A Novel Link Quality Assessment Method for Mobile Multi-Rate Multi-Hop Wireless Networks

Jinglong Zhou, Martin Jacobsson, Ertan Onur and Ignas Niemegeers
Faculty of Electrical Engineering, Mathematics, and Computer Science
Delft University of Technology, Mekelweg 4, 2600 GA Delft, The Netherlands
Emails: {J.Zhou, M.Jacobsson, E.Onur, I.Niemegeers}@ewi.tudelft.nl

***Abstract*—Accurate and fast wireless link quality assessment (LQA) for wireless channels would bring in huge benefits for mobile multi-hop and multi-rate wireless ad hoc and sensor networks in the form of improved end-to-end performance. In this paper, we propose a novel LQA method based on cross-layer information. The algorithm is implemented in a real test bed, which is based on IEEE 802.11, and achieved a significant LQA improvement up to 50 % in mobile scenarios without introducing overhead. The effectiveness of accurate and fast LQA is demonstrated by feeding it into the routing layer to enable the route decisions to adapt faster and more accurately during changing situations, especially in mobile scenarios. The experiment results show that the proposed LQA method can lead to faster and smarter routing decisions and higher end-to-end throughput compared to traditional methods in the mobile scenario.**

I. Introduction

For mobile wireless ad hoc and sensor networks, the link quality affects the performance since higher layer application's performance are very sensitive to the link dynamics. Thus, lots of higher layer applications require better link quality assessment (LQA), such as data rate adaptation, routing, QoS provisioning, gateway selection, video transmission optimization and so on. However, indoor wireless channels are dynamic in nature, link quality can change dramatically in a short time period due to small space, complex environment and person movements. Traditional LQA methods, which are based on one layer information, can not provide an accurate link quality estimation due to limited knowledge. Therefore, a combination of information sources from different layers is required to help the higher layers to improve the performance. This suggests a cross-layer design.

A direct application of LQA is link quality-based routing. Hopcount-based routing may produce routes with poor links. Hence, some prior proposals suggested to try to minimize the number of transmissions needed to reach from the destination to the source, such as expected transmission count (ETX) proposed by De Couto *et al.* [1]. Furthermore, when using a multi-rate network, such as IEEE 802.11g, not only the packet losses are considered for route selection, but also the data rate, such as the expected transmission time (ETT) [2].

This work is supported by the Freeband PNP2008 and IST MAGNET Beyond research projects.

In both ETX and ETT as well as most other works, link quality estimation is based on hello packet delivery ratio. Slow reaction to link dynamics and inaccuracy are the problems of such probe-based estimations. It may take long for a node to learn the quality of a new link or to adapt to the link conditions. Hello packets probing is usually done with fixed lower data rates. For the IEEE 802.11, probing is done in 1-2 Mbps, while the actual data rate can be 54 Mbps and 200 Mbps for IEEE 802.11g and IEEE 802.11n networks, respectively. Furthermore, the real data packet size can be 1500 Bytes compared to 40 Bytes probe hello packet. Therefore, the traditional LQA, which is based on hello packet counting becomes very inaccurate [3]. To alleviate the problems of hello packets, Kim *et al.* proposed to use unicast instead of broadcast packets to probe the links and for the others to overhear the pass-by traffic [4]. When unicast packets are used, the data rate difference may be alleviated but the connectivity lessens. This is because the communication range for high data rates is shorter and the periodic probe packets are still too infrequent.

Unfortunately, most of the previous works consider only stationary networks, eg. [1],[2]. Nevertheless, wireless mobile ad hoc networks are intended to be mobile and many such scenarios are envisaged. Examples include people roaming around in a building with all kinds of personal wireless devices and other fixed devices in the building. In such scenarios, a LQA method still must be fast enough and detect link changes and failures in the mobile environment in due time.

