A Testbed for Radio Propagation Measurements in Wireless Sensor Networks

Semester Thesis

Daniel Burgener

danielbu@student.ethz.ch

Distributed Computing Group
Computer Engineering and Networks Laboratory
ETH Zurich

Supervisors:
Philipp Sommer
Prof. Dr. Roger Wattenhofer

September 8, 2011

Abstract

Wireless Sensor Networks (WSNs) consist of several sensor nodes which communicate with each other over a wireless channel. There are many real-world deployments of WSNs, for example they are used for environmental monitoring in the mountains as well as in cities to monitor the air pollution. There are different models which try to describe the wireless channel between nodes within WSNs, but they often fail because they do not consider all possible effects which may occur in reality.

In this semester thesis, we build a testbed to characterize wireless channels in real WSNs. The testbed allows for partially automated characterization with respect to different parameters.

Based on the developed testbed, measurements are conducted in two different environments (indoor and anechoic chamber). Initial tests show that the measurement results in an anechoic chamber are similar to the models whereas in a real environment the measured values differ significantly. Furthermore, the characteristics of the wireless channels do not only depend on the environment but also on the nodes.

Acknowledgments

First of all I would like to thank Prof. Dr. Roger Wattenhofer for giving me the opportunity to write this semester thesis in his research group.

I would also like to thank my advisor Philipp Sommer for his support during this semester thesis. He helped me with the different programming and database languages (TinyOS, Java, Python and SQLite) and gave me valuable tips. I am convinced that the gained knowledge will be very useful for my master thesis and my career as engineer.

Contents

Chapter 1

Introduction

1.1 Motivation and Aims of this Semester Thesis

In an ideal environment we neither have any obstacles causing reflections, shadowing or scattering nor have we active devices causing interferences. This allows to predict the behavior of wireless channels by means of simple models. But in a real environment the models differ very much from the reality. To get an understanding of the influencing factors of a wireless channel in a real environment, we aim to build a testbed which allows to reliably measure different channel characteristics such as Receive Signal Strength Indicator (RSSI), Link Quality Indication (LQI) and Packet Reception Rate (PRR). Furthermore, the testbed should allow to measure, partially automated, the influence of different parameters (output power level, radio channel, payload length etc.).

1.2 Structure of the Semester Thesis

This semester thesis is organized as follows: Related work and two simple propagation model are described in Chapter 2. In Chapter 3 the layout of the testbed is described. This includes the description of the individual hardware and software components used in the testbed. First measurement results of tests run on the testbed are shown in Chapter 4. Finally, Chapter 5 concludes the semester thesis and gives an outlook.

Chapter 2

Related Work and Radio Propagation Models

2.1 Related Work

The sensor node Tmote Sky [1] is widely used for investigating experimentally the performance of WSNs (among others [2], [3]). The height of the nodes was found to have a major impact on link performance [2]. Furthermore it appears that the channels are completely symmetrical in an outdoor setup because RSSI and LQI were measured on both the forward and reverse channel, with no significant difference. The Tmote Sky was also used to evaluate the impact factors on RSSI accuracy for localization [3]. They investigated the impact of a series of parameters such as the operating frequency, the transmitter-receiver distance, the variation of transceivers and the antenna orientation. Their results showed that using only RSSI data to measure the distance is not possible due to the unreliability of the obtained data. Experiments with another hardware platform, but using the same radio chip as the one on the Tmote Sky node (CC2420 [4]), were conducted in an indoor sensor network testbed to characterize the signal strength properties [5]. Their results demonstrate that the relative antenna orientation between receiver-transmitter pairs is a major factor in signal strength variability, even in the absence of multipath effects. Based on their results, they conclude that direct distance prediction from raw RSSI data in 3-D indoor environments is impossible.

2.2 Radio Propagation Models

The system analyzed in this semester thesis consists of wireless communication links between transmitter and receiver. These links are influenced by several factors such as path loss, interference and absorption. The link budget is the sum of all gains and losses in the communication link between transmitter and receiver [6]. If the link budget is computed in dB, all factors can be added or subtracted. In the following two different pass loss models are described.

2.2.1 Free-Space Model

The simplest model is the free-space model which comprises only the direct-path signal without any multipath components [7]. Furthermore, it is assumed that there are no obstacles between transmitter and receiver. The link budget of the free-space model is:

$$P_r = P_t \, L_p \, G_t \, G_r \tag{2.1}$$

$$P_{r\,dB} = P_{t\,dB} + L_{p\,dB} + G_{t\,dB} + G_{r\,dB} \tag{2.2}$$

where:

P_r = received power
P_t = transmitter output power
L_p = path loss
G_t = total transmitter gain
G_r = total receiver gain

The path loss L_p in the free-space model is the factor $(\lambda/4\pi d)^2$ where d is the distance between the transmitter and the receiver and λ the wavelength of the signal. The communication range corresponds to a circle around the transmitter. If the pass loss is calculated in dB the path loss is:

$$L_{p\,dB}(d) = -20log(d) - 20log(4\pi) + 20log(\lambda) \propto -20log(d) \tag{2.3}$$

2.2.2 Two-Ray Model

The two-ray model takes into account that there is not only the direct path which the signal takes, but also a path where the signal is reflected, the ground reflection path [7]. The path loss of the two-ray model is $L_p(d) = (h_t h_r/d^2)^2$ where h_t and h_r are the hight of the transmitter and the receiver, respectively. In dB the pass loss is:

$$L_{pdB}(d) = 20log(h_t) + 20log(h_r) - 40log(d) \propto -40log(d) \tag{2.4}$$

Chapter 3

Testbed

3.1 Overview

The aim of the testbed is to characterize IEEE 802.15.4 radio channels [8]. We use Tmote Sky nodes [1] to do this characterization. The testbed, illustrated in Figure 3.1, consists of several Tmote Sky nodes which are either connected directly or via USB hub with the computer. The USB link allows for a reliable connection between nodes and computer. A Java application, running on the computer, controls the nodes and stores the data sent and received from the nodes in a database (DB). Additional to the USB interface, the nodes feature a 802.15.4 radio interface. In the testbed, one node is used as transmitter of radio messages and the other nodes as receivers. The wireless channel between transmitter and receivers is characterized by means of messages which are sent by the transmitter and possibly received by the receivers.

Some relevant features of the Tmote Sky node are described in Section 3.2. The communication between the computer and the nodes as well as the communication between nodes is described in Section 3.3.1. The software running on the nodes is described in Section 3.3.2. The Java application, the DB and the Python script, used for the data evaluation, are described in Section 3.3.3.

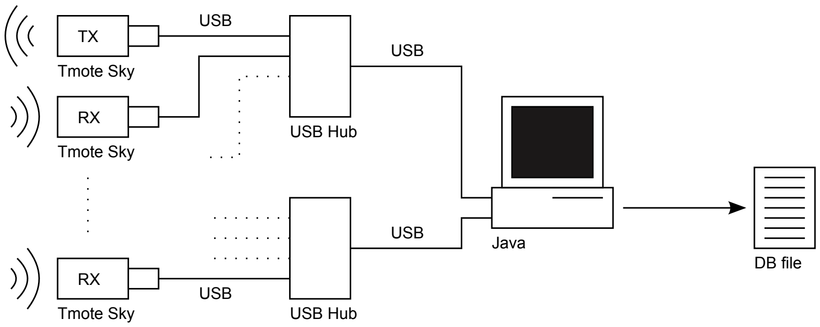


Figure 3.1: Testbed overview.

3.2 Hardware

3.2.1 Tmote Sky

The Tmote Sky node is a low power wireless module [1]. It features a wired USB interface and a wireless IEEE 802.15.4 radio interface [8]. The USB signals are converted on the node to UART signals. Therefore, in the following both terms are used for the computer-node interface. The 802.15.4 radio transceiver CC2420 from TI [4] operates in the 2.4GHz ISM band and has 16 channels within this band. Figure 3.2 illustrates the arrangement of the 16 channels. The WiFi channels are also illustrated in the Figure 3.2. Depending on the selection of the 802.15.4 channel and the 802.11 channels, interference may occur.

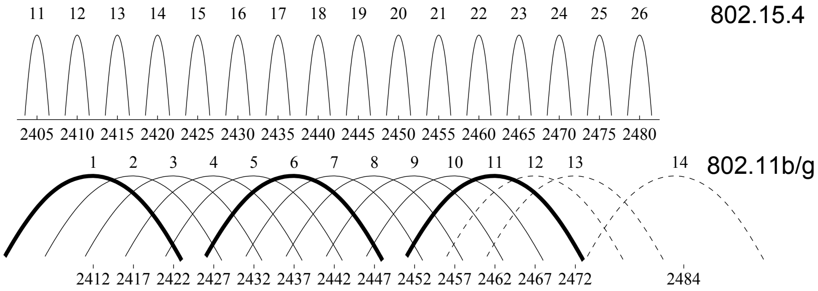


Figure 3.2: 802.11b/g and 802.15.4 frequency channels in the 2.4GHz ISM band. Each 802.11 channel is 22MHz wide, while 802.15.4 channels are 2MHz wide [9].

The RF output power of the CC2420 can be programmed by setting the Power Amplifier (PA) level. The relation between the PA level and the RF output power is shown in Table 3.1.

Table 3.1: PA level vs. output power.

PA level	31	27	23	19	15	11	7	3
Output Power [dBm]	0	-1	-3	-5	-7	-10	-15	-25

The 802.15.4 packet format, specified in [8], is shown in Figure 3.3. The preamble field is used for synchronization of the incoming message and consists of four bytes all set to `0x00`. The Synchronization Header (SHR) includes the preamble field and the Start of Frame Delimiter (SFD) (1 byte set to `0xA7`) which indicates the end of the synchronization field. The length field, included in the PHY Header (PHR), specifies the total number of bytes in the PHY Service Data Unit (PSDU) which consist of the packet's payload and the 2-byte Cyclic Redundancy Check (CRC).

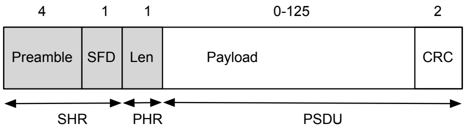


Figure 3.3: 802.15.4 packet format. Field sizes are in bytes [9].

RSSI

The RSSI is a signal strength indication and does not represent the 'quality' or 'correctness' of the signal. Therefore a high RSSI may result from an interfering device using the same frequency. The CC2420 has a built-in digital RSSI calculation [4]. Figure 3.4 shows where in the demodulation chain the RSSI is calculated. To calculate the RSSI, the received signal level is estimated. The $RSSI_{\text{VAL}}$ value is averaged over the first 8 symbols following the SFD ($128\mu s$) and available as a 8 bit, signed 2's complement value. The RSSI range is approximately from $-100dBm$ to $0dBm$ and has a typical accuracy of $\pm 6dB$ [4]. The input power P at the RF pins can be calculated based on the following equation:

$$P = RSSI_{\text{VAL}} + RSSI_{\text{OFFSET}}[dBm] \tag{3.1}$$

The $RSSI_{\text{OFFSET}}$ is found empirically and is approximately $-45dBm$. In the next sections, we refer to P and not to $RSSI_{\text{VAL}}$, when we refer to the RSSI value.

