that is calculated by checking how many other links contend with that particular link and how that leads to retransmissions and how the medium backoff procedure of 802.11 DCF further delays the transmission on that link. Therefore it gains knowledge about the interference range of nodes and their ETX values.

The transmission contention degree *TCD(k)* is the average time where the outgoing queue of link *k* is not empty. With this method also retransmitted packets are considered. The TCD is calculated by the rate data arrives on that link and the rate the data leaves on that link. These rates depend on the links own loss rate and the loss rate and TCD value of the previous adjacent link in that path. Figure 3 shows that relationship and equation 7 shows how it is calculated. The complete contention degree of a link is called total *TCD* and is the sum $I(k)$ of the TCD value from all links that are in the contention domain of link k.

$$TCD(k+1) = min(1, TCD(k) \cdot \frac{ETX(k+1)}{ETX(k)}) \tag{7}$$

Now to calculate the EDR value of link k we need its ETX value and further on its $I(k)$ value. The data rate available on link k is reduced by $ETX(k) \cdot I(k)$ due to interference from neighbouring links. Let Γ be the optimal data rate of link k. Γ is influenced by several mechanisms like packet overhead, backoff procedure overhead and RTS/CTS/ACK procedures. To estimate these influences a reduction factor r is added. The EDR of an entire ad hoc path is calculated from the bottleneck link k' with the largest reduction of data rate. That means the one with the highest loss rate and leads to the following formula.

$$EDR(k) = \frac{r\Gamma}{ETX(k) \cdot I(k)} \tag{8}$$

However the presented EDR value does not take into account that the contention window size changes if transmission errors occur, but this is defined in the 802.11 DCF standard. This may result in even higher data reduction rates than those calculated by EDR at this point. Therefore the TCD equation is changed so that it contains the changes in the contention window of nodes and with that EDR is able to precisely take care about the MAC backoff mechanism.

EDR is compatible with DSR and AODV [13] routing algorithms. The developers tested it with DSR with ETX values added to the routing request packets. Overall EDR needs slightly more space in the packets and slightly more processing time at the nodes. The tests showed that EDR performed better than ETX in all cases. However the developers mentioned some possible improvements regarding to the link loss model applied. Future work will be about *temporally correlated link loss*.

D. MIC and LIBRA

The next representatives of MAC-aware routing metrics and algorithms are the metric MIC and the routing protocol LIBRA presented in [14]. MIC consists of two separate metrics that have been put together. The first one is IRU, *Interference-aware Resource Usage*, that deals with differences in transmission rates and loss rates of wireless links. The second one is CSC, *Channel Switching Cost* that cares about reducing intraflow interference between two adjacent nodes by using different channels for the transmission.

Now the $IRU_{ij}(c)$ value between node i and node j communicating over channel c is the sum of the expected transmission time ETT between these nodes and a set of nodes that may be involved in interference when a transmission on that link takes place. This takes into account the interflow interference between link (i, j) and this set of nodes. If (i, j) has a transmission active, no node from the set is allowed to send due to RTS/CTS and other mechanisms. The implementation of ETT into the equation captures the quality changes of links regarding transmission rates and loss ratios. When choosing a minimum weight path with IRU, network usage as well as network load, overall throughput and delay are improved.

The CSC value covers the multi-channel ability of mesh nodes. Nodes that are carrying a flow and are communicating over different channels can improve the throughput mentionable because adjacent nodes do not interfere with each other. In general, CSC penalizes a link if it is sending on the same channel as its predecessor on that path.

$$MIC(p) = \alpha \cdot \sum_{\text{link } l \in p} IRU_l + \sum_{\text{node } i \in p} CSC_i \tag{9}$$

By merging both components the new metric of *interference and channel-switching* (MIC) results. Equation 9 shows the exact relationship. It may happen that both components are conflicting with each other in certain situations. When a path p_1 uses more different channels and because of that has nearly no intraflow interference and another path p_2 on the other hand has more high bandwidth links that are sending on the same channel it becomes difficult to determine what path to prefer now. Therefore α is introduced to be able to set a focus to one of both choices. The higher the α value is the more MIC focuses on load balancing. A smaller α prefers per-flow performance without caring about interflow interference. Important characteristic of MIC is that it is not isotonic. That means it is not possible to calculate minimum weight paths without an exponential algorithm. However, the routing algorithm proposed by the MIC developers solves the non-isotonic property, by creating a virtual network ontop of the real network as briefly explained later, that allows running algorithms without exponential runtime.

