Humboldt University Berlin
Computer Science Department
Systems Architecture Group

Rudower Chaussee 25
D-12489 Berlin-Adlershof
Germany

Phone: +49 30 2093-3400
Fax: +40 30 2093-3112
http://sar.informatik.hu-berlin.de

Towards using 900 MHz for Wireless IEEE 802.11 LANs Measurements in an Indoor Testbed

HU Berlin Public Report
SAR-PR-2008-05

März 2008

Author(s):
Matthias Naber, Moritz Grauel

Towards using 900 MHz for Wireless IEEE 802.11 LANs
Measurements in an Indoor Testbed

Matthias Naber
Systems Architecture Group
Computer Science Department
Humboldt-University Berlin
Email: naber@informatik.hu-berlin.de

Moritz Grauel
Systems Architecture Group
Computer Science Department
Humboldt-University Berlin
Email: grauel@informatik.hu-berlin.de

March 25, 2008

Contents

1 Abstract

To explore further use cases for wireless mesh networks – evading congested frequency bands is required. Achieving soft real time or simply bridge larger distances – the requirements simply exceed the physical specifications given by today's IEEE 802.11 networks. Some saw their opportunity to modify existing IEEE 802.11 hardware to operate at different frequency bands. How to determine possible improvements?

The "Wireless Testbed Framework" is a set of tools to ease and automate testing and evaluating of new hardware in an existing, experimental wireless mesh network. It is targeted to small and embedded devices running openWRT or similar embedded Linux variants. The main purpose is to manage measurement scripts and schedule the execution of those scripts as well as to collect the measurement results.

Many measurements on wireless link performance have been made. This paper presents link performance results – comparing different frequency bands at 900 MHz, 2400 MHz and 5000 MHz – performed with our automated measurement framework. The Ubiquity SR9 900 MHz WiFi adapter is compared to off the shelf WiFi hardware. The link quality and the bit error patterns of the different frequencies are analyzed and compared to each other. This paper also indicates whether this technology can be of any benefit to wireless community mesh networks.

2 Introduction

Wireless networks based on the IEEE 802.11 standards are present everywhere in today's urban areas. The technology has evolved from its early state and has become a widely adopted consumer product in home and office environments. The increasing demand of being online at all times brings up a rising number of companies, who provide wireless internet access in many urban areas [1][2]. There are also some free of charge networks installed and operated by individuals or bigger projects like the "Freifunk" [3] or the "RoofNet" [4][5] projects.

Nevertheless IEEE 802.11 based networks are reaching technological limits in link utilization and throughput. Especially urban indoor environments, like bureau buildings and trade exhibition, are troublesome due to many concurrent networks and WiFi's physical properties.

First, we will give a brief introduction to the theoretical benefits, which we expect from operating IEEE 802.11 at lower frequencies in section 3.1. We will verify existing IEEE 802.11 technologies in the formerly unused frequencies around 900 MHz and compare them to recent 2.4 and 5 GHz WiFi technologies. We will start with some benefits and explanations of theoretical radio models shown in section 3.2 and later presenting real measurements. We will introduce these measurements we performed in chapter 5 and the corresponding results in chapter 6. Along with our measurements we created an extensible and dynamic framework to support us in this and further measurements. This paper will also introduce the measurement framework together with the used hardware in chapter 4. We will also discuss the question, whether switching to different frequencies is of any benefit to community based wireless mesh networks in urban areas like the BerlinRoofNet[5] and similar projects in our conclusion (chapter 7).

3 Expected results

In this section, we will briefly introduce the theoretical benefits of lower frequencies and what we can expect from IEEE 802.11 devices operating at 900 MHz.

3.1 Electromagnetic Waves

The theory of electromagnetic waves (propagation) goes back to James Clerk Maxwell. Between 1861 and 1864, he developed 4 famous formulas which had an enormous influence to science. His theory of electrical and magnetic fields explains electric current, electric charge and their relationship to each other (as well as many other things). A few years later, in 1888, Heinrich Hertz was the first person who artificially generated electromagnetic waves. He saw no practical use for radio waves because of the poor propagations and he predicted that it will never carry voice[6].

Electromagnetic waves, as the name predicts, have electric and magnetic components which propagate through space. They can be reflected, scattered and diffracted by obstacles. This results in a multitude of complex waves, with variants of amplitudes, different phases and different frequencies. To get an exact solution to this problem you would have to solve Maxwell's equations. But this is not possible for real problems due to missing parameters and an extremely high complexity of the resulting formulas.

To solve those mathematical problems, simplifications and approaches have been developed. Most of them are based on ray models that have a deterministic behavior and are basically dependent on parameters like ranges and frequencies. In the next section we will present some of these and we give some predictions for IEEE 802.11 networks at 900 MHz based on those models.

3.2 Radio Models

There are a lot of different propagation models for radio waves. Reaching from simple and highly inaccurate models like the "Free Space Path Loss Model" (FSPL) to much more sophisticated models like "Rayleigh Fading" and "Shadow Fading".

The basic ones use very few parameters, so predicting the behavior of the specific propagation is very easy, although with weak accuracy. Some models are only a function of distance and frequency. It should be clear that these results will not reflect the reality because of their high level of abstraction. They do not even consider the environment but idealize it into an open and empty space.

First, we will make some predictions based on the "Free Space Model." The free space model is derived from the Coulomb equation, which is part of the previously mentioned Maxwell equations. It describes how electric fields behave in free space vacuum

for statically positioned electrically charged particles. Assuming there is a source of an electromagnetic field, the received power decreases exponentially by a factor of 2.

The "Free Space Path Loss" (in dB) is defined as

$$L(d) = 20 \cdot log_{10} f + 20 \cdot log_{10} d - 20 \cdot log_{10} \left(\frac{c}{4\pi} \right) \tag{3.1}$$

Where f is the frequency of the signal, d is the distance to the sender and c is the speed of light. Obviously the path loss at any given distance is less at lower frequencies compared to higher ones. Based on FSPL we expect significantly better signal to noise ratios using 900 MHz.

The actual improvement which will be achieved by changing IEEE 802.11 from 2400 MHz to 900 MHz is difficult to predict, due to the fact that IEEE 802.11 at 900 MHz is quite rare and unused these days. Developed by NCR in 1990, acquired by AT&T 1991, AT&T released the first 900 MHz wireless devices in 1995 named WaveLan[7]. As viewed from today, they had no success with this technique. But it held a lot of innovations which were used for the later IEEE 802.11 standards. In 1995, AT&T claimed to reach ranges of 800 ft (245 m) [8].

The hardware vendor of the wireless equipment we used for our measurements claims to reach 20 km non-line-of-sight with the same frequencies [9]. This appears to be very optimistic. We decided to compare some basic radio propagation models and their results for the three different frequencies we want to measure.

Longer wavelengths have better abilities to penetrate obstacles. This has already been shown by Hata-Okumura. Their model tries to predict the propagation of frequencies in ranges of 150 MHz to 1920 MHz at longer ranges than we used for our indoor measurements. Nevertheless this model predicts that the attenuation is a function of frequency and distance and it was especially developed for urban areas. Furthermore it depends on antenna heights which are not in our research interest.

Path loss (in dB) according to the Hata-Okumura Model[8]:

$$\begin{aligned} L(d) = \quad & 69.55 + 26.16 \cdot log_{10} \tfrac{f}{[MHz]} - 13.82 \cdot log_{10} \tfrac{h_b}{[m]} \\ & - a(h_m) + (44.9 - 6.55 \cdot log_{10}(h_t)) \cdot log_{10}(d) \end{aligned} \tag{3.2}$$

$$a(h_m) = \quad (1.1 \cdot log_{10}(f) - 0.7) \cdot h_m - (1.56 \cdot log_{10}(f) - 0.8) \tag{3.3}$$

Where f is the frequency and d is the distance to the sender. h_t and h_m are the heights of the transmitter and the receiver antennas. a depends on the environment. The above a is for a midsize city.

Another accepted model is the "dual slope model". It was developed by Beyer and Jakoby in 1997 who discovered, that the propagation probability behaves differently at close distances compared to far ranges. Therefore a breakpoint distance indicating the distance where the far field effects appear is needed. The equation is as follows[8]:

$$L_1(d) = \quad 10 \cdot \gamma_1 \cdot log_{10} (4\pi df) - a_0 \qquad \text{for } d < d_{Br} \tag{3.4}$$

$$L_2(d) = \quad L_1(d_{Br}) + 10 \cdot \gamma_2 \cdot log_{10} \left(\tfrac{d}{d_{Br}} \right) \quad \text{for } d \geq d_{Br} \tag{3.5}$$

Where f is the frequency, d is the distance and d_{Br} is the breakpoint distance between the near and the far field. a_0 represents the attenuation difference between the calculated

path loss and the path loss of free space propagation. γ_1 and γ_2 are attenuation constants for the near and the far field.

We used a breakpoint distance of 20 m for our estimations. The resulting plot can be seen in Figure 3.1. These analytical models show that the mean path loss will be less the lower the frequencies are. The available bandwidth should be equal for each of the used frequency bands – according to "Shannon's theorem" there should be the same theoretical capacity for each channel with the same bandwidth.

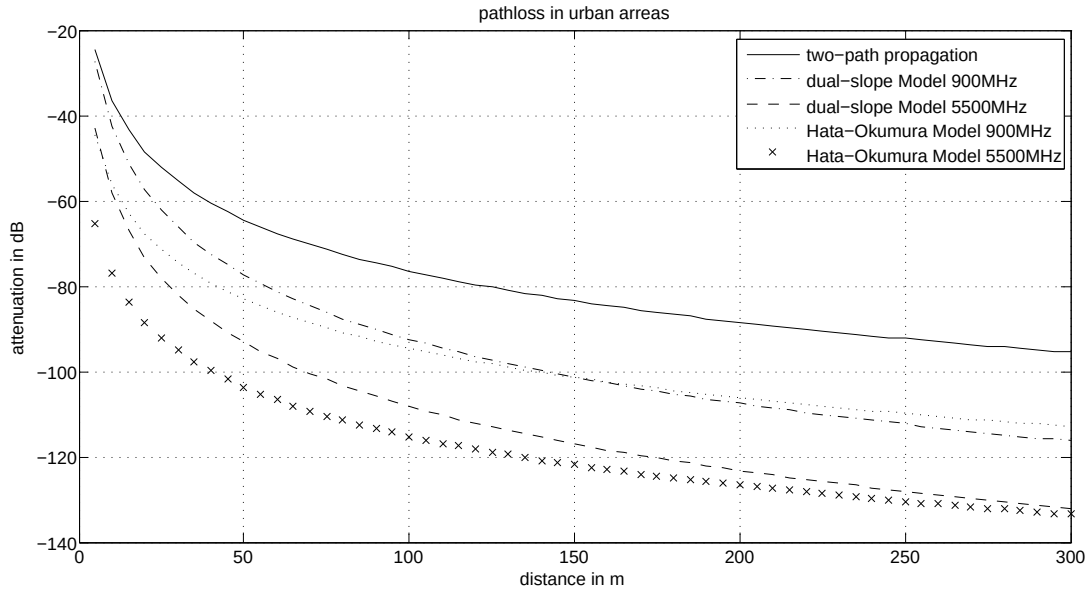

Figure 3.1: Analytic results, various propagation models

All those estimations indicate the wireless link should have a better signal to noise ratio for lower frequencies. This is quite reasonable if we compare the possible range of low frequency signals in common, e.g. acoustic signals. These waves behave similar to electromagnetic waves and the attenuation is also proportional to f^2 where f is the waves frequency [10]. But these estimations which we obtained from the free space model do not hold for urban areas – especially indoor environments as already described in section 3.1. There are many propagation models for urban areas. We chose some simple ones to get a more detailed prediction. Figure 3.1 indicates that there may be a gain in link quality for urban areas due to constructive overlay of waves which increase the signal strength whereas in free space there is only a line-of-sight connection. The reality is going to look more complex and difficult because of diffraction, refraction, attenuation and fading.

3.3 Conclusion

Summing up the previously mentioned models we get a clear view about the theoretical results of using different frequencies. The new 900 MHz devices should have a much better performance. They provide higher ranges, better path losses, penetrate better through objects and last but not least, the band is not so crowded compared to the 2.4 GHz industrial, scientific, and medical band (ISM) band. According to the graphs we could

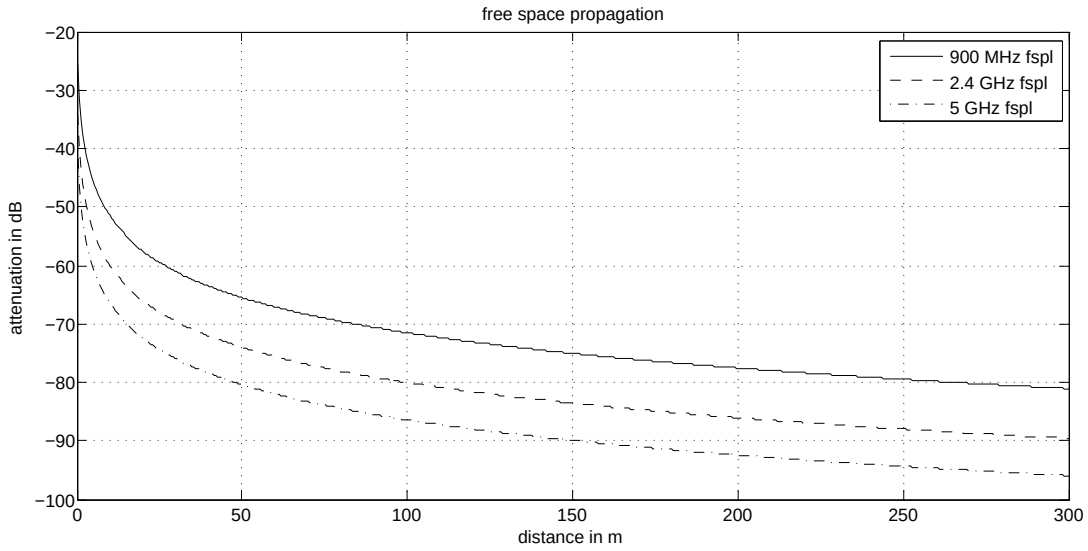

Figure 3.2: Analytic results free space propagation

assume a factor of 3 or 4 in range when comparing 900 MHz and 5000 MHz links. In other words, there should be an estimated gain of +20 dB to +25 dB in path loss for a given distance.

4 Methodology

In this section, we will first present our wireless testbed framework. Afterwards, the hardware used for the measurements will be introduced.

4.1 Measurement Framework

It came out that measuring in a widespread, existing network is a fragile task. The existing solutions and tools were not satisfying for our needs. Existing tools like "emulab" [11] or "orbit-lab" [12] are to heavy weight for our embedded network nodes. Other attempts are made on workstations with proprietary software, which is inappropriate for our case [13]. The method formerly used in our network based on lots of local and remote shell scripts. This "tool chain" was neither very reusable nor reliable. The existing scripts were either built solely for one purpose or hastily rewritten for another. This has proven to be an error prone way of controlling tasks which is why we created a more dynamic framework for conducting our measurements on embedded devices.

We wanted to create a stable and reusable environment which can be easily extended and has very little impact on the resources of the measuring nodes. To achieve this goal, we created a very light weight and portable daemon to run on the network nodes. The daemons are controlled by a server, which runs on the researchers workstation or a department's server. Depending on the researcher's preferences, the daemon can either be scripted on the command line via configuration files or easily controlled through a web based interface.

By using the command line way, the user has to create his schedule by hand. A big advantage lies in using relative times, which results in an easy reuse of existing scripts. Once the script is created, it allows reproduction of measurements in a very convenient way.

In turn our web interface provides an intuitive frontend which is fully platform independent. It is used to manage projects and the associated measurements. Measurements can easily be scheduled for execution at a certain point in the future. It also provides arbitrary parameters for the measurements to quickly set up long series of measurements with varying parameters like the used channels or bit rates. The included node list (Figure A.6) is as a very helpful tool to carry out jobs to specific nodes with little administrative work required. If the address of a node changes due to a reset or a replacement, you only need to update it once, instead of changing every single script.

The web interface even can visualize some basic results as long as the data is collected in standard pcap file format (Figure A.7).

The basic workflow is as follows: In most cases, the researcher will start by either creating his measurements or by modifying an existing one. The process can be streamlined a lot, by making the complete measurement on a node ready to work with only a single

script. Ideally, interesting aspects like frequency or transmit power can be controlled via parameters to the script. Once this script is created, it is useful to log into a node and test the scripts correct functionality.

After the script is debugged and ready to run, our tools can be used to simplify the execution of the measurement. The easiest way is to access the web interface. The first step is to create a project (Figure A.1). The purpose of a project is to group the measurements in order to provide a better overview. After creating a project, it can be populated with measurements (Figure A.2). The researcher creates a new measurement with a recognizable name. He can also enter a short description of the measurement, which will be displayed as a tool tip at the measurements overview.