LQI

For the determination of the LQI, the actual signal strength is not relevant. But the signal quality is often influenced by the signal strength because a strong signal will be less affected by noise and therefore result in a higher LQI. The CC2420 can determine an average correlation value for each received radio message [4]. Its calculation is based on the first 8 symbols, what corresponds to 4 bytes, after the SFD. The correlation value is represented as an unsigned 7-bit value. According to the datasheet [4], the correlation value can be seen as a "chip error rate". The demodulator correlation threshold value, required before SFDs are searched, is set to the default value of 20. Figure 3.4 illustrates at which point in the demodulator chain the calculation of the correlation value takes place.

According to the 802.15.4 specification [8], the minimum and maximum LQI values (`0x00` and `0xFF`) should be associated with the 802.15.4 signals of the lowest and highest quality. There are different possibilities how to derive the LQI value based on its definition in the specification. We use the correlation value of the CC2420 as LQI indicator which is in the range of 50 to 110. Therefore, in the following sections the correlation value of the CC2420 is referred to as LQI.

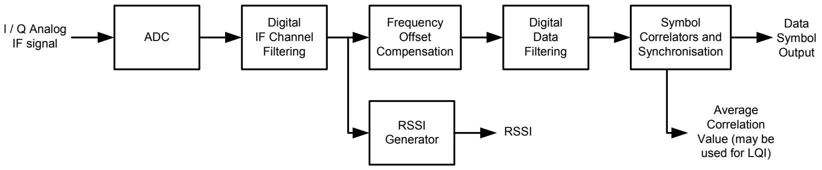

I / Q Analog IF signal
ADC
Digital IF Channel Filtering
Frequency Offset Compensation
Digital Data Filtering
Symbol Correlators and Synchronisation
Data Symbol Output
RSSI Generator
RSSI
Average Correlation Value (may be used for LQI)

Figure 3.4: Demodulator simplified block diagram [4].

3.3 Software

3.3.1 Interface Computer - Tmote Sky

This chapter describes the protocol running over the USB and the radio. First, some communication scenarios are illustrated for a general understanding of the message flow. Afterwards the individual messages are explained in more detail.

Communication Scenarios

In the following, three different communication scenarios between computer and nodes are depicted. The text below the envelope represents the type of the message. The first communication scenario is the setting of the node's radio channel (Figure 3.5). The computer initiates the setting of the radio channel. After the node has set the new radio channel, it acknowledges the message.

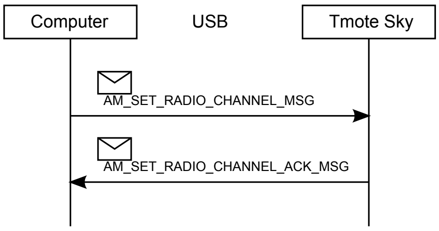


Figure 3.5: Communication scenario: Set radio channel of node.

The next scenario illustrates the transmission of a message over the 802.15.4 radio (Figure 3.6). Again, the scenario is initiated by the computer. If the node was able to transmit the incoming message, it sends an acknowledgment back to the computer. Other nodes receiving the transmitted message, extend it with some additional information (RSSI, LQI etc.), and forward the extended message via USB to the computer. The two computers illustrated in Figure 3.6 may be the same.

In case of the reception of a corrupted message, the node changes the message type and forwards the message to the computer (Figure 3.7). A message is corrupted if the CRC value is not correct.

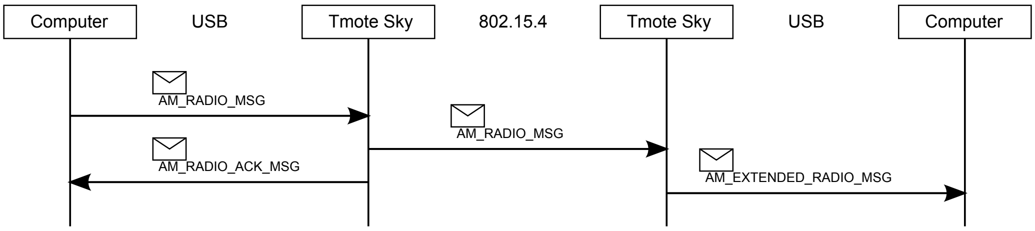


Figure 3.6: Communication scenario: Sending of 802.15.4 radio message with successful reception.

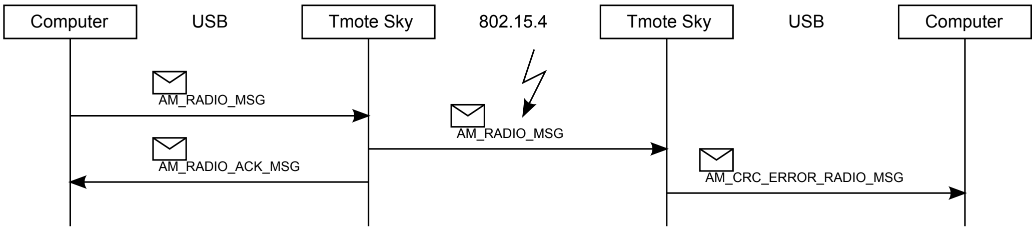


Figure 3.7: Communication scenario: Sending of 802.15.4 radio message with erroneous reception.

AM RADIO MSG

Messages of type *AM RADIO MSG* are sent from a computer via USB to a node, where they are transmitted over the radio to other nodes. This message type is used to characterize the wireless channel. The payload of a radio or UART message of type *AM RADIO MSG* consists of four fields (Table 3.2). The *Counter* contains a four byte number used as a unique message identifier. The field *TX delay* allows to force the node to delay the message before sending it. The *PA level* field defines the output power of the radio to be used to transmit the message (Table 3.1). The *random payload* contains a definable number of bytes to enlarge the message payload with random values.

Table 3.2: Payload of an *AM RADIO MSG* message.

Nb of Bytes	4	1	1	n
Description	Counter	TX delay [ms]	PA level	random payload

AM RADIO ACK MSG

The message type *AM RADIO ACK MSG* is used by the node to acknowledge the successful transmission of an *AM RADIO MSG* message. The acknowledgment is sent via USB to the computer. A message of type *AM RADIO ACK MSG* only confirms the successful transmission of a radio message, but does not contain any information about the successful reception of the message by other nodes. The payload of a message of type *AM RADIO ACK MSG* is illustrated in Table 3.3. It contains exactly the same payload as a message of type *AM RADIO MSG*. We refer to the *AM RADIO MSG* message description for further information about the individual fields.

Table 3.3: Payload of an *AM RADIO ACK MSG* message.

Nb of Bytes	4	1	1	n
Description	Counter	TX delay [ms]	PA level	random payload

AM EXTENDED RADIO MSG

The message type *AM EXTENDED RADIO MSG* is used by the node to forward *AM RADIO MSG* messages, received on the radio, via USB to the computer. Only incoming messages with a valid CRC are forwarded as an *AM EXTENDED RADIO MSG* message. For messages with an invalid CRC we refer to the *AM CRC ERROR RADIO MSG* message. The incoming message is extended with RSSI, LQI, CRC flag and the length of the random payload. The random payload itself is not part of the message. The RSSI value can be calculated by means of Equation (3.1).

Table 3.4: Payload of an *AM EXTENDED RADIO MSG* message.

Nb of Bytes	4	1	1	1	1	1	1
Description	Counter	TX delay [ms]	PA level	$\mathrm{RSSI}_{\mathrm{VAL}}$	LQI	CRC	random payload length

AM CRC ERROR RADIO MSG

The message type *AM CRC ERROR RADIO MSG* is used by the node to forward radio messages which were received with an invalid CRC. Those corrupted messages are forwarded via USB to the computer. Because any byte of the incoming message may be corrupted, it is not possible to determine the message type. Therefore all fields of the payload in Table 3.5 except $\mathrm{RSSI}_{\mathrm{VAL}}$, LQI and CRC may be corrupted. Because in this testbed only messages of type *AM RADIO MSG* are transmitted over the radio, it is assumed that the corrupted message is of type *AM RADIO MSG*.

AM SET RADIO CHANNEL MSG

Messages of type *AM SET RADIO CHANNEL MSG* are sent from a computer via USB to a node. The message payload is depicted in Table 3.6. This type

Table 3.5: Payload of an *AM CRC ERROR RADIO MSG* message.

Nb of Bytes	1	4	1	1	1	1	1	1
Description	AM type	Counter	TX delay $[ms]$	PA level	RSSI_{VAL}	LQI	CRC	random payload length

of message is used to set the radio channel of a node. For the range of radio channels we refer to Section 3.2.1.

Table 3.6: Payload of an *AM SET RADIO CHANNEL MSG* message.

Nb of Bytes	1
Description	radio channel

AM SET RADIO CHANNEL ACK MSG

Messages of type *AM SET RADIO CHANNEL ACK MSG* are sent from a node via USB to a computer. This type of message is used to acknowledge a previously received message of type *AM SET RADIO CHANNEL MSG*. The *acknowledge* field in Table 3.7 is '1' if the setting of the channel was successful, else it is '0'.

Table 3.7: Payload of an *AM SET RADIO CHANNEL ACK MSG* message.

Nb of Bytes	1	1
Description	acknowledge	radio channel

3.3.2 Tmote Sky

The operating system TinyOS version 2.1.1 is running on the nodes. The node software is based on the TinyOS example project *BaseStation*. By default, TinyOS rejects messages received with a CRC error. Therefore two TinyOS files had to be adapted such that those messages are accepted as well (CC2420ReceiverP.nc and CC2420TinyosNetworkP.nc). The same software is running on all nodes, independent whether they are transmitter or receiver. The behavior of the node is illustrated in a simplified manner in Figure 3.8. For detailed information we refer to the source code.

An event is thrown for all incoming messages. Within the event the received message is stored in a buffer or queue after a short verification of the message type. Afterwards, the message is further processed by a corresponding task. Messages received on the radio are stored in the UartOutputQueue and, later on, processed by the UartSendTask. Depending on the CRC flag, the UartSendTask converts the message type and extends the message as described in Table 3.3 and Table 3.4. The UartOutputQueue does not only contain messages received on the radio, but also messages generated after the successful transmission over the radio (*AM RADIO ACK MSG*). This type of message does not undergo any modification by the UartSendTask because it has already been modified by the RadioSendDone event.

