Michael Schober, BSc

Implementation and Design of a UWB-Based Network Simulation Platform for Indoor Localization Systems

MASTER THESIS

to achieve the university degree of

Diplom-Ingenieur

Master's degree programme: Information and Computer Engineering

submitted to:

Graz University of Technology

Supervisor

Ass.Prof. Dipl.-Ing. Dr.techn. Christian Steger

Institute for Technical Informatics

Advisor

Ass.Prof. Dipl.-Ing. Dr.techn. Christian Steger

Dipl.-Ing. Lukas Greßl

Graz, April 2019

EIDESSTATTLICHE ERKLÄRUNG

Ich erkläre an Eides statt, dass ich die vorliegende Arbeit selbstständig verfasst, andere als die angegebenen Quellen/Hilfsmittel nicht benutzt, und die den benutzten Quellen wörtlich und inhaltlich entnommenen Stellen als solche kenntlich gemacht habe. Das in TUGRAZonline hochgeladene Textdokument ist mit der vorliegenden Masterarbeit identisch.

Datum

Unterschrift

Kurzfassung

Die UWB Technologie ermöglicht eine sichere und genaue Positionserfassung von mobilen Geräten. Da für diese Positionserfassung eine hohe Bandbreite benötigt wird, kann es bei größeren Netzwerken schnell zu einer Überlastung des Kommunikationskanals kommen. Speziell Echtzeit-Lokalisierungssysteme welche in Gebäuden verwendet werden benötigen eine hohe Frequenz für die Erfassung der Positionen. Dies führt zu einer zusätzlichen Belastung des Kommunikationskanals.
Ziel dieser Masterarbeit war es, eine Simulationsplattform für solche Lokalisierungssysteme zu erstellen, welche für den späteren Systementwurf verwendet werden kann. Die Plattform kann dazu verwendet werden, die Auswirkungen der Netzwerkgröße auf den UWB Kanal abzuschätzen und unterschiedliche Systemkonfigurationen zu testen. Weiters wurden die Personen welche mit dem Netzwerk lokalisiert werden sollen und deren Einfluss auf physikalischen Kanalparameter berücksichtigt um eine möglichst genaue Abschätzung der Realisierbarkeit der simulierten Systeme zu gewährleisten. Die Plattform ermöglicht es auch gewisse Bewegungsmuster der zu lokalisierenden Personen zu erstellen damit auch dynamische Systeme evaluiert werden können. Die Simulationsresultate geben Auskunft über die Kanalqualität, die Anzahl der Personen die geortet werden können und die noch zur Verfügung stehende Kanalkapazität.

Im letzten Kapitel werden unterschiedliche Konzepte von Systemrealisierungen getestet und ausgewertet um die Vor- und Nachteile von den Systemkonfigurationen zu zeigen. Zusätzlich wird die Genauigkeit und der Bandbreitenverbrauch von unterschiedlichen Lokalisierungsmethoden simuliert, die eine Abschätzung des Messfehlers ermöglicht. Dies soll als Hilfestellung für einen späteren Systementwurf dienen.

Abstract

The UWB-technology allows a secure and accurate localization of mobile devices. Since a localization requires a high bandwidth, networks of a bigger size can quickly lead to an overload of the used UWB-channel. Especially real-time localization systems that are used for an indoor localization of persons do have special requirements from a localization frequency point of view. This leads to an additional load on the UWB-Channel.
The goal of this thesis was to create a simulation platform that can be used for creating models of such indoor localization systems. This platform can be later used for evaluating the impact of the network size on the corresponding UWB-channel. The platform includes the simulation of obstacles and persons within the room and the impact on the UWB-channel of the localized devices. For making the simulations as realistic as possible, also movement patterns of the persons can be created and evaluated. The simulation platform can benchmark the systems from a channel-quality, channel-capacity and a localization throughput perspective.

In the last chapter, different system approaches are simulated. Based on the simulation results, the advantages and disadvantages of the different system configurations can be evaluated. Additionally, different localization methods are simulated and evaluated from a measurement accuracy and a bandwidth consumption perspective. This creates reverence data that can be used for a later localization system development.

Danksagung

Diese Diplomarbeit wurde im Jahr 2018 am Institut für Technische Informatik an der Technischen Universität Graz, in Kooperation mit NXP Semiconductors Austria GmbH durchgeführt.

Mein besonderer Dank gilt Herrn Dipl.-Ing. Lukas Greßl und Rias Al-Kadi MSc MBA von NXP Semiconductors Austria GmbH. Herr Greßl hat mich bei der Ausführung und Dokumentation meiner Masterarbeit maßgeblich unterstützt. Herr Al-Kadi unterstütze mich beim Konzeptentwurf und der Evaluierung. Des Weiteren bedanke ich mich bei Herrn Ass.Prof. Dipl.-Ing. Dr.techn. Christian Steger für die Betreuung und Begutachtung der Arbeit. Zusätzlich möchte ich mich noch bei Herrn Dipl.-Ing. David Veit und Dipl.-Ing. Thomas Wilding bedanken, welche mir bei der Überprüfung der Simulationsergebnisse geholfen haben.

Mein weiterer Dank gilt meiner Familie und meinen Freunden die mich während des Verfassens dieser Arbeit unterstützt haben. Besonders bedanke ich mich bei David Lenzhofer, Philipp Rosmann und Thomas Stranger für das sorgfältige Korrekturlesen meiner Masterarbeit.

Graz, im April 2019 — Michael Schober

Contents

List of Figures

List of Tables

Chapter 1

Introduction

The term Ultra-Wideband (UWB) refers to a wireless communication technology that uses short pulses in the time domain for the encoding of data. For keeping the pulses as short as possible, a wide frequency spectrum is required. Analogue to narrowband technologies the UWB technology uses a high frequency narrowband carrier for the transmission of the data. A technology is named UWB if the required bandwidth is at least 20 % of the carrier center-frequency or if the bandwidth is bigger than 500 MHz.

Wireless sensor networks consist of multiple devices reporting to a common central unit responsible for collecting the captured data. Such data can be for example: room temperature, air pressure or humidity. This data can be measured very easy by modern sensors, but it's hard to track the location of the sensor itself. The advantage of the UWB technology is, the capability of making high accurate time measurements. This means also the time of flight (TOF) a signal can be measured with a high accuracy. By combining multiple TOFs to different reference stations, the location of a device can be determined. This was not possible with a comparable accuracy by using any prior art narrowband technology. So the UWB technology opens new possibilities for location aware sensor data.

1.1 History of Ultra-Wideband

UWB in its general meaning was first used by Guglielmo Marconi in 1901 for transmitting Morse code over the Atlantic Ocean. For the encoding of the symbols he used short time pulses created by a spark gap radio transmitter. Because of the little time consumption of the sparks, a high frequency bandwidth was required. So the first data transmission system based on the UWB technology was born. The advantages of the UWB technology regarding its potential high data rates and the possibility of using it for a TOF measurement were not considered at that time. Also, disturbances caused by the pulses didn't matter because no wireless communication technologies did exist at that time.
About 50 years later the pulse-based technology gained an interest of the military for creating impulse based radars applications. Pioneers of the UWB communication were Henning Harmuth from the Catholic University of America and Gerald Ross with K. W. Robins working for the Sperry Rand Corporation [1]. Because of its high bandwidth consumption

that interfered with other technologies, the UWB technology was restricted to military and defence applications from 1960 to 1990. Through the ongoing developments in the microelectronic industry, the integrated circuits became faster and smaller. This increased the interest on the UWB technology also for more commercial applications. The possibility of making commercial application based on UWB caused developers to make pressure on the Federal Communications Commission (FCC) for allowing the UWB technology for a commercial use. After about 10 years, the FCC approved the usage of commercial UWB applications under strict limitations from an output power and bandwidth perspective [1].

1.2 Problem Statement and Motivation

Offering location aware data in general is a problem that hasn't been solved with a high accuracy by using narrow band technologies. For example, a Bluetooth based indoor localization system offers a location accuracy of roughly $2\,m$, therefore it requires a high density of broadcast stations that are also known as anchors. Also, a localization based on Bluetooth is not secure because it's based on the received signal strength indication principle (RSSI). The RSSI based distance measurement compares the received signal strength with the radiated signal strength, thus the distance between two devices can be estimated. Since the measurement based on the signal strength, there is no prevention against replay attacks e.g. like a simple radiation of the Bluetooth carrier. Since UWB based systems offer a localization accuracy in the centimetre domain, UWB systems are an auspicious technology for serving use cases that require an accurate and secure indoor localization.

The UWB technology requires a big bandwidth for the data transmission. Out of that reason, it has limitation from a channel capacity perspective. A big number of anchors is required if hundreds of mobile devices, also called nodes, need to be tracked within a building. The number of messages that need to be transmitted within a system scales with the number of anchors and nodes. The throughput boarders of such a network can be reached quite easily if the system was not designed properly. Independent from the outcome, making a feasibility study of a system based on prototypes leads to high development and material cost. Out of that reason, systems that require a big number of anchors and nodes need to be simulated first before prototypes are made. This thesis is about creating a platform for making an analysis about small and big scale UWB network that can be used for indoor localization. Also, this platform can be used for evaluating different system approaches from a infrastructure and throughput perspective.

Chapter 2

Related Work

This chapter will give an overview about the ranging technologies that are used in modern UWB systems. These techniques can be used either for a peer to peer distance measurement or for making an indoor localization. Also, the state-of-the-art network simulation platforms and some communication protocols that are evaluated are explained.

2.1 Ultra-Wideband Based Distance Measurement

A UWB distance measurement is based on measuring the time that is required for a message to travel from one device to the other. This measurement technique is also known as a Time of Flight (TOF) measurement. Because both devices have a very accurate internal clock, high accurate timestamps can be stored when a messages was transmitted and received. Figure 2.1 shows the message exchange that is required for doing a TOF measurement.

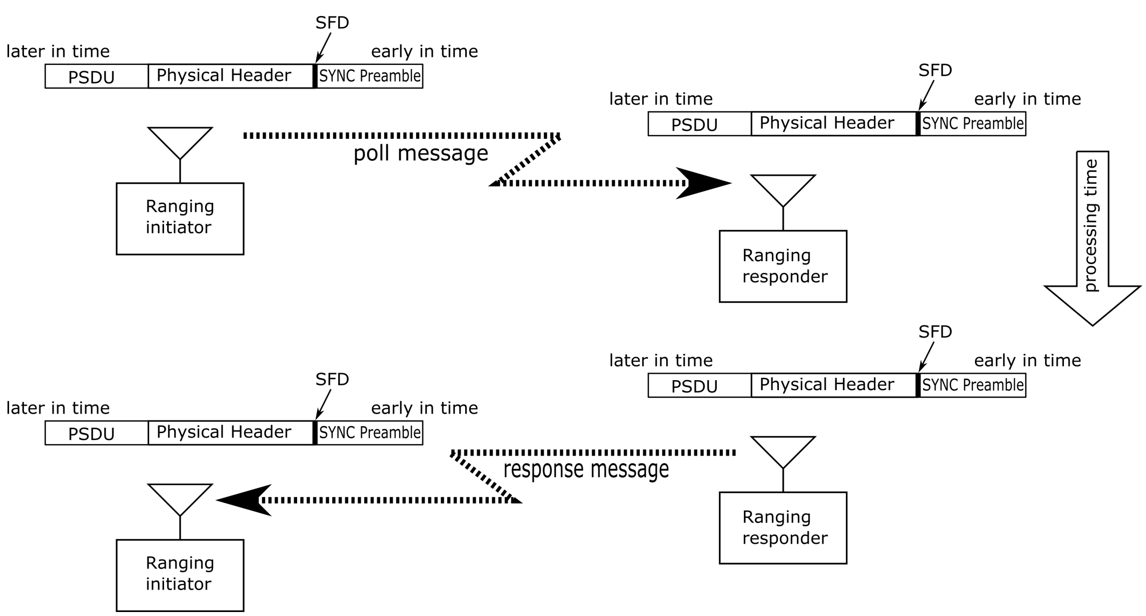

Figure 2.1: The setup of a UWB ranging frame exchange.

An active TOF measurement as specified in the IEEE 802.15.4a requires two devices, a ranging originator also known as initiator and a responder. The initiator sends out a poll message that is received by the responder. After the reception the responder processes the message and checks its payload. If the processing unit says the message, inclusive payload was received without any errors, the payload content needs to be checked if the message was really meant for the responder. The MAC is defined in the IEEE 802.15.4 standard [2]. Only if the received destination address fits to the address of the devices, the device may respond with the source address as destination. When the message is transmitted on the initiator side a timestamp is stored based on the transmission time of the SFD marker. When the SFD marker of a successfully processed and interpreted message is received on the responder side, the responder also stores a timestamp. Based on the ranging protocol layers, the responder configures and transmits its response message and stores the SFD transmission timestamp analogous to the initiator. Also, the initiator processes and interprets the received response messages and stores the SFD timestamp if the message was received with proper content.