LIBRA means *Load and Interference Balanced Routing Algorithm* and consists of four main components. The *Neighbour Management* component enables each node to measure all ETT values over each of it's channels to each of its neighbours. The *Routing Control* tells each node to propagate routing information to other nodes in the network. The *Route Determination* lets each node calculate minimum weight paths from the gathered informations and finally the *Routing Table Management* builds up routing tables at each node, containing these minimum weight paths. Distance Vector Routing as well as Link State Routing can be used as basic routing protocols for implementing neighbour management and routing control.

Finding a route with MIC is more complicated and needs the network to be split into a virtual network. For each radio a node owns two virtual nodes are introduced. That means a node that is able to send over two channels is represented as four nodes in the virtual network. One node for each direction it may communicate over a certain channel. We will not discuss the Virtual Network Model here but an important result of it is that LIBRA can be used with Dijkstra or Bellmann-Ford and does not create any loops when used with Link-State routing. Further on, the CSC part of the metric may be extended to be able to care about intraflow interference that covers more than two consecutive hops on a path. Also dynamical channel configuration is mentioned by the developers. This means that a wireless interface is able to change a channel while it is carrying a flow. Currently this is not efficient cause the process of changing the channel tends to be slow, but in the future it may be a possibility to further boost the performance.

During the performance tests the simulations showed that LIBRA came close to optimal performance and performed much better than ETT or Hop Count in most cases. The multichannel ability was the main reason for that performance boost. This was measured in testbeds with multiradio and multichannel interfaces. However MIC needs more performance at the nodes and uses a pretty complicated network model.

E. WCETT and the MR-LQSR Protocol

Now, by further improving the already introduced metric ETT by three more features, the authors enable it to become a true MAC-aware path metric. First, a path of n hops is treated as the sum of all the hops ETT on that path, like shown in the following formula.

$$WCETT = \sum_{i=1}^{n} ETT_i \tag{10}$$

This ensures that adding hops to a path also increases the value of the path metric overall. The next characteristic added is the channel diversity. Channel diversity refers to several hops operating on different channels. Simply adding up the ETT values on a path would not reflect the channel diversity. Each hop may be on a different channel, so now only those hops on the same channel are counted. This sum is called X_j and reflects the sum of transmission times of hops on channel j. The maximal throughput of a path will be decided by the channel with the highest X_j value. The highest X_j value means the lowest throughput. This leads to the following calculation:

$$WCETT = \max_{1 \leq j \leq k} X_j \tag{11}$$

Using formula 11 enables WCETT to prefer channels that are more channel diverse than others. However, using only this equation would not always penalize adding new hops to a path, because adding hops not using a bottleneck channel would not affect the metric. By combining equation 10 and 11 and taking their weighted average, WCETT is penalizing longer paths by summing up their ETT values, and it is taking into account the channel diversity of that path by taking it's max X_j value. The parameter β is used to implement the weighting of both values. This results to the following equation:

$$WCETT = (1 - \beta) * \sum_{i=1}^{n} ETT_i + \beta \max_{1 \leq j \leq k} X_j \tag{12}$$

The first part of the equation can be seen as the delay occuring on a path, the second part represents the throughput on it. By choosing different values for β, one is able to shift the importance of these two characteristics.

The routing protocol that integrates WCETT is called MR-LQSR, *Multi Radio Link Quality Source Routing*. It works like a normal LQSR routing protocol, just with some changes. First of all it of course uses the WCETT metric to find the best paths. For implementation and testing MR-LQSR has been integrated into a loadable Windows driver, called MCL (Mesh Connectivity Layer). This driver functions as an additional layer between layer 2 and layer 3 of the network stack.