Messages received on the UART are either stored in the RadioOutputQueue or in the SetRadioChannel buffer. Incoming messages of type *AM RADIO MSG* are stored in the RadioOutputQueue. Later on, the RadioSendTask transmits the message without any modification over the radio. The applied output power depends on the *PA level* field in the message payload (Table 3.2). If the incoming message is of type *AM SET RADIO CHANNEL MSG*, it is processed by the SetRadioChannelTask. The task sets the radio channel according to the *radio channel* field in the message payload (Table 3.6). Afterwards an acknowledge message is sent over the UART.

3.3.3 Computer

Java

The Java application is running on a computer. It uses the WSN Control Library[1] which facilitates connecting and interfacing Tmote Sky nodes. The application can send and receive messages from Tmote Sky nodes connected via USB. All received and sent messages are stored in the DB according to their message type. Before a new test can be run, the unique IDs of the nodes have to be linked with the USB port names. Therefore an *AM SET RADIO CHANNEL MSG* message is sent to all nodes connected to the computer. Based on their reply message *AM SET RADIO CHANNEL ACK MSG*, which contains the node ID in the message header, the individual nodes are linked with the USB port names. After the successful linking, the actual test can be started. In the following, the power sweep test used for the measurements in Chapter 4 is explained.

PowerSweepTest The power sweep test transmits 1000 *AM RADIO MSG* messages for each PA level in Table 3.1. The node with the ID 0 is the transmitter and all the other nodes are receivers. Therefore the destination address of the *AM RADIO MSG* message is the broadcast address `0xFFFF` instead of a specific node ID. During the test, the position of the individual nodes is fix. The delay between two consecutive messages with the same PA level is $500ms$. If the PA level is changed after 1000 transmitted messages, an additional delay of $1s$ is introduced. During the whole test, the radio channel is channel 26. The *Counter*

[1] The WSN Control Library has been written by Philipp Sommer

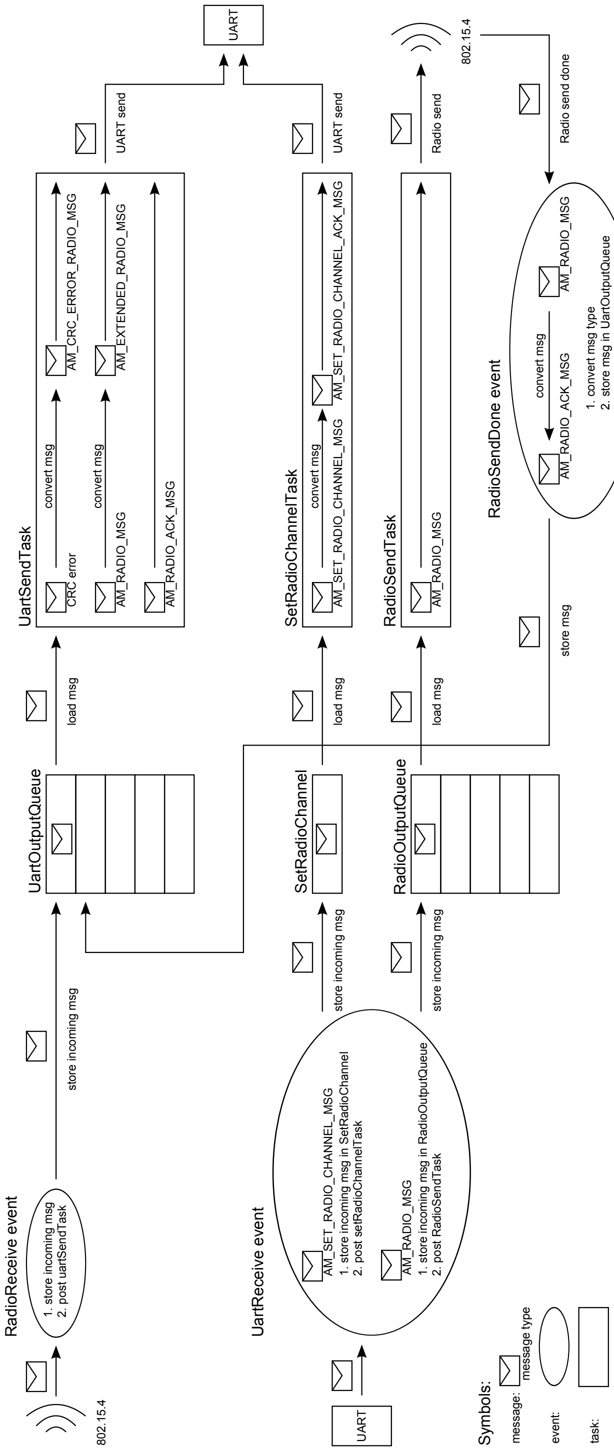


Figure 3.8: Tmote Sky software overview.

value is incremented after each transmitted message, starting with 0. The length of the *random payload* is 10 bytes and the *TX delay* is zero (Table 3.8).

Table 3.8: Power sweep test: Payload *AM RADIO MSG* message.

Nb of Bytes	4	1	1	10
Description	Counter	TX delay	PA level	random payload
Value	incrementing counter	0	changed after 1000 msg, see Table 3.1	random payload of 10 bytes

Database

SQLite is used as the database (DB) for storing messages sent and received by the Java application. The DB contains nine tables. The content of the DB tables is illustrated in Appendix B. The tables *startTimeTable* and *testEnvironmentTable* contain information about the measurement setup. Furthermore, each message type listed in Section 3.3.1 has its specific DB table. In those DB tables, each payload field of such a message is represented as a column. Additionally, each DB table contains the following four columns:

- time
- gateway
- source
- destination

The *time* represents the difference, in milliseconds, between the time of sending/receiving a message and the start time of the measurement stored in the DB table *startTimeTable*. The *gateway* corresponds to the ID of the node which has sent or received the message. The *source* column contains the ID of the transmitter of the message. The *destination* column can either be a specific node or the broadcast address (`0xFFFF`). In case of messages not intended for radio transmission (e.g. *AM SET RADIO CHANNEL MSG* messages), *source* and *destination* are the same as the *gateway*. In the following, the individual tables of the DB are described.

startTimeTable The *startTimeTable* contains only a single entry, the absolute start time of the measurement (Table B.1).

testEnvironmentTable The *testEnvironmentTable* represents the test environment (Table B.2). Each node used during the test has an entry in this table. The entry contains information about the USB port the node is connected to and the node's position. The position of the node has to be specified in the Java code prior to the test.

radioMsgTable Entries in the *radioMsgTable*, illustrated in Table B.3, represent messages of the type *AM RADIO MSG* which were sent by the computer to a node. An *AM RADIO MSG* message is sent via USB to the node where it is forwarded over the radio with the defined output power.

radioAckMsgTable Successful radio transmissions of *AM RADIO MSG* messages are acknowledged by the node. Those acknowledgments with message type *AM RADIO ACK MSG* are stored in the DB table *radioAckMsgTable* (Table B.4). The columns in the table are the same as in the *radioMsgTable.*

extendedRadioMsgTable *AM RADIO MSG* messages which were received by the nodes with a valid CRC are stored in this table (Table B.5). The original message is extended by the node with some information about the message (RSSI, LQI etc.). Those additional information are also stored in the *extendedRadioMsgTable.*

crcErrorRadioMsgTable *AM RADIO MSG* messages which were received by the nodes with an invalid CRC are stored in this table (Table B.6). The original message, which is completely or partially corrupted, is extended with some information about the message (RSSI, LQI etc.). Those additional information are also stored in the *extendedRadioMsgTable.*

setRadioChannelMsgTable The table *setRadioChannelMsgTable* contains messages of type *AM SET RADIO CHANNEL MSG* which were sent by the computer to a node to modify the node's radio channel. The table is illustrated in Table B.7.

setRadioChannelAckMsgTable The *setRadioChannelAckMsgTable* table contains messages of type *AM SET RADIO CHANNEL ACK MSG* which were sent by the nodes after they received an *AM SET RADIO CHANNEL MSG* message. The *ack* field in Table B.8 is '1' if the setting of the radio channel was successful, else the field is '0'.

unknownMsgTable Radio messages with a valid CRC but an unknown message type are stored in the *unknownMsgTable* DB table (illustrated in Table B.9). The payload of an unknown message cannot be interpreted. Therefore the complete messages is stored in the DB table.

Python

Python is used for the evaluation of the measurement data stored in the DBs. A single DB file contains the measurement results of a fix node arrangement. For the evaluation, several DB files are required such that each receiver node has been placed exactly once at each position. The transmitter node with the ID 0 has to be always at the same position (Figure 3.9).

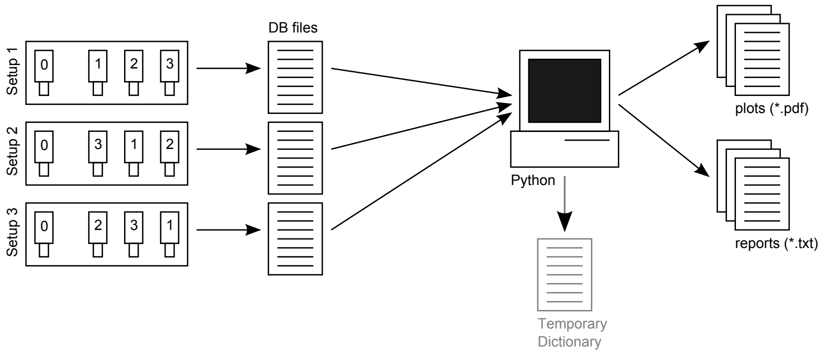


Figure 3.9: Overview Python.

The Python script consecutively scans the DB files of each setup. For each combination of radio channel, node ID, distance between transmitter and receiver and the output power of the transmitter, an object is generated and stored in a temporary dictionary[2] with the key
'ch*radioChannel*_id*nodeId*_*distance*cm_*outputPower*dBm'.
For example 'ch26_id2_100cm_-3dBm' denotes the key for the statistical information about the received packets of node 2 placed at distance $100cm$ where the transmitter has send packets on the radio channel 26 with $-3dBm$ output power. Each object in the temporary dictionary contains the statistical informa-

[2] A Python dictionary is an unordered set of *key: value* pairs, with the requirement that the keys are unique.

tion listed in Table 3.9.

Table 3.9: Statistics of a specific node, at a specific position, receiving messages which were sent with a specific output power on a specific radio channel.