Inside a measurement, the researcher creates tasks (Figure A.3). A task essentially contains a single command that will be executed on an arbitrary number of nodes at a chosen point of time in the future and can be stopped anytime later. The start- and stop-times can either be entered as an offset in seconds from the moment the task was sent to the node or as a discrete point in time via a date and time chooser. All the nodes known to the system are displayed in a list with a checkbox next to their name, which is used to activate this task on that specific node (Figure A.4).

Variables can be created and modified at the tasks overview. They are accessed inside tasks by simply writing `${Variablename}`.

After creating all the tasks, they can be submitted to the nodes. Therefore, each task contains a list of nodes. If a node shall execute a specific task, he has to be activated in the tasks node list. This makes it possible to only run a subset of a measurement. Pressing the commit button will send the tasks to the associated nodes and the web interface will inform the user about the results. At this point, the nodes can be separated from the wired backbone network because all the necessary scheduling will be done on the nodes itself. It is also possible to submit further tasks either to enhance the measurement or to schedule later ones.

After the measurement has finished, the results can be collected at the overview of started tasks (Figure A.5). Every output that was created by a single task is stored within its own directory. This directory will be wrapped into an archive and sent to the server. On the server side it can either be visualized (if it is a regular pcap file) or simply be downloaded by the researcher. If the measurement creates output outside of its own working directory, the researcher has to collect the data manually.

This concludes the outline of the basic workflow with our tools. At the moment there is one further step to it. In preparations for his measurements, the researcher has to start the daemon on each node. In the future, this tedious work will no longer be necessary because we hope to integrate the daemon in the default start-up configuration of all our nodes.

Our goal was neither to implement fully automated active nor passive network monitoring with our framework. For that purpose, very good tools like [14] or [15] are already existing.

4.2 Hardware

Since projects like “Berlin Roof Net” (BRN) are mainly interested in the performance of embedded devices like home routers, we conducted all of our measurements on such devices. We used two different platforms. We had 6 Netgear WGT home router modified to use the Ubiquity SR9 network interface card (NIC). The Netgear WGT has a Broadcom 200 MHz MIPS based CPU and 16MB of RAM that runs openWRT embedded Linux. The WGTs were located in different bureaus at different stories of the computer science department of the Humboldt-University.

In addition to those devices, we had 3 nodes based on the Metrix Soekris board, an AMD Geode based x86 system with two mini-PCI slots also running Linux. The two mini-PCI slots were populated with one Ubiquity SR9 and one Atheros based IEEE 802.11a/b/g card (Winstron CM9). The Soekris (SK) devices were very valuable to our measurements, since a direct comparison between the three frequencies was possible without moving or exchanging the measuring device with another one. The SK devices have two antennas that are located 10 cm apart. They were the main platform for our measurements.

4.3 Ubiquity SR9

The Ubiquitiy SR9 is a mini-PCI WiFi card based on an Atheros chip. It converts the native analogous signal outputted from the onboard Atheros chip down to 900 MHz by analog transformation. There is also an amplifier on the card which allows sending data at a very high transmission power of up to 900 mW. Its receive sensitivity is set to -93 dBm.

According to the datasheet, provided by the cards manufacturer [9], the transmission power depends on the selected bit rates. Table 4.1 shows the transmission power and the required current of the card at all the supported rates.

Table 4.1: TX power

	Data rate	Current	TX Power
802.11b	1 Mbps	1.2A	28 dBm
	2 Mbps	1.2A	28 dBm
	5.5 Mbps	1.2A	28 dBm
	11 Mbps	1.2A	28 dBm
802.11g OFDM	6 Mbps	1.2A	28 dBm
	9 Mbps	1.2 A	28 dBm
	12 Mbps	1.2 A	28 dBm
	18 Mbps	1.2 A	28 dBm
	24 Mbps	1.2A	28 dBm
	36 Mbps	1.1 A	26 dBm
	48 Mbps	1.0 A	23 dBm
	54 Mbps	0.85 A	21 dBm

Since the SR9 is originally a regular Atheros based NIC and all the modifications are transparent to the operating system, the card can be operated with the regular Atheros drivers. In our case, this is the MadWifi driver[16]. The MadWifi driver does not recognize that it operates at 900 MHz. It still reports 2.4 GHz frequencies to the operating system and the transmit power is also displayed incorrectly. For the power offset the manufacturer supplies an offset table. To obtain the frequency for each available channel we created our own table (see Table 6.2).

4.4 Software

To generate packets and therefore traffic, we used raw sockets. Since the MadWifi driver allows raw frame injection, we could generate arbitrary content, wrap it as a packet and pass it to the driver. We did so by the means of a very simple program written in C. We generated a unique byte sequence of 10 bytes to recognize our packets, appended a counter and padded the rest of the packet up to a definable size with random bytes. On the receiver side, we were filtering for our specific byte sequence. The counter allowed us to see directly, which packets were lost. All further information, like time stamps or signal strength, are provided by the driver in either the radiotap or the athdesc header.

For our bit error measurement (section 5.6) we chose a different approach. We used the click modular router framework (CMF) developed at the MIT. The CMF allows the user to build modular components of a router and *click* them together. For a more detailed introduction to click see [17].

4.5 Measurement Setups

To get expressive and reproducible results, we decided to conduct most of our measurements with the "Metrix Soekris Boards" which, as described in section 4.2, had 2 independent wireless NICs. This allowed us to directly compare 900 MHz, using the Ubiquity SR9, to 2.4 GHz and 5 GHz, using a standard wireless NIC.

Virtually all of our measurements were performed at late times and weekends, when the building was deserted to prevent unnecessary interferences. During work hours the building is crowded and many people are using the wireless network of the university. This would have led to some disadvantages of the 2.4 GHz measurements.

Due to the fact that the department of computer science is placed at a remote area, our results should contain as little external influences as possible for bigger cities like Berlin. And since the frequencies used by the SR9 are not located in a free band, the remote location was also beneficial to avoid legal issues and to not disturb the legal owners of the frequencies.

5 Measurements

The following sections will describe the measurements we performed. It explains the setups and all its parameters. The results of the measurements will be presented later in chapter 6.

5.1 Baseline Measurement

We decided to start our measurements with simple baseline tests. This had two purposes. First, we wanted to get in touch with the used hardware. Second and more important we wanted to determine whether the use of the SR9 was possible at all. The frequencies used by the SR9 are not located in an ISM band. Therefore it was likely that the owners of the frequencies were using it themselves in such a way that our measurements in this frequency band will be noticeably affected.

We placed two nodes 6m apart from each other in an office room. The sender was set to transmit at rates of 1, 6, 11 and 54 Mbps each for 15 minutes. We performed those tests at 900 MHz, 2.4 GHz and 5 GHz each to compare the different frequencies. We performed these measurements on each of the four supported 900 MHz channels so that we would find channels that are more congested by external sources than others.

5.2 Radio Crosstalk Measurement

We used a similar setup to search for crosstalk between the two radios in the SK nodes similar to the ones reported by measurements like [18]. Therefore we used our baseline setup and made simple throughput measurements on both radios simultaneously as well as using each of the radios on its own. We were mainly looking for an impact on packet loss and signal strength instead of only monitoring raw data throughput. We were expecting a decrease in throughput, since the processor of the node is already highly loaded when transmitting only one stream of data at higher transmission rates.

5.3 Shakeboard Measurement

We borrowed a shakeboard ("Roedelbrett") from the Artificial Intelligence Department of our Institute [19]. This device is nothing more than a wooden board that is movable in one dimension by a stepper motor. The frequency and the amplitude of this motion (along with some presets) can be adjusted via a little control application. The Robotics Department is using this device to test balancing of bipedal robots. We used it to measure small scale fading. We set the board to a periodical sinus movement with maximal amplitude and placed a receiving node onto the moving board. That way we could measure the influence of small movements on the links.

Sadly it is not possible to choose amplitude and frequency precisely. Especially at higher frequencies, the small stepper motor had difficulties to move the board properly. Therefore it neither ran perfectly periodic nor with perfectly the same amplitude. The period decreased with increasing frequency.

For our measurements, we set it up to about 0.5 Hz. At that rate the amplitude was approximately 17 cm. The sender and the receiver were 3 m ($\pm$ 8.5 cm) apart from each other. The sender was transmitting at 54 Mbps.

We chose this transmission rate because we expected only slight variations in the received signal strength. Therefore we were hoping that faster transmission rates could amplify the effects since they use a weaker modulation and even a small decrease in signal to noise ratio could lead to severe packet loss.

Again, this was repeated for each of the three frequencies.

5.4 Distance Path Loss Measurement

To measure the performance in relation to the distance, we placed nodes at both ends of a 50 m long, straight corridor in our departments building. One of the nodes was set to transmit for 5 minutes each at the four speeds 1, 6, 11 and 54 Mbps. Two other nodes – positioned 1 m apart from each other – where relocated 2 meters after each measurement so that we have covered the whole distance of 50 meter at a granularity of one meter. To rule out influence by people, walking through the corridor (as measured by [20]), we performed these measurements only at weekends, where the building is usually deserted.

5.5 Spectrum Analyzer Measurement

Since our results up to this point could not keep up with our expectations, we decided to take a deeper look into the workings of the SR9. The MadWifi driver can provide only very limited insight into this. We borrowed a SPECTRAN HF-6080 spectrum analyzer [21] from the Chaos Computer Club Berlin [22]. Equipped with a spectrum analyzer, we were looking for periodical interference or a very high noise level in the used frequency range. We also verified the specifications according to the datasheet.

We measured the emitted energy by connecting the spectrum analyzer directly to the antenna plug of the wireless cards. We set the card to all 16 transmit power levels the driver accepted and measured the emitted energy at the antenna plug. To put our results into perspective, we did the same thing with a regular IEEE 802.11 WiFi card set to 2.4 GHz and to 5 GHz.

Finally we let the spectrum analyzer probe the used frequency range in order to find any background noise.

5.6 Bit Error Measurement

To get a more detailed insight into the behavioral characteristics we took a more detailed insight into propagations probability. Therefore we deactivated the MadWifi drivers dropping of corrupt packets whose “Cyclic redundancy check” (CRC) failed by simply setting `/proc/sys/net/ath0/monitor_crc_errors` to “`1`”. As a result of this change, we were

able to also receive those packets whose checksum is "wrong" due to transmission errors. Those errors result in an arbitrary number of bit errors. We decided against monitoring "physical errors" because they mostly result in truncated packets due to the hardware stopping to receive. There is no use in analyzing them: Firstly, we estimated a specific error distribution to occur. This means, if the packet is truncated – the received fragment will contain error patterns similar to those ones who were not truncated. So not analyzing these packets would not affect our results. Secondly, most truncated packets are caused by collisions of concurrent transmissions. But we wanted to do comparable measurements of the hardware. For there are different amounts of congestion in the 3 used frequency bands – especially the 900 MHz band is virtually empty in comparison to the 2400 MHz band – we decided again not to take these truncated packets into consideration.

We used one node which was steadily sending packets of a given size of 1500 bytes with a content that is generated randomly by a "click script" and two nodes for receiving these data. The first node sent its data as broadcast so that there are no acknowledgments. The transmission feedback was then written into a file to compare it to the received data of the other nodes. The two other nodes received the data and wrote it directly into a file. The did nothing more than to filter out all packets not originated by the first node. The packets held mostly random data. The non-random parts of the packets contain necessary information, specifically the receivers mac address, the senders mac address and a short identifier, which is needed to classify the packet as one of "our" packets. For later analysis, each packet also contains an additional unique time stamp written into the payload which allows finding related packets.

For this measurement we decided to choose a rate of 54 Mbps and placed the nodes on spots where the link quality was moderate. We have to note, that the area was not deserted. As shown in [20] this may already have influence on link quality. But since this time we wanted to measure errors, this was a benefit to us.

More about the specifics on error models of wireless networks can be found in [20] and [23].

6 Results

In this section, we will present the results of our measurements. The results will be presented in the same order as they were introduced in the previous section.

6.1 Baseline Measurement

Our initial findings could not match our expectations. Especially at OFDM rates the links were very unstable in terms of link quality. The same node setup with regular 2.4 GHz frequencies showed better performance all the time. The packet loss was less, the received signal strength was more stable and as a result the throughput was higher.

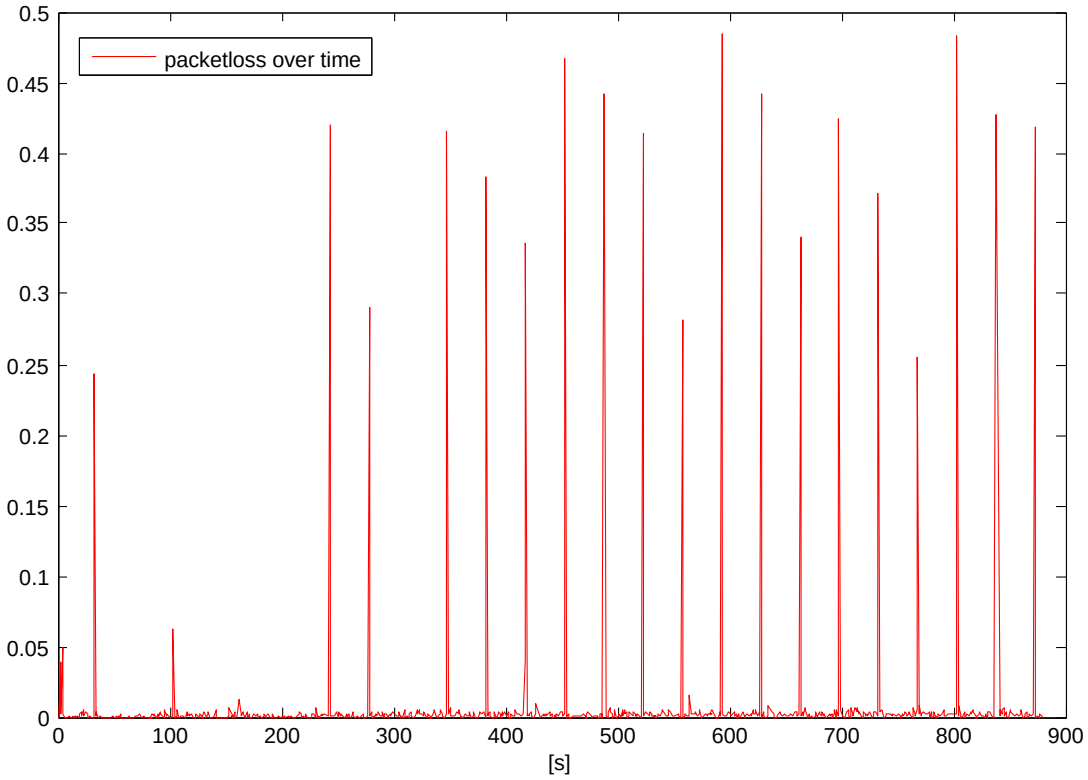

Figure 6.1: Packet loss over time at 11 Mbps

At 1 and 11 Mbps the results were pretty much the same on all the frequency ranges. However there were measurable differences at 6 and especially at 54 Mbps. The performance of the 900 MHz link is notably the worst of the three.

One noticeable result of the measurements is shown in Figure 6.1. This figure shows the packet loss at 11 Mbps over a period of 15 minutes. Overall the packet loss is fairly

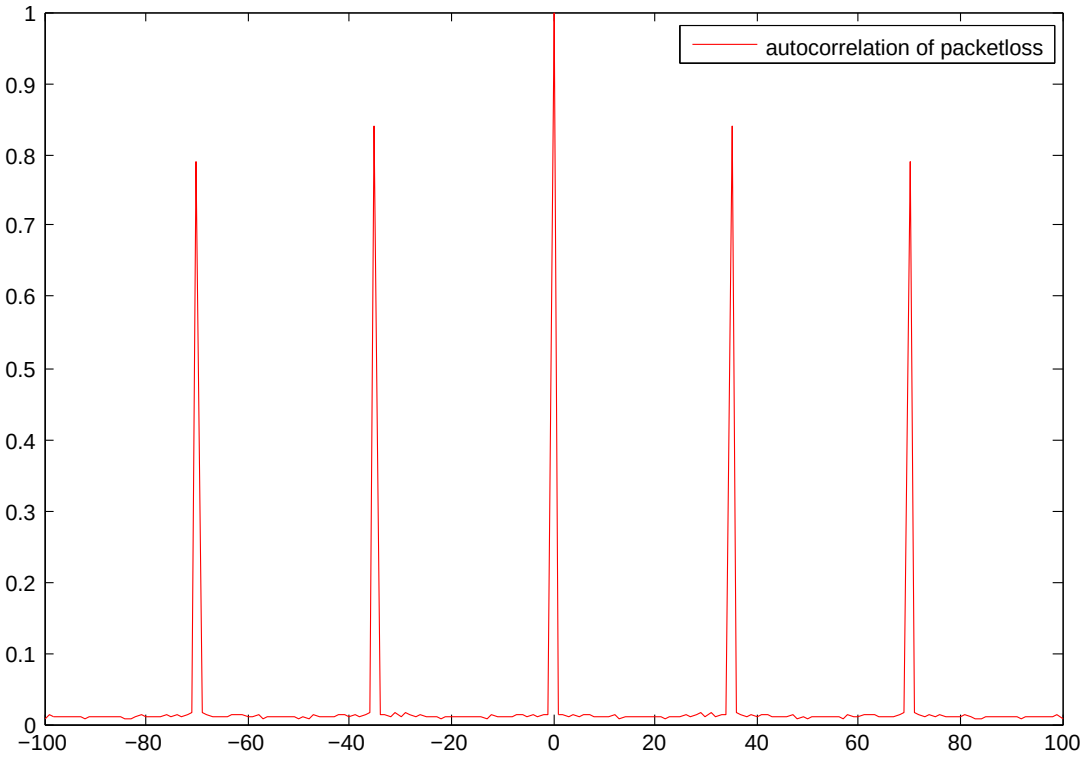

Figure 6.2: Autocorrelation of the packet loss at 11 Mbps

low. However, there are remarkable peaks in the packet loss. Every 33 seconds the link falls down until a significant amount of the packets were dropped. The autocorrelation of the loss rate is shown in Figure 6.2. Obviously the peaks are highly periodic with a correlation factor of over 0.8 at a period of 33 seconds. Those peaks can be found in our entire baseline measurements, although not all the time this distinctive. However, neither at 5 GHz nor at 2.4 GHz we could observe something similar during our baseline measurements.