2.1.1 Single Sided - Two-Way Ranging

A message exchange shown in Figure 2.1 is also known as Single Sided -Two-Way Ranging (SS-TWR) [3]. The message exchange of SS-TWR in the time domain is shown in Figure 2.2. The initiator sends the poll message at *t_send_poll*. The responder receives the

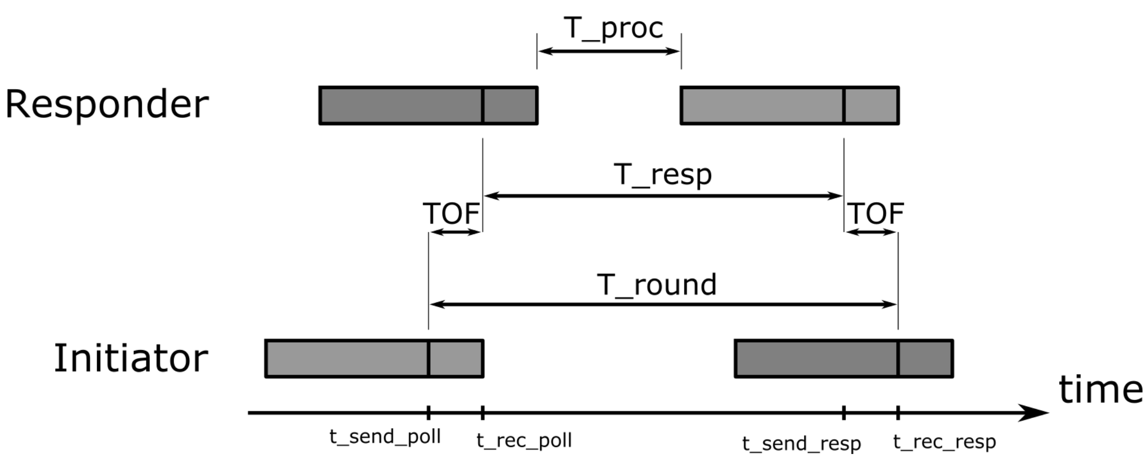

Figure 2.2: The timing diagram of a Single Sided - Two-Way Ranging.

message a signal propagation time later at *t_rec_poll*. Both devices store the SFD marker timestamp when the message is sent or received. When the poll message was fully received, the responder starts processing it. If the UWB-frame was received without any error, the responder prepares and sends the response message. The response message is received a TOF later by the initiator. Also for the response message the SFD times are stored by both devices, the corresponding timestamps are called *t_send_resp* and *t_rec_resp*. The time that passes from sending the poll message to the reception of the response message is called round-trip time. The time that passes from the reception of the poll message until the response message is sent is called response time. The time span that is required by the responder for checking the poll message and preparing the response message is called processing time. The processing, response and round-trip time are symbolized in Figure 2.2 by the variables T_proc, T_resp and T_round. The time span that passes from the transmission to the reception of a UWB-frame is symbolized by the variable TOF. All these variables need to be calculated based on the SFD timestamps that are stored by the devices. Equation 2.1, 2.2 and 2.3 show how the round-trip, processing and response time can be calculated based on the timestamps stored by the devices. The time T_msg_frame is the predefined length of a UWB-frame.

$$T\_round = t\_rec\_resp - t\_send\_poll \tag{2.1}$$

$$T\_resp = t\_send\_resp - t\_rec\_poll \tag{2.2}$$

$$T\_proc = T\_resp - T\_msg\_frame \tag{2.3}$$

The TOF can be calculated by applying equation 2.4.

$$TOF = \frac{T\_round - T\_resp}{2} \tag{2.4}$$

Since the calculation of the TOF requires both the response and the processing time, both of these times need to be stored on one device. The responder can achieve this by chaining its predefined processing time in the PSDU field of the response message, so that the initiator can calculate the TOF without sending an additional message.

For getting an accurate SS-TWR ranging result, the reference clock the SFD timestamps are generated on, needs to be very accurate. A state-of-the-art system can achieve a RX timestamp accuracy of about $0.1\,ns$ by using a sampling frequency of $500\,MHz$, the TX timestamp error is negligible. The resulting TOF error is shown in equation 2.5.

$$TOF_{err} = \frac{(T\_round + \Delta err) - (T\_resp - \Delta err)}{2} = \Delta err \tag{2.5}$$

Equation 2.5 only holds if both devices are running on very accurate clocks with a negligible clock-drift. Since most UWB systems need to operate in a non-constant environments from a temperature perspective, the clock-drift can cause measurement errors of up to several nanoseconds depending on the message length and the processing time.

2.1.2 Double Sided - Two-Way Ranging

For compensating the clock-drift based error of a SS-TWR a new ranging method called Double Sided - Two-Way Ranging (DS-TWR) has been introduced [4]. Figure 2.3 shows the message exchange for preforming a DS-TWR. Basically, a DS-TWR consists out of two SS-TWR where one message is shared between the first and the second SS-TWR. The initiator starts with transmitting the poll message that is received by the Responder. The responder answers with the response message which is received by the initiator. After the reception of the response message the initiator sends out the final message which is again received by the responder. The first SS-TWR is made by sending the poll and response message. The second SS-TWR sequence uses the response message as poll message and

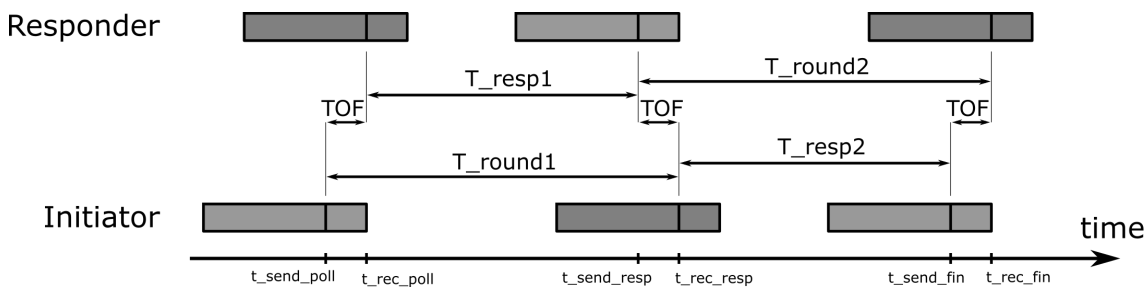

Figure 2.3: The timing diagram of a Double Sided - Two-Way Ranging.

sends a final message which is used as response message. The timestamps t_send_fin and t_rec_fin are stored when the final messages are transmitted and received. The equations 2.7, 2.9, 2.6 and 2.8 show how the round-trip times can be calculated.

$$T\_round1 = t\_rec\_resp - t\_send\_poll \tag{2.6}$$

$$T\_resp1 = t\_send\_resp - t\_rec\_poll \tag{2.7}$$

$$T\_round2 = t\_fin\_resp - t\_send\_resp \tag{2.8}$$

$$T\_resp2 = t\_send\_fin - t\_rec\_resp \tag{2.9}$$

Assuming the initiator clock is drifting by the factor k_a and the responder clock is drifting by the factor k_b in comparison to an ideal clock. k_a and k_b are defined in the equations 2.10 and 2.11, where α and β are the relative clock-drifts.

$$k_a = 1 + \alpha \tag{2.10}$$

$$k_b = 1 + \beta \tag{2.11}$$

The equations 2.12 and 2.13 show the influence of the clock-drift on the SS-TWR.

$$TOF_{SS-TWR} = \frac{T_{round1} \cdot k_a - T_{resp1} \cdot k_b}{2} \tag{2.12}$$

$$TOF_{SS-TWR} = \frac{T_{round1} \cdot k_a - T_{resp1} \cdot k_b}{2} \approx TOF + (\alpha - \beta) \cdot T_{resp1} \tag{2.13}$$

Because it's very unlikely that $alpha$ and $beta$ are equal and the response time is at least a message length, the TOF error can become very big in comparison to DS-TWR. Considering a difference of $1\,ppm$ between α and β and a response time of $1\,ms$, the clock-drift based error is $1\,ns$.

Equation 2.14 shows how the clock-drift compensated TOF can be calculated based on a DS-TWR.

$$TOF_{DS-TWR} = \frac{T\_round1 \cdot k_a \cdot T\_round2 \cdot k_b - T\_resp1 \cdot k_b \cdot T\_resp2 \cdot k_a}{T\_resp1 \cdot k_b + T\_round1 \cdot k_a + T\_resp2 \cdot k_a + T\_round2 \cdot k_b} \tag{2.14}$$

The assumption shown in equation 2.15 is valid for the denominator because it adds only a small scaling factor to the TOF.

$$T\_resp1 \cdot k_b + T\_round2 \cdot k_b \approx T\_resp2 \cdot k_b + T\_round1 \cdot k_b \tag{2.15}$$

So equation 2.14 can be formed to equation 2.16 and 2.17.

$$TOF_{DS-TWR} = \frac{k_a \cdot k_b}{2 \cdot k_b} \frac{T\_round1 \cdot T\_round2 - T\_resp1 \cdot T\_resp2}{T\_round2 + T\_resp2} \tag{2.16}$$

$$TOF_{DS-TWR} = \frac{T\_round1 - T\_resp1}{2} \cdot k_a = k_a \cdot TOF = TOF + TOF \cdot \alpha \tag{2.17}$$

Since UWB is used for short distance measurement the clock-drift based error can be neglected for DS-TWR.

2.2 ALOHA Protocol

The ALOHA protocol is an asynchronous communication protocol that has been invented in 1970 for the creation of the ALOHAnet network. It was first used as a communication protocol between the University of Honolulu, Hawaii, and the Islands of Hawaii. Initially it was only planned to be a wireless networking protocol but it also became also the basis of the Ethernet-protocol. Figure 2.4 shows the device internal state chart of the ALOHA-Protocol. In the "Idle"state, the device is waiting for a message that needs to be

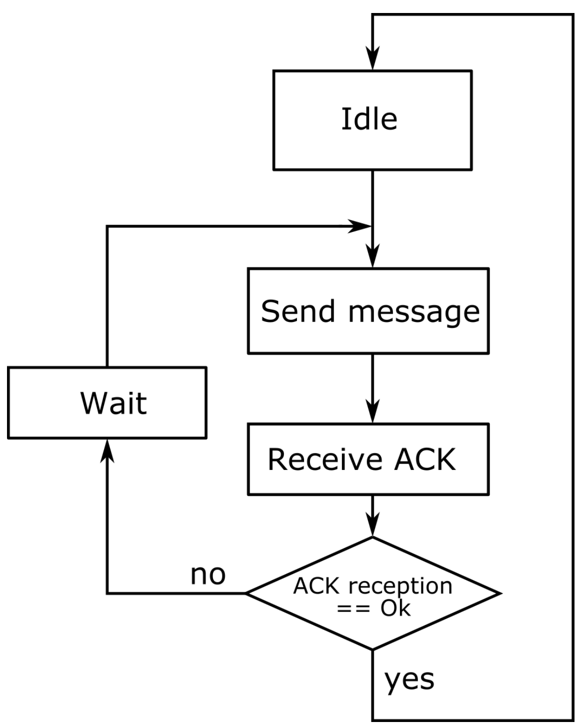

Figure 2.4: State chart of the ALOHA protocol.

transmitted, during this state no RF message transmission triggered by the device itself is

going on. When a message should be transmitted the device is going into the "Send message"state. During this state, the channel is block by the device until the transmission was completed and the device switches into to "Receive ACK"state. When the target device has successfully received the transmitted message, it responses with an acknowledgement (ACK) message that is received by the transmitting device. If the reception of the ACK message was okay the device can switch back into the "Idle"state. When the reception of the ACK message was erroneous the device switches in the "Wait"state. Possible reasons for an unsuccessful received ACK message could be that it wasn't sent at all or that the transmission was disturbed by a different interfering device. If ACK message was not send at all the transmission of first message was already erroneous. In any case, the device needs to wait a random time during the "Wait"state and restart the transmission afterwards. This wait and retransmit procedure is repeated until the reception of the ACK message was successful. The waiting time needs to be random because if it would be constant, devices would disturb each other's until the drift of the internal clock would lead to an shift of the transmitted messages in a way that they don't overlap any more.
Figure 2.5 shows three devices transmitting messages based on the ALOHA protocol. The

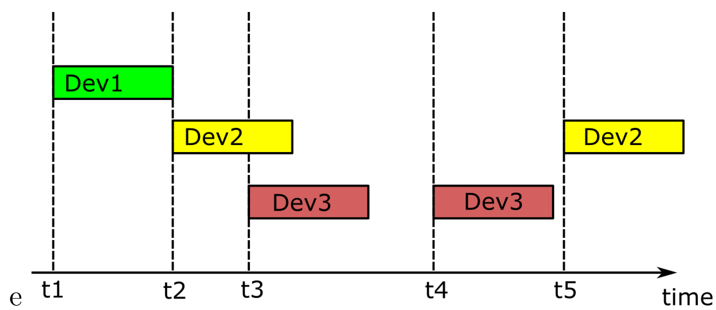

Figure 2.5: Timing diagram of the ALOHA protocol.

devices are not synchronized, so the starting time of each device is individual. The first message is transmitted by "Dev1", since no messages of another device is overlapping with this message, the transmission of "Dev1"is successful and there is no need to repeat it. "Dev2"starts transmitting the second message at the time "t2"and "Dev3"starts transmitting its message at "t3". Since the messages of "Dev3"and "Dev2"are overlapping, both message transmissions are erroneous and need to be repeated. The second attempts of the message transmissions takes place at "t4"and "t5". Since both messages are transmitted without any disturbances, the reception of the ACK messages are okay and there is no need for a retransmission.