The testbed were WCETT has been tested consists of 23 wireless nodes in a typical office building. Each node uses two different radios tuned to different channels. The first tests were made with only one radio enabled, to show how WCETT would perform in a normal single radio environment. Results for this were an improvement by 16% over ETX and by 38% over simple shortest-path routing. Another important result was, that the longer the path a transmission used was, the bigger the improvement achieved by using WCETT was. Overall, even in a single radio environment WCETT was able improve the throughput.

In a multiradio environment WCETT achieved the following numbers. The throughput compared to ETX increased by about 89% and compared to shortest-path routing even by 254%. This shows how WCETT is able to exploit the additional capacity offered by the second radio in each node.

	Throughput	Multichannel	Testbed
ETX	better than Min-Hop-Count	NO	29-nodes real world
ETP	$\sim$ 60% > ETX	NO	100-nodes simulated
EDR	finds a better path than ETX in 90% of all cases	NO	??-nodes simulated
WCETT	$\sim$ 89% > ETX $\sim$ 254% > Min-Hop-Count	YES	23-nodes real world
MIC	better than WCETT, not far from optimum	YES	160-nodes simulated

TABLE I
THROUGHPUT, MULTICHANNEL ABILITY AND TESTBED COMPARISON

V. CONCLUSION

It is obvious that there are many possible approaches to achieve better throughput in wireless mesh networks. All these approaches can be pretty different covering completely different aspects of the network. I introduced only a very small amount of current MAC-aware routing approaches as well as ETX, a metric that is often used as basic starting point for building up new metrics and to compare the performance gains.

To be able to compare the different approaches and to rate them we have to find criterias that allow us to measure their quality. The most important criteria is the improvement of throughput a certain metric achieves. All of the routing metrics introduced above do compare themselves by showing how much more throughput they deliver regarding to others. But also end-to-end delay and especially for a MAC-aware routing metric in a wireless lan, channel utilization are important comparable characteristics. Unfortunately, mostly only the throughput has been evaluated and published.

Table I shows the evaluation results of each metric, as they were published in the corresponding papers, regarding the throughput. It was not possible to exactly determine the numbers because the evaluation results were not always comparable. For example, the evaluation of ETX did not mention the exact numbers, but it showed that ETX performed better than Minimum Hop Count. The column Multichannel tells if the metric really uses the multichannel ability of the network interfaces. Testbed finally just shows what kind of testbed was used.

The first MAC-aware metric introduced is ETP that measures the expected throughput of a wireless link and closely incorporates the bandwidth sharing of 802.11 DCF. It is compatible with multiradio and multichannel network interfaces but does not explicitly use all their features. Then we took a look at a new routing algorithm MaLB proposed by the developers of ETP that focuses on load balancing over wireless mesh network gateways and uses a slightly improved ETP as metric. MaLB using ETP proved to perform well in networks where congestion on certain links is a problem. MaLB is able to spread the traffic and with that improve the overall throughput using a new cost function together with a graph theoretical approach. Next is EDR, a metric closely related to ETX. It defines a new model of measuring the contention degree of certain nodes by checking what nodes interfere with each other. Further on it closely looks at the medium backoff mechanism and takes into account the raising contention window size. EDR has been tested in a single radio environment but may be able to be implemented in multiradio environments as well. It can be used with any source routing or any distance vector routing algorithms. The metric MIC is our next candidate. MIC focuses on load balancing as well but takes a further step by really incorporating the possibility of multichannel radios and uses a cost function for channel switching. It is presented together with the new routing scheme LIBRA that solves the non-isotonic property of MIC. A virtual network is created to find routing paths in the real network. MIC is able to exploit the advantages of multiradio and multichannel environments and even can be configured to support dynamic channel switching.