In this paper, we discuss a cross-layer design for mobile IEEE 802.11 multi-hop networks. Several link quality indicators from different layers are combined to achieve a better LQA method. The generated cross-layer link quality information is evaluated and fed into the routing layer to improve the route selection and thereby also the end-to-end throughput. We also analyze the effect of the transmission rate on the LQA and routing layer. We have two major contributions: 1) We proposed a new LQA method, which can accurately estimate the link quality for all IEEE 802.11b/g data rates. Our experimental results show significant performance enhancements compared to the traditional hello packet probing. 2) We implemented the cross-layer architecture in a real test bed, in which accurate LQA is provided to the routing layer and demonstrated how our proposed LQA leads to faster and smarter route decisions and higher end-to-end throughput

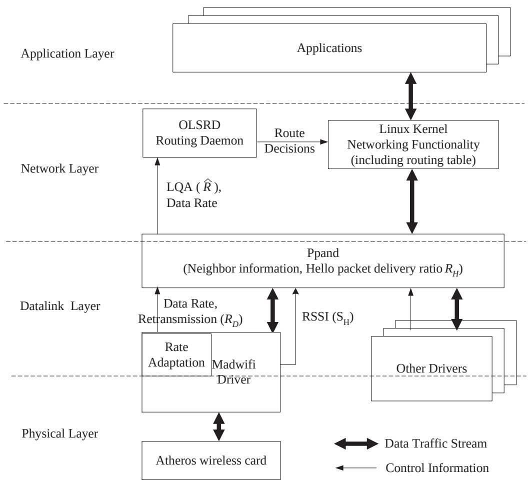

Fig. 1. Test bed architecture.

without extra overhead. Further, we also analyze the effect of using a multi-rate network on end-to-end throughput.

The paper is organized as follows. We present the cross-layer based test bed implementation in Section II. In Section III, we discuss the LQA mechanism in detail and introduce our novel LQA method. Section IV presents the experimental results and the route improvements. This paper is concluded in Section V.

II. Test-Bed implementation

Previous work demonstrate that current simulation models can not accurately reflect the channel conditions. Therefore, we built a test bed and evaluated our proposed LQA method using real measurements. The test bed architecture is shown in Fig. 1

A. Hardware

The test bed hardware consisted of four laptops equipped with an IEEE 802.11b/g card (3Com's OfficeConnect 108Mb 11g PC Card), which has a wireless chip from Atheros supported by the open source *Madwifi* driver [5]. The laptops ran the Linux 2.6.20 kernel and Madwifi driver version 0.9.2.1. Received Signal Strength Indicators (RSSI) values were obtained from the driver by Linux's IWSPY interface. To obtain the information related to the retransmissions of data packets and the data rate used for each neighbor, we adapted the driver. For rate adaptation, we used the *SampleRate* rate adaptation algorithm [6]. We changed several parameters as shown in Table I to optimize SampleRate to respond faster to link dynamics. Default power levels (15dbm) were used.

B. Software

Ppand is the software designed to maintain the neighbor list and combine all the cross-layer information. As can be seen in Fig. 1, it sits in between the network interfaces and the rest of the networking stack as a layer 2.5 implementation. Its purpose is to discover neighbors and monitor the quality of the links to those neighbors using LQA mechanisms. Ppand

TABLE I
Test bed Configuration parameters.

SampleRate	Values
Smoothing factor	0.65
Information store time	2 s
Minimum rate switch time	1 s
OLSRD	Values
Topology control message interval	1 s
Topology control message valid interval	3 s
Ppand	Values
Hello packet interval	1 s
α Hello packet delivery ratio	0.2
α Hello packet RSSI	0.2
α Data packet delivery ratio	0.5
Background traffic	20 UDP packets/s
Data packet size	1500 Bytes

generates and processes hello messages. To make sure that data traffic does not delay hello packets, we used a priority queue on each of the interfaces that gives hello packets the highest priority. For the routing layer, we used the Optimized Link State Routing Daemon *OLSRD* [7], [8] of version 4.10.0. We adapted the software and parameters (see Table I) to get the link quality information from Ppand instead of using the daemon's own LQA mechanism. OLSRD then uses its topology control messages to share the link quality information with the rest of the network.

C. Cross-layer Interaction

Direct communication between layers via the primitives results in low efficiency. Layer independent decisions may produce unnecessary overhead. Therefore, we propose a separate plane that communicates with all the layers. The Ppand program become the perfect place to hold this plane. It stores and processes all cross-layer information, then, forwards the data to the relevant layers. The Madwifi driver play the role of Physical and Link layer, so the basic physical channel information such as RSSI and the link layer information such as the available link types, number of transmitted and retransmitted packets with neighbors are forwarded to the Ppand. The LQA method in Ppand estimates the link quality and forwards to the routing layer. If necessary, Ppand also forwards the LQA information to other layers for other cross-layer optimization. OLSRD uses the LQA calculated by the Ppand to make route decisions and give those decisions to the Linux kernel routing table. The data packets sent from application layer are directed to the kernel routing functionality and are then sent to the next hop based on the routing table via the Madwifi driver and the Atheros wireless card.