2.2.1 Slotted-ALOHA Protocol

The difference between the ALOHA and the Slotted-ALOHA protocol is that for the Slotted-ALOHA protocol the devices are synchronized.That means that there given times when a transmission may start. Figure 2.6 shows the same message exchange of the ALOHA protocol with the Slotted-ALOHA. As before "Dev1"can transmit again without any disturbances. But now since the starting time of "Dev2"and "Dev3"is the same, the advantage is that a message can only interfere with the messages of the same slot. This means, with ALOHA a message of "Dev3"can interfere with two other messages, even if

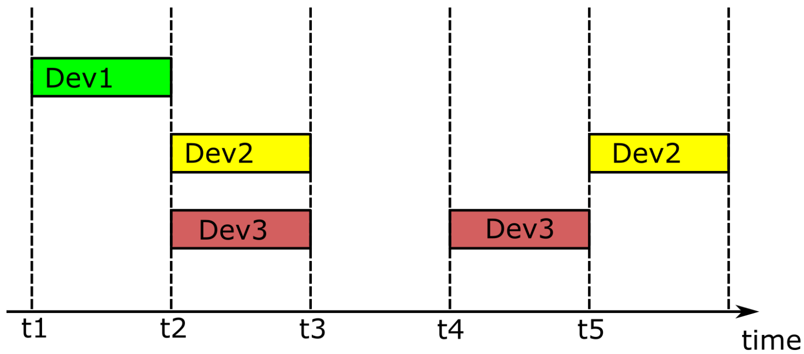

Figure 2.6: Timing diagram of the Slotted-ALOHA protocol.

these messages aren't interfering with each other. For Slotted ALOHA this is no more possible which has a positive effect on maximum message throughput.

2.2.2 Throughput Rate calculation

The big difference between Pure ALOHA and Slotted ALOHA is the so called "vulnerable period". This means the time span where no message transmission may start for not interfering with an ongoing transmission. Let's assume the all messages have the same length and are transmitted randomly. Also, an overlapping of two message leads automatically to a collision where both of the two messages need to be retransmitted [5].

2.2.2.1 Pure ALOHA

Figure 2.7 shows the definition of the vulnerable time "t_vol". The earliest time the interfering message may start is $t\_start - t\_msg$, this means $t\_vol = 2 \cdot t\_msg$, where "t_msg"is the transmission times span of a message. Equation 2.18 shows the definition

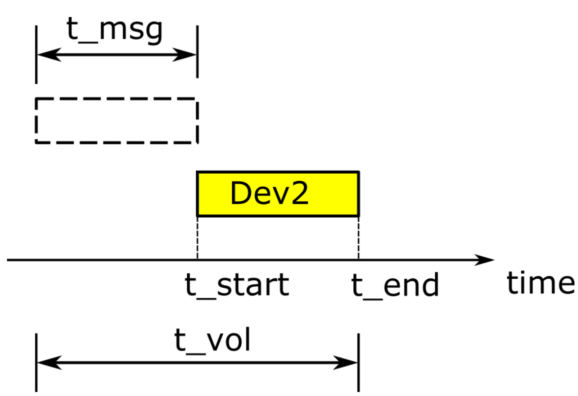

Figure 2.7: Timing diagram of the vulnerable time.

of the throughput S. G is the number of messages that are transmitted per second, $Pr\{no\ clollision\}$ is the probability that no message is transmitted during the vulnerable period.

$$S = G \cdot Pr\{no\ collisions\} \tag{2.18}$$

Assuming the messages are transmitted based on a Poisson distribution, the likelihood of messages received in a given time is shown in equation 2.19.

$$Pr\{k\} = \frac{\lambda^k}{k!} \cdot e^{-\lambda} \tag{2.19}$$

λ is the expected number of messages during a given period based on the number of messages transmitted per second, k is the number of messages that should be transmitted during the given period. The expected number of messages transmitted during the vulnerable period λ is $2 \cdot G$. For the case that no collision shall occur, the number of messages that shall be transmitted during the vulnerable period λ is 0, this means $k = 0$. The probability of not getting a collision is shown in equation 2.20.

$$Pr\{no\ collision\} = \frac{(2 \cdot G)^0}{0!} \cdot e^{-2 \cdot G} \tag{2.20}$$

This means, the throughput S of a given expected transmission G per vulnerable period can be calculated by applying equation 2.21.

$$S = G \cdot e^{-2 \cdot G} \tag{2.21}$$

This function has global maximum for $G = 0.5$, the calculation of the maximum throughput S_{max} is shown in equation 2.22.

$$S_{max} = \frac{1}{2} \cdot e^{-1} \approx 18.4\% \tag{2.22}$$

2.2.2.2 Slotted ALOHA

All the assumptions made for the Pure Aloha protocol also hold for Slotted ALOHA. The only difference is that the vulnerable time for slotted aloha is halved because a collision only occurs if two or several transmissions take place in the same slot. This influences the factor $\lambda = G$ which increases the maximum throughput rate. Equation 2.23, 2.24 and 2.25 show how the collision likelihood, the throughput calculation and the maximum throughput have changed.

$$Pr\{no\ collision\} = e^{-G} \tag{2.23}$$

$$S = G \cdot e^{-G} \tag{2.24}$$

$$S_{max} = e^{-1} \approx 36.8\% \tag{2.25}$$

The comparison of the throughput calculation results shows that Slotted ALOHA has a speed up by the factor of 2 in comparison to Pure Aloha. Nevertheless, the infrastructure/system cost of a Slotted ALOHA system are bigger because the communicating devices need to have reference time for their slot generation which can also lead to a higher power consumption.

2.3 Wireless Personal Area Network - Media Access Control

The Wireless Personal Area Network (WPAN) Media Access Control (MAC) is defined in the IEEE 802.15.8 Standard [6]. It is needed for having a specification of the following networking mechanism:

- Device synchronization
- Device discovery
- Establishing of a communication channel
- Providing broadcast, multicast and unicast infrastructure
- Definition of channel access mechanisms

For providing these mechanisms a new frame structure called "Superframe structure" is needed.

2.3.1 Superframe Structure

A Superframe consists of different phases, each of this phases has got a dedicated time frame. Figure 2.8 shows the different phases of a Superframe.

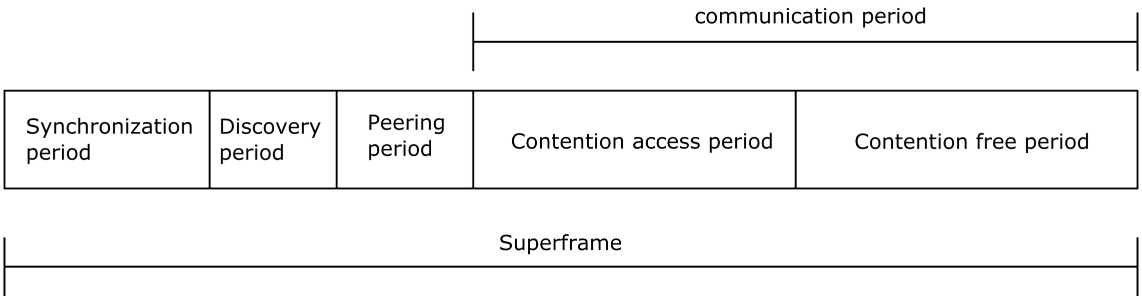

Figure 2.8: Schematic drawing of a Superframe according to IEEE 802.15.8 [6].

During the *Synchronization period* the initial network synchronization takes place. This frame is needed for having a common time basis in the network. All the later communication uses the synchronization timestamp as reference. It is based on random-access mechanism for initial network synchronization.

Immediately after the network synchronization has finished, the device discovery starts. There are two types of discovery, the One-way discovery and the Two-way discovery. In the One-way discovery a device either shows its presence to other devices by transmitting its discovery information or it receives the discovery information of other devices. Two-way discovery means that a device can request the discovery information of other devices or device groups by transmitting a "Discovery Request" including its on discovery information.

The *Peering period* starts directly after the device discovery. Peering means a procedure to setup a communication link between devices that have been discovered in the device discovery phase. The MAC supports one-to-many, one to one and many-to-many peering.

The *Contention access period* is like the previous periods also based on a random-access mechanism. During this phase management messages are transmitted, including data packets and discovery management packets.

The *Contention free period* performs communication without any previously needed random-access schemes because the communication slot negotiations have already taken place.

2.3.2 Cyclic-Superframe

A Cyclic-Superframe (CSF) consists of a periodic sequence of Superframes (SF). The components of the SFs within the CSF can be active or inactive. The only part of the SFs that must be always active is the Synchronization period. This leads to $2^4 = 16$ possible active and inactive combinations of SF components. A CSF consist out of two different SFs also known as pattern A and pattern B, where each pattern can be one combination out of the 16 possibilities. A CSF can contain up to 4096 SFs which need to consist either out of pattern A or B. If the CSF size is 1 only one SF pattern is used. Figure 2.9 shows

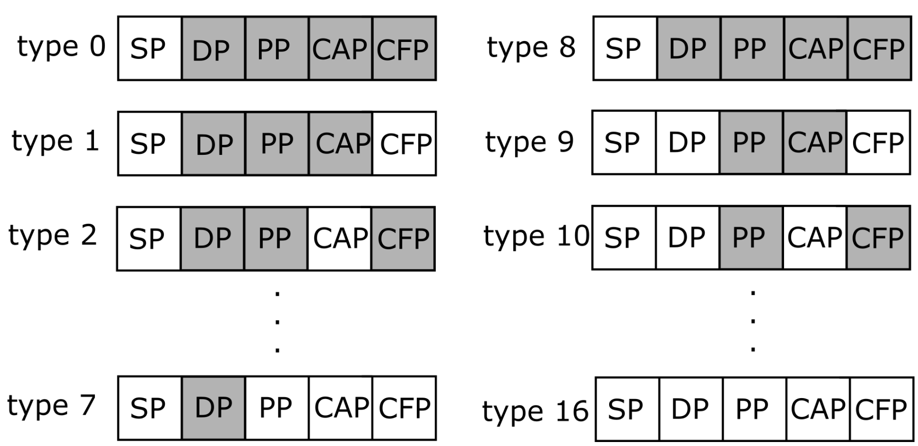

Figure 2.9: The possible combinations Superframe active and inactive periods.

how a a SF can be configured. All periods up to the Synchronization periods represent one bit of the type. The Contention free period (CAP) is the least significant bit and the Discovery period (DP) is the most significant bit. If a period is inactive the bit is 0 and the frame is marked grey.

Figure 2.10 shows the MAC consisting out of a periodic sequence of CSFs. As previously mentioned, each CSF consists of two patterns named pattern A and pattern B. Pattern A and pattern B are repeated n and m times, n and m can be chosen independently from each other until the sum of both frames is smaller than 4096. A pattern consists of a SF with active and inactive periods for power saving goods. In an inactive period the transceiver can be turned off so the transmission/reception power is saved.

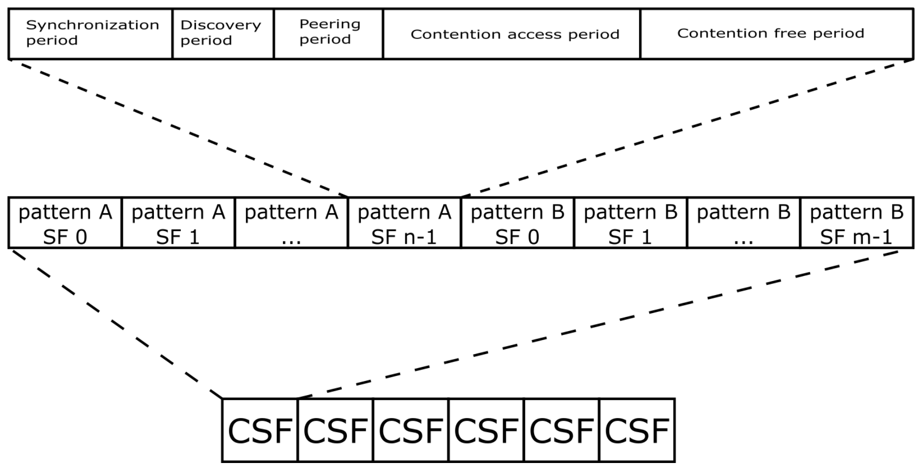

Figure 2.10: The structure of a Cyclic-Superframe consisting out of patterns.