Variable	Description
lRssiValidPkts	List with the RSSI values of all valid packets[1] received [dBm]
lRssiCorruptedPkts	List with the RSSI values of all corrupted packets[2] received [dBm]
lLqiValidPkts	List with the LQI values of all valid packets[1] received
lLqiCorruptedPkts	List with the LQI values of all corrupted packets[2] received
avgRssiValidPkts	Average RSSI value of all valid packets[1] received [dBm]
avgRssiCorruptedPkts	Average RSSI value of all corrupted packets[2] received [dBm]
avgLqiValidPkts	Average LQI value of all valid packets[1] received
avgLqiCorruptedPkts	Average LQI value of all corrupted packets[2] received
nbOfValidPkts	Number of valid packets[1] received
nbOfCorruptedPkts	Number of corrupted packets[2] received
nbOfLostPkts	Number of lost packets
nbOfSentPkts	Number of sent packets

[1] valid packets = packets received with a valid CRC
[2] corrupted packets = packets received with an invalid CRC

The number of packets which were sent with the specific output power is determined based on the entries in the *radioAckMsgTable* table. This table only contains packets which were sent by the transmitter node. To determine the valid packets (RSSI, LQI and quantity), the two tables *extendedRadioMsgTable* and *radioAckMsgTable* are joined based on the unique *counter* columns. Furthermore, only those entries are selected which:

- were sent by the transmitter node (*source* in *radioAckMsgTable* = 0)
- were sent with the specific output power (*txPower* in *extendedRadioMsgTable*)
- were received by the specific node (*gateway* in *extendedRadioMsgTable*)

To determine the corrupted packets (RSSI, LQI and quantity), the two tables *crcErrorRadioMsgTable* and *radioAckMsgTable* cannot be joined based on the *counter* columns because the *counter* column in the *crcErrorRadioMsgTable* table may be corrupted. Therefore two tables are joined based on the *time* column. The *time* column of the *crcErrorRadioMsgTable* table has to be between *time* - 100*ms* and *time* + 500*ms* of the *radioAckMsgTable* table. Only those entries are selected which:

- were sent by the transmitter node (*source* in *radioAckMsgTable* = 0)
- were sent with the specific output power (*txPower* in *radioAckMsgTable*)
- were received by the specific node (*gateway* in *crcErrorRadioMsgTable*)

The number of lost packets can be calculated based on the number of received packets (valid and corrupted) and the number of sent packets. The temporary dictionary is updated after all DB files have been analyzed. Based on the temporary dictionary, the following plots are generated:

- RSSI vs. output power (one plot for each position and radio channel combination)
- RSSI vs. distance (one plot for each node and radio channel combination)
- LQI vs. output power (one plot for each position and radio channel combination)
- LQI vs. distance (one plot for each node and radio channel combination)
- Packet statistics (valid, corrupted, lost) vs. distance (one plot for each node, radio channel and output power combination)
- Packet statistics (valid, corrupted, lost) vs. output power (one plot for each node, radio channel and distance combination)
- Valid packets vs. distance (one plot for each node and radio channel combination)
- Valid packets vs. output power (one plot for each distance and radio channel combination)
- LQI vs. RSSI (one plot for each node)
- Cumulative Distribution Function (CDF) of RSSI and LQI (one plot for each node)

Additionally, for each plot a report text file, containing all data used to generate the plots, is generated.

Chapter 4

Measurements on the Testbed

4.1 Overview

Different tests were conducted with the testbed in two environments. The first test took place in a normal room of a one family-house where six receiver nodes were placed between $0.5m$ and $4.0m$ from the transmitter node. The second test took place in an anechoic chamber which is designed to reduce the reflections of electromagnetic waves. Furthermore, the anechoic chamber is also insulated from exterior sources of noise. In the anechoic chamber, tests were conducted once with four receivers and once with only one receiver. In all tests the transmitter and receiver nodes were placed horizontally. Table 4.1 shows the nodes used for the individual tests. All nodes are of the same hardware version, only node 4 is from a different series. The node with the ID 0 was always used as transmitter. During all tests, transmitter and receivers were using the radio channel number 26.

Table 4.1: Hardware version of Tmote Sky nodes.

Node ID	Hardware version
0	4M 94V-0 H014-4787 0650
1	4M 94V-0 H014-4787 0650
2	4M 94V-0 H014-4787 0650
4	2M 94V-0 H014-4787 0549
5	4M 94V-0 H014-4787 0650
6	4M 94V-0 H014-4787 0650
7	4M 94V-0 H014-4787 0650

4.2 Indoor Test

4.2.1 Test Setup

The indoor test was conducted in a room of $6.0m \times 4.0m \times 2.5m$ ($L \times W \times H$). Seven nodes were used for the indoor test. All nodes were placed in a row and at the same height of $80cm$ from the ground. The node with the ID 0 was the transmitter and placed at $0cm$. The nodes with the IDs 1, 2, 4, 5, 6 and 7 were receivers and placed at the positions $50cm$, $100cm$, $150cm$, $200cm$, $250cm$ and $400cm$ (Figure 4.1). The test consisted of six rounds. In each round the node arrangement was changed so that each node, except node 0, was once at each position. The power sweep test (Section 3.3.3) was run in each round, generating one DB file per round.

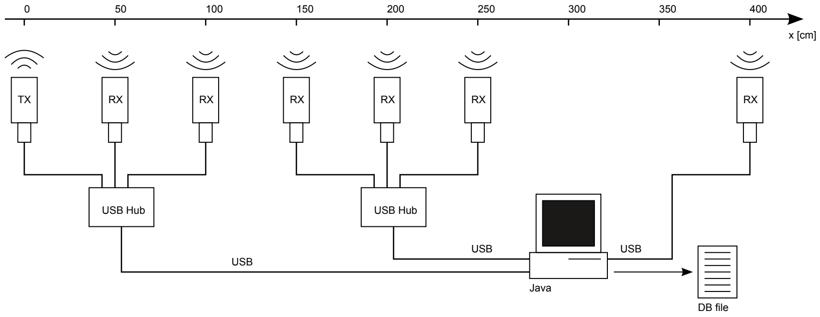


Figure 4.1: Setup indoor test.

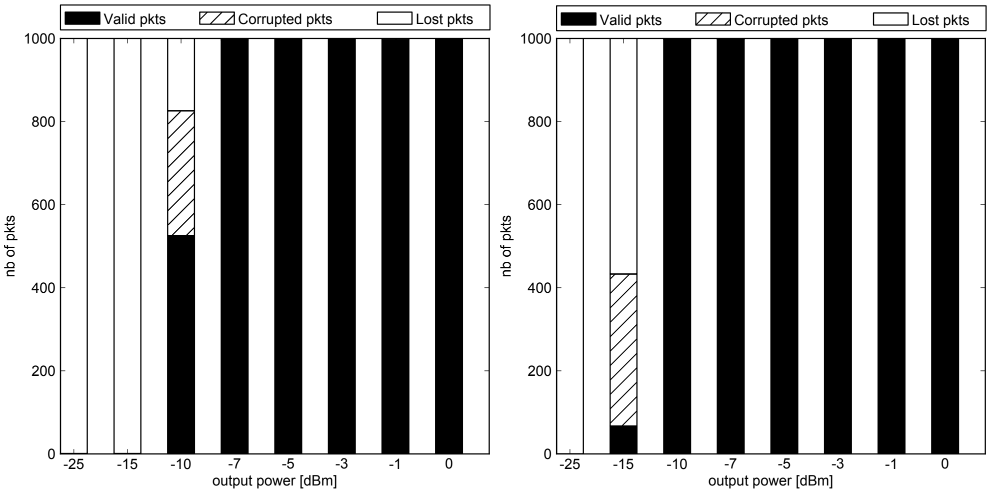
4.2.2 Results

The measurement results stored in six DB files, one for each round, are visualized by means of Python. In the following some selected plots are shown and described. Some more plots are listed in Appendix C.1.

Packet Statistics vs. Output Power

Figure 4.2 shows the packet statistics of the receiver nodes 2 and 5 at a distance of $150cm$ from the transmitter. Comparing the statistics where the output power of the transmitter is $-10dBm$, node 5 has a PRR of 99.8% and node 2 has a PRR of only 52.4%. An even more significant difference exists between node 6 and 7 at a distance of $250cm$ and an output power of $-15dBm$ illustrated in Figure 4.3. While node 7 receives almost all packets (PRR = 99.2%), node 6 does not receive any valid packets.

As expected the PRR of a node drops off at lower transmission powers. But although all nodes are of the same hardware version, the nodes do not always



(a)

(b)

Figure 4.2: Packet statistics vs. output power at 150*cm* of a) node 2 and b) node 5.

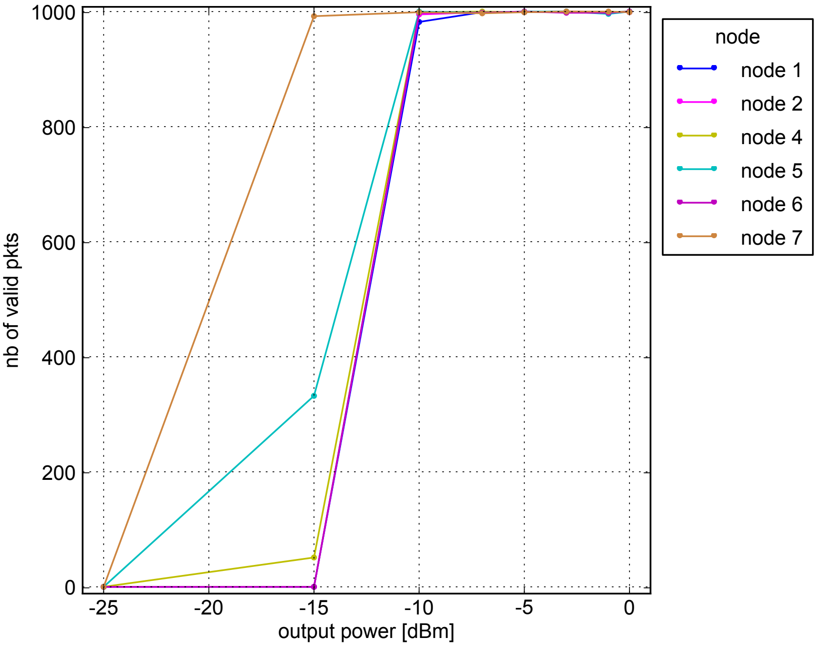
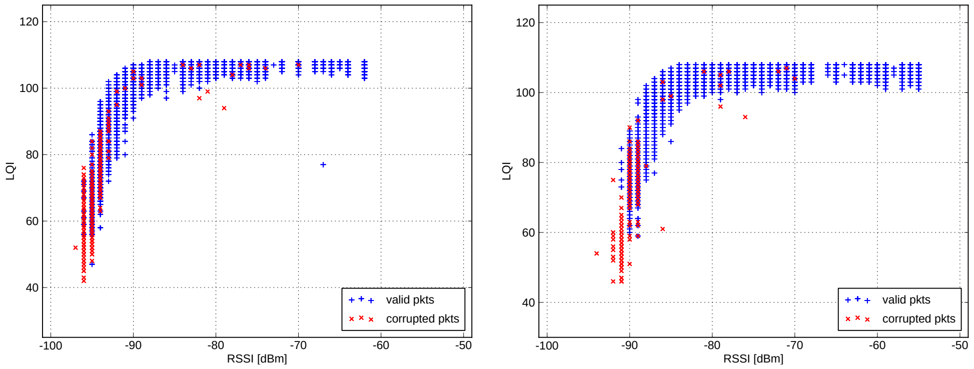


Figure 4.3: Valid packet vs. output power at 250*cm*.

perform the same. For some constellations one node may receive all packets without any CRC errors, while another node does not receive any packets, even not corrupted ones. The reason for this may be bad soldering of the radio chip or the variation among the same type of transceivers which was also observed in [3] and [5].