With a periodicity this high, it is very unlikely that this is a radio phenomenon. For a result of fading, it is far too periodic. Fading phenomena tend to be more stochastic. Another alternative would be interference with external sources. But this is unlikely, too. Periodic beacons are usually much more frequent and our subsequent measurements with a spectrum analyzer showed no signs of periodical interfering signals (see section 6.5).

This allows us to draw the conclusions, that this is most likely the result of some hardware or software malfunction. The radio crosstalk measurement (section 6.2) supports the theory of a malfunction in the device.

Another outstanding result is the sensitivity towards multipath fading. Slight movements of the receiver can lead to drastic changes in the packet loss of the link. Single persons moving inside the room can have the same result. We observed a link with a packet loss of less then 5 percent a second that fell to clearly over 80 percent packet loss after we moved the receiver box a few centimeters. To measure this phenomenon in a more controlled fashion, we came up with the idea to use a shakeboard (see section 6.3).

Apart from that, there were no remarkable results in our baseline measurements. It might be noteworthy that the MadWifi driver and therefore our nodes were not running as stable as we had hoped. We had to repeat lots of measurements due to node crashes. Especially changes to the used channel or the transmission rate tend to break the driver.

6.2 Radio Crosstalk Measurement

One of our primary goals was getting enough knowledge to make a reasonable comparison between IEEE 802.11 at 900 MHz and the IEEE 802.11 at the standardized frequencies. Therefore it is very valuable, to measure both frequencies at the same time. We made multiple measurements with devices, that contained only one radio or both radios, respectively. We were not able to notice any differences between measurements conducted with only one radio and measurements with two radios, transmitting at the same time. As long as the two radios transmit at different bands, according to our results there is no significant influence between the two transmissions with regards to received signal strength or packet loss.

The mean loss rates and the mean received signal strength were fluctuating around the same values regardless of a second transmission being in place from the second interface.

However, the significant peaks in packet loss and as a result of this a decreasing throughput which we detected and described in section 6.1 can be observed at other frequencies as well if, and only if, we transmit at two frequencies simultaneously. The peaks occur at the same time. We assume that the software stalls for an instant which results in missing packets.

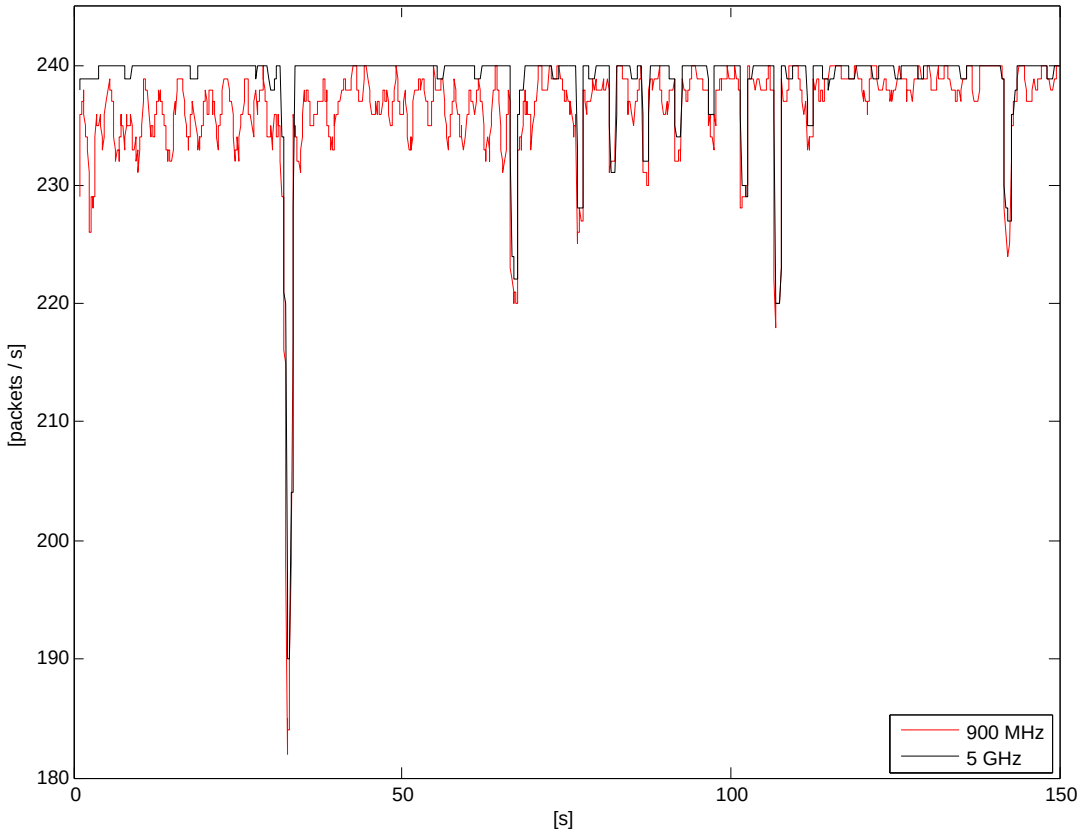

Figure 6.3: Throughput of 900 MHz and 5 GHz simultaneous transmission at 6M

Figure 6.3 shows the throughput of a link at 900 MHz and a link at 5 GHz both

transmitting simultaneously. Obviously the 900 MHz link is not as stable as the 5 GHz link. They both achieve similar throughput. But the interesting thing about the two links is that both links decrease simultaneously significant in throughput.

Nevertheless we made most of our follow up measurements at 900 MHz and one of the other two bands simultaneously. Those sharp declines are disappointing, but since we could not observe any other influence between simultaneous transmissions, to us, the benefit of having a direct comparison between the frequencies outweighs this drawback.

6.3 Shakeboard Measurement

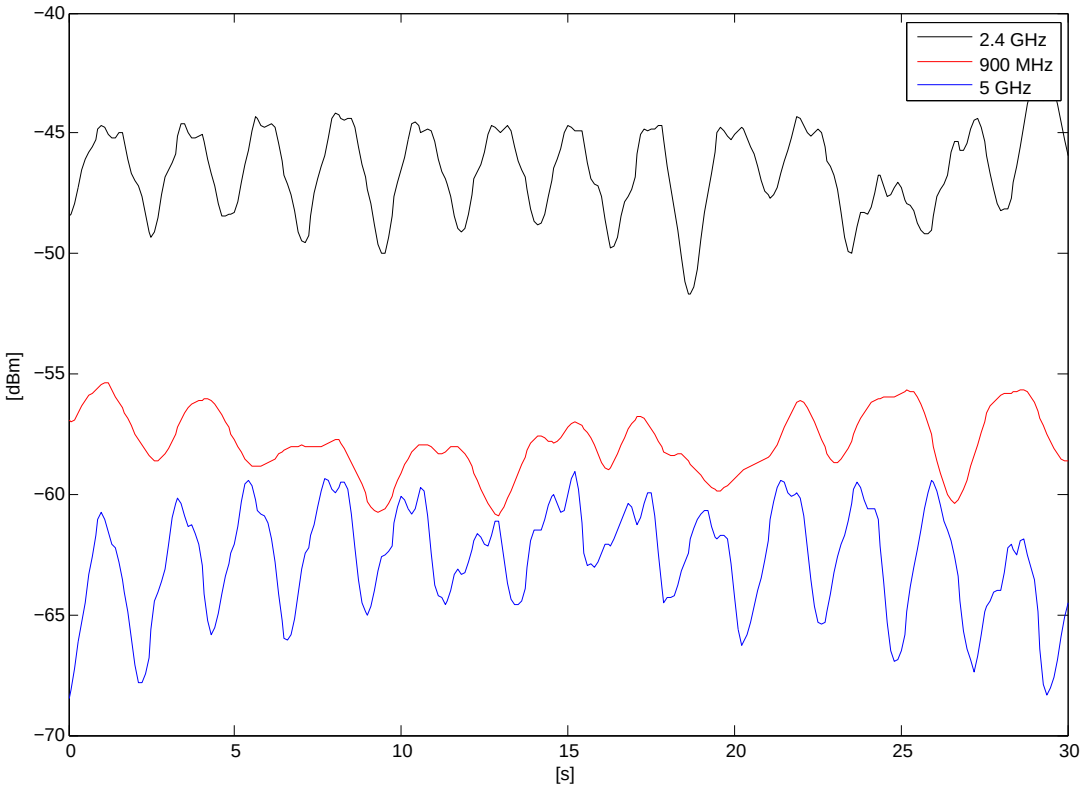

Figure 6.4: Development of the signal strength during periodic motion over 30 seconds

The results of the shakeboard measurements were pretty surprising. We were expecting to see an impact but we were not expecting such clear and obvious results. The received signal strength clearly yields a highly regular pattern. During long periods of time, the signal oscillates around a stable value. Figure 6.4 shows the development of the signal strength over 30 seconds on each of the three measured bands. It should be noted, that those three measurements were performed one after another. Therefore the waves in Figure 6.4 do not align perfectly.

It is particularly interesting that all three bands react similar to the movement. With the chosen movement range of about 17 cm we were expecting to see clear results at 2.4 GHz, since the wavelength at that band is 12.5 cm. A horizontal movement along one entire wavelength should reveal clear results. At 5 GHz we were even moving through nearly three complete wavelengths (6 cm). Finally at 0.9 GHz, we were moving the receiver through just half a wavelength (33 cm).

Nonetheless results are similar to each other. It appears though, that one can see the three distinct wavelengths that are passed through in the middle of the 5 GHz curve in Figure 6.4.

Obviously the 900 MHz signal is least affected by movement. This is probably due to the fact, that the receiver is not moving an entire wavelength.

Although the signal strength shows clear evidence of the movement, this does not hold

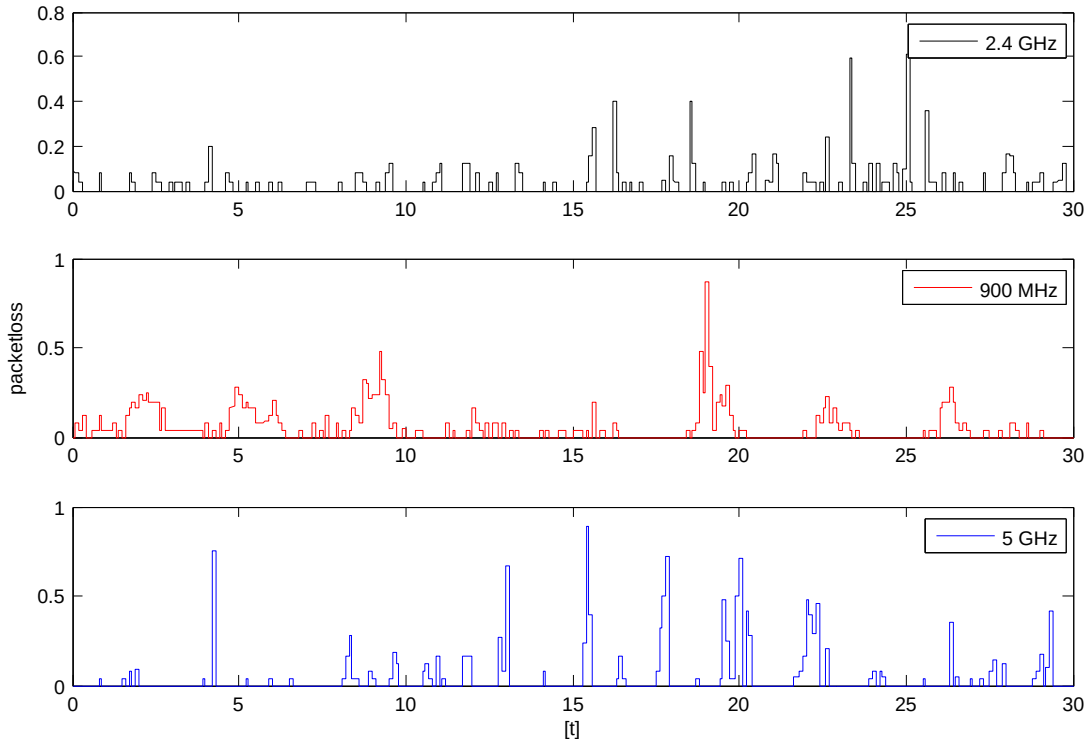

Figure 6.5: Development of the loss during the same period of time as in Figure 6.4

for the actual quality of the link. Figure 6.5 shows the packet loss over the exact same period of time as Figure 6.4. The packet loss of the 900 MHz link is the only one that has clear sinus like elements in it. The plot at 5 GHz is also periodic but with much sharper peaks. The packet loss at 2.4 GHz does not seem to be periodic which is highly surprising since the signal strength showed a near perfect sinus shape.

One explanation might be the fact, that the received signal at 2.4 GHz was the strongest overall and the resulting loss is mostly due to multipath instead of signal fading.

It is also important to note, that regarding all the different frequencies, the 900 MHz link was suffering most from periodic movement. This is a bit surprising, since the received signal strength was varying the least at 900 MHz.

It appears as if the SR9 is more vulnerable to multipath as regular WiFi cards. This assumption is also supported by our baseline measurements (section 6.1).

6.4 Distance Path Loss Measurement

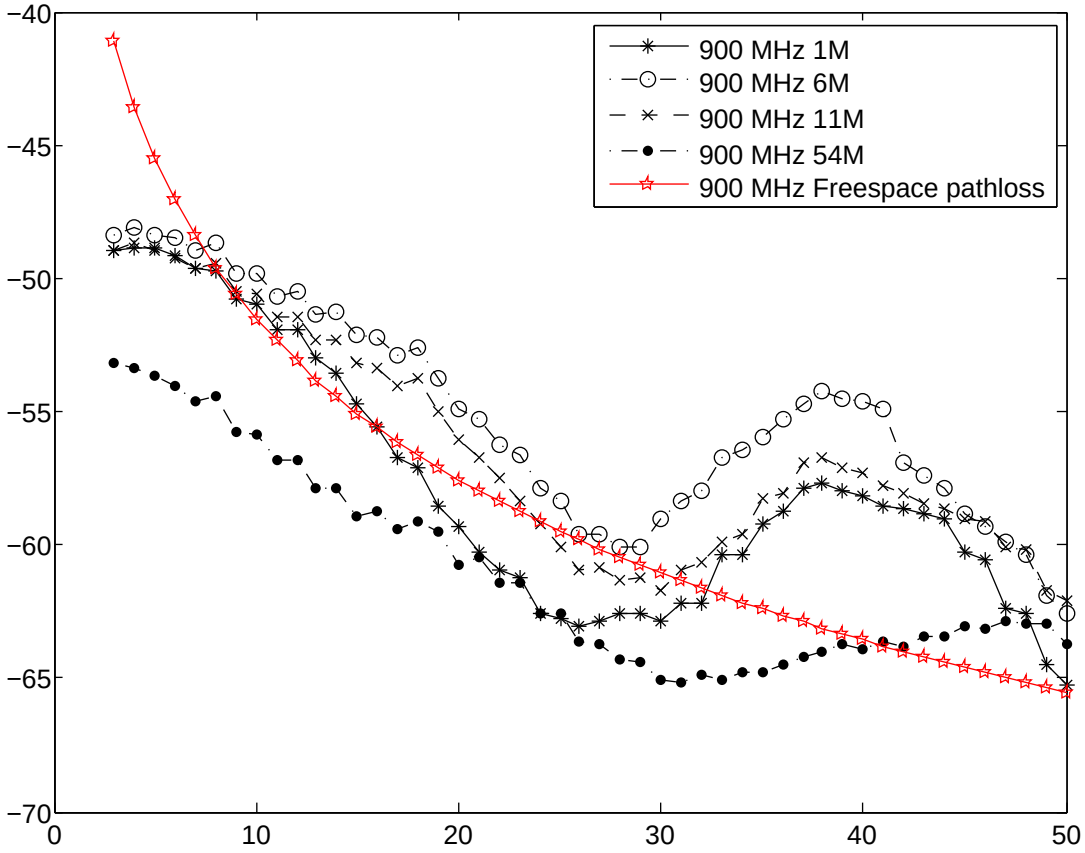

Figure 6.6: Development of the received signal strength along a corridor at 900 MHz

The results of the distance path loss measurements along a corridor are very well in line with our previous findings. The results in Figure 6.6 are filtered to a certain extend and at some points are interpolated. Sadly, we had to interpolate some values, because some data points were not usable, since the measuring boxes crashed unnoticed. We filled those gaps with linear interpolation. Thereafter the results were filtered, to get rid of unreasonable peaks. First of all, the results are pretty sound. As Figure 6.6 shows, the received signal strength slowly decreases with the distance to the sender. The slight increase of the signal strength after 28 meter is very likely a result of some constructive reflections along the corridor. Nevertheless the general trend is surprisingly close to the theoretical model of free space path loss, which is also in Figure 6.6 as a reference. The signal strength at 54 Mbps is below that of the slower links because the SR9 is transmitting with less energy at higher data rates.