2.3.2.1 Ultra-Wideband Physical Layer

The physical layer (PHY) of the UWB technology is specified in the IEEE 802.15.8 [6]. According to the standard the UWB has two different modulations modes the BPM-BPSK and the OOK modulation. The operating channels are between $3.1\,GHz$ and $10.6\,GHz$. In this section only the BPM-BPSK modulation frame will be explained since it's the more common data encoding method. Figure 2.11 shows a UWB Frame according to the IEEE 802.15.8 standard. The first part that is transmitted of a UWB-frame is the

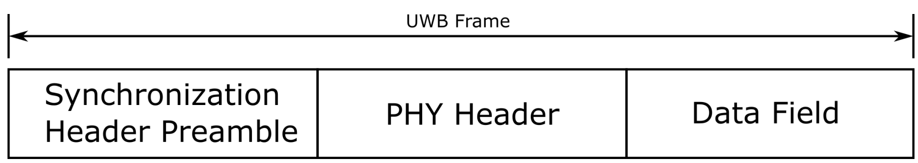

Figure 2.11: The different sequences of a UWB-frame.

synchronization header (SHR) preamble. This SHR preamble consists of a SYNC frame and a Start of Frame Delimiter (SFD). The SYNC frame can consist out of 64 to 4096 symbol repetitions where the same symbol is repeated during the whole SYNC frame. The number of symbol repetitions that is used is depending on the environment. A harsh environment requires a higher number of symbols repetitions because this makes the preamble is easier to detect. A symbol is a sequence of codes of the ternary alphabet (-1, 0, 1). The codes of a symbol are generated in a way that they have a perfect periodic autocorrelation property, so a physical channel can be split up into 48 virtual channels based on the symbols and their spreading sequence [6].
The SFD is sent after the SYNC field and marks the end of the SHR that is followed by beginning of the PHR. Also, the SFD is needed for getting a marker for the timestamp of the UWB Frame.

The Physical Header (PHR) information is needed for a successful decoding of the rest of the UWB data also known as PHY service data unit (PSDU). The PHR consists of 19 bits and contains the following information:

- The SHR preamble length
- The length of the PSDU payload
- The PSDU data rate
- Six bits used as parity check information for detecting channel errors

The PHR is Reed-Solomon encoded and is transmitted with the same data rate as the PSDU.

The data field can have a length of up to 1023 bytes and can be transmitted with data rates ranging from 110 kb/s to 27.24 Mb/s. It is first Reed-Solomon (RS) encoded with additional 48 parity bits. Then this data is convolutional encoded before it's modulated on the carrier. The PSDU frame contains the data that is needed for the UWB MAC protocol.

2.3.2.2 Ultra-Wideband Superframe

The UWB technology has different properties from timing and collision detection perspective compared to narrowband technologies. Because of this, the IEEE 802.15.8 standard has defined a different Superframe structure for UWB based WPAN.

Figure 2.12 shows the Superframe structure that shall be used according to the IEEE 802.15.8 standard [6].

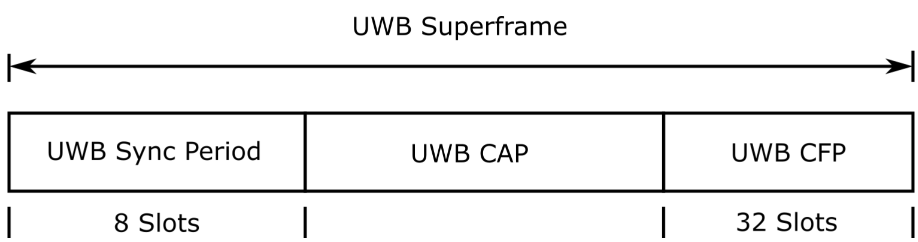

Figure 2.12: The Superframe structure that is used for the UWB MAC.

The UWB Superframe only consists of three periods:

- UWB Sync Period
- UWB Contention Access Period (CAP)
- UWB Contention Free Period (CFP)

The difference between the UWB Superframe to the narrowband Superframes is that there is no more a peering and device detection period. Also, all periods up to the UWB CAP are slot based because the CSMA-CA MAC scheme that is used for narrowband technologies is not feasible for UWB technologies. The UWB Sync Period consists of 8 slots, each of these slots is again separated in 4 possible starting points. Table 2.1 shows the 4 possible starting points in a UWB Sync slot. The different starting points are needed for getting a increase of the throughput, because if several devices would start at the same slot on the same channel a collision would occur. If two devices are sending with a delay of

Start time delay code	delay value in preamble symbols
0b00	0
0b01	2.25
0b10	4.5
0b11	6.75

Table 2.1: The possible delays in a slot of the Sync Period.

e.g. 2.25 symbols the first device will be heard because the correlation unit of the receiving devices locks on the first few symbols. This means, a reduction of the collision likelihood by approximately the factor 4. The total length of the slot is only increased by 7 symbols which is an acceptable tradeoff since each slot has at least the length of 128 symbols.

The CAP is based on a random-access scheme because a CSMA-CA scheme is not feasible for the UWB PHY. The CAP is used for initiating the data communication and the ranging that takes place during the CFP. Also peering takes place during the CAP. Since CAP is based on random-access without listening on the channel, there are no dedicated time slots for the starting point of the communication.

The ranging takes place in the previously agreed slots of the CFP, where each slot has a size of $2.25\,ms$ [6]. During this slot a ranging/data communication between the devices of the WPAN can take place. Since the CFP has got only 32 slots and also data communication takes place during this period, only a limited number of ranging can be made during the CFP [7].

2.4 OMNeT++ Simulation

OMNeT++ is a network simulation platform, that can be used, extended and adapted for free by changing the source code files. The platform is based on discrete event based object-orientated environment. OMNeT++ can be used for a detailed simulation of wired and wireless communication networks. The simulation possibilities reach from protocol modeling to simulating multiprocessor systems with distributed hardware [8]. OMNeT++ is not a simulation program itself, instead it provides an infrastructure that can be used for setting up a network inclusive environment that can be evaluated. The models of the network components can be reused and also extended if needed. Each sub-module of a component has got several gates that can be uses as a component interface. The behaviour of these gates can be configured individually which allows for a deep simulation depth. Such a gate has a predefined path that can be used for simulating a signal travelling from the source to the destination. This path can be used e.g. for simulating a wireless bidirectional channel. OMNeT++ also supports parallel simulation of multiple distributed devices. This allows the simulation and detection of collisions, e.g. if two devices are communicating at the same time on the same channel this will be detected and an erroneous communication and an event is triggered [9].

2.4.1 OMNeT++ IEEE 802.15.4 Simulation Model

Also, a big advantage of OMNeT++ is that a protocol stack can be implemented very fast by reusing components of protocols that already exists. Figure 2.13 shows the protocol stack of an OMNeT++ simulation that has been used for implementing the IEEE 802.15.4 standard that is very common for sensor networks [10].

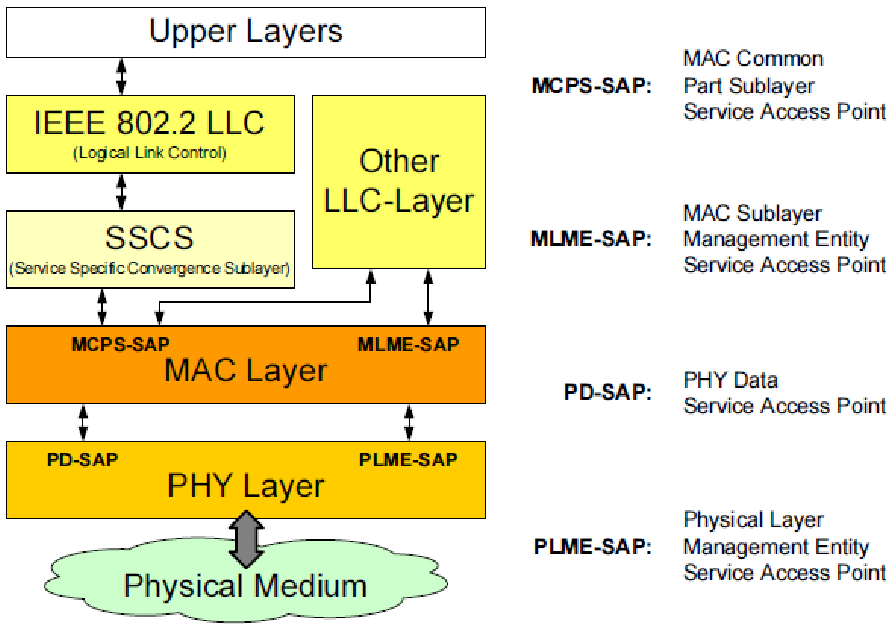

Figure 2.13: Protocol definition with interfaces.

In Figure 2.14 there are several Layers, the "PHY Layer" is used for the data transmission, it manages the transceiver and selects the frequency band of the configured channel. Such a "PHY Layer" also reports if the carrier/data of an external device is detected and only activates a frame transmission if the channel is free. This check is needed for a "Carrier Sense Multiple Access - Collision Avoidance" (CSMA-CA) access schemes which is implemented in the "MAC Layer". The "MAC Layer" is also responsible for encapsulating the data coming from the higher protocol layers. If data is received by the "PHY Layer" it is forwarded to "MAC Layer" where it is then processed and again forwarded to the higher layers. In this example the UWB Protocol is implemented according to IEEE 802.15.4 in the "Service Specific Convergence Sublayer". This layer can be seen as a kind of wrapper for the IEEE802.2 and allows to reuse and adapt the standard so that it matches for the IEEE802.15.4. The application is running on the "Upper Layers", this includes also the UWB ranging protocol and the power management e.g. the duty cycling ratio.

2.5 Channel Interferences

For a multi-node RF communication system, channel interferences are a limiting factor for the maximum throughput that can be achieved. In principle interferences can be split up in two groups, the noise caused by the environment that is present over all frequency bands and in the channel interference. The environmental noise is decreasing the Signal to Noise Ration (SNR). The SNR should be as big as possible, so filter circuits need to be applied for filtering the Noise before it reaches the sample unit. An in-band disturber can't be filtered that easy, especially if it uses the same physical and logical channel. Figure 2.14 shows a Matlab simulation of a multiple access scenario where two signals are transmitted at the same time.

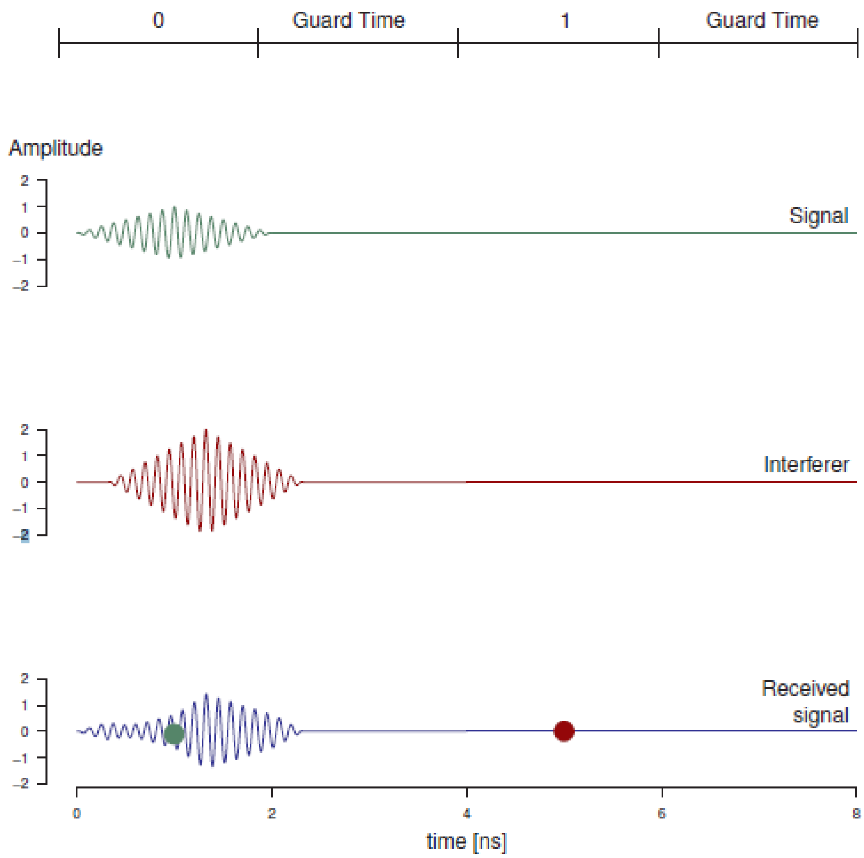

Figure 2.14: Destructive in-band signal interference.

The data is encoded by pulse positions, if the pulse occurs at the "0"-slot, it is interpreted as 0, if the pulse occurs at the "1"-slot it is interpreted as a 1. The guard time is needed for decreasing the number of wrongly interpreted data caused by timing variations. In the graphic there are two independent traces plotted. The trace of the wanted signal is called "Signal"the trace of the interfering signal is called "Interferer". The Interferer can be either another active device or an reflection that is cause by the environment. The trace called "Receive signal"is the signal that an RX station would receive if the "Signal"and the "Interferer"are transmitted at the same time. Such a scenario can happen if e.g. the Line of Sight (LoS) of a signal is blocked and there is an indirect path with a good reflection. In this scenario the phase difference of the interference is big enough to make a destructive interference which would have a negative impact on the package error rate. A Matlab based simulation of the signals has the advantage that the channel of the transmitted signal can be simulated very accurate which can include multiple interference sources that can be within or outside of the UWB-channel [11]. An in-channel interference can be stronger than the original signal depending on the properties of the interfering devices. Depending on the signal strength, the number of overlapping frames and the data encoding an overlapping of UWB-frames can increase the package error rate (PER) drastically [11]. The impact of multiple parallel communicating devices on the PER should be considered during simulations and the protocol development for getting more meaningful data. The disadvantage of such a detailed simulation model is the impact on the simulation speed. Depending on the network size the simulation run-time can be increased drastically by making such a detailed simulation.