RSSI vs. LQI

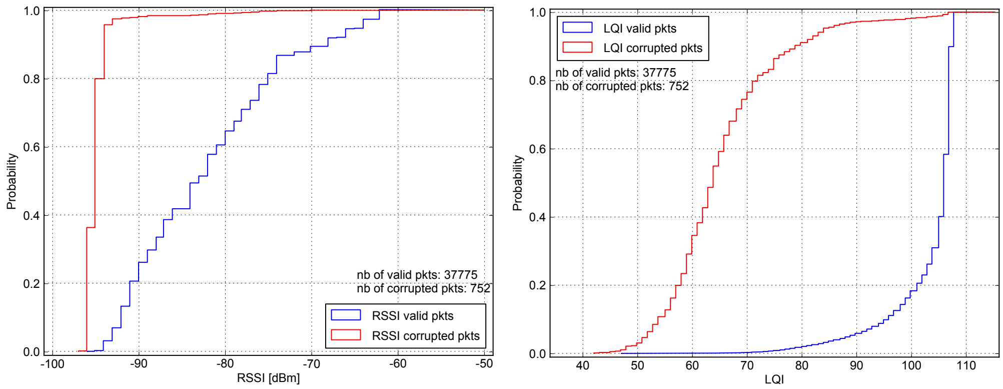
Figure 4.4 shows the relation between LQI and RSSI of two nodes. The plots contain the node's measurement results of all distances and output power levels. The LQI value tends to increase rapidly for low RSSI values before it saturates for higher RSSI values. But the behavior of the LQI vs. RSSI is not the same for all nodes. While node 4 already receives packets with a RSSI value of around $-95dBm$, node 6 starts receiving packets at a about $-90dBm$. The majority of the corrupted packets received by node 4 are located at low RSSI values while the valid packets are uniformly distributed over the whole RSSI range (Figure 4.5(a)). E.g. only 2% of the corrupted packets have an RSSI above $-85dBm$. Valid packets are located mainly at high LQI values (Figure 4.5(b)).



(a) (b)

Figure 4.4: LQI vs. RSSI of valid and corrupted packets of a) node 4 and b) node 6.

The variation between transceivers observed at the PRR is also visible in the LQI vs. RSSI plots which are not the same for the different nodes. Furthermore, these plots show that neither a high RSSI nor a high LQI guarantees the reception of a valid packet. The reason for an invalid CRC in case of a high RSSI may be a temporary noisy environment. In case of a high LQI we expect a successful reception of a packet. One cause for corrupted packets with high LQI may be the fact that the LQI is calculated based only on the 8 first symbols following the SFD. If, for example, the 9th symbol after the SFD is corrupted, the LQI may be high although the received data is corrupted. The range of LQI values



(a) (b)

Figure 4.5: Cumulative Distribution Function (CDF) of node 4: a) RSSI and b) LQI.

for valid packets is between 50 and 110 and corresponds to the range indicated in the datasheet [4]. Looking at the plots of Figure 4.4 one may conclude that there is a strong relation between RSSI and LQI, because there are only a few points with low LQI and high RSSI. In a more noisy environment, e.g. a nearby WiFi access point using the same frequency band, these kind of points may be more frequent. An interferer can increase the RSSI value and corrupt the transmitted data causing a lower LQI.

RSSI vs. Output Power

Figure 4.6 shows the mean RSSI of all nodes at different output powers. The standard deviation for most points is below $1dBm$. Even though all nodes are of the same hardware version, their RSSI level differ up to $9dBm$ (e.g. at $250cm$ and $-10dBm$ output power, node 7 has a RSSI of $-83.1dBm$ and node 2 $-92.2dBm$). Even larger differences were measured at larger distances (Figure C.6(b)). The node which has the highest RSSI depends on the distance, e.g. at distance $50cm$, the RSSI of node 7 is lower than almost all the other nodes. But at distance $250cm$ node 7 has, over the whole output power range, the highest RSSI of all nodes. The slope of all nodes is about 1, i.e. if the output power is increased by $1dBm$ the RSSI of the individual nodes is also increased by $1dBm$.

RSSI vs. Distance

The behavior of the RSSI versus distance is illustrated in Figure 4.7. The RSSI does not decrease monotonically but has a lower value at $150cm$ than at $200cm$. This behavior is present at all nodes and over the whole output power range (Figure C.3, Figure C.4).

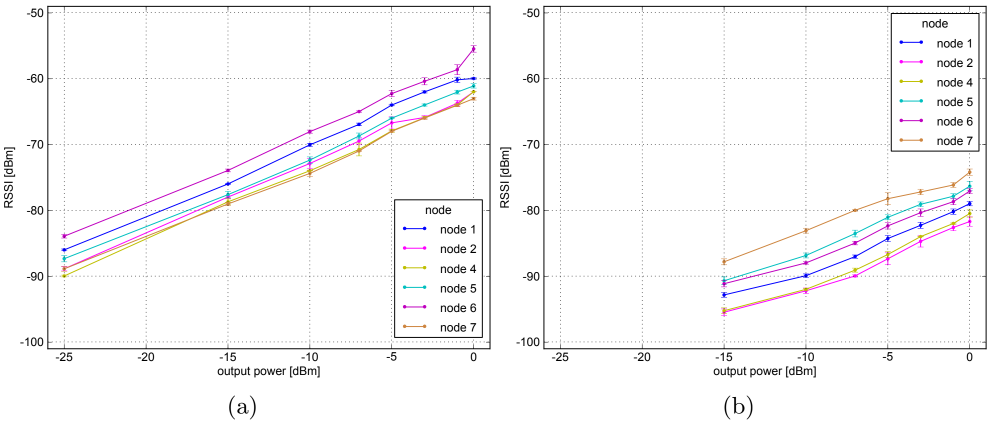


Figure 4.6: RSSI mean vs. output power of valid and corrupted packets at a) $50cm$ and b) $250cm$. The error bars indicate the standard deviation.

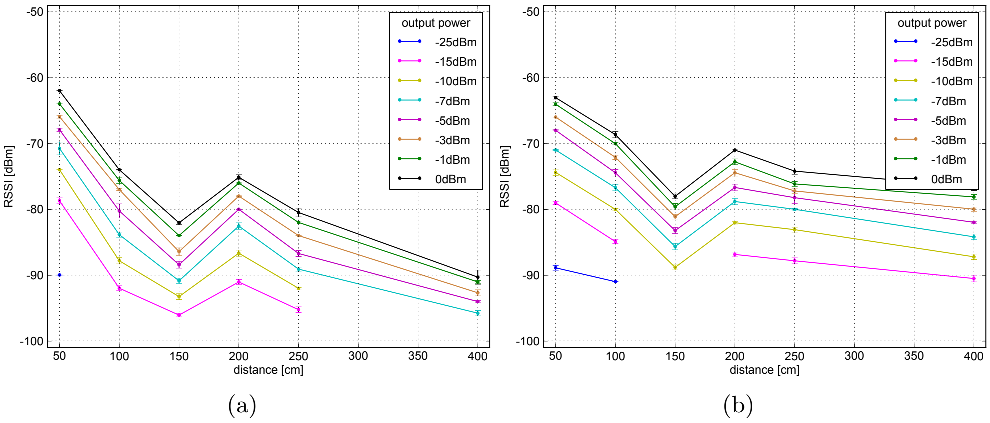


Figure 4.7: RSSI mean vs. distance of valid and corrupted packets of a) node 4 and b) node 7. The error bars indicate the standard deviation.

The form of the RSSI level with respect to the distance is almost the same for all output power levels and has also been observed in [3]. Because of this behavior, which is similar for all nodes, we assume that the non-monotonic decrease was caused by interference. If the non-monotonic decrease were caused by an active external disturbance source, e.g. a WiFi access point, this source most probably would behave always the same an thereby cause a constant RSSI level at a specific position. This is obviously not the case. Because the models presented in Section 2.2 do not consider interference, it does not make sense to apply them on the measurement results.

4.3 Anechoic Chamber Test with 4 Nodes

4.3.1 Test Setup

The anechoic test was conducted in an anechoic chamber of $7.2m \times 3.6m \times 3.4m$ ($L \times W \times H$). Five nodes were used for this test. The arrangement of the nodes in the anechoic chamber is shown in Figure 4.8. All nodes were placed at a height of $160cm$ from the ground. The node with the ID 0 was the transmitter and placed at $0cm$. The nodes with the IDs 1, 2, 4, and 6 were receivers and placed at the positions $100cm$, $200cm$, $300cm$ and $400cm$ (Figure 4.9). During the test, the computer generating and recording packets was placed outside the anechoic chamber. The test consisted of four rounds. In each round the node arrangement was changed so that each node, except node 0, was once at each position. The power sweep test (Section 3.3.3) was run in each round, generating one DB file per round.

4.3.2 Results

In the following some selected plots are shown and described. Some more plots are listed in Appendix C.2.

RSSI vs. LQI

Figure 4.10 shows the relation between LQI and RSSI of two nodes. The plots contain the node's measurement results of all distances and output power levels. The behavior is similar to the one observed in the indoor test (Figure 4.4). Node 4 starts receiving valid packets at a RSSI level of $-95dBm$ and node 6 at about $-90dBm$. The results of the two environments are different in terms of the location of the corrupted packets. In the anechoic chamber almost no corrupted packets were received with LQI values above 100 (Figure 4.10 and Figure 4.11(b)), while in the indoor test corrupted packets were present until the upper LQI limit at about 110 (Figure 4.4 and Figure 4.5(b)). All corrupted

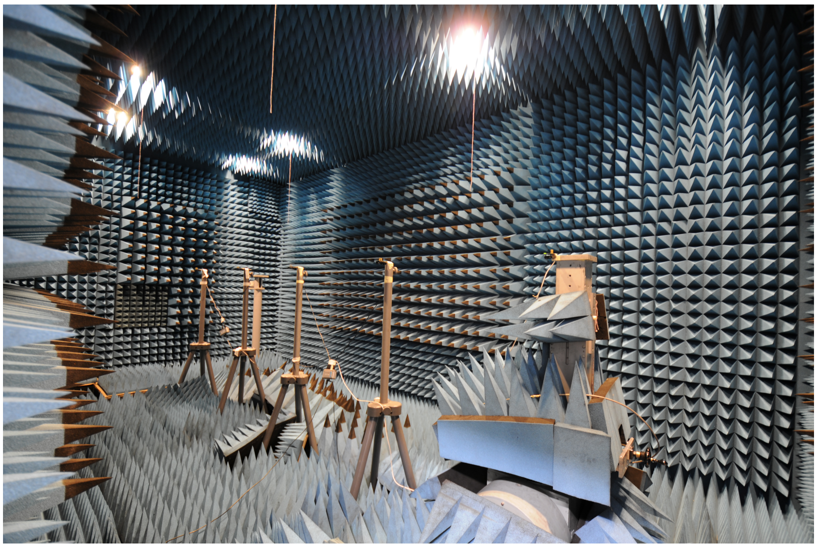


Figure 4.8: Setup of tests in the anechoic chamber. The transmitter node 0 is placed on the stand at the right-hand side of the picture. The four receiver nodes are placed on tripods in a row, on the left-hand-side of the transmitter node.