III. Link Quality Assessment

A. Traditional LQA Indicators

Various link quality indicators, such as retransmission count, hello packet delivery ratio or received packet RSSI can be used to assess the link quality [9]. In this paper, we combine the hello packet delivery ratio, R_H and hello packet RSSI, S_H to estimate the packet delivery ratio, $\widehat{R}$ and compare it with the actual data packet delivery ratio, R_D. The measurements of

R_H S_H and R_D are described as follows. The hello packet transmission interval is known to all nodes. Therefore, the delivery ratio is measured at the receiver side. The Exponential Weighted Moving Average (EWMA) formula, R_H^t = αX_t + (1-α)R_H^{t-1}, was used to calculate the R_H. A received or lost hello packet will cause X_t to be 1 or 0 respectively. The smoothing factor α is to filter the R_H. A larger α will result in fast LQA but also less accuracy due to easier effected by random events and the use of less history. The α value used in the experiments in this paper is deduced by our prior experiments and presented in Table I. Since R_H is used by a lot of previous work, we will call it the *traditional method* in the rest of this paper.

Further, every received hello packet causes Ppand to read the latest RSSI from the driver. A lost hello packet is assigned the minimum receivable RSSI of -95 dBm since this is also a indication of bad link quality. The EWMA formula was used to calculate the RSSI, the smoothing factor for RSSI is also presented in Table I. To measure R_D for each link in both directions, all nodes produced connectionless data traffic among neighbors referred as background traffic. We sent 20 UDP packets per second and each second we calculated the R_D based on number of retransmission from the Madwifi driver. We also used the EWMA formula to filter second level R_D to generate a continuos R_D. In some of our experiments, data traffic is specially generated to measure the R_D as benchmark.

B. Proposed LQA method

Each link quality indicator has its own advantage. For example, R_H is simple and stable, while S_H does not need to inject new packets into the network, and hence, it is costless. S_H is fast, does not depend on packet type and data rate and can be used with other methods to improve the accuracy and speed. R_D is the most accurate one. However, in real networks, there is not always data traffic for every link. Therefore, R_D is not always available.

To leverage the advantages of all measures, we propose the following optimal method to estimate the link quality based on different situations:

1) When there is enough data traffic on the link, use the feedback about the MAC-level retransmissions, R_D,

2) When a node first joins the network, use S_H to estimate the packet delivery ratio,

3) Use the S_H to compensate the difference between hello and data packets on transmission rate and packet size.

4) Adapt to link dynamics by using S_H, which is faster to react to link changes. Based on 2), 3), 4), we propose a formula as follows:

$$\widehat{R} = C\left(1 - \frac{S_H}{S_M}\right) R_H, \quad (1)$$

where C is a multiplier and S_H is normalized with the minimum RSSI level, $S_M = -95$ dBm. Two extra rules to this formula are, (1) the generated $\widehat{R}$ will still be compared to 1. If the combination is larger than 1, the link quality is estimated to 1. (2) if the RSSI is larger than -60 dbm, the $\widehat{R}$ is generated as 1 no matter what the R_H value is. The multiplier C is a constant that adapts to different conditions such as multi-data rates and packet sizes. In this paper, this formula with the two extra rules are called the *proposed method*.

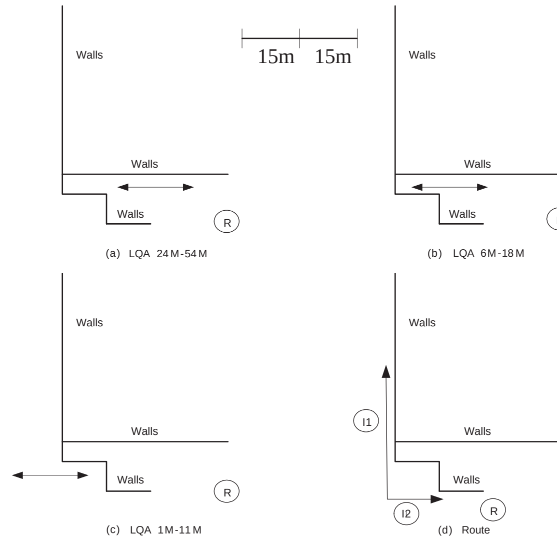

Fig. 2. Experiment topology (arrows are the trajectories).