2.6 MiXiM based UWB Simulation

MiXiM is a modeling framework that can be integrated into OMNeT++. It was created for simulating wireless networks with dynamic or static channel properties. The framework offers detailed models of wave propagation and interferences. Also, several MAC protocols are implemented in the MiXiM Framework.

Figure 2.15 shows the Architecture of an OMNeT++ simulation where MiXiM is used for simulating the UWB PHY. The UWB PHY is implemented in the PhyLayerUWB class

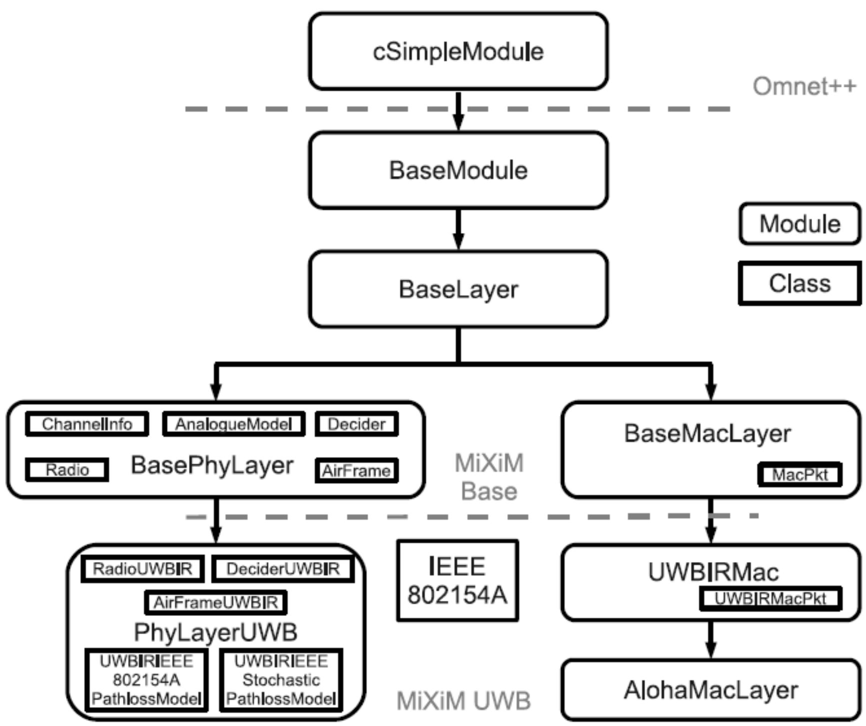

Figure 2.15: UML diagram of a MiXiM UWB class.

that has got several members for simulating the path loss. The "UWB-IR IEEE 802.15.4A PathlossModel" is used to simulate the static path loss according to the IEEE Standard

[12]. Since the IEEE model is very detailed, a high computational power is needed for considering all the channel parameters. Out of that reason there is another channel model called "UWB IEEE Stochastic PathlossModel". This model is based on statistics and is much easier to handle which improves the simulation timings. The UWB radio inclusive the power consumption is simulated in the "RadioUWBIR"class. This class has 4 internal states: "Idle", "Receive", "Transmission"and "Switching". This states are important for simulating the power consumption in more details because UWB is depending on the state a very power-hungry technology. Since many UWB devices are based on a half-duplex RF technology, receiving and sending in parallel is not possible. The Switching state is needed for simulating the power consumption when the devices switches from the receiving to the sending state. The Idle state is important for having a kind of standby power consumption when the device is neither sending nor receiving. Also this states are important for defining the requirements of the external supply.
The big advantage of a MiXiM based Omnet++ UWB device model is the layer setup that can be modified easily. E.g. the "AlohaMacLayer"in Figure 2.15 could be exchanged with another already existing slotted protocol for making power consumption analysis because of the detailed device models.

2.6.1 OMNeT++ Limitation

The grade of detail OMNeT++ simulations are working with, might be an advantage for the most small size network simulation, but for big networks with hundreds of nodes the OMNeT++ starts crashing. An UWB passive key access scenario made of cars consisting out of 4 anchors per car lets the OMNeT++ simulation crash after adding 20 cars plus the corresponding keys. Out of that reason, for simulating big networks a different layer of abstraction needs to be considered.

Chapter 3

Design

This chapter gives an overview about the different design decisions that could be considered based on the related work and the Target use case.

3.1 Design requirements

The Simulation platform should be usable as a tool that can be used for an analysis of UWB based localization systems. The platform shall have following properties:

- Realistic network simulation based on the IEEE standard models
- Possibility to scale the system size
- High-level and low-level simulation of the devices in the system
- External (non-UWB) channels can be used as backbone network
- Power analysis of network nodes
- Modularity

3.1.1 Scalability

Scalability is an important property for simulation in general, the more detailed a simulation is the higher is the computational power that is needed. This can lead to an infrastructural bottle neck and a crash, just because the needed resources for the simulation could not be allocated. Scalability can be limited by following factors:

- Computational power
- Memory consumption
- Simulation Time

The computational power is required in principle for any simulation, the bigger the network gets, the higher is the computational effort. Depending on the simulation approach, the computational power that is required can have a different dependency with the size of the

system that is simulated. Especially for network simulation the computational power that is required can scale exponentially if the simulation model was not chosen properly.
In the most simulation approaches, there is a tradeoff between required memory and the computational power. For example, a simulation model that stores all the previously made simulation might be good from computational power perspective because many results are cached and can be reused, but therefore the required memory will increase continuously. Also, it needs to be considered that given physical memory borders exist, e.g. memory limitations from speed perspective. The bigger the amount of the stored data, the bigger is the required time for a recall. Additionally, bigger sized memory is much slower than smaller size memory, every data that can be stored for a short period in the cache is an advantage from run time perspective. Since fast cache memory is very expensive and very limited, a good caching algorithm is needed for keeping data as global as possible and still as local as needed. Data that is used very frequently could be also forced to be stored in very fast memory e.g. CPU cache if the compiler isn't already doing so.
For keeping the simulation time small, either the simulation-based algorithms could be improved for reducing the number of calculations, or speed of the system needs to be increased. So, there is also a tradeoff between the hardware afford and the timing. If a simulation doesn't run fast enough on a single core, it can be executed on several cores in parallel which will result in a speed up. For a high grade of parallelism, it needs to be checked that there is no long critical path in the algorithms that can be only handled by a single core because of the data dependencies, because only when every task has finished also the simulations finishes. If the computational power of a single system is no more sufficient, clusters networks can be used for distributing the work load between several systems. The advantage of a cluster in comparison to a server is that several distributed terminals can be used instead of central server. This means, the terminals are only offering the rest of their available computational power that is currently not needed by the user. So, the already available hardware can be used without increasing the system costs.

3.1.2 Evaluation of Communication Protocols

One of the central tasks of a network simulation is an evaluation of different communication protocols. UWB protocols can be benchmarked based on the following properties:

- Maximum data throughput
- Power consumption of the infrastructure
- Number of nodes that can be tracked
- Power consumption per node
- Power consumption of the nodes
- Requirement of additional RF PHYs
- Average message count that is needed for tracking a node

For testing and evaluating different communication protocols, the simulation platform should be as modular as possible. Figure 3.1 shows a possible setup of the layers for a

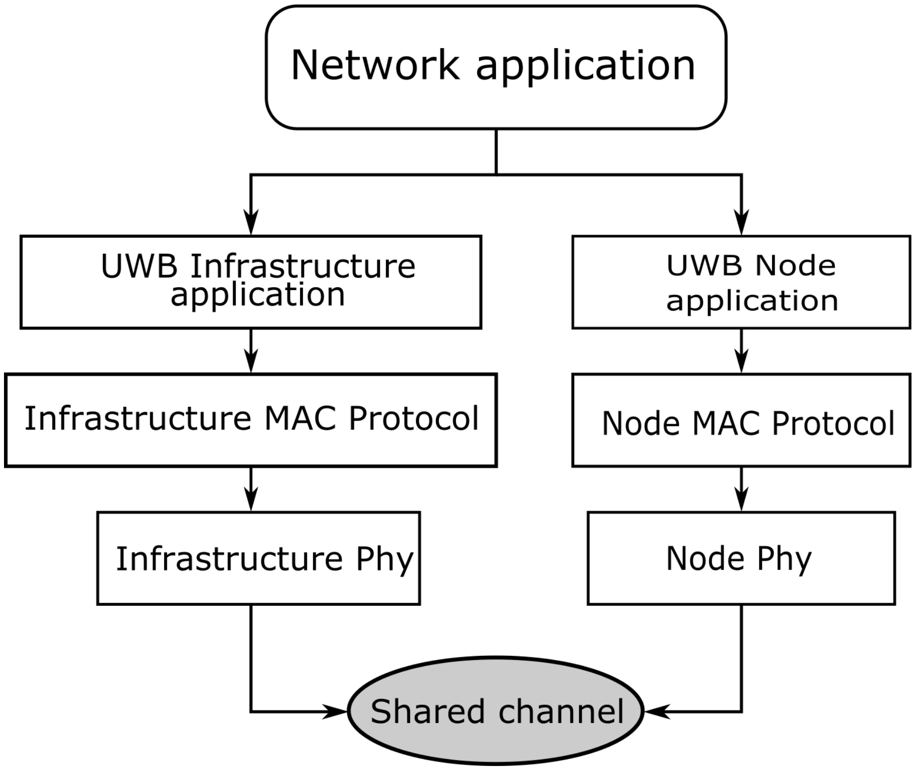

Figure 3.1: Possible layout for Protocol evaluation.

modular exchange of the Networking Protocol. On top of all UWB use cases there is a network application that specifies the required data communication and ranging that needs to take place for fulfilling the requirements. Since and UWB network can consist of multiple devices of different types, these different types need to be considered. An infrastructure device has different properties than a low power network node. This means, also, the device application needs to be adapted accordingly. In Figure 3.1 there are only two device applications symbolized, a node application and an infrastructure application. Basically every different device type in a network needs its own device application for dealing in a proper way with its interfaces and limitations. The following properties of a devices should be considered for developing the device application and the MAC protocol implementation:

- Power supply
- RX gain
- TX output power
- Additional interfaces
- Computational power

Most Infrastructure will have a wired power supply, this means they are not limited in their power consumption. Also, infrastructure can have a wired interface to a central unit that collects all the information for a later post processing, this means also a high potential computational power. An UWB node can be depending on its application very limited in power and size, this needs to be considered in duty cycling ratio of the RF interfaces and also in the RX gain and TX output power. The bigger a device can be the bigger antennas can be used, this means bigger devices will have a better RX gain because of the antenna design. Also the TX output power can be limited if the antenna design was not properly chosen. Even if there are FCC regulations on the output power, many devices are not able to reach this boarder because of the weak output drivers. Also, low power and small size designs will be very limited in their additional interfaces, because every additional device adds an additional power consumption and connector to the system. The physical layer is responsible for the data modulation which means all the communication passes this layer. Out of that reason, the power consumption of the RF interface should be measured in the physical device layer e.g. the "Node Phy"and the "Infrastruture Phy"in Figure 3.1. The computation overhead of the application and protocol can be measured in the corresponding layers. This means, if a application or protocol is changed the impact on the power consumption can be directly measured because of layer internal measurement which can be later used for a benchmarking. All devices in the network may have different PHY, MAC, and application layers, but they all use a common channel which is symbolized as "Shared channel"in Figure 3.1. This channel is responsible for combining all signals generated by the devices PHYs. Also, the signal propagation and interferences need to be considered in the channel. Especially for a UWB system a channel model can become very complex because of the high time resolution of the technology. Figure 3.2 shows an

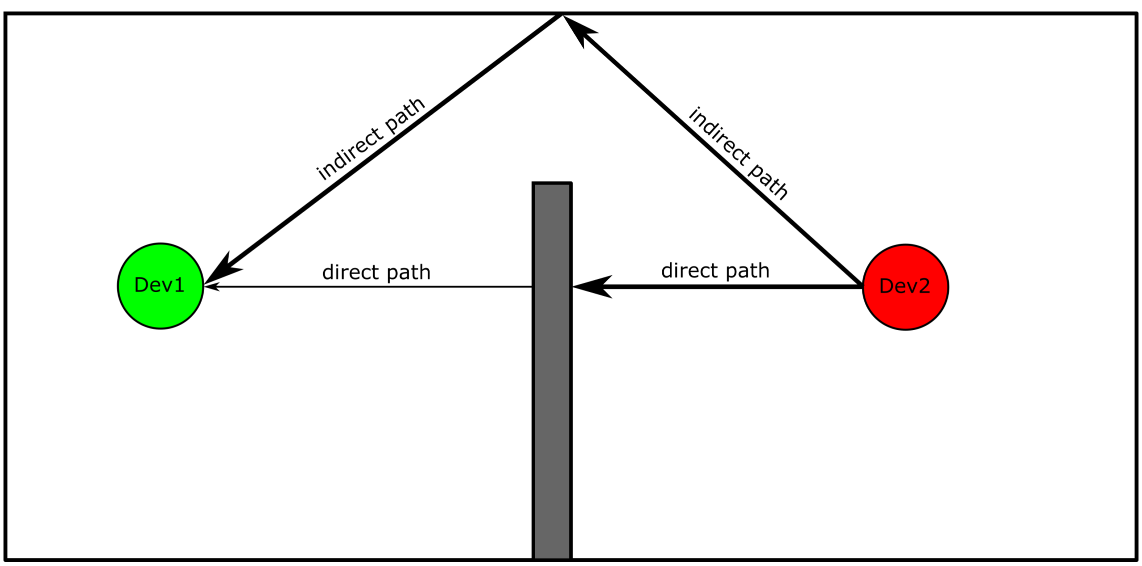

Figure 3.2: Example of a multipath UWB-channel.