Figure 6.7 shows the results from 2.4 and 5 GHz from the exact same setup. The general trend is fairly obvious and not surprising. The free space path loss is used as a reference again. The signal trend at 5 GHz matches the free space path loss very well. The 2.4 GHz signal is a bit stronger than expected but not unusually high.

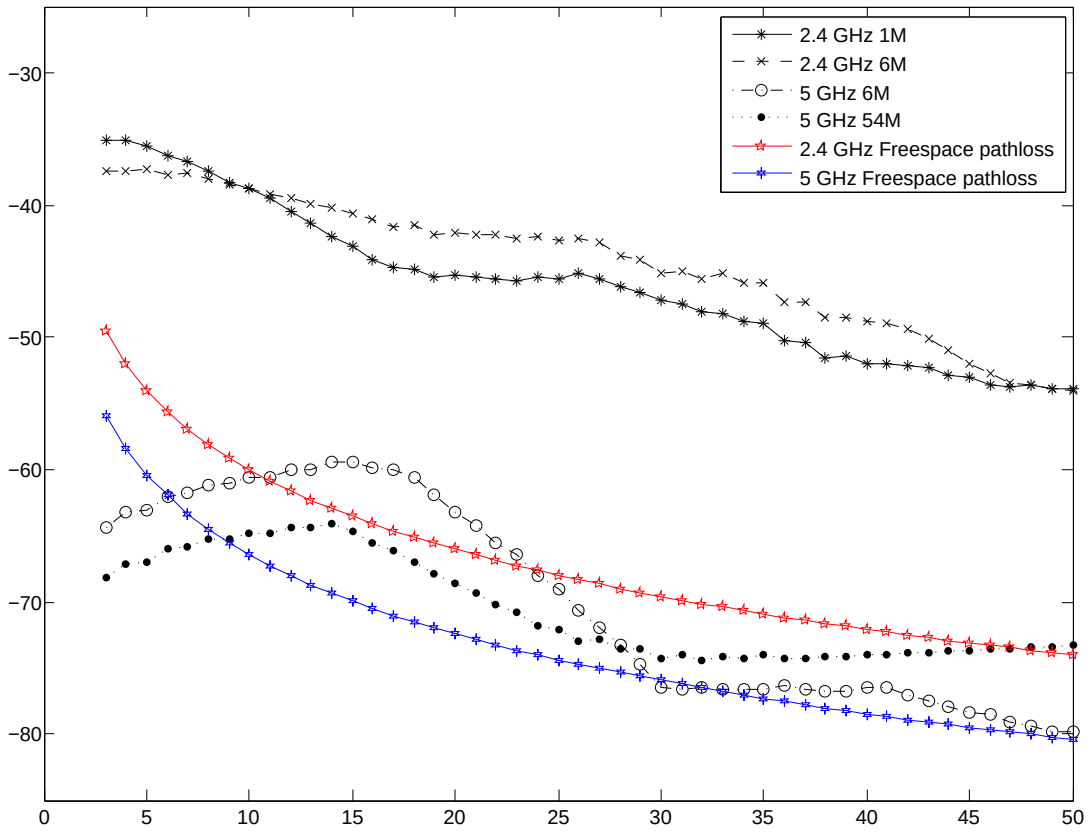

Figure 6.7: Development of the received signal strength along a corridor at 2.4 and 5 GHz

6.5 Spectrum Analyzer Measurement

The measurements with the spectrum analyzer provided us with three very important insights. Our SR9 is emitting way too less energy, the used band is virtually empty and the datasheet has to be taken with a grant of salt.

As Table 6.1 shows, the emitted power at 900 MHz is far below that of the other frequencies. And it is orders of magnitude below the specified transmit power of 28 dBm.

Since none of the values matches those of the datasheet, the calibration of the spectrum analyzer is very doubtful. Nonetheless the transmitted energy is by far the least at 900 MHz albeit it should be the highest of the three.

We can only speculate about the reasons. We measured another SR9 card with similar results, to exclude a defect device. Our best guess is a power shortage in the cards energy supply. According to the specification of the Mini PCI bus, devices are allowed to consume up to 2 W [24]. However, the SR9 consumes up to 3.9 W during transmission ($1.2A \times 3.3V = 3.9W$). To put that into contrast, the Winstron CM9 card consumes about 0.16 W.

We tried to measure the 3.3 V lanes of the Mini PCI bus with a volt meter. In theory, if the card can not get enough power, the current should collapse. During our measurements with the volt meter, the current was constant at 3.3 V. It should be noted, that the volt

meter measurements are very inaccurate. The volt meter is pretty dull, so that flickering results can not be accurately measured. In addition to that, the Mini PCI port is not suited to be measured directly with a rather blunt volt meter.

That said, we were not able to prove this thesis but we also could not disprove it completely. But if our thesis is right and the card is suffering on a power shortage, this could lead to all kinds of strange results.

Despite that the measurements with spectrum analyzer also provided us with some promising results. Figure 6.8 shows a waterfall diagram of the frequency range between 900 MHz and 930 MHz over one minute. This measurement took place at a regular Friday afternoon. One can clearly see a pretty strong signal with -70 dBm at about 920 MHz. According to the "Bundesnetzagentur", this is either a carrier of the "Deutsche Bahn AG" or may be in use by the military [25]. Apart from that, the frequency range used by the SR9 is clearly empty.

Since we did not want to collide with this signal, we were measuring all the supported channels of the SR9. Our findings are listed in Table 6.2. The channels in parenthesis are officially not supported by the SR9. However the MadWifi driver accepted them and the SR9 transmits far outside of its specified range. Another thing worth mentioning is the fact that the channels are ordered inversely. The lowest channel has the highest frequency and vice versa. This is clearly important for our measurements, because we do not want to interfere with the detected signal at 920 MHz.

Regardless of those findings, we will still refer to the four officially supported channels as the only supported channels. We were not performing any measurements on any channel outside the vendor's specification.

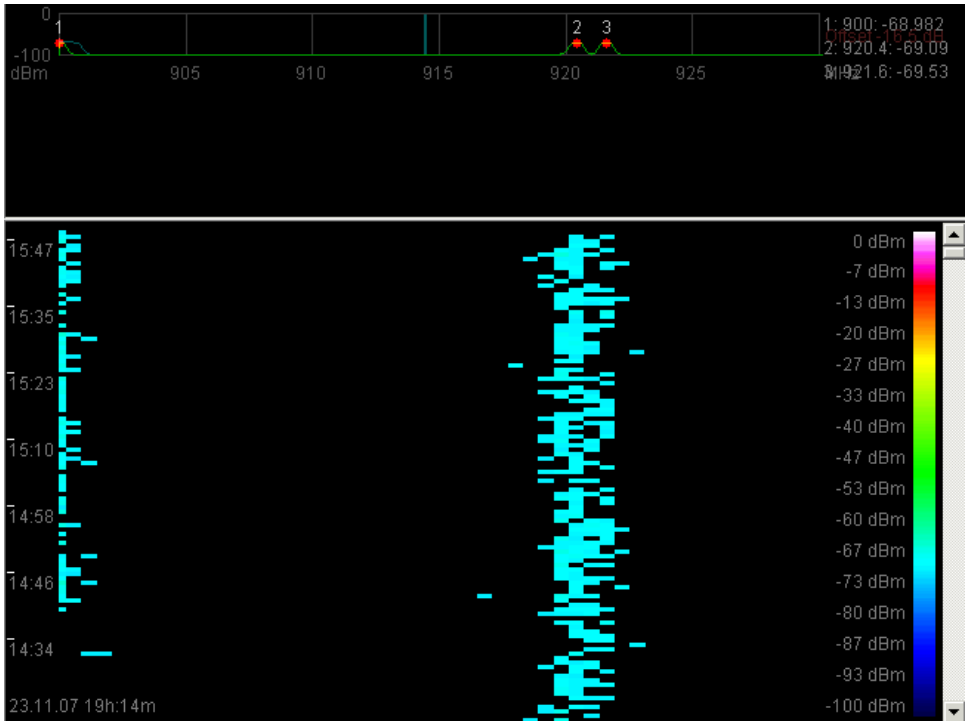

Figure 6.8: Waterfall diagram of the used spectrum

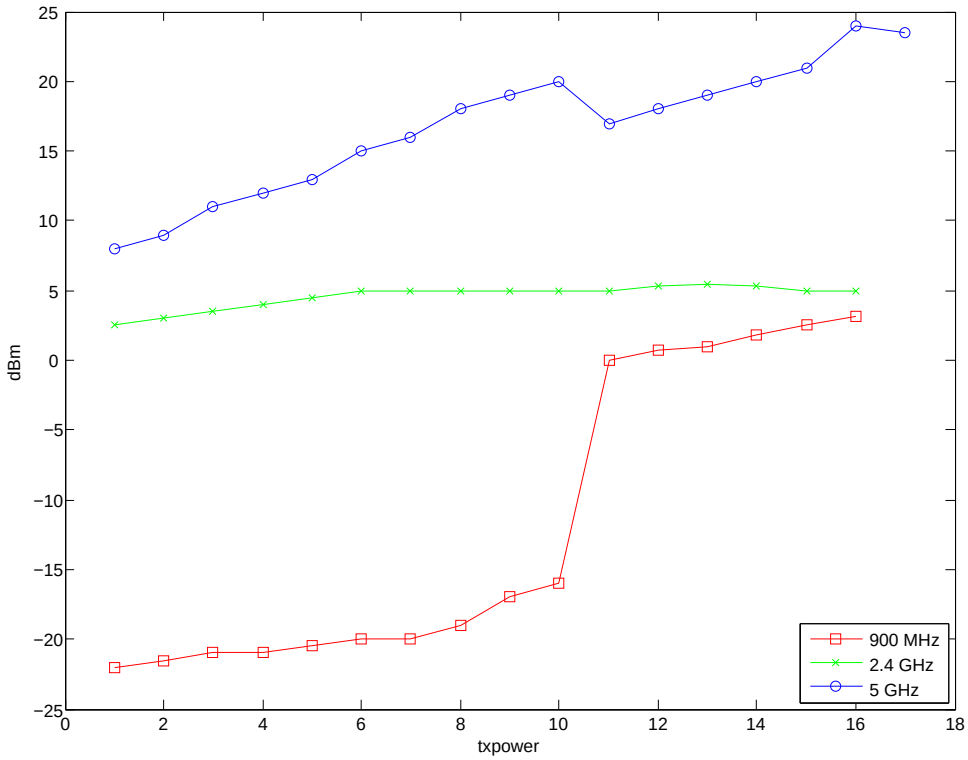

Figure 6.9: Transmission power at the different MadWifi txpower settings

Table 6.2 also shows, that all channels are 10 MHz width. This is only true for non OFDM rates. While transmitting at OFDM rates, the card uses 20 MHz. Due to some sort of echoing within our spectrum analyzer, we were not able to measure the exact location of the channels at OFDM rates. Each signal in the spectrum analyzer is echoed at 20 MHz to both sides. Therefore it was impossible to locate the beginning and the end of the channel at OFDM rates because the analyzer only displayed a wide, pulsing range.

Table 6.1: TX energy results at 6 Mbps

txpower	900 MHz (dBm)	2.4 GHz (dBm)	5.2 GHz (dBm)
1	-22	2.5	8
2	-21.5	3	9
3	-21	3.5	11
4	-21	4	12
5	-20.5	4.5	13
6	-20	5	15
7	-20	5	16
8	-19	5	18
9	-17	5	19
10	-16	5	20
11	0	5	17
12	0.7	5.3	18
13	1	5.4	19
14	1.8	5.3	20
15	2.5	5	21
16	3.1	5	24
17	n/a	n/a	23.5

6.6 Bit Error Measurement

A summary of the results of the bit error measurements can be explained best with the help of some figures. First, Figure 6.10 shows a link at 900 MHz. One can see approximately 12000 Bits on the x-axis (1500 Byte payload * 8 Bits per Byte) and on the y-axis the probability of of a corrupt bit on this exact position. 21533 packets were sent. The two plots represent two different and independent links of two independent nodes. Within the points a moving average of the bit error probability is plotted with a windows size of 100 bits. The link quality is obviously different, although the nodes had an equal distance to the sender: they were placed about 2 meters apart from each other.

One can see two obvious things: First, there are great periodic peaks on specific bit positions and smaller oscillations all over the graph. The bigger periodic peaks start at about 2500th bit position and then occurs regularly with a periodicity of about 1050 bits, the smaller one has a period of about 200 bits. The first link shows this effect in a much stronger way than the second one, but the oscillation can still be seen. The Second thing, we can extract from this figure is a constant level of bit errors at about 0.04, this acts a lower threshold for the first plot, and a lower threshold of about 0.05 for the second plot, respectively – there are virtually no values underneath this line. A correlation of these two links and their autocorrelation can be seen in Figure 6.11. The link names used here are the same in the above mentioned plot (Figure 6.10). This means ‘link 1’ and ‘link 2’ are representing the same links – the third one indicates the correlation to each other. One can see a great correlation at a lag of about 1050 with a correlation factor of 0.7 for link 1 and a correlation factor of 0.4 for link 2, respectively. The high correlation of this

Table 6.2: Channels

Channel	Frequency in MHz (using DSSS modulation)
(1)	927 - 937
(2)	922 - 932
3	916 - 926
4	912 - 922
5	907 - 917
6	901 - 911
(7)	897 - 907
(8)	892 - 902
(9)	887 - 997
(10)	882 - 892
(11)	877 - 887

two links reveals that this effect may have an external source. If a specific bit is corrupt for one link, it is likely that the other link did not receive it correctly, too.

Similar measurements have been made by others. Although the same frequencies were used, they could not observe any tendency for bad bits at a particular bit position within a packet [26] or they simply showed no characteristics [27] at all.

To get a better comparison to the different frequencies used, we did the same measurement for the 2.4 GHz link, too. Therefore, we used the same setup of nodes as for the 900 MHz measurements. In this case, we send 14365 packets. Basically we can see a slightly different behavior compared to the last example (Figure 6.12). There are still some oscillations within the probability of the bit errors over the bit position, but it shows a different periodicity. The main peaks are about 1000 bits apart, but the smaller ones can not be seen clearly. Another difference is the factor by which the error probability increases. In Figure 6.10 the first link jumps from an error probability of about 0.05 to 0.3 – which is a factor of 6, and from 0.1 to 0.25 for the second link, which is factor 2.5. In Figure 6.12 these steps are smaller but still increase by a factor of about 0.5. But, as a main difference the socket amount of bit errors increases towards the end of the packet. This behavior can not be seen in the 900 MHz link of Figure 6.10 so far.

Last but not least, the same measurement for the 5 GHz band (Figure 6.13), using channel 36 with the same wireless NIC as used for the 2.4 GHz measurement. Here we can not see any of the above mentioned points. There is no significant deviation depending on the bit position and the distribution of errors seems to be completely random.