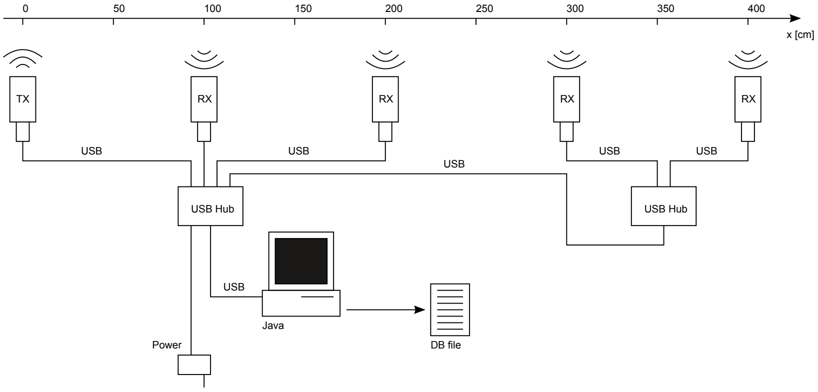
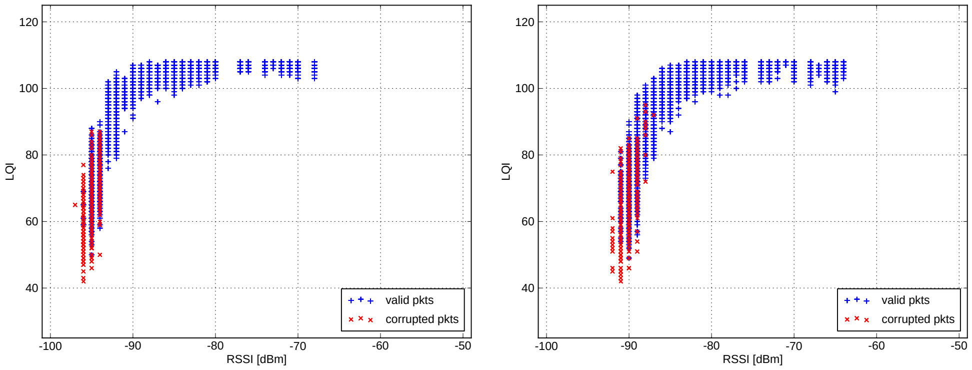


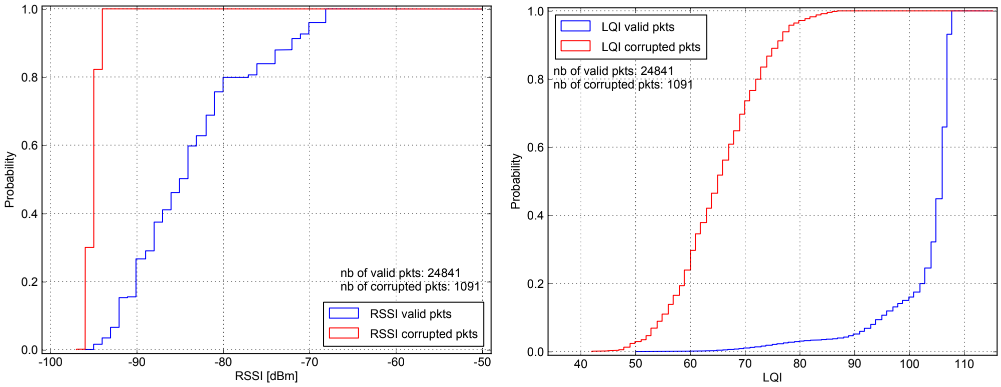
Figure 4.9: Setup of tests in the anechoic chamber with four nodes.

packets received by node 4 are located at RSSI values below $-96dBm$ while the valid packets are again uniformly distributed over the whole RSSI reception range (Figure 4.11(a)).



(a) (b)

Figure 4.10: LQI vs. RSSI of valid and corrupted packets of a) node 4 and b) node 6.



(a) (b)

Figure 4.11: Cumulative Distribution Function (CDF) of node 4: a) RSSI and b) LQI.

The reason for the absence of corrupted packets at high LQI values is probably the non-existence of exterior noise sources due to the insulation of the anechoic chamber. That is, if the LQI of the first 8 symbols used for the LQI calculation is high, most probably the same will be the case for the remaining symbols.

RSSI vs. Distance

The behavior of the RSSI versus distance is illustrated in Figure 4.12. The RSSI does not decrease monotonically but has a lowest value at 200cm for node 6 and at 300cm for node 2.

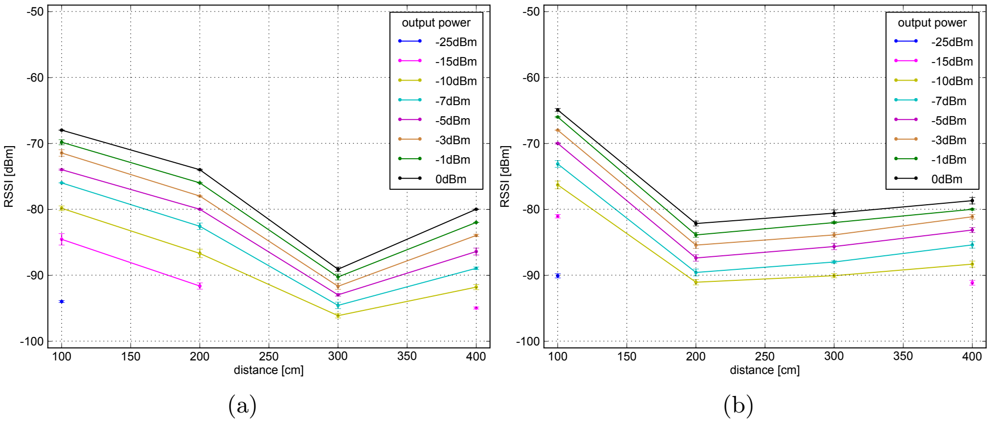


Figure 4.12: RSSI vs. distance of valid and corrupted packets of a) node 2 and b) node 6. The error bars indicate the standard deviation.

For a specific node, the form of the RSSI level with respect to the distance is almost the same for all output power levels. This result is similar to that of the indoor test. The presence of exterior noise sources within the anechoic chamber is very unlikely due to the insulation. Therefore, external noise can be excluded as reason for the non-monotonic decrease of the RSSI. One possible explanation of the non-monotonic decrease is the presence of interference due to the infrastructure for placing the nodes (tripods) and the cabling of the nodes (USB hubs and cables). While in the indoor test all nodes showed the same form of the RSSI level with respect to the distance (Figure 4.7), this is not the case for the test in the anechoic chamber.

4.4 Anechoic Chamber Test with 1 Nodes

4.4.1 Test Setup

This test was done in the same anechoic chamber as the previous test (Section 4.3), but only two nodes were involved. The transmitter was placed at 0cm and the single receiver node with the ID 6 was place at a new position in each round: 100cm, 200cm, 300cm and 400cm (Figure 4.13). The power sweep test was run in each round, sending 100 messages per output power level and round.

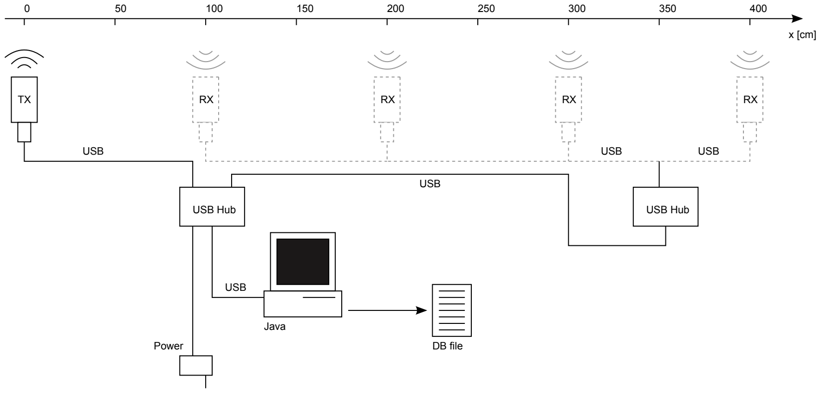


Figure 4.13: Setup of tests in the anechoic chamber with one node.

4.4.2 Results

In the following some selected plots are shown and described. Some more plots are listed in Appendix C.3.

RSSI vs. Distance

Figure 4.14(a) shows the behavior of RSSI vs. distance with a single receiver. Compared to the tests with four nodes (Section 4.3), no obvious influences of interferences on the RSSI are observable. This is reasonable because there is line of sight between transmitter and receiver. The RSSI level is monotonically decreasing independent of the output power. This behavior allows to compare the measurement results with the two models presented in Section 2.2. In the comparison, only the model's path loss L_p is considered. To compare the models and the measured RSSI levels, the models are fixed at the position of the first measured RSSI level, i.e. at $100cm$. For illustration reason, only the measurement with an output power level of $0dB$ is compared with the models (Figure 4.14(b)). The free-space model matches better than the two-ray model. This is plausible because the anechoic chamber reduces the reflections of electromagnetic waves reducing the influence of a reflection path signal. However, it has to be noted that the measured RSSI levels represent a short distance with few measurement points only. Therefore the validity of the model-measurement comparison is limited.