simple example of two UWB devices named "Dev1"and "Dev2"in a room communicating with each other. Dev2 is sending a message to Dev1, an obstacle between these devices is blocking a big part of the signal. The room has good signal reflection properties. The direct path signal from Dev2 to Dev1 is damped. This means, the first path of the signal received by Dev1 is very weak. Since the room has very good reflection properties, the indirect path reflected by the wall is nearly not weakened and out of that reason stronger than the first path. For a narrowband technology this indirect path would lead to an advantage from SNR perspective because it is stronger than the first path and out of that reason the probability of receiving a good signal is higher than with the direct path only. But since the UWB technology is used for distance measurement and additional indirect path of the signal can have a big impact on the measurement. The TOF measurement generates the timestamps on the first path that is detected, if the direct path its much weaker than the indirect path the receiver of Dev1 won't be able to receive it because of the AGC, this means the indirect pact will be measured as the first path. Out of that reason the measured TOF will be the signal propagation time of the indirect path which leads to an erroneous measurement. Even if the direct path can be detected and the indirect path arrives just a short time later, the receiver won't be able to distinguish between both signals and only one signal will be detected. For a sampling frequency of $500\,MHz$ the minimum delay between two signals needs to be at least $1/500\,Mhz = 2\,ns$ for detection two independent signals. Out of that reason, the network environment needs to be considered in the simulation.

3.1.3 Evaluation of Measurement Methods

Beside of the communication protocol, also different measurement methods can be evaluated in the network simulation. The most common ranging techniques are:

- Single Sided - Two-Way Ranging
- Double Sided - Two-Way Ranging
- Time difference of arrival (TDOA) measurement
- Spy Ranging
- Asymmetric Ranging

The ranging method can be implemented in a sub-layer of the device application, since every technique requires a different message set and timing, also the Mac Layer needs to support the ranging method and its corresponding message exchange. Some very generic MAC protocols won't be able to handle every ranging Method. Out of that reason e.g. the Asymmetric Ranging won't be applicable with the ALOHA protocol at least not without a big impact on the ranging error. The ranging protocols in combination with a communication protocol can be benchmarked under consideration of the following aspects:

- Ranging error rate
- Localization accuracy
- Localization throughput
- Message count per localization
- Infrastructural requirements

The ranging error rate can be seen as reference value for the average error rate of ranging method by using a given protocol. E.g. a SS-TWR is a very simple, two message based ranging method that is erroneous if at least one of the two messages are disturbed. This makes the SS-TWR very robust in comparison to an Asymmetric Ranging that can consist dozens of messages and also crashes if at least one message is disturbed.
Different ranging method will have a different accuracy especially if the ALOHA protocol is used more complex ranging mechanism will have an increased stochastic measurement error which leads to a worse localization accuracy.
The localization throughput benchmarks how many nodes can be tracked with a given ranging method and protocol in a given time frame.
The message count per localization benchmarks the ranging method itself by considering the channel capacity. This means, the lower the message count per localization is, the better is the method for networks with a high node count.
Not every ranging method can be applied by using an UWB only system. This means, also the required synchronization infrastructure needs to be taken in account for more complex ranging localization techniques e.g. the TDOA measurement.

3.1.3.1 Asymmetric Ranging

Figure 3.3 shows the message exchange of an Asymmetric Ranging between a device called "Initiator"and two responding devices [13]. The Initiator sends out a poll message that is received by devices named "Responder1"and "Responder2". Both devices belong to a device group, which means, the poll message is also meant for both of the devices. This means, also both devices are responding to the poll message. For avoiding a collision, there is a given response order. In the example of Figure 3.3, Responder1 sends out the first response message, and Responder2 sends the second response message after a pre-defined guard time for avoiding clock-drift based collisions. When that time has passed, the last group message should be sent, the Initiator responds with the final message. By exchanging these messages DS-TWR sessions have been performed, one between the Initiator and Responder1 and another between the Initiator and Responder2. In comparison to two conventional DS-TWR sessions, only 4 instead of 6 messages have been send, this has the

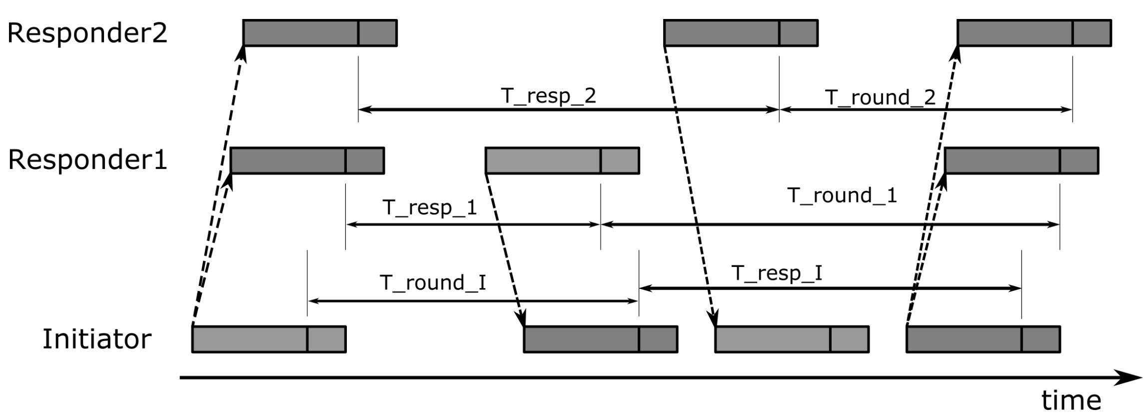

Figure 3.3: The timing diagram of the Asymmetric Ranging.

advantage of reducing the channel consumption. Equation 3.1 can be used for calculating the number of needed messages for preforming a ranging between a given number of initiators and responders.

$$N_{msg} = (2 + N_{responders}) \cdot N_{initiators} \tag{3.1}$$

Equation 3.2 shows the number of messages that is needed for ranging between a given number of initiators and responders by using DS-TWR.

$$N_{msg} = 3 \cdot N_{initiators} \cdot N_{responders} \tag{3.2}$$

The comparison of the required number of messages between the Asymmetric Ranging and the DS-TWR shows, that the message ratio for a big number of responders is converging to 1/3. The Asymmetric Ranging has also got disadvantages, the biggest one is, that the measurement accuracy decreases by an increasing number of responders in the network. Also, the responders need to know their response time slot. This requires either an additional communication interface or additional messages on the UWB interface if the response order is not given in the program, which is hard to achieve for a dynamic network of anchors.

3.1.3.2 Spy Ranging

Spy Ranging (SR) consists of a SS-TWR, this means, it has the same message count as a SS-TWR. The only difference is that all active anchors that are placed in the room are also listening to the ranging session. Figure 3.4 shows the message exchange of a SR session between one node and two anchors. The node named "Node"is the initiator of the ranging session, the master anchor named "Anchor M"is the responder. The slave anchor named "Anchor S"is just listening to the ongoing communication and does not participate on the SS-TWR between the node and the master. Since the position of anchors is static in the room, also the TOF between the anchors is constant until there is no blocking object added between the anchors. This means, the TOF between the anchors can be measured once and stored until the room changes. TOF_M_S is the variable that symbolizes the TOF between the anchors of the SR. When the slave anchor receives the poll message sent by the node, it stores the timestamp $t\_rec\_poll\_S$. The timestamp $t\_rec\_resp\_S$ is stored when the slave anchor receives the response message, transmitted by the master anchor. The timestamps of the node and the master anchor are stored analogous to the responder and initiator timestamps of a SS-TWR. Based on the timestamps equation 3.3 can be formed.

$$t\_rec\_resp\_S - t\_rec\_poll\_S + TOF\_N\_S = TOF + T\_resp\_M + TOF\_M\_S \tag{3.3}$$

The variables TOF and T_resp_M can be generated based on the master anchor and node timestamps. By combining all the timestamps equation 3.4 can be formed.

$$\begin{aligned} t\_rec\_resp\_S - t\_rec\_poll\_S + TOF\_N\_S = {} & T\_resp\_M + TOF\_M\_S + \\ & \frac{(t\_rec\_resp\_N - t\_send\_poll) - (t\_send\_resp - t\_rec\_poll\_M)}{2} \end{aligned} \tag{3.4}$$

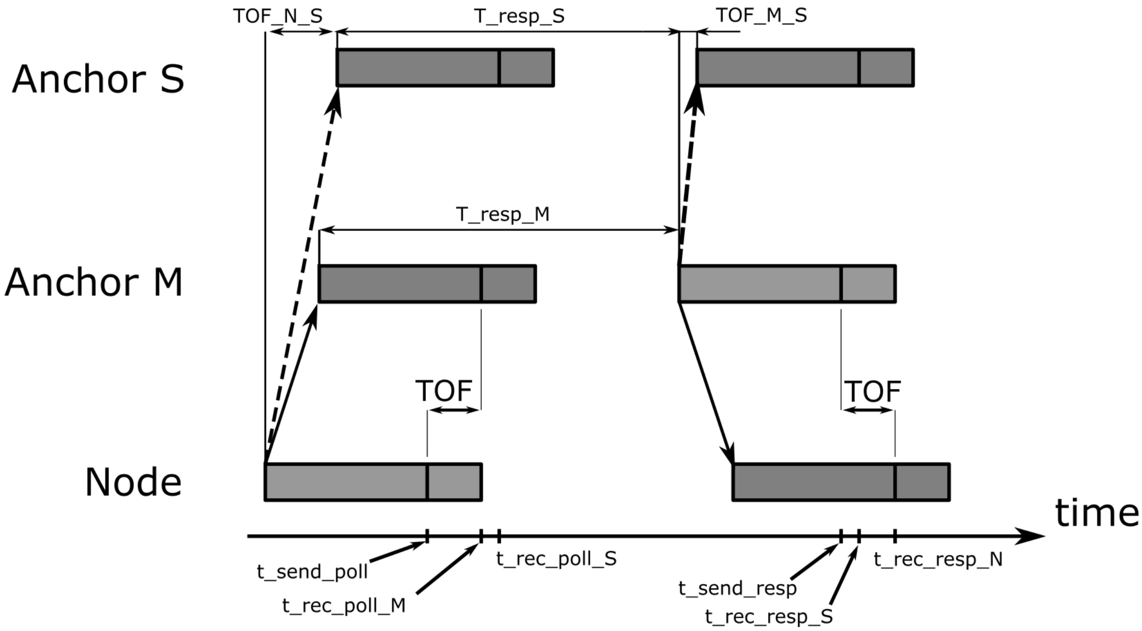

Figure 3.4: The timing diagram of the Spy Ranging.

By reforming equation 3.4 the TOF between the node and the slave anchor can be calculated wich is shown in equation 3.5.

$$\begin{aligned} TOF\_N\_S = T\_resp\_M + TOF\_M\_S - t\_rec\_resp\_S + t\_rec\_poll\_S \\ + \frac{(t\_rec\_resp\_N - t\_send\_poll) - (t\_send\_resp - t\_rec\_poll\_M)}{2} \end{aligned} \tag{3.5}$$

The advantage of the Spy ranging in comparison to SS-TWR is, that the node only needs to perform one SS-TWR to one anchor of the network instead of ranging to all the anchors of the network. This means, the number of exchanged messages is linear to the number of nodes in the network and independent to the number of anchors. This property is important from a power consumption and from a channel capacity perspective. For multi-anchor networks the number of messages is decreased significantly in comparison to the SS-TWR. If the clock-drift needs to be compensated, this can be either done based on the preamble, or by using a SDS-TWR for the ranging.