To get more into the behavior of these periodical oscillations, we took some more measurements in the 900 MHz band but this time at different rates. Since we assumed a constant external jamming source the error characteristics should differ as we change the rates and modulations. The redundancy increases for lower transmission rates – so if there were external interferences – the effect on the links should decrease by lowering the rates. Surprisingly, nothing changed at all. If we take a look at the 11 Mbps measurement in Figure 6.16, we can see, that there is an equivalent error pattern present. The 11 Mbps transmission rate uses DSSS modulation, the 54 Mbps links uses OFDM modulation,

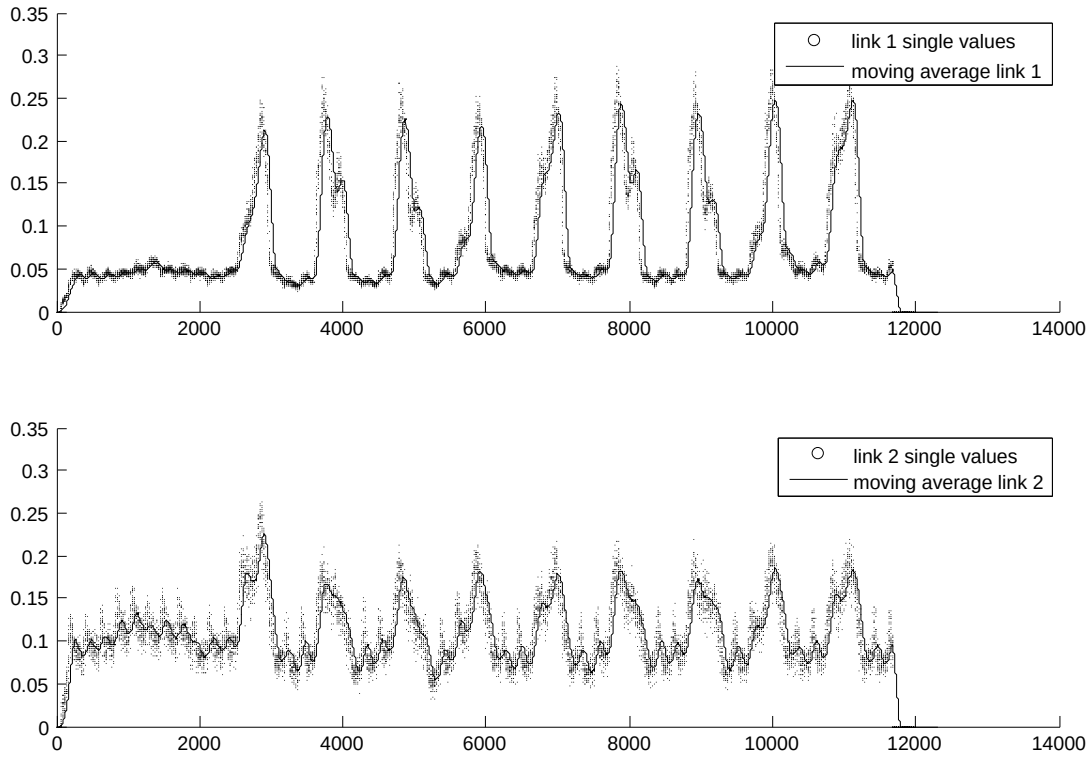

Figure 6.10: Bit error quantity/bitposition in the 900 MHz band, 54 Mbps

respectively. Figure 6.16 reveals that even the lag of the peaks is the same, in numbers: an autocorrelation shows a factor of 0.5 at a lag of about 1050 bits – this is virtually the same as seen in the 54 Mbps link. Another remarkable circumstance is that this link is nearly perfect. It has an excellent quality in packet error rate: about 0.82 of the packets of link 1 were received correctly and about 0.92 of link 2 were totally correct. A cumulative distribution function of the links quality is shown in Figure 6.14. 0.999 of packets of link 1 had 0.97 correct bits, and 0.997 of link 2, respectively

But nevertheless these above mentioned effects appear, which leads us to the assumption that these error characteristics are hardware related. Figure 6.15 holds one point worthy of mention: link 1 shows a similar error pattern as the link 1 of Figure 6.12 – But Link 2 behaves like most of the other measured links. This measurement had the same setup and the same hardware as the other ones in this section, but we had no explanation for this behavior.

To get a better comparison of these plots, we merged them together into one figure. In Figure 6.19 we can see that they all behave in a similar manner. The peaks of different measurements seems to have a small drift, so that the peaks occur at different bit positions – peaks of the same measurements always occur at the same position. The label shows which links belong together. The span between two peaks is approximately 1050 bits.

Further analysis of the errors did not reveal additional characteristics. The burst error length has an exponential decline with mostly short bursts and only a few longer ones Figure 6.18.

It remains unclear why this pattern occurs on different rates, even on different WiFi

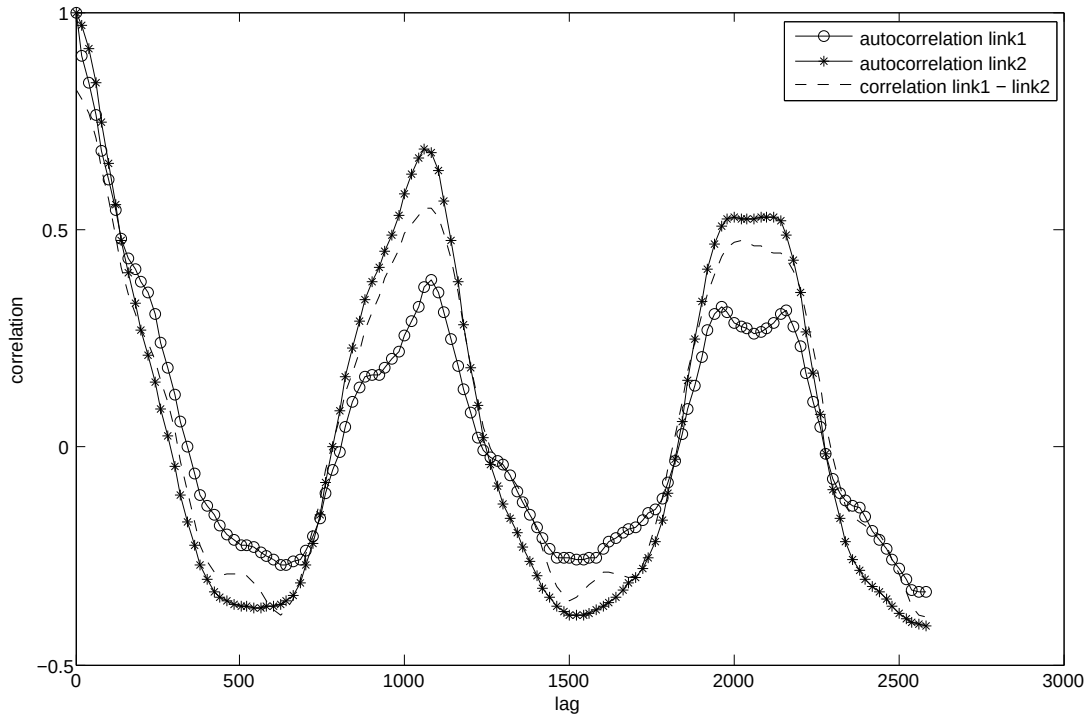

Figure 6.11: Bit error correlations in the 900 MHz band, 54 Mbps

hardware in different frequency bands. The equivalent error distribution for two independent links only leads to the conclusion that this is an effect generated by the sender. To be more specific, equal patterns were generated even if we had exchanged the sender for another one. Both cards had Atheros chips and therefore used the very same driver and both cards were powered on simultaneously.

In smaller dimensions, regarding bit errors within packets – this effect is really remarkable, as we have just revealed. In larger scales, observing the error rates of complete packets over time – a totally different behavior occurs. The graph (Figure 6.20) does not show remarkable anomalies. Even changing the transmission rates and choosing different frequencies, therefore different hardware is used, did not affect the results significantly. The graph shows some significant changes of link quality that may be caused by large scale fading. Similar measurements had been made by [23].

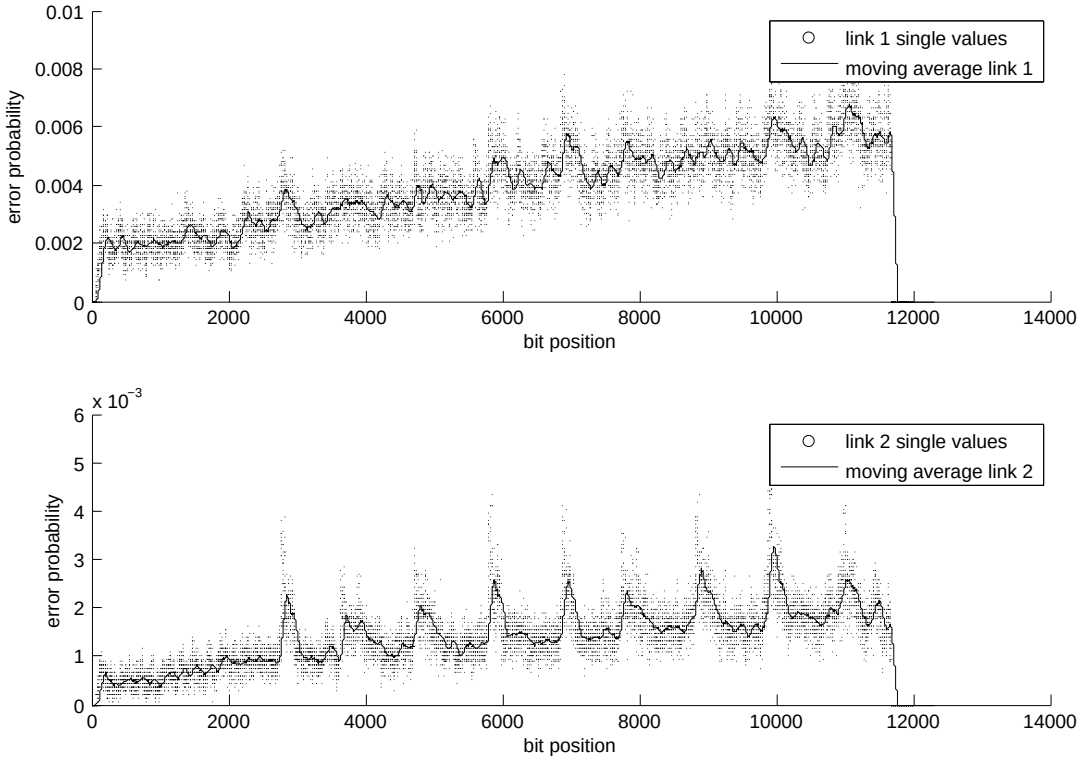

Figure 6.12: Bit error quantity/bitposition in the 2.4 GHz band

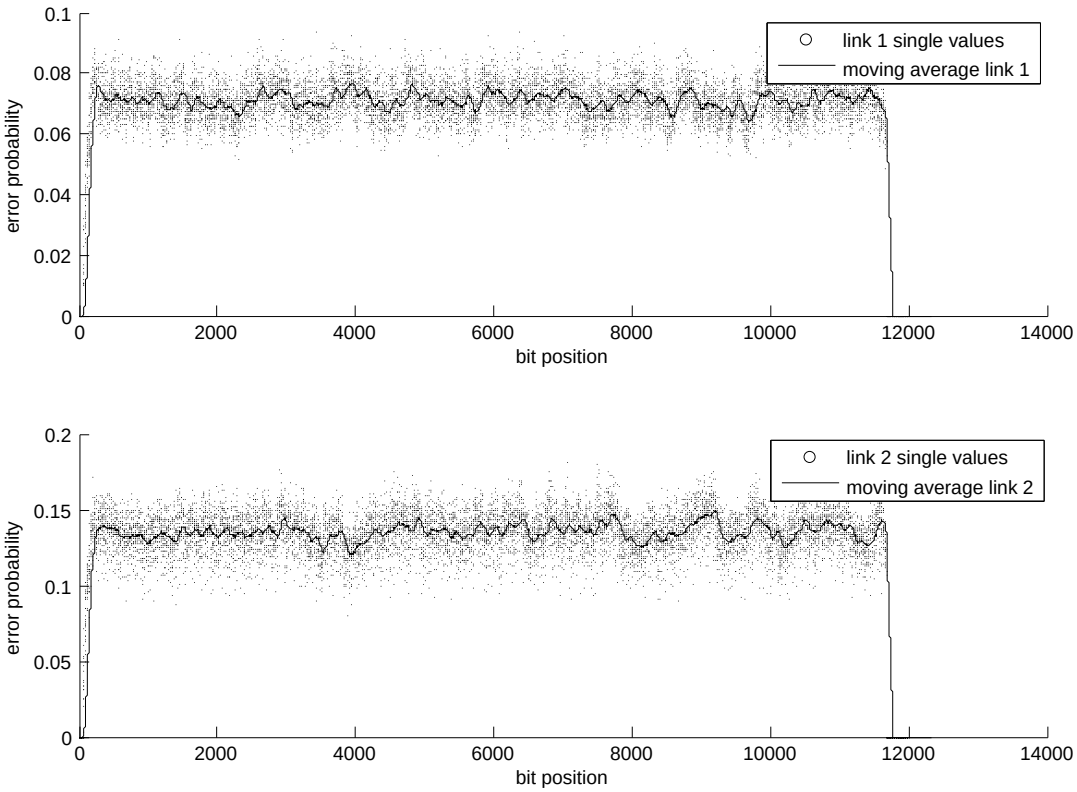

Figure 6.13: Bit error quantity/bitposition in the 5 GHz band

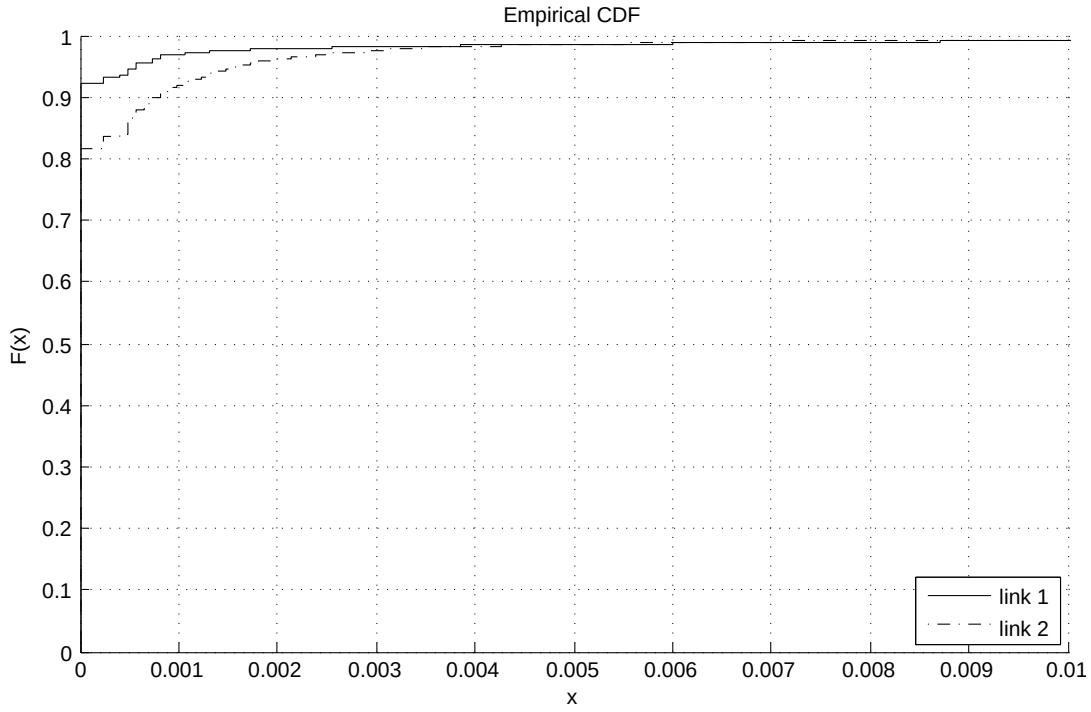

Figure 6.14: Cumulative distribution function: probability of bit errors per complete packet in the 900 MHz band, 11 Mbps