We carried out real measurements using two nodes to get the link quality statistics for measuring the optimal value of the multiplier C. The experiment scenario is shown in Fig. 2 (a) (b) (c) and is an indoor environment. One of the nodes was stationary (R in Fig. 2) and the other was mobile (the mobile node's moving range is marked as arrow and other lines are the walls in the figure). The path is designed so that mobile node moves from the point where it starts to drop packets until reaching the point where it can not receive any data packets from the stationary node. That is, the sender tries to experience the packet delivery ratio variance from 0 to 1 and other way around. The speed of the mobile node is one meter per second. Both nodes constantly send hello and data packets at the same time. R_H, S_H and R_D are measured for ten minutes by taking one sample per second. The link quality is estimated for different C values. For each C value, we calculate the average absolute difference between $\widehat{R}$ and R_D. The lowest difference yields the optimal value of the multiplier, *C**.

C. Performance comparison

To show the effectiveness of the proposed estimation method, samples are collected at a data rate of 36Mbps for 300 seconds. As can be seen in Fig. 3, our proposed $\widehat{R}$ estimates R_D better than the traditional method R_H. R_H hardly predicts the link quality variance in this scenario. For 36 Mbps, *C* = 2.9*.

The average absolute percentage deviation D is defined as

$$D^{\text{method}} = \frac{100}{M} \sum_{i=1}^{M} |r_{d_i} - r_i^{\text{method}}|,$$

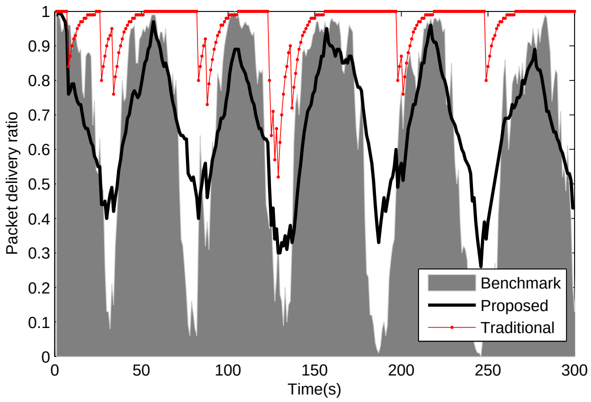

Fig. 3. The effectiveness of the estimation (Rate is 36 Mbps, $C^* = 2.9$).

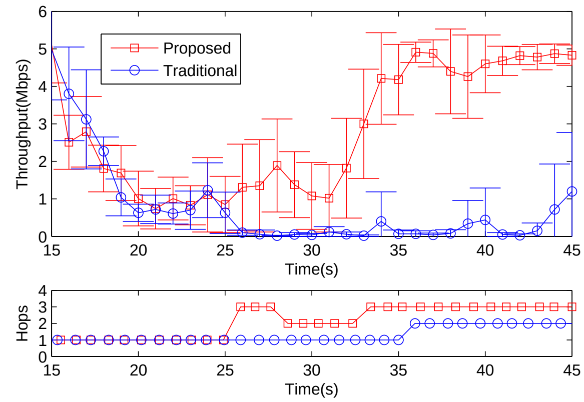

Fig. 5. The throughput with time (ETX, constant 36Mbps).

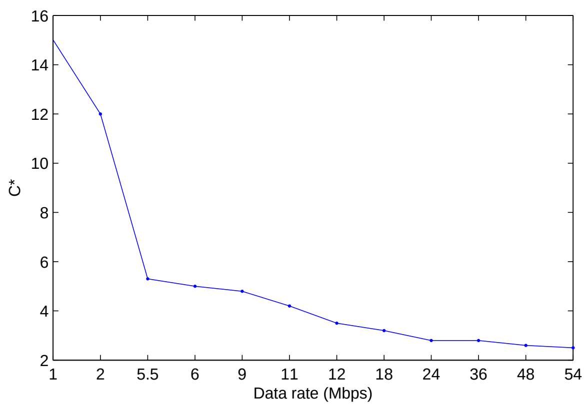

Fig. 4. The best constant value for 12 data rate.

where M is the number of samples, r_{d_i} is the sampled data packet delivery ratio, r_i^{method} is the estimated sample delivery ratio where `method` is either the traditional hello packet estimation, R_H or, our proposed $\widehat{R}$. Calculation based on the data in Fig. 3 produces $D^{\widehat{R}}$ = 14 % and D^{R_H} = 32 %. Hence, our method estimates the link quality twice better than the traditional R_H. The reason why is that for a certain scenario, there is some correlation between the RSSI and R_D. When the channel changes very fast, RSSI adapts quickly. However, RSSI tends to fluctuate more. Therefore R_H is multiplied in Eq.1 to generate more stable predictions. For different configurations, such as data rate, packet size, scenario and so on, this mapping relation between RSSI and R_D may also change, which leads to different C*. Based on prior learning experiments, we can choose the appropriate C* for each application.