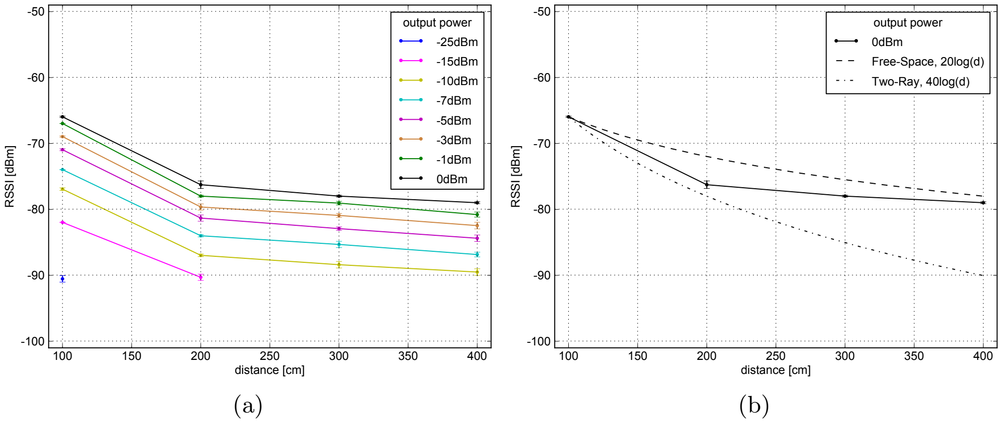


Figure 4.14: a) RSSI mean vs. distance of valid and corrupted packets of node 6. The error bars indicate the standard deviation. b) Comparison of the RSSI measurement results with the free-space and two-ray model.

Chapter 5

Conclusion

5.1 Summary and Achievements

In this semester thesis, we built a testbed to characterize the wireless channel between nodes of WSNs. The testbed allows for measuring different properties of the channel such as RSSI, LQI and PRR. Furthermore, the testbed allows for fast adaption of different test parameters (output power level, radio channel, payload length, number of nodes etc.).

Measurements conducted on the testbed have shown that under certain circumstances Tmote Sky nodes of the same hardware version behave very differently. For example while one node receives all transmitted packets, another node at the same position may not receive any packets. Furthermore, the RSSI level with respect to the distance between transmitter and receiver depends on the environment. In a real environment the RSSI level does not allow for reliable distance measurement due to its non-monotonic behavior. Even in an anechoic chamber where reflections are reduced, the presence of other nodes influences the RSSI significantly. Moreover, RSSI and LQI measurements of the radio chip are not always reliable because their values are calculated based only on a fraction of the whole packet. That is, a high LQI value does not automatically result in the reception of a valid packet. While in the anechoic chamber a high LQI corresponds to a valid packet, in a noisy environment external interference may corrupt the packet after the radio chip's LQI calculation resulting in an invalid packet with a high LQI. In this semester thesis only a few factors of influence were examined. There are many other factors on which a reliable communication between nodes depends, e.g. orientation of the antenna, radio frequency and packet length.

5.2 Future Work

In the described tests, the testbed was used with only one transmitter and multiple receivers. However, the testbed is designed such that more than one node can be used as a transmitter. This would allow for the sending of quasi-simultaneous packets by multiple transmitters and with adapted power levels. The sending would be quasi-simultaneous because only one node after the other can be addressed via USB. But with some modifications it would be possible to transmit multiple packets synchronously so that the behavior of the receivers on the presence of multiple transmitters can be analyzed. For example, the transmitters can be synchronized by connecting the individual nodes with a common line to trigger the sending or by implementing a synchronization algorithm. Another direction of study could be the modification of additional parameters such as payload length or radio channel allowing a more detailed channel characterization. Furthermore, characterization could take place in the presence of jammers causing intentional interference.

Appendix A

List of Abbreviations

DB	database
CDF	Cumulative Distribution Function
CRC	Cyclic Redundancy Check
ISM	Industrial, Scientific and Medical
LQI	Link Quality Indication
PA	Power Amplifier
PRR	Packet Reception Rate
SFD	Start of Frame Delimiter
SHR	Synchronization Header
PHR	PHY Header
PSDU	PHY Service Data Unit
RF	Radio Frequency
RSSI	Receive Signal Strength Indicator
TI	Texas Instruments
TX	Transmit
UART	Universal Asynchronous Receiver/Transmitter
USB	Universal Serial Bus
WSNs	Wireless Sensor Networks

Appendix B

Database Tables

Table B.1: DB: startTimeTable.

Column Name	Data Type	Description
startTime	DOUBLE	absolute start time in milliseconds[1]

Table B.2: DB: testEnvironmentTable.

Column Name	Data Type	Description
nodeId	INTEGER	node ID
gwIdx	INTEGER	Gateway index of the node
gwIdentifier	INTEGER	Gateway ID (corresponds to the node ID)
usbPort	TEXT	USB port name
x	INTEGER	x position in cm
y	INTEGER	y position in cm
z	INTEGER	z position in cm

Table B.3: DB: radioMsgTable.

Column Name	Data Type	Description
time	DOUBLE	relative time of sending[2]
gateway	INTEGER	gateway ID (corresponds to the node ID)
source	INTEGER	source in message header
destination	INTEGER	destination in message header
counter	INTEGER	counter value
txDelay	INTEGER	TX delay in milliseconds
txPower	INTEGER	PA level
rndLength	INTEGER	length of the random payload

[1] difference, measured in milliseconds, between the current time and midnight, January 1, 1970 UTC.

Table B.4: DB: radioAckMsgTable.

Column Name	Data Type	Description
time	DOUBLE	relative time of sending[2]
gateway	INTEGER	gateway ID (corresponds to the node ID)
source	INTEGER	source in message header
destination	INTEGER	destination in message header
counter	INTEGER	counter value
txDelay	INTEGER	TX delay in milliseconds
txPower	INTEGER	PA level
rndLength	INTEGER	length of the random payload

Table B.5: DB: extendedRadioMsgTable.

Column Name	Data Type	Description
time	DOUBLE	relative time of sending[2]
gateway	INTEGER	gateway ID (corresponds to the node ID)
source	INTEGER	source in message header
destination	INTEGER	destination in message header
counter	INTEGER	counter value
txDelay	INTEGER	TX delay in milliseconds
txPower	INTEGER	PA level
rssi	INTEGER	$\mathrm{RSSI}_{\mathrm{VAL}}$
lqi	INTEGER	LQI
crc	INTEGER	CRC flag
rndLength	INTEGER	length of the random payload

[2] difference, in milliseconds, between the time of sending/receiving and the absolute start time Table B.1

Table B.6: DB: crcErrorRadioMsgTable.

Column Name	Data Type	Description
time^v	DOUBLE	relative time of sending[2]
gateway^v	INTEGER	gateway ID (corresponds to the node ID)
source^c	INTEGER	source in message header
destination^c	INTEGER	destination in message header
amType^c	INTEGER	type of message
counter^c	INTEGER	counter value
txDelay^c	INTEGER	TX delay in milliseconds
txPower^c	INTEGER	PA level
rssi^v	INTEGER	RSSI_{VAL}
lqi^v	INTEGER	LQI
crc^v	INTEGER	CRC flag
rndLength^c	INTEGER	length of the random payload

^{v} column is always valid
^{c} column may be corrupted

Table B.7: DB: setRadioChannelMsgTable.

Column Name	Data Type	Description
time	DOUBLE	relative time of sending[2]
gateway[3]	INTEGER	gateway ID (corresponds to the node ID)
source[3]	INTEGER	source in message header
destination[3]	INTEGER	destination in message header
radioChannel	INTEGER	radio channel

[3] *gateway*, *source* and *destination* have the same value

Table B.8: DB: setRadioChannelAckMsgTable.

Column Name	Data Type	Description
time	DOUBLE	relative time of sending[2]
gateway	INTEGER	gateway ID (corresponds to the node ID)
source	INTEGER	source in message header
destination	INTEGER	destination in message header
radioChannel	INTEGER	radio channel
ack	INTEGER	acknowledge flag

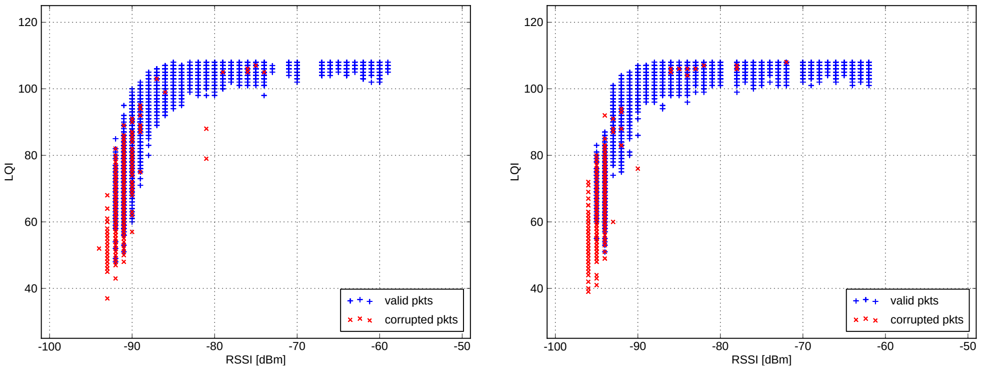
Table B.9: DB: unknownMsgTable.

Column Name	Data Type	Description
time	DOUBLE	relative time of sending[2]
gateway	INTEGER	gateway ID (corresponds to the node ID)
hexDump	TEXT	complete message including header

Appendix C

Measurement Plots

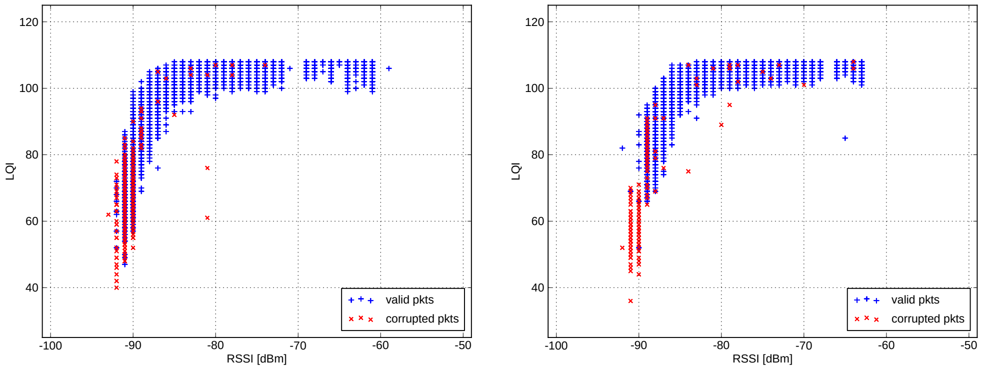
C.1 Indoor Test



(a)

(b)

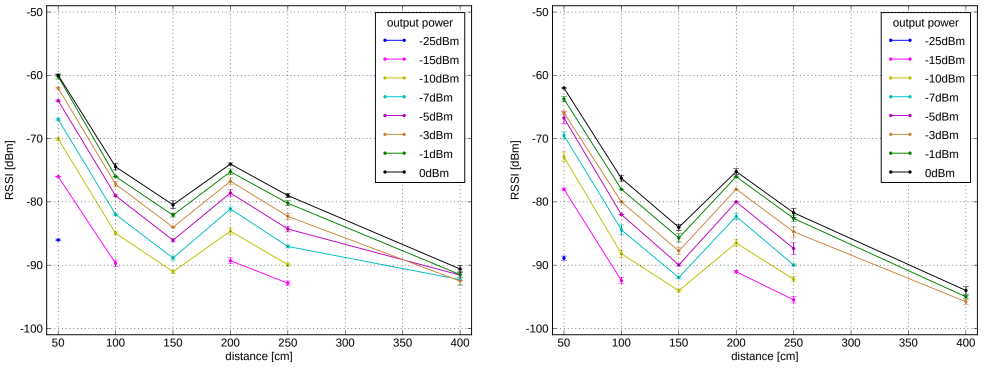
Figure C.1: LQI vs. RSSI of valid and corrupted packets of a) node 1 and b) node 2.