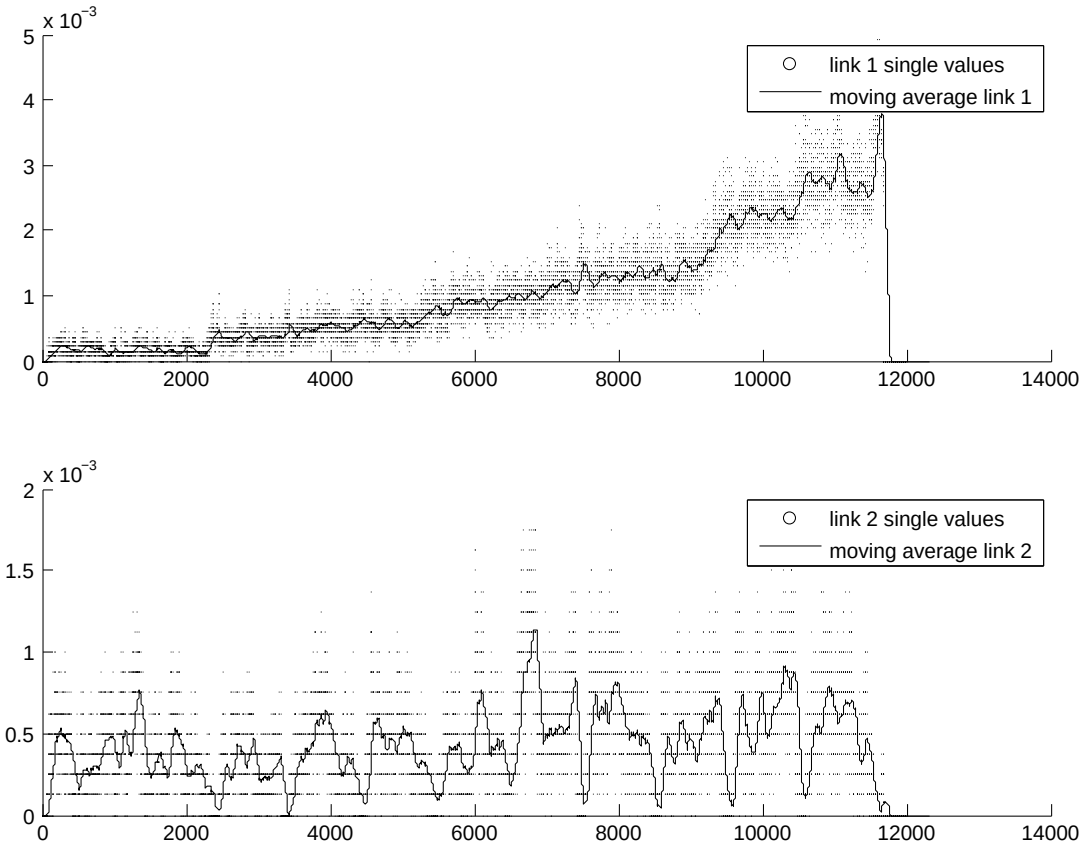

Figure 6.15: Bit error quantity/bitposition in the 900 MHz band, 18 Mbps

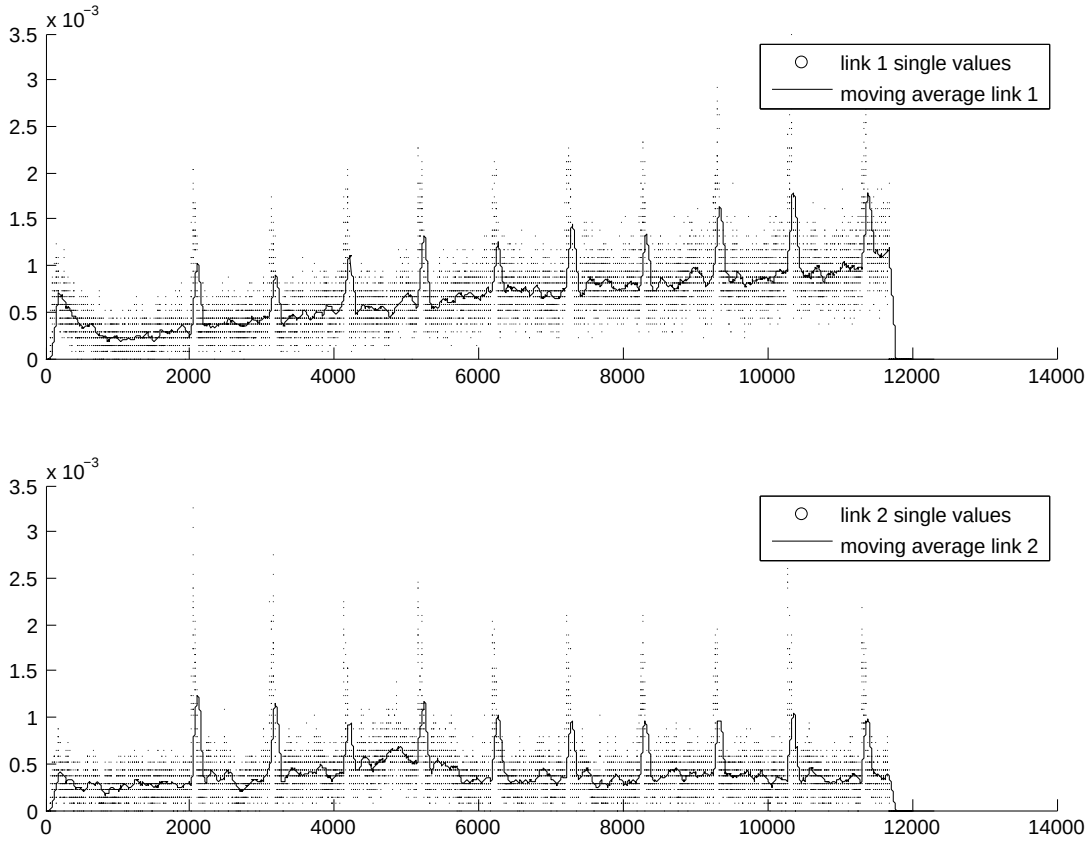

Figure 6.16: Bit error quantity/bitposition in the 900 MHz band, 11 Mbps

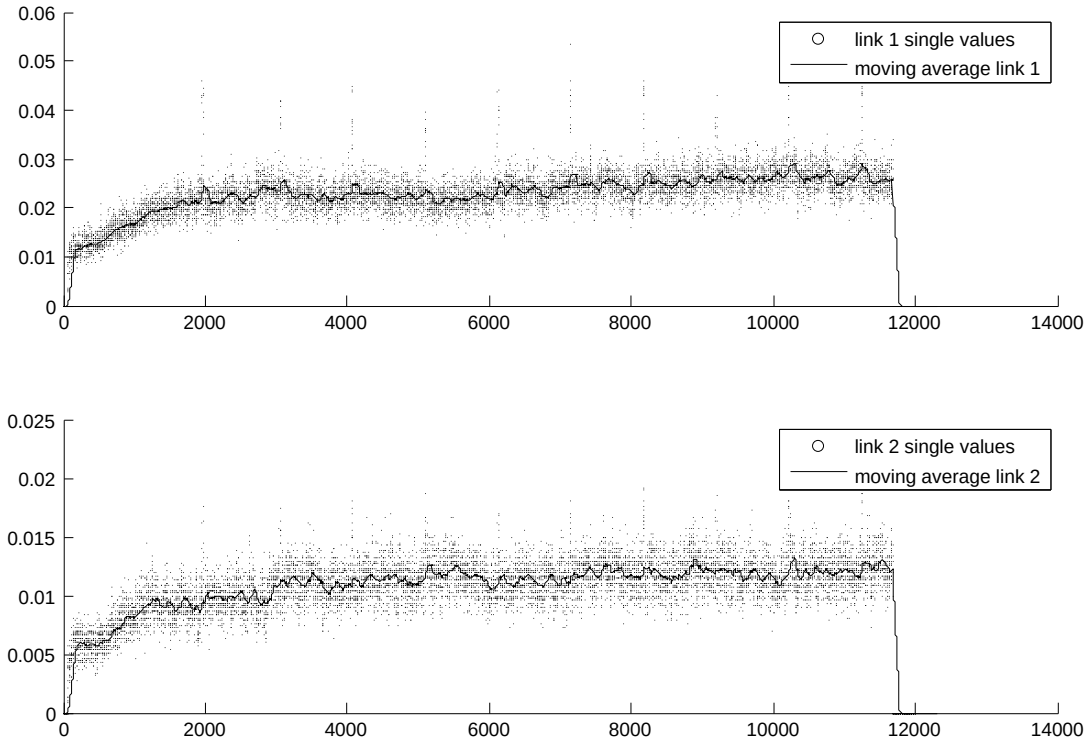

Figure 6.17: Bit error quantity/bitposition in the 900 MHz band, 1 Mbps

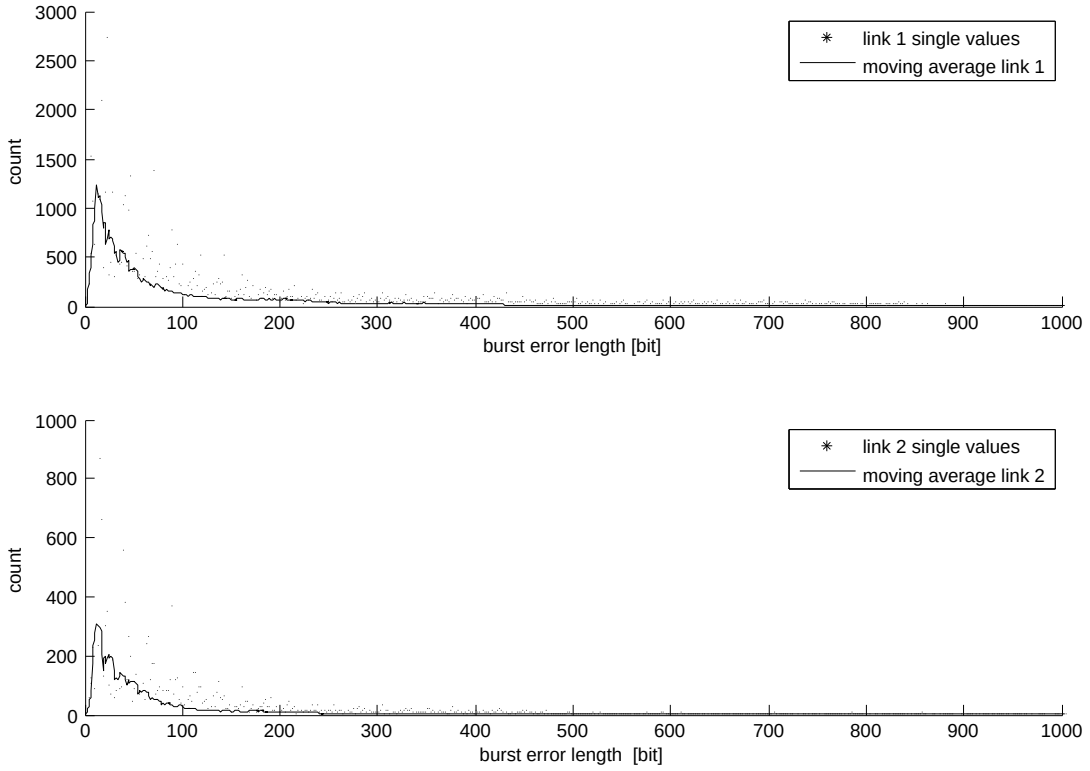

Figure 6.18: Burst error length in the 900 MHz band, 54 Mbps

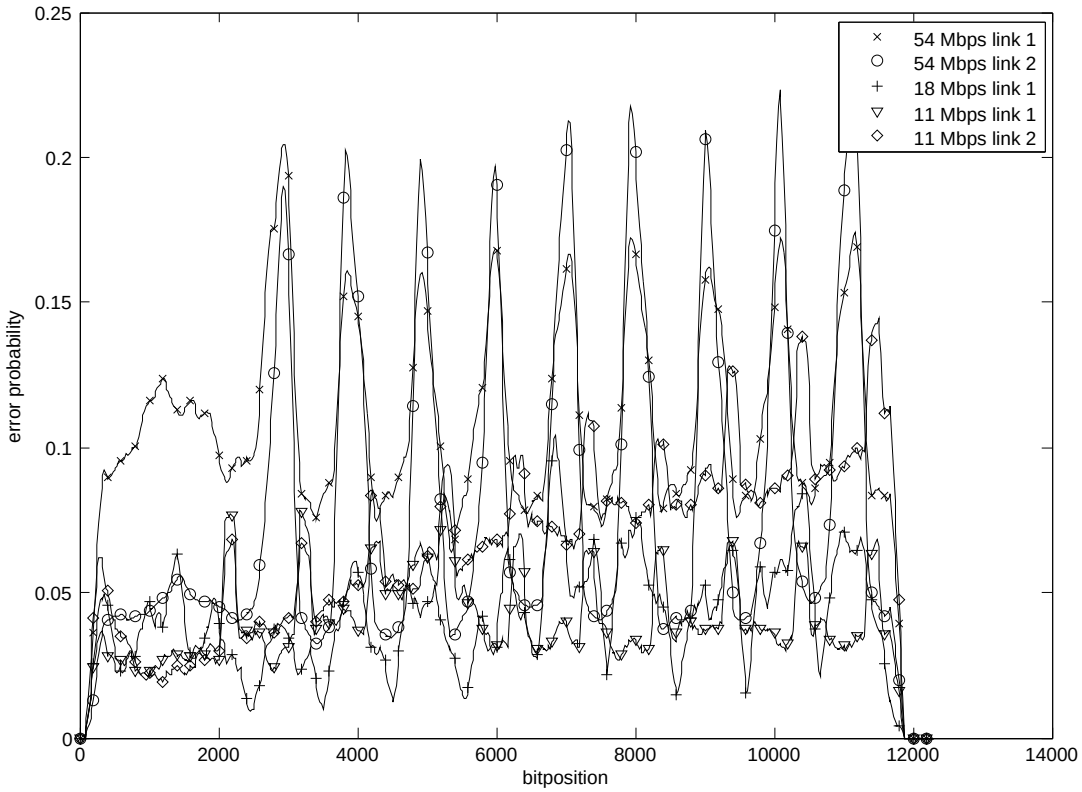

Figure 6.19: Bit error quantity/bitposition in the 900 MHz band, several rates

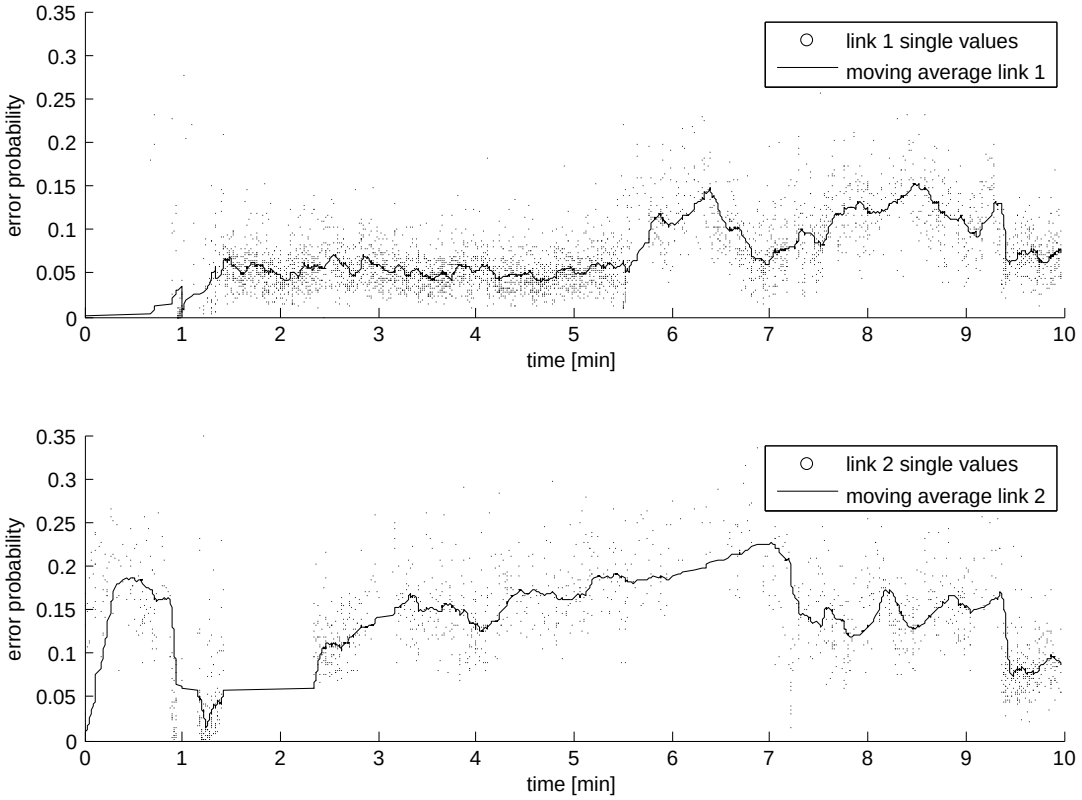

Figure 6.20: Bit error probability of whole packets 54 Mbps, channel 36, 5 GHz band

7 Discussion

We will first discuss the conclusions of our measurements with the Ubiquity SR9 and then briefly discuss the future of our measurement framework. Finally, we will talk about future work that can be based upon our findings.

7.1 Conclusion

Our findings yield the conclusion, that we can not recommend the use of the Ubiquity SR9 for the BRN. Despite the obvious advantage of using a completely unpopulated frequency range, we could not find any reasons to use the SR9 in an indoor environment. Measurements like [28] and [29] showed advantages in outdoor environments. Predominantly greatly improved distance could be achieved. But we could not find any advantages for indoor environments. Even worse, the advantages are coming with some serious drawbacks. The most significant are very high energy consumption (compared to regular IEEE 802.11 cards) and unstable links.

Even if we pass over the serious problems of unstable drivers and insufficient transmit power there are lots of disadvantages left. Especially in mobile urban mesh environments, the unstable links (probably caused by heavy problems with multipath) combined with the penalty to battery lifetime practicaly rule out the use of the SR9.

This is particularly disappointing since the theoretical model promised lots of advantages, as we already discussed in section 3.3.

Our results indicate that, there seems to be a much higher gain in performance for an urban mesh in improved and specialized protocols like [30] or better error correction [20] than in the switch to lower frequencies.

7.2 Framework

The wireless testbed framework was a valuable tool for our measurements. However at its current state, there are some serious limitations that decrease the value significantly. There are some rough edges that occurred while we were using it on a regular basis. But the most important weakness is the daemon process. It is currently not included in the default run level of the nodes and therefore it has to be started manually before every measurement. Since the nodes were not that stable during the measurements, we had to restart them frequently. Resulting in many annoying manual restarts of the daemon. The framework, together with setup instructions, will be available through sartrac [31]. If there are researchers willing to try and use it, we might provide support and subsequent improvements.