To analyze the performance of the our proposed estimation method for different data rates, we carried out more experiments. The optimal multiplier values for twelve IEEE 802.11 rates are shown in Fig. 4. Depending on the maximum range that can be achieved by a specific data rate, the experiments are grouped into three scenarios as shown in Fig. 2 (a),(b),(c).

When higher data rates are used, a smaller $C*$ produces more accurate estimates (see Fig. 4). The data rate does not affect the delivery ratio of hello packets because the hello packet use the lowest data rate and a fixed small packet size. When the experiment conditions are the same, the data packet delivery ratio decreases when the data rate is increased and $C*$ decreases. Notice that C^* converges to a value around 2.5, as the rate increases. The experiment result of 48 Mbps and 54 Mbps also got almost 50% improvement, and this improvement decrease with the data rate, which is due to the smaller difference between data packets and hello packets. We also evaluated the effect of the packet size, mobility pattern, traffic load and speed of the node on $C*$. Due to the space limit, we cannot present them in this paper.

IV. Routing Improvement

To compare the impact of different LQA methods on the routing layer, we carried out experiments with the scenario shown in Fig. 2 (d) and configuration for three different mechanisms. To maintain a stable link, three stationary laptops were placed on tables one meter above the floor at positions R, I1, I2, two of them are the intermediate nodes and the other one is the receiver node. The mobile node is referred to as the sender. The sender follows the trajectory shown in Fig. 2 (d). It waits 30 seconds (warmup period to stabilize the SampleRate mechanism, then takes a movement of 45 seconds following the trajectory. This process is repeated ten times with almost exact time control. We can assume each walk is independent. The sender keeps sending UDP packets as fast as it can via the routing table provided by OLSRD. We plot the instantaneous throughput with time. The ETX and ETT routing metric is implemented based on [1] and [2] as $ETX = \frac{1}{P_f \times P_r}$ and $ETT = ETX \times \frac{L}{B}$ in which P_f and P_r is forward and reverse packet delivery ratios of the link, L and B is packet size and bandwidth respectively.

In the first experiment, all nodes used fixed data rate of 36 Mbps. Two link quality metrics are used separately, our proposed $\widehat{R}$ and the traditional R_H. We plot the average throughput obtained in ten runs with a 95% confidence interval in Fig. 5. To analyze the route choice influenced by the two mechanisms, we also plot the number of hops taken to reach the receiver. At the beginning of the experiments, the two mechanisms made the same choice: one-hop route. Thus, in the first 15 seconds, the performance is almost the same, so we do not plot this period and performance comparison is only based on 15 to 45s for all experiments. From 25 to 35 seconds, the throughput achieved by our proposed $\widehat{R}$ is higher than traditional R_H, because our proposed LQA influences the routing layer to use the interim nodes (multi hop paths) while the traditional LQA continues to suggest the routing layer to use the low throughput one-hop route. Even though the receiver and sender do not have a line of sight and the link quality has already become worse, hello packets still have good delivery ratio and causes a slow reaction to the link dynamics. After the 35^{th} second, the proposed LQA chooses the three-hop route while the traditional LQA chooses the two-hop route. Therefore, the throughput of the traditional LQA is still poor. We calculate the average throughput for these two methods in Table II and mark the first experiment as the ETX scenario.