An important result of this survey is that it is possible to gain nice boosts in performance by incorporating the MAC-layer. A major deficit of current network interfaces is the missing ability to use a wider range of the bandwidth because of the lack of multichannel usage. By enabling nodes to communicate on more than one channel it is possible to reduce interference and raise the throughput. The best performances have been achieved with multichannel nodes and a routing protocol that avoids congested zones in networks.

In my opinion future work will focus on research with multichannel and multiradio enabled nodes. Mesh networks will have to deal with much more traffic and so the protocols should be able to perform efficient load balancing around congested areas. With routing protocols that are able to recognize congestion and then react with channel switching and with that lowering congestion in a whole area, the performance of the whole network rises. However calculating those routing schemes needs more time and has much more complexity then simple common routing algorithms have. Nevertheless its obvious that future routing protocols will be incorporating MAC-layer mechanisms.

References

[1] *Self-Organizing Neighborhood Wireless Mesh Networks*, Microsoft Research. [Online]. Available: http://research.microsoft.com/mesh/

[2] *IEEE 802.3 standard*, IEEE 802.3 Working Group. [Online]. Available: http://standards.ieee.org/getieee802/

[3] *IEEE 802.11 standard*, IEEE 802.11 Working Group. [Online]. Available: http://standards.ieee.org/getieee802/11/

[4] D. S. J. D. Couto, D. Aguayo, J. Bicket, and R. Morris, "A high-throughput path metric for multi-hop wireless routing," *Wirel. Netw.*, vol. 11, no. 4, pp. 419–434, 2005.

[5] C. Perkins and P. Bhagwat, "Highly dynamic destination-sequenced distance-vector routing (DSDV) for mobile computers," in *ACM SIGCOMM'94 Conference on Communications Architectures, Protocols and Applications*, 1994, pp. 234–244. [Online]. Available: citeseer.ist.psu.edu/perkins94highly.html

[6] *DSR Routing Protocol*. [Online]. Available: http://www.cs.cmu.edu/ dmaltz/dsr.html

[7] R. Carter and M. Crovella, "Measuring bottleneck link speed in packet-switched networks," -, Tech. Rep. 1996-006, 15, 1996. [Online]. Available: citeseer.ist.psu.edu/carter96measuring.html

[8] R. Draves, J. Padhye, and B. Zill, "Routing in multi-radio, multi-hop wireless mesh networks," in *MobiCom '04: Proceedings of the 10th annual international conference on Mobile computing and networking*. New York, NY, USA: ACM, 2004, pp. 114–128.

[9] V. P. Mhatre, H. Lundgren, and C. Diot, "Mac-aware routing in wireless mesh networks," in *Wireless on Demand Network Systems and Services, 2007. WONS '07. Fourth Annual Conference on*, 2007, pp. 46–49. [Online]. Available: http://dx.doi.org/10.1109/WONS.2007.340470

[10] V. Mhatre, H. Lundgren, F. Baccelli, and C. Diot, "Joint mac-aware routing and load balancing in mesh networks," in *CoNEXT '07: Proceedings of the 2007 ACM CoNEXT conference*. New York, NY, USA: ACM, 2007, pp. 1–12.

[11] *ns-2 network simulator*. [Online]. Available: http://www.isi.edu/nsnam/ns/

[12] D. S. J. De Couto, D. Aguayo, J. Bicket, and R. Morris, "A high-throughput path metric for multi-hop wireless routing," in *MobiCom '03: Proceedings of the 9th annual international conference on Mobile computing and networking*. New York, NY, USA: ACM, 2003, pp. 134–146. [Online]. Available: http://portal.acm.org/citation.cfm?id=938985.939000

[13] C. Perkins, "Ad-hoc on-demand distance vector routing," 1997. [Online]. Available: citeseer.ist.psu.edu/article/perkins97adhoc.html

[14] Y. Yang, J. Wang, and R. Kravets, "Interference-aware load balancing for multihop wireless networks," University of Illinois at Urbana-Champaign, Tech. Rep., 2005. [Online]. Available: http://www.cs.uiuc.edu/research/techreports.php?report=UIUCDCS-R-2005-2526