7.3 Future Work

The performance of the SR9 as indicated by our measurements is disappointing in an indoor environment. Therefore it is interesting to see if this is a general problem with the hardware or if the devices perform significantly better in an outdoor environment.

The high energy consumption of the SR9 suggests different results in a device capable of providing more energy than an embedded router can. As a result, a measurement with PCs could yield interesting results as well.

The interesting patterns in bit error position could yield to very interesting results. It would be valuable to search for their cause. If they are not just a problem of some specific devices, they might provide precious input for better error correction algorithms. One might even consider adding those regular patterns to network simulators.

Acknowledgment

We would like to thank Henryk Plötz for the spectrum analyzer. We also thank the Artificial Intelligence Workgroup for the shakeboard.

We really appreciate the constructive input by Anatolij Zubow, Mathias Kurth and all the other people at the Systems Architecture Group.

Bibliography

[1] "T-mobile hotspots." [Online]. Available: http://www.hotspot.de/

[2] "BT fon." [Online]. Available: http://www.btfon.com/

[3] "Freifunk." [Online]. Available: http://start.freifunk.net/

[4] "MIT roofnet." [Online]. Available: http://pdos.csail.mit.edu/roofnet/doku.php

[5] "Berlin roofnet." [Online]. Available: http://sarwiki.informatik.hu-berlin.de/BerlinRoofNet

[6] A. Goldsmith, *Wireless Communications*. Cambridge University Press, August 2005. [Online]. Available: http://www.amazon.ca/exec/obidos/redirect?tag=citeulike09-20\&path=ASIN/0521837162

[7] "WaveLAN devices and drivers." [Online]. Available: http://www.hpl.hp.com/personal/Jean_Tourrilhes/Linux/Linux.Wireless.drivers.html#WaveLAN

[8] T. S. Rappaport and T. Rappaport, *Wireless Communications: Principles and Practice (2nd Edition)*. Prentice Hall PTR, December 2001. [Online]. Available: http://www.amazon.ca/exec/obidos/redirect?tag=citeulike09-20\&path=ASIN/0130422320

[9] "Ubiquiti SR9 data sheet." [Online]. Available: http://ubnt.com/downloads/sr9_datasheet.pdf

[10] "ISO 9613-1 extract, Attenuation of Sound in Air." [Online]. Available: http://www.kayelaby.npl.co.uk/general_physics/2_4/2_4_1.html

[11] E. Eide, L. Stoller, and J. Lepreau, "An experimentation workbench for replayable networking research," in *4th USENIX Symposium on Networked Systems Design & Implementation*, 2007, pp. 215–228. [Online]. Available: http://www.usenix.org/events/nsdi07/tech/eide.html

[12] D. Raychaudhuri, I. Seskar, M. Ott, S. Ganu, K. Ramachandran, H. Kremo, R. Siracusa, H. Liu, and M. Singh, "Overview of the orbit radio grid testbed for evaluation of next-generation wireless network protocols," vol. 3, 2005, pp. 1664–1669 Vol. 3. [Online]. Available: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=1424763

[13] R. Riggio, N. Scalabrino, D. Miorandi, and I. Chlamtac, "Janus: A framework for distributed management of wireless mesh networks," in *TRIDENTCOM 2007*, 2007. [Online]. Available: http://www.wing-project.org/_media/publications:tridentcom2007_janus.pdf?id=start\&cache=cache

[14] K.-H. Kim and K. G. Shin, "On accurate measurement of link quality in multi-hop wireless mesh networks," in *MobiCom '06: Proceedings of the 12th annual international conference on Mobile computing and networking*. New York, NY, USA: ACM Press, 2006, pp. 38–49. [Online]. Available: http://portal.acm.org/citation.cfm?id=1161095

[15] L. Yuan, C.-N. Chuah, and P. Mohapatra, "Progme: Towards programmable network measurement," August 2007. [Online]. Available: http://www.sigcomm.org/ccr/drupal/?q=node/250

[16] "MadWifi." [Online]. Available: http://madwifi.org/

[17] "MITs Click Modular Router Project." [Online]. Available: http://www.read.cs.ucla.edu/click/

[18] J. Robinsony, K. Papagiannakiz, C. Diotz, X. Guo, and L. Krishnamurthy, "Experimenting with a multi-radio mesh networking testbed," Tech. Rep.

[19] "Department of Artificial Intelligence, Humboldt-University Berlin." [Online]. Available: http://www.ki.informatik.hu-berlin.de/

[20] D. Eckhardt and P. Steenkiste, "Measurement and analysis of the error characteristics of an in-building wireless network," in *SIGCOMM*, 1996, pp. 243–254. [Online]. Available: http://citeseer.ist.psu.edu/eckhardt96measurement.html

[21] "Spectran spectrum analyser." [Online]. Available: http://www.elektrosmog.de/Spektrumanalysator.htm

[22] "Chaos Computer Club e.V." [Online]. Available: http://www.ccc.de/

[23] D. Dhoutaut, A. Regis, and F. Spies, "Integration of physical phenomena into an experiment-based propagation model," in *PE-WASUN '06: Proceedings of the 3rd ACM international workshop on Performance evaluation of wireless ad hoc, sensor and ubiquitous networks*. New York, NY, USA: ACM Press, 2006, pp. 98–105. [Online]. Available: http://portal.acm.org/citation.cfm?id=1163627

[24] "miniPCI specification." [Online]. Available: http://members.datafast.net.au/dft0802/specs/mpci10.pdf

[25] "Frequenznutzungsplan," Bundesnetzagentur, pp. 292–293, May 2006.

[26] D. A. Eckhardt and P. Steenkiste, "A trace-based evaluation of adaptive error correction for a wireless local area network," *Mob. Netw. Appl.*, vol. 4, no. 4, pp. 273–287, December 1999. [Online]. Available: http://portal.acm.org/citation.cfm?id=337845

[27] H. Dubois-Ferriere, D. Estrin, and M. Vetterli, "Packet combining in sensor networks," in *SenSys '05: Proceedings of the 3rd international conference on Embedded networked sensor systems*. New York, NY, USA: ACM Press, 2005, pp. 102–115. [Online]. Available: http://portal.acm.org/citation.cfm?id=1098918.1098930

[28] "Sr9 / mikrotik study pmp 900 mhz network performance investigation," Ubiquiti Networks. [Online]. Available: http://ubnt.com/downloads/SR9_Mtik_PMP.pdf

[29] "Sr9 / mikrotik study ptp 900 mhz network performance investigation." [Online]. Available: http://ubnt.com/downloads/SR9_Mtik_PtP.pdf

[30] A. Zubow, M. Kurth, and J.-P. Redlich, "Multi-channel opportunistic routing." ENSTA, April 2007. [Online]. Available: http://www.ew2007.org/papers/1569013580.pdf

[31] "Sartrac." [Online]. Available: https://sartrac.informatik.hu-berlin.de/

A Screenshots

This section provides screenshots of the webinterface.