In the second experiment, we turn on the rate adaptation mechanism. Hence, the data rate between each link is determined by the SampleRate algorithm, the data rate selection will not effect the LQA since LQA is only related to the hello packet. We still use the ETX routing metric to decide the route. Since the procedure is the same as the first one, we do not plot the throughput with time for this scenario but only provided the average throughput for different LQA methods. We mark the second experiment with Rate scenario in Table II. As can be seen in Table II, our proposed LQA still has better performance than the traditional LQA due to the selection of the better path. It is shown in previous work that rate adaptation does not outperform the fixed data rate in some scenarios [10]. This explains why for the proposed method, second experiment performance slightly worse than ETX scenario which use fixed data rate 36 Mbps while rate adaptation select some lower data rate which caused the lower throughput. For the traditional method, since one-hop is selected for a long time, rate adaptation can select some lower data rate that is more suitable for long distance and non line of sight transmission, thus delivery more packets than the ETX scenario which use 36 Mbps all the time and can hardly deliver any packets when the distance is long.

In the last experiment, the rate adaptation mechanism was also turned on, and both LQA and data rate were used in the ETT route selection. We name this experiment the ETT scenario in Table II. The result is very similar to the second experiment and our proposed LQA still outperforms the traditional LQA. Because the data rate selection is done by SampleRate and the variance of the data rate influences the ETT route selection also. The slow reaction of rate adaptation also effect the route decision and that is why the ETT scenario's throughput is lower than Rate scenario. In the route record, the ETT route selection seldom select three hop route both in proposed and traditional LQA experiment.

Based on these experiments, it can be concluded that our proposed LQA can achieve better performance than traditional LQA with a faster response to link dynamics and differentiate the data and hello packet delivery ratio. Rate adaptation can only improve end-to-end throughput compared to fixed data rate in some scenarios. Moreover, in ETT routing in which the data rate is considered, the accuracy and responsiveness of rate adaptation is even more important, which suggest the proposed LQA should also feed into the rate adaptation mechanism.

TABLE II
ROUTING THROUGHPUT PERFORMANCE COMPARISON.

	ETX	Rate	ETT
Proposed (Mbps)	3.3	2.974	1.85
Traditional (Mbps)	0.75	2.0	1.36
Improvement	340%	48%	36%

V. CONCLUSION

In this paper, we proposed a cross-layer design based link quality assessment (LQA) method and used it in the routing layer. We combined the available information sources to a new combined LQA method. The real experiments on our test bed showed the significance of an accurate LQA method in a mobile multi-rate IEEE 802.11 network. The proposed estimation method improves the throughput performance without any overhead in the network. As future work, we plan experiments to test the performance of LQA in more scenarios (e.g. stationary, outdoor) and conditions to investigate their impact to the optimal value in our proposed LQA method. Moreover, a self learning protocol is being developed to input the self learning mechanism for the optimal multiplier selection in environmental adaptive LQA. Further, our proposed LQA can be used for other applications, such as data rate adaptation.

REFERENCES

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

[2] J. Bicket, D. Aguayo, S. Biswas, and R. Morris, "Architecture and evaluation of an unplanned 802.11b mesh network," in *Proc. ACM Mobicom'05*, Cologne, Germany, August 2005, pp. 31–42.

[3] H. Lundgren, E. Nordstrom, and C. Tschudin, "The gray zone problem in IEEE 802.11 b based ad hoc networks," in *ACM SIGMOBILE Mobile Computing and Communications Review*, vol. 6, no. 3. ACM Press New York, NY, USA, 2002, pp. 104–105.

[4] K. S. K. Kim, "On accurate measurement of link quality in multi-hop wireless mesh networks," in *Proc. ACM Mobicom'06*, Los Angeles, CA, USA,, September 2006, pp. 38–49.

[5] "Madwifi driver." [Online]. Available: http://www.madwifi.org

[6] J. Bicket, "Bit-rate selection in wireless networks," Master's thesis, Massachusetts Institute of Technology, 2005.

[7] T. Clausen and P. Jacquet, "RFC3626: Optimized Link State Routing Protocol (OLSR)," *Internet RFCs*, 2003.

[8] A. Tonnesen, T. Lopatic, H. Gredler, B. Petrovitsch, A. Kaplan, and S.-O. Tcke, "Olsrd: An adhoc wireless mesh routing deamon," 2008. [Online]. Available: http://www.olsr.org

[9] J. Zhou, M.Jacobsson, and I.Niemegeers, "Cross layer design for enhanced quality personal wireless networking," in *Proc. Med-hoc-net 2007*, Corfu, Greece, June 2007.

[10] S. Wong, H. Yang, S. Lu, and V. Bharghavan, "Robust rate adaptation for 802.11 wireless networks," in *Proc. ACM Mobicom'06*, Los Angeles, CA, USA, 2006, pp. 146–157.