3.1.3.3 Time Difference of Arrival Measurement

The Time Difference of Arrival (TDOA) measurement is a one message based localization technique, that uses the different arrival times of a single message for estimating the position of its sender. Figure 3.5 shows the message exchange of a TDOA measurement. The "Initiator"sends a broadcast that is received by all the static devices, also known as anchors. Based on the different positions of the anchors, the message propagation time changes. So the anchor that is the closest to the Initiator will receive the message first. In Figure 3.5 "Anchor1"is closer than "Anchor2"to the node, the time that passes from the message reception of Anchor1 to the message reception of Anchor2 is called TDOA. The position of the initiator by setting up an equation system based on the position of the

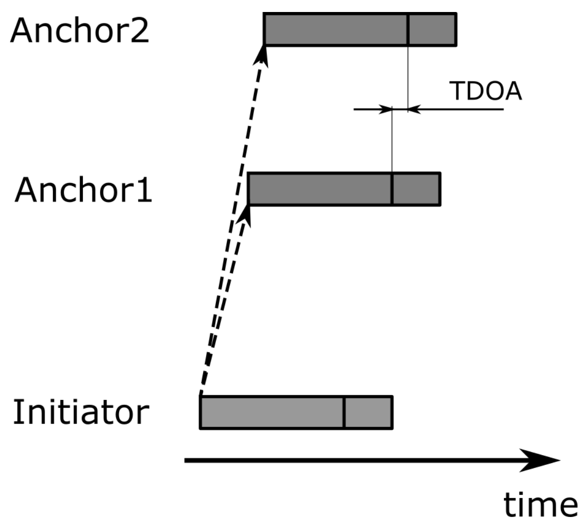

Figure 3.5: The timing diagram of the Time Difference of Arrival measurement.

anchors and the position of the node. Such an approach is named multilateration [14] and is a common technique that is also used for global positioning systems (GPS) [15]. For calculation the 3D position of a device, at least 4 anchors are needed because the arrival time of one anchor is used as reference timestamp. For a TWR based localization system 3 anchors are sufficient for calculating the 3D position of the initiator. Additionally, to the increased amount of anchors, the TDOA measurement requires a common clock since it is based on differences in the arrival time. This common clock can be either achieved by having a wired clock reference or by using a wireless synchronization scheme. Both approaches lead to additional infrastructure/development cost and increase the system complexity. Another disadvantage of the TDOA measurement is that it hasn't got a replay attack prevention. This means, the broadcast transmitted by the initiator can be blocked and repeated when a position needs to be faked. This can lead to a problem for systems where a higher security is required.

3.1.4 Simulation of dynamic Channel influences

Beside to the static channel parameters mentioned in section 3.1.2, there are also dynamic channel parameters that need to be taken in account for a UWB network simulation. Figure 3.6 gives an example of dynamic influences on the UWB-channel. The Figure shows a central anchor name "A1"that is communicating with three nodes called "D1","D2"and "D3". The nodes are attached to persons called "Person1", "Person3"and "Person4". Additionally, there is a person named "Person2"in the room that has no attached node. For simplicity the static channel parameters, like the walls of the room, have been neglected in this example because they are explained in section 3.1.2. Depending on its age, a human consists of about 60 % water. This means, because of its size, the human body can have a big impact on microwave signals. Since the UWB signal has its carrier frequency between about $3\,GHz$ and $10\,GHz$, every UWB-channel can be influenced by the human body attenuation if the body is placed in the signal path. In the scenario shown in Figure 3.6 the node D2 has got a LOS path to anchor A1, this is the best-case scenario with the best

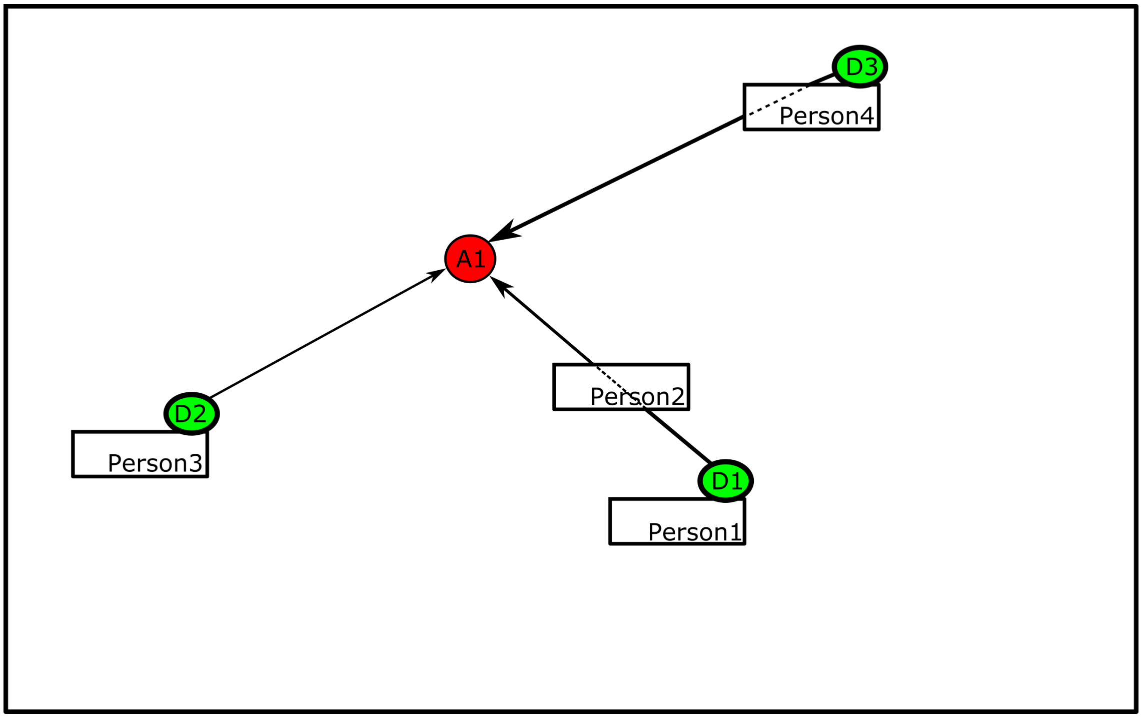

Figure 3.6: An example of dynamic channel influences.

measurement accuracy and SNR. The signal of D1 that is attached to Person1 is blocked by Person2. This means, that the LOS signal is damped by roughly $20dB$ compared to LOS scenario. As soon as Person2 moves out of the signal path, the LOS is unblocked and the measurement accuracy will be increased. If the scenario of D1 is really a LOS depends on the geometrics of the room, the higher the room is the bigger is the probability for getting a LOS signal. Depending on the height of the node the direct signal path will not hit Person2. Out of that reason, if the anchor density is the same, rooms with a high ceiling e.g. big halls, will have a better average SNR rate, in comparison to rooms with a low ceiling e.g. living rooms. The node D3 is block by Person4. This scenario will occur with a probability of about 50% if a node is attached to a person. Depending on the position of the node on the human body, about 50% of signal paths will be block. Also, this probability is very hard to decrease by the room height. Because of the proximity of the node to the human body the ceiling needs to be tens meters high for avoiding this effect, which has again an impact on the signal quality and the localization accuracy. The body attenuation D3 scenario is also about $20\,dB$.
Since the human body has such an impact on the UWB signal, it needs to be considered for a network simulation, especially because of the high likelihood of the occurrence.

3.1.5 Evaluation of Timing requirements

Depending on the measurement method the timing, when UWB-frames are sent, has a big impact on the measurement accuracy. Especially if it comes to more complex message exchanges like Asymmetric Ranging, the timing requirements need to be considered. Also, for more complex communication protocols, the inaccuracy of the clocks needs to be considered for the timeslot agreement.

Figure 3.7 shows an example where a central synchronization unit is used for generating

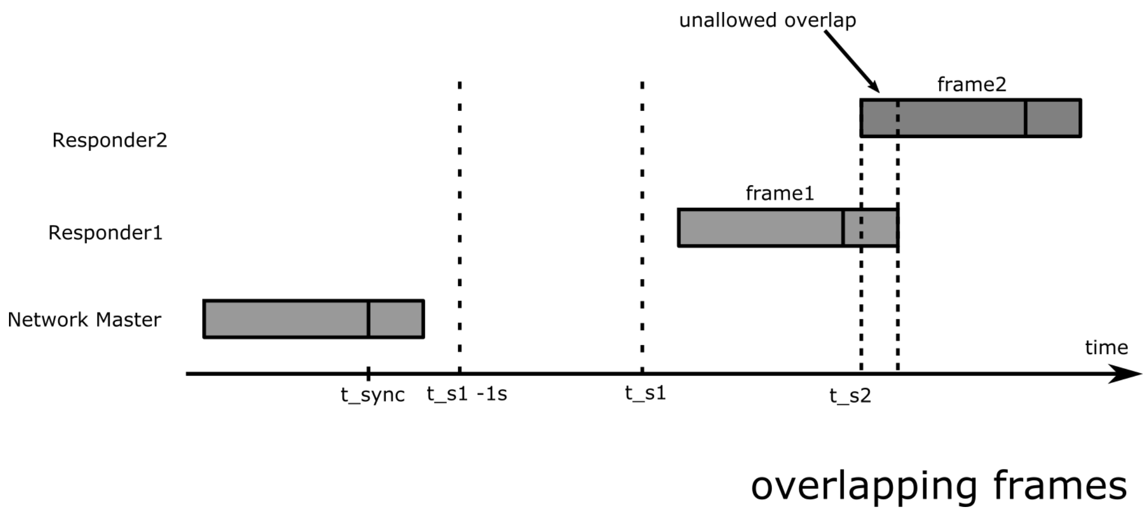

Figure 3.7: An example of an error caused by wrong timing.

a reference time. In this example, a synchronization unit called "Network Master"sends a reference broadcast for a network synchronization. Based on this broadcast, the network nodes named "Responder1"and "Responder2"need to respond according to their given time slot. The start time of the slot of Responder1 is called t_s1 the start time of the slot of Responder2 is called t_s2. Between the broadcast and the slot of Responder1 passes at least $1\,s$ in this example. The slots of Responder2 follows directly after the slot of Responder1. If Responder1 has temperature-based clock-drift of just $10\,ppm$, the clock-drift based time difference is at least $10\,\mu s$, if the clock of Responder2 is not drifting, This leads to an frame overlapping of $10\,\mu s$. For a maximum possible clock-drift of $10\,ppm$ the overlapping time of the UWB-frames can reach up to $20\,\mu s$ if both clocks are drifting in the correct direction. Out of that reason, either the clock-drift based effects needs to be compensated, or a guard time needs to be considered during the design of the communication protocol.

3.2 Matlab Based Simulation

A Matlab based simulation has the advantage that channel models according to the IEEE standard do already exist [16]. This means, instead of writing a new channel model, the IEEE channel model can be adapted according to conditions of the room and the environment of the network participants. Figure 3.8 shows a possible layout of a Matlab based UWB network simulation. The network application is responsible for realizing and defining the use case that shall be simulated. Based on the network application a dedicated amount of device localization needs to be triggered. The network application also defines the additional communication that needs to be transmitted for fulfilling the use case specifications. In the next layer the application that runs on the devices itself is

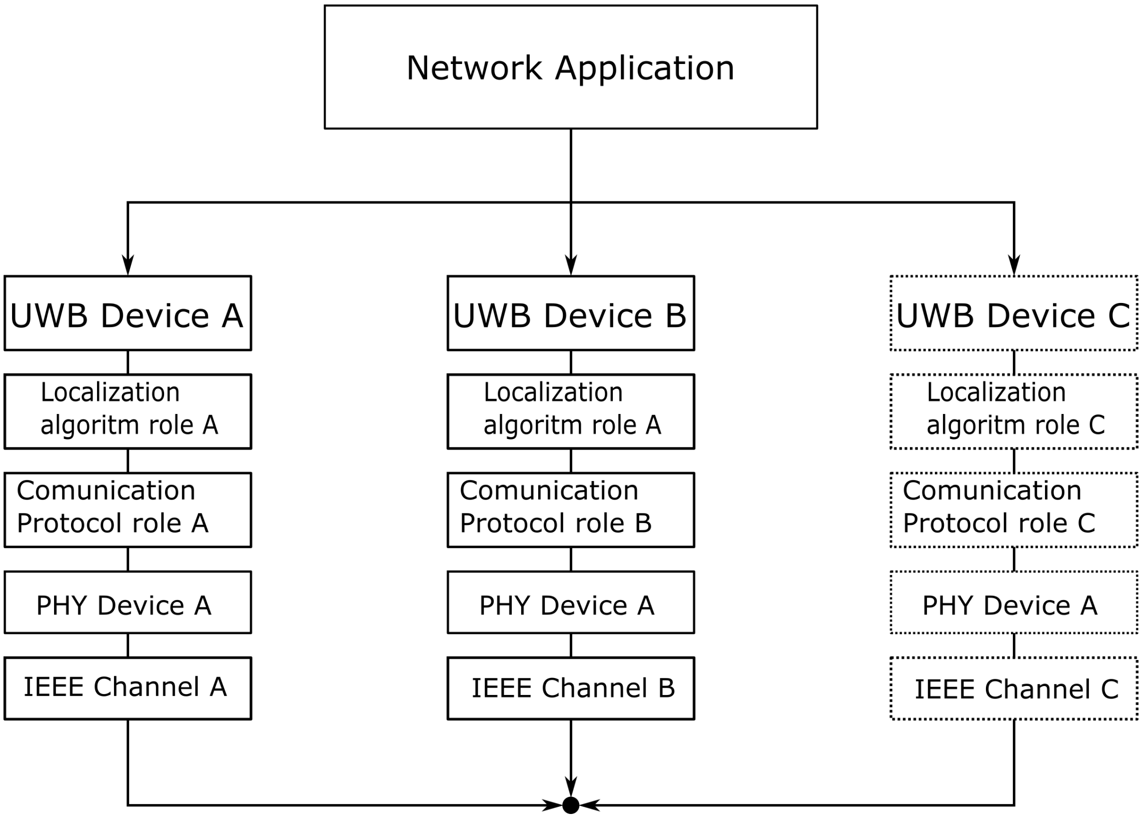

Figure 3.8: Layout of a Matlab based UWB Simulation model.