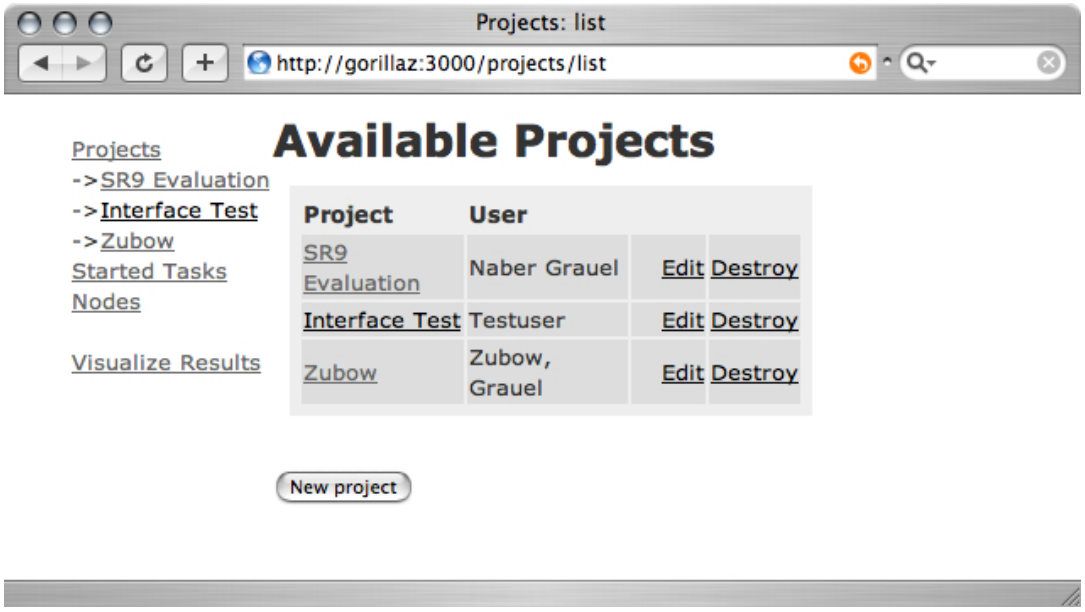

Figure A.1: Screenshot of the project overview page

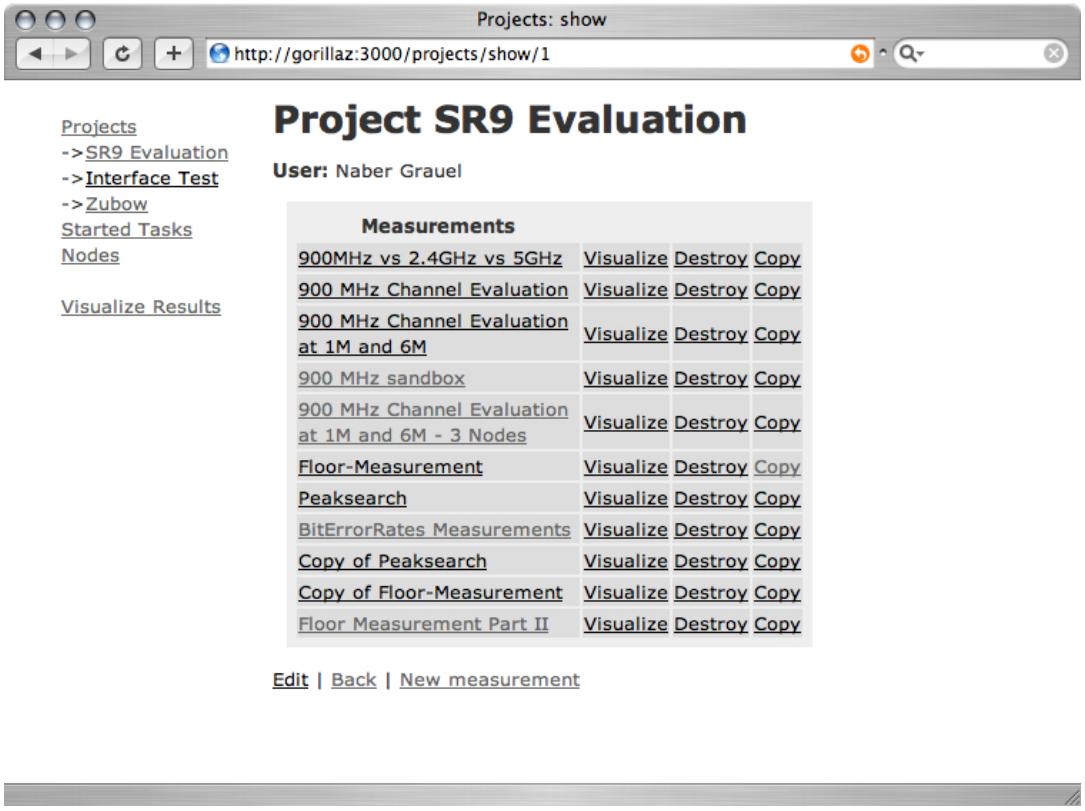

Figure A.2: Screenshot of the project details page

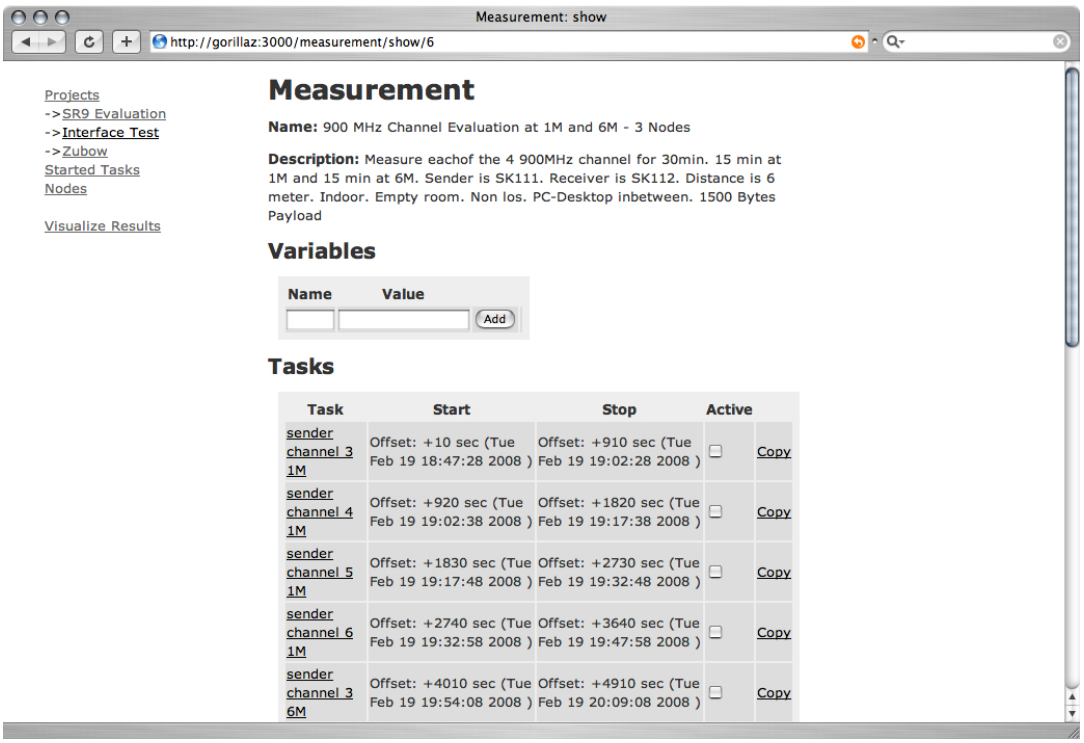

Figure A.3: Screenshot of a typical measurement page

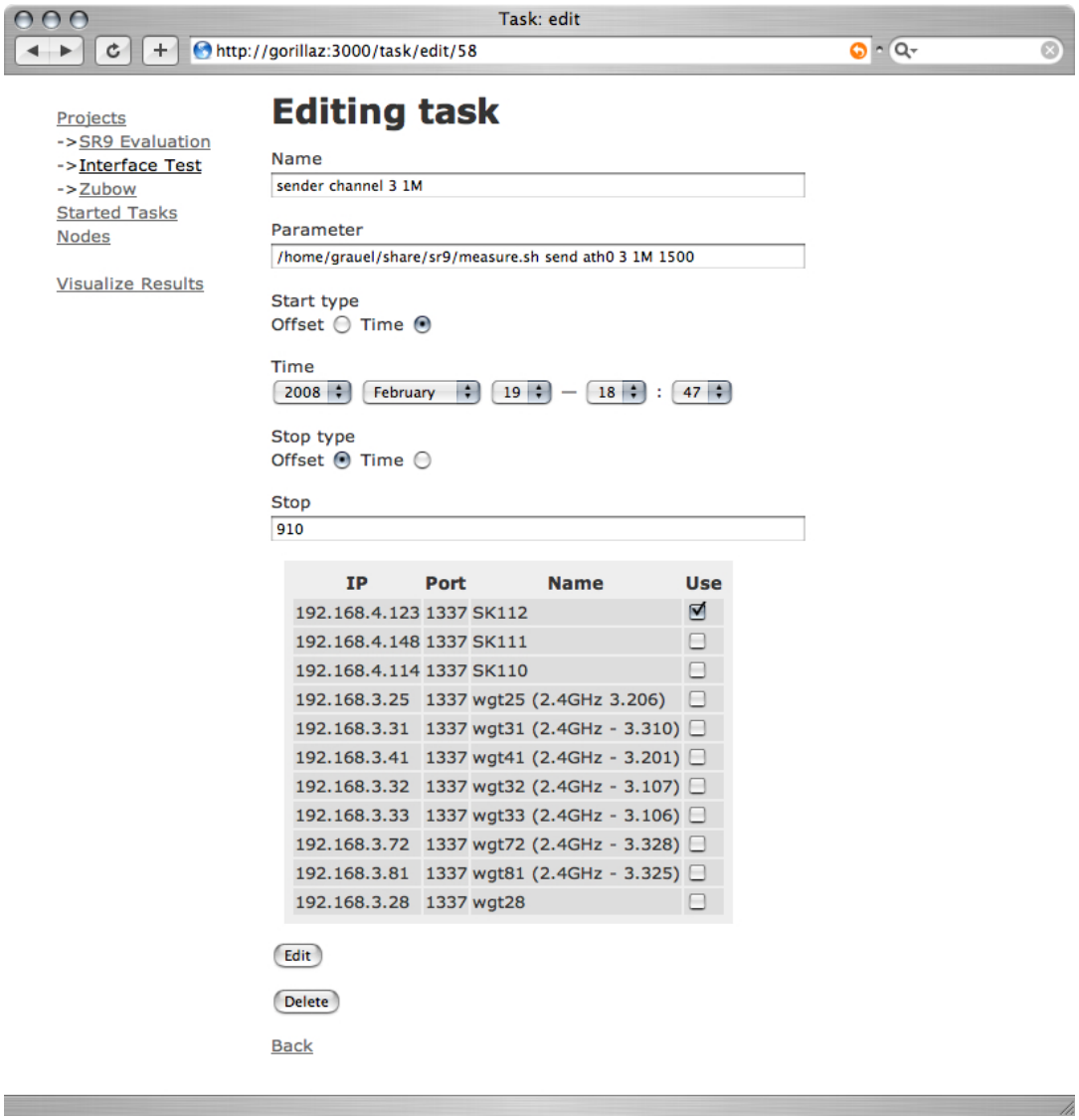

Figure A.4: Screenshot of a task editing page

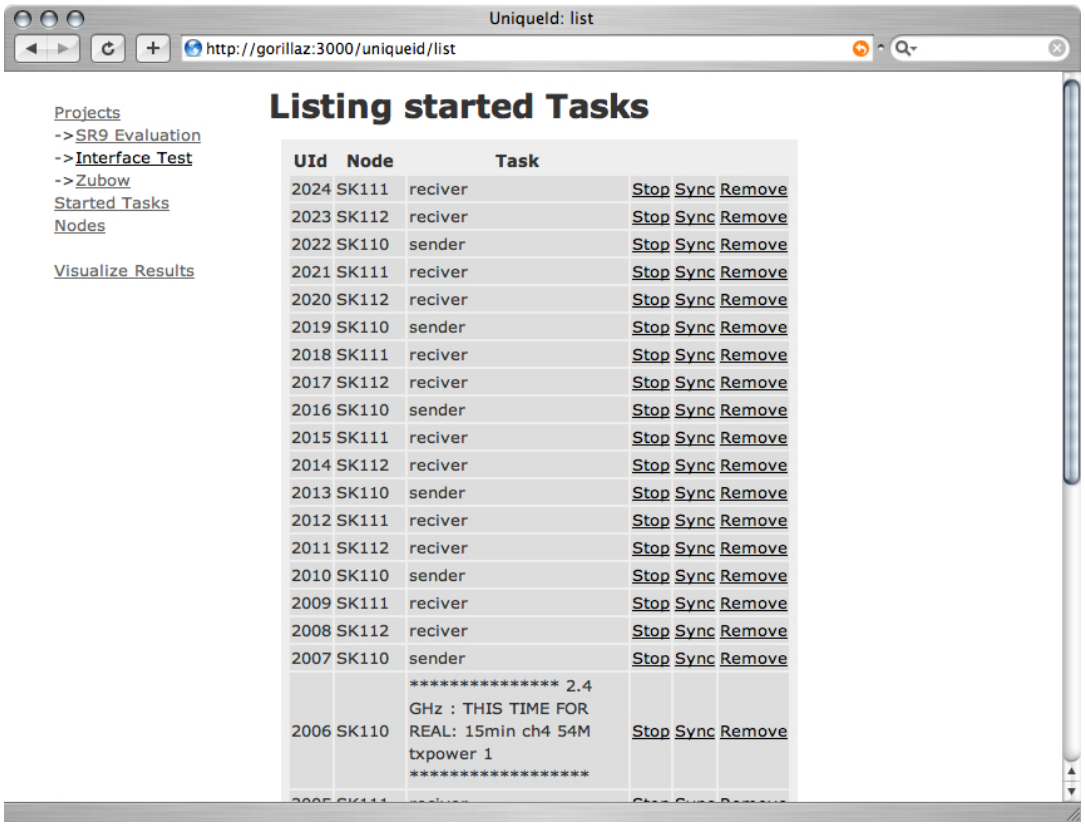

Figure A.5: Screenshot of the started tasks page

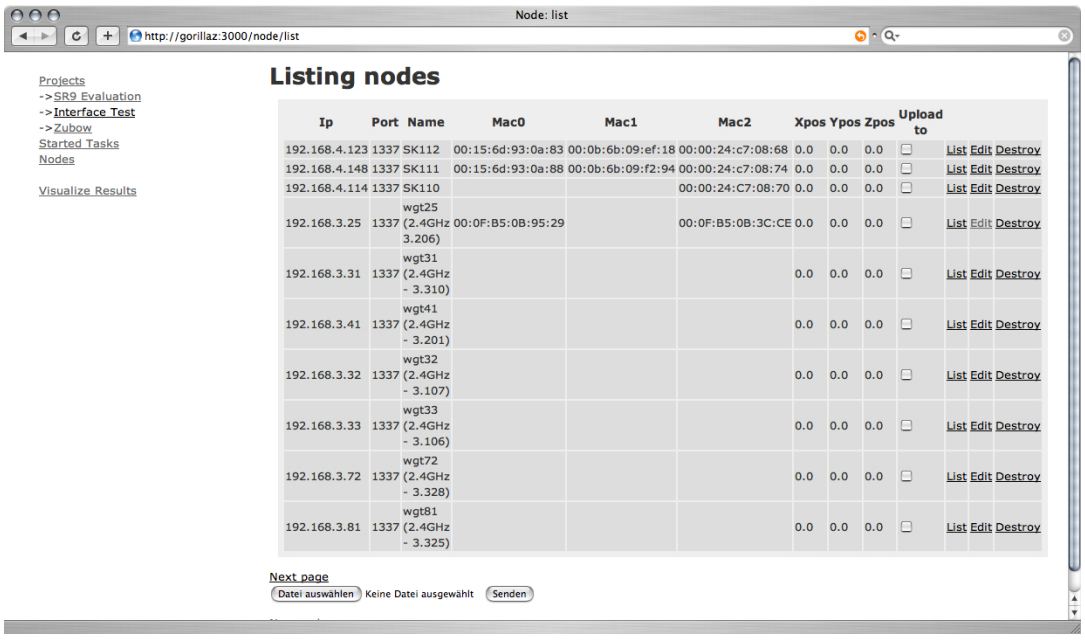

Figure A.6: Screenshot of the node overview page

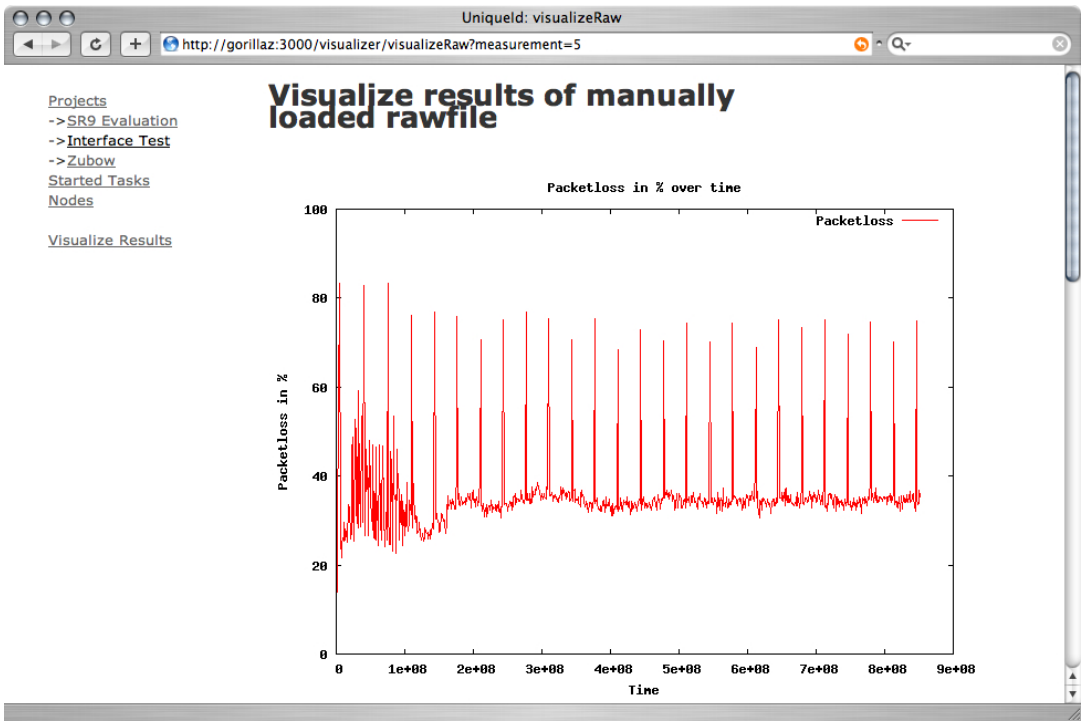

Figure A.7: Screenshot of a visualization page

B Publications

Reports published by Humboldt University Berlin, Computer Science Department, Systems Architecture Group.

1. SAR-PR-2005-01: Linux-Hardwaretreiber für die HHI CineCard-Familie. Robert Sperling. 37 Seiten.
2. SAR-PR-2005-02, NLE-PR-2005-59: State-of-the-Art in Self-Organizing Platforms and Corresponding Security Considerations. Jens-Peter Redlich, Wolf Müller. 10 pages.
3. SAR-PR-2005-03: Hacking the Netgear wgt634u. Jens-Peter Redlich, Anatolij Zubow, Wolf Müller, Mathias Jeschke, Jens Müller. 16 pages.
4. SAR-PR-2005-04: Sicherheit in selbstorganisierenden drahtlosen Netzen. Ein Überblick über typische Fragestellungen und Lösungsansätze. Torsten Dänicke. 48 Seiten.
5. SAR-PR-2005-05: Multi Channel Opportunistic Routing in Multi-Hop Wireless Networks using a Single Transceiver. Jens-Peter Redlich, Anatolij Zubow, Jens Müller. 13 pages.
6. SAR-PR-2005-06, NLE-PR-2005-81: Access Control for off-line Beamer – An Example for Secure PAN and FMC. Jens-Peter Redlich, Wolf Müller. 18 pages.
7. SAR-PR-2005-07: Software Distribution Platform for Ad-Hoc Wireless Mesh Networks. Jens-Peter Redlich, Bernhard Wiedemann. 10 pages.
8. SAR-PR-2005-08, NLE-PR-2005-106: Access Control for off-line Beamer Demo Description. Jens Peter Redlich, Wolf Müller, Henryk Plötz, Martin Stigge. 18 pages.
9. SAR-PR-2006-01: Development of a Software Distribution Platform for the Berlin Roof Net (Diplomarbeit / Masters Thesis). Bernhard Wiedemann. 73 pages.
10. SAR-PR-2006-02: Multi-Channel Link-level Measurements in 802.11 Mesh Networks. Mathias Kurth, Anatolij Zubow, Jens Peter Redlich. 15 pages.
11. SAR-PR-2006-03, NLE-PR-2006-22: Architecture Proposal for Anonymous Reputation Management for File Sharing (ARM4FS). Jens Peter Redlich, Wolf Müller, Henryk Plötz, Martin Stigge, Torsten Dänicke. 20 pages.
12. SAR-PR-2006-04: Self-Replication in J2me Midlets. Henryk Plötz, Martin Stigge, Wolf Müller, Jens-Peter Redlich. 13 pages.
13. SAR-PR-2006-05: Reversing CRC – Theory and Practice. Martin Stigge, Henryk Plötz, Wolf Müller, Jens-Peter Redlich. 24 pages.
14. SAR-PR-2006-06: Heat Waves, Urban Climate and Human Health. W. Endlicher, G. Jendritzky, J. Fischer, J.-P. Redlich. In: Kraas, F., Th. Krafft & Wang Wuyi (Eds.): Global Change, Urbanisation and Health. Beijing, Chinese Meteorological Press.
15. SAR-PR-2006-07: 无线传感器网络研究新进展 (State of the Art in Wireless Sensor Networks). 李刚 (Li Gang), 伊恩斯•彼得•瑞德里希 (Jens Peter Redlich)

16. SAR-PR-2006-08, NLE-PR-2006-58: Detailed Design: Anonymous Reputation Management for File Sharing (ARM4FS). Jens-Peter Redlich, Wolf Müller, Henryk Plötz, Martin Stigge, Christian Carstensen, Torsten Dänicke. 16 pages.
17. SAR-PR-2006-09, NLE-SR-2006-66: Mobile Social Networking Services Market Trends and Technologies. Anett Schülke, Miquel Martin, Jens-Peter Redlich, Wolf Müller. 37 pages.
18. SAR-PR-2006-10: Self-Organization in Community Mesh Networks: The Berlin RoofNet. Robert Sombrutzki, Anatolij Zubow, Mathias Kurth, Jens-Peter Redlich, 11 pages.
19. SAR-PR-2006-11: Multi-Channel Opportunistic Routing in Multi-Hop Wireless Networks. Anatolij Zubow, Mathias Kurth, Jens-Peter Redlich, 20 pages.
20. SAR-PR-2006-12, NLE-PR-2006-95: Demonstration: Anonymous Reputation Management for File Sharing (ARM4FS). Jens-Peter Redlich, Wolf Müller, Henryk Plötz, Christian Carstensen, Torsten Dänicke. 23 pages.
21. SAR-PR-2006-13, NLE-PR-2006-140: Building Blocks for Mobile Social Networks Services. Jens-Peter Redlich, Wolf Müller. 25 pages.
22. SAR-PR-2006-14: Interrupt-Behandlungskonzepte für die HHI CineCard-Familie. Robert Sperling. 83 Seiten.
23. SAR-PR-2007-01: Multi-Channel Opportunistic Routing. Anatolij Zubow, Mathias Kurth, Jens-Peter Redlich, 10 pages. IEEE European Wireless Conference, Paris, April 2007.
24. SAR-PR-2007-02: ARM4FS: Anonymous Reputation Management for File Sharing. Jens-Peter Redlich, Wolf Müller, Henryk Plötz, Christian Carstensen, 10 15 pages.
25. SAR-PR-2007-03: DistSim: Eine verteilte Umgebung zur Durchführung von parametrisierten Simulationen. Ulf Hermann. 26 Seiten.
26. SAR-SR-2007-04: Architecture for applying ARM in optimized pre-caching for Recommendation Services. Jens-Peter Redlich, Wolf Müller, Henryk Plötz, Christian Carstensen. 29 pages.
27. SAR-PR-2007-05: Auswahl von Internet-Gateways und VLANs im Berlin RoofNet. Jens Müller. 35 Seiten.
28. SAR-PR-2007-06: Softwareentwicklung für drahtlose Maschennetzwerke – Fallbeispiel: BerlinRoofNet. Mathias Jeschke, 48 Seiten.
29. SAR-SR-2007-07, NLE-SR-2007-88: Project Report: Anonymous Attestation of Unique Service Subscription (AAUSS). Jens-Peter Redlich, Wolf Müller, 19 pages.
30. SAR-PR-2007-08: An Opportunistic Cross-Layer Protocol for Multi-Channel Wireless Networks. Anatolij Zubow, Mathias Kurth, Jens-Peter Redlich, 5 pages. 18th IEEE PIMRC, Athens, Greece, 2007.

31. SAR-PR-2007-09: 100% Certified Organic: Design and Implementation of Self-Sustaining Cellular Networks. Nathanael A. Thompson, Petros Zerfos, Robert Sombrutzki, Jens-Peter Redlich, Haiyun Luo. ACM HotMobile'08. Napa Valley (CA), United States, Feb 25-26, 2008.
32. SAR-SR-2007-10, NLE-SR-2007-88: Project Report: Summary of encountered Security / Performance / Scalability problems with uPB and Wireless Thin Client Architecture. Jens-Peter Redlich, Wolf Müller, 26 pages.
33. SAR-PR-2008-01:On the Challenges for the Maximization of Radio Resources Usage in WiMAX Networks. Xavier Perez-Costa*, Paolo Favaro*, Anatolij Zubow, Daniel Camps* and Julio Arauz*, Invited paper to appear on 2nd IEEE Broadband Wireless Access Workshop colocated with IEEE CCNC 2008. *NEC Laboratories Europe, Network Research Division, Heidelberg, Germany
34. SAR-PR-2008-02: Cooperative Opportunistic Routing using Transmit Diversity in Wireless Mesh Networks. Mathias Kurth, Anatolij Zubow, Jens-Peter Redlich. 27th IEEE INFOCOM, Phoenix, AZ, USA, 2008.
35. SAR-PR-2008-03: Evaluation von Caching-Strategien beim Einsatz von DHTs in drahtlosen Multi-Hop Relay-Netzen - Am Beispiel eines verteilten Dateisystems. Felix Bechstein. Studienarbeit.
36. SAR-PR-2008-04: Precaching auf mobilen Geräten. Sebastian Ehrich, Studienarbeit, 23 Seiten.