(a)

(b)

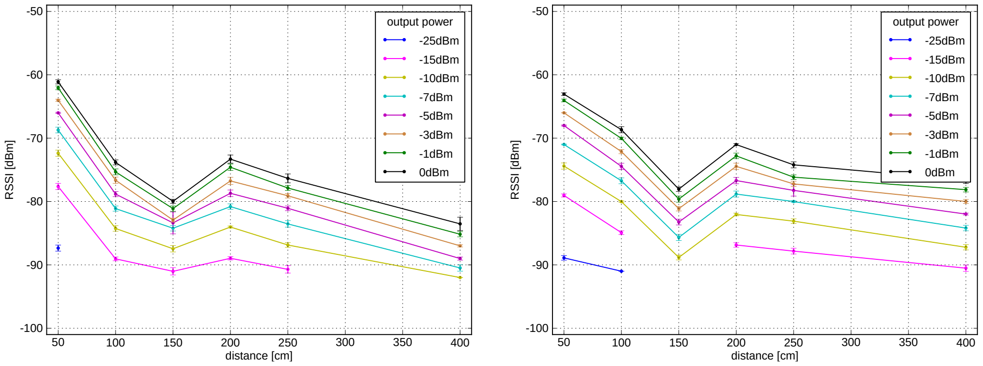
Figure C.2: LQI vs. RSSI of valid and corrupted packets of a) node 5 and b) node 7.



(a)

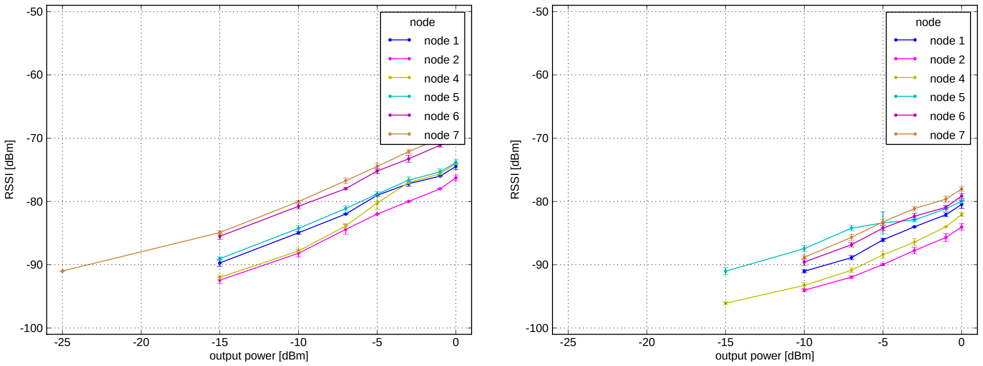
(b)

Figure C.3: RSSI mean vs. distance of valid and corrupted packets of a) node 1 and b) node 2. The error bars indicate the standard deviation.



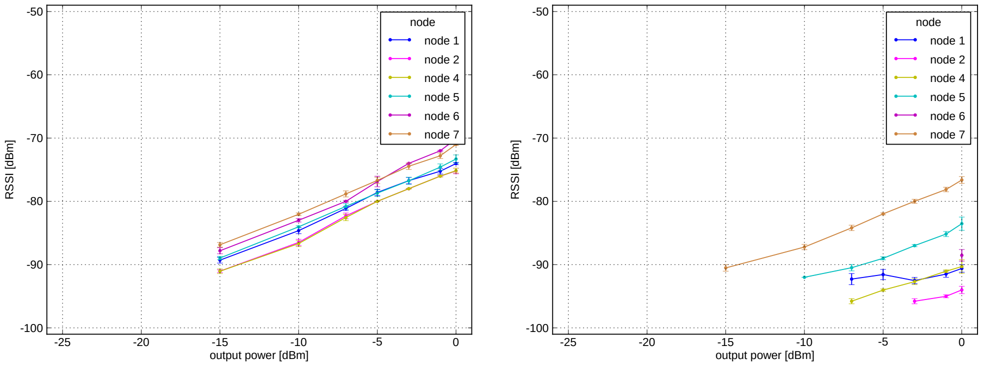
(a) (b)

Figure C.4: RSSI mean vs. distance of valid and corrupted packets of a) node 5 and b) node 7. The error bars indicate the standard deviation.



(a) (b)

Figure C.5: RSSI mean vs. output power of valid and corrupted packets at a) 100cm and b) 150cm. The error bars indicate the standard deviation.

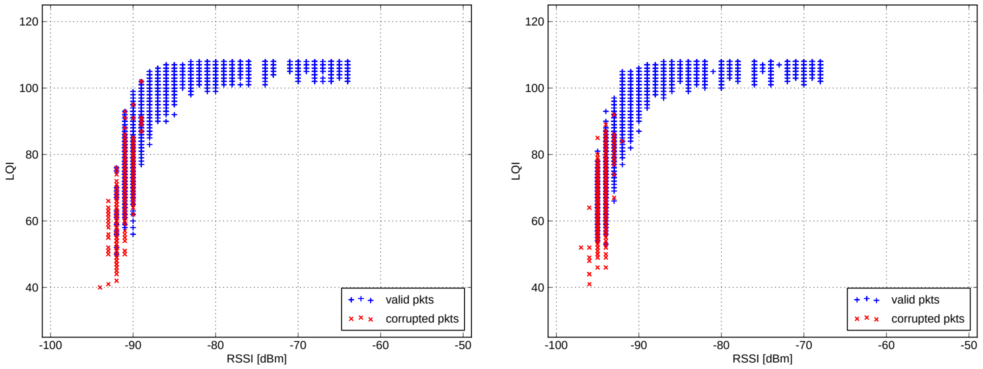


(a)

(b)

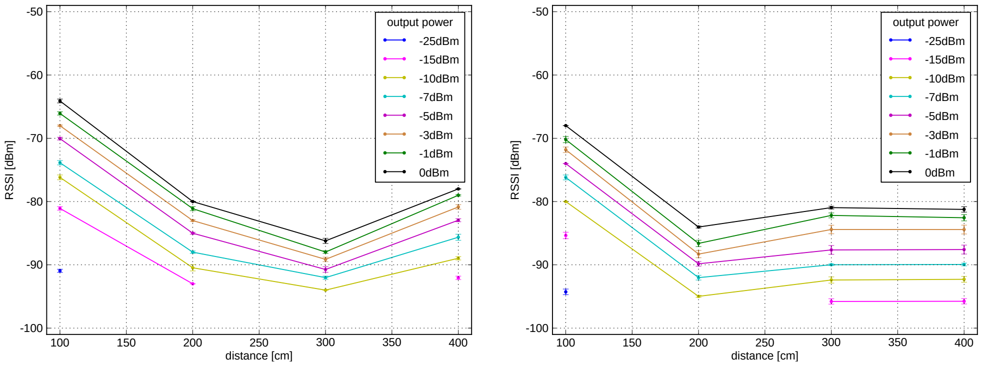
Figure C.6: RSSI mean vs. output power of valid and corrupted packets at a) $200cm$ and b) $400cm$. The error bars indicate the standard deviation.

C.2 Anechoic Chamber Test with 4 Nodes



(a) (b)

Figure C.7: LQI vs. RSSI of valid and corrupted packets of a) node 1 and b) node 2.



(a) (b)

Figure C.8: RSSI mean vs. distance of valid and corrupted packets of a) node 1 and b) node 4. The error bars indicate the standard deviation.

C.3 Anechoic Chamber Test with 1 Nodes

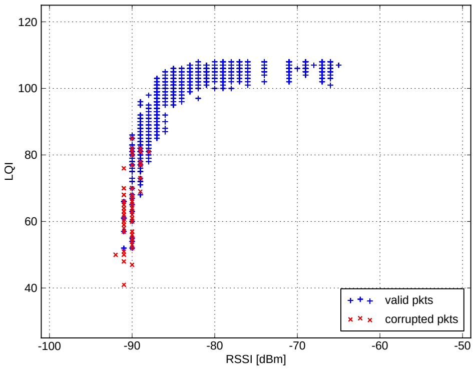


Figure C.9: LQI vs. RSSI of valid and corrupted packets of node 6.

Bibliography

[1] Moteiv Corporation. *Data Sheet: Ultra low power IEEE 802.15.4 compliant wireless sensor module*, 12-November 2006.

[2] M.M. Holland, R.G. Aures, and W.B. Heinzelman. Experimental investigation of radio performance in wireless sensor networks. In *2nd IEEE Workshop on Wireless Mesh Networks (WiMesh)*, pages 140–150, 2006.

[3] Tsenka Stoyanova, Fotis Kerasiotis, Aggeliki Prayati, and George Papadopoulos. Evaluation of impact factors on rss accuracy for localization and tracking applications. In *Proceedings of the 5th ACM international workshop on Mobility management and wireless access (MobiWac)*, pages 9–16, 2007.

[4] Texas Instruments. *Data Sheet: 2.4 GHz IEEE 802.15.4 / ZigBee-ready RF Transceiver*, swrs041b edition, 19-March 2007.

[5] Dimitrios Lymberopoulos, Quentin Lindsey, and Andreas Savvides. An empirical characterization of radio signal strength variability in 3-d ieee 802.15.4 networks using monopole antennas. In *EWSN*, pages 326–341, 2006.

[6] J.S. Seybold. *Introduction to RF propagation.* Wiley, 2005.

[7] J.B. Anderson. *Digital transmission engineering.* IEEE Series on Digital & Mobile Communication. IEEE Press, 2005.

[8] IEEE 802. *Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low-Rate Wireless Personal Area Networks (LR-WPANs).* IEEE Computer Society, October 2003.

[9] Chieh-Jan Mike Liang, Nissanka Bodhi Priyantha, Jie Liu, and Andreas Terzis. Surviving wi-fi interference in low power zigbee networks. In *Proceedings of the 8th ACM Conference on Embedded Networked Sensor Systems (SenSys)*, pages 309–322, 2010.