specified, where multiple types of devices are possible. For example, a indoor localization system would have two main types of devices, static devices that act as reference point for the localization algorithm and dynamic devices that needs to be tracked. Within these device groups there can be sub-groups of e.g. ultra low power devices or devices that need to be tracked with a very high update frequency. Also, anchors can differ in their role of the network. Since every device has running a different application, also the localization algorithm can differ from device to device. A low power device might use a low power algorithm e.g. TDOA where a device that needs to be tracked with a high accuracy and redundancy might use SDS-TWR. An anchor also needs to be able to play the counter part of every node in the network, so these devices can run multiple localization algorithms at the same time, depending on the ranging method of the nodes in the network. As mentioned in section 3.1.2 also the protocol plays a role for the accuracy of the ranging. Since there are plenty of available network protocols that can be mixed for getting different properties from power consumption and accuracy perspective, they also need to be implemented in the Matlab model. Also, here the anchor needs to play the counter-part to every ranging protocol, so it also needs to run multiple protocols in parallel. The PHY of a device can be simulated based on a Matlab based model of the device PHY. Since the most companies create an accurate Matlab PHY model during the IC development process, this model can be reused in the Matlab based simulation frame worked for getting a realistic PHY layer simulation. Matlab has also the advantage, that

there are already existing channel models that are standardized by the IEEE organization. These are only very generic channel models but they can be used as reference for simulation the channel of a device. It needs to be considered that the device channel doesn't only differ from device to device, the channel is also depending on the current location of the device and the location of other devices inclusive surrounding.
In a real system a channel is always depending on two device channels, the channel of the transmitter and the channel of the receiver, also the reflections that need to be considered are depending on the positions of both devices.

3.3 OMNeT++ Based Simulation

The OMNeT++ simulation framework offers many predefined modules that can be used for simulating different layers of a network with a little adoption afford in comparison to implementing the layers from scratch.

Figure 3.9 shows the simulation model architecture that could be used for an OMNeT++ based network simulation.

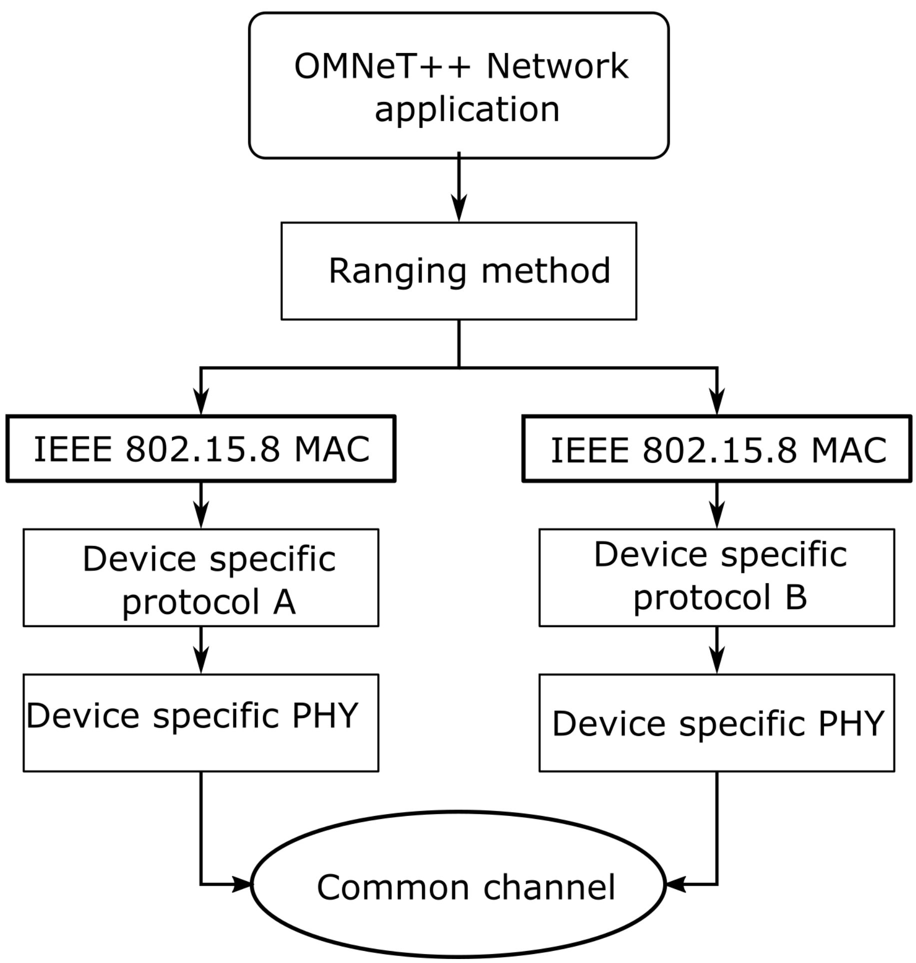

Figure 3.9: Layout of a OMNeT++ based network simulation.

The block "OMNeT++ Network application"specifies the application that shall be simulated. The number of nodes and the environment can be defined here. Based on the use case and the environment, a localization method needs to be selected, this method can be defined in the "Ranging method"layer. Now the generic parameters are set up and the device specific roles must be given. In Figure 3.9 there are two different device roles shown, but the model can be extended to the amount of device types that is required. All the devices must use the same MAC layer that is defined in the IEEE 802.15.8 standard, else the higher layers that are responsible for the post processing of the data won't be able to determine the devices that are participation on the ranging session [6]. The network protocol is the same for the network, but every device can have a different role within this protocol. Out of that reason, every device type has got an own protocol layer e.g. named "Device specific protocol A". This layer is responsible for making the decision on the time when a message should be sent and when the response is expected. When the protocol layer has defined the message transmission times, these times are interpreted by the device specific PHY layer that modulates the signal according to the given input. This signal is then transmitted on the common device channel, this can be also reused for the OMNeT++/MiXiM channel models.

3.3.1 OMNeT++ based Simulation Properties

In comparison to other Network simulation approaches the OMNeT++ has the advantage of the low afford that needs to be made. Also, a graphical simulation is very easy to achieve because of the already existing visualization libraries. Nevertheless, an OMNeT++ has the limitation of maximum network size that is at roughly about 100 devices, which is not sufficient for big scale networks. Also, the channel models do have limited details in comparison to a Matlab based simulation where the IEEE based UWB-channels can be reused. But most probably the biggest problem with an OMNeT++ based simulation model is that the device internal state and timing aren't considered, which is also the case for the Matlab based simulation. This means, the network can be simulated with an ideal device behaviour, but the device internal communication overhead caused by the interpretation of additional data coming from an external interface won't be considered. Especially for low power and low cost infrastructure simulation this could become a critical factor. If the protocol defined timing requirements can't be fulfilled by the hardware the protocol is running on. The simulation of the network won't be realistic because the real throughput bottle neck of the network can't be simulated. The same conditions hold for power consumption simulation, it's not sufficient to simulate only the power consumptions that are caused by the device PHY. Also, the computational overhead of an encryption or a secure element belonging to the device needs to be considered.

3.4 Python Based Simulation

Python offers a free usable network simulation library that can be used as base framework for making customized network simulations [17]. The framework is named "pynsim"and is a very generic framework. This means, all the protocol and MAC layers need to be added by hand. Also the available modules are very limited in comparison to the OM-NeT++ simulation tool. This means, the Python based simulation framework has the same properties as the OMNeT++ simulation from a simulation detail perspective, but it has a more limited library. The only advantage is that the network size can reach a bigger scale which can be an advantage for simulating networks of of thousands of components.

3.5 SystemC Based Simulation

SystemC is a programming language that is used for modelling and the simulation of complex electronic circuits. A model can contain hardware as well as software components. In comparison to a pure hardware description language like VHDL and Verilog-HDL, SystemC is mainly used to operate on a higher layer of abstraction. This leads to a big speed up in the simulation time. Out of that reason also more complex hardware can be simulated without making the simulation times impracticable long. The SystemC offers also the modulation of synthesizable circuits by writing code in the so-called register transfer layer. A big advantage of SystemC is that it is open source and based on programming language C++. The SystemC development platform is based on dedicated C++ libraries that can be included in the modulation models. These libraries allow the simulation of parallel running processes and interprocess communication as well as the modulation of system synchronization. SystemC is very often used as a high-level modelling language of protocols and interfaces for ensuring their functionality before implementing them in firmware or hardware [18].

3.5.1 SystemC Model Creation

Figure 3.10 shows the work flow that is required for creating a SystemC based model. A SystemC model consists of three parts:

- Files for the model description
- Ordinary C++ libraries
- SystemC libraries

The simulation description happens in ordinary C++ files. In these files the System, consisting of different modules and submodules, is defined. Every module has a given task and interfaces. When an input of an interface is triggered, the module reacts accordingly. Depending on the specification of the module, an output is triggered based on a time event or a triggered input. The Ordinary C++ libraries are needed for reusing the already existing libraries which reduces the afford of the module definition. The SystemC libraries are needed for having a simulation engine in the background. These libraries give a time basis for the system which is needed for an evaluation of the system. Also the SystemC libraries are needed for having data channels for inter-module communication, which is needed for a correct handling of events and interfaces that are needed for a realistic system simulation.

Figure 3.10 shows how a SystemC simulation is created. The module definitions are combined with the ordinary C++ libraries and SystemC libraries that are needed for the system descriptions. The C++ compiler compiles the source files which are later linked together by an ordinary C++ linker. This creates a binary file that can be executed. A step by step analysis of the executable can be made either by using a debugger or by programming a debug sequence in advance during the model creation. The execution results in a simulation traces that can be evaluated afterwards. Such traces can be used e.g. for the evaluation of different ranging algorithms and protocol like it is mention in the sections 3.1.2 and 3.1.3.

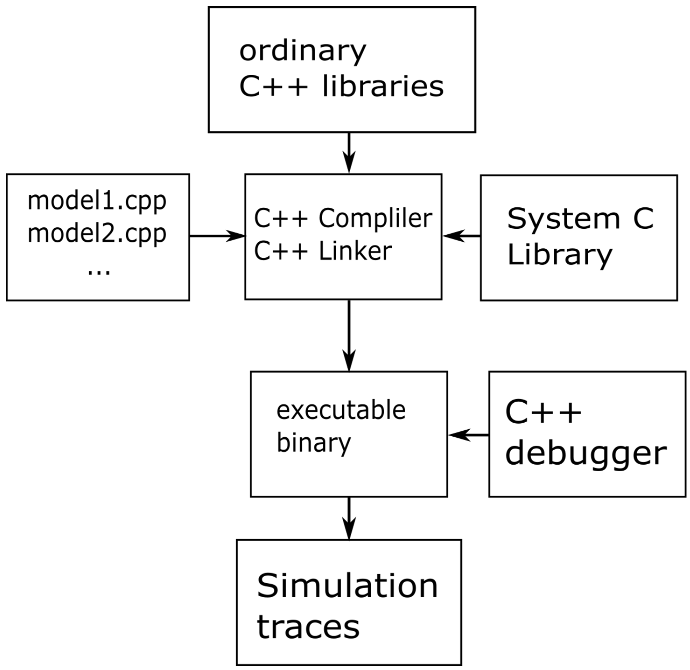

Figure 3.10: The needed steps for creating a SystemC based model [19].

SystemC has many advantages in comparison to a Python based and OMNeT++ based simulation model, especially possibility of simulating device internal layer enables new debugging possibilities. Nevertheless, SystemC has big disadvantages in comparison to Matlab if it comes a more detailed channel simulation. SystemC based channels are simulating the Data and MAC layer of a channel but not the PHY layer. This means, a frame collision can be detected based on the timestamps, but the physical effects of the environment can't be estimated based by using a SystemC model.

3.6 Combination of SystemC and Matlab

Sections 3.2, 3.3 and 3.5 show, that every simulation platform has its advantages and disadvantages. Matlab is the best from a channel simulation perspective because of the reusable channel models that are defined by the IEEE organization. Also the Matlab based device models that are created for the validation of the UWB PHY can be reused by using a Matlab based channel model. The advantage of SystemC is that the device internal timings and data communication can be simulated which leads to a more realistic timing because the device internal limitations are also considered. Another advantage of SystemC is, since it is a modelling language, it's likely that a model of the device already exists because it was used for specifying the internal hardware and states. This means,

the device implementation afford is also reduced.
Since the Matlab based channel models and the SytemC based device models are the best in comparison to the other technologies, combining both models is a promising solution for a UWB network Simulation.

Figure 3.11 shows a network simulation consisting of a SystemC based network model interfacing with a Matlab based channel model. The SystemC based network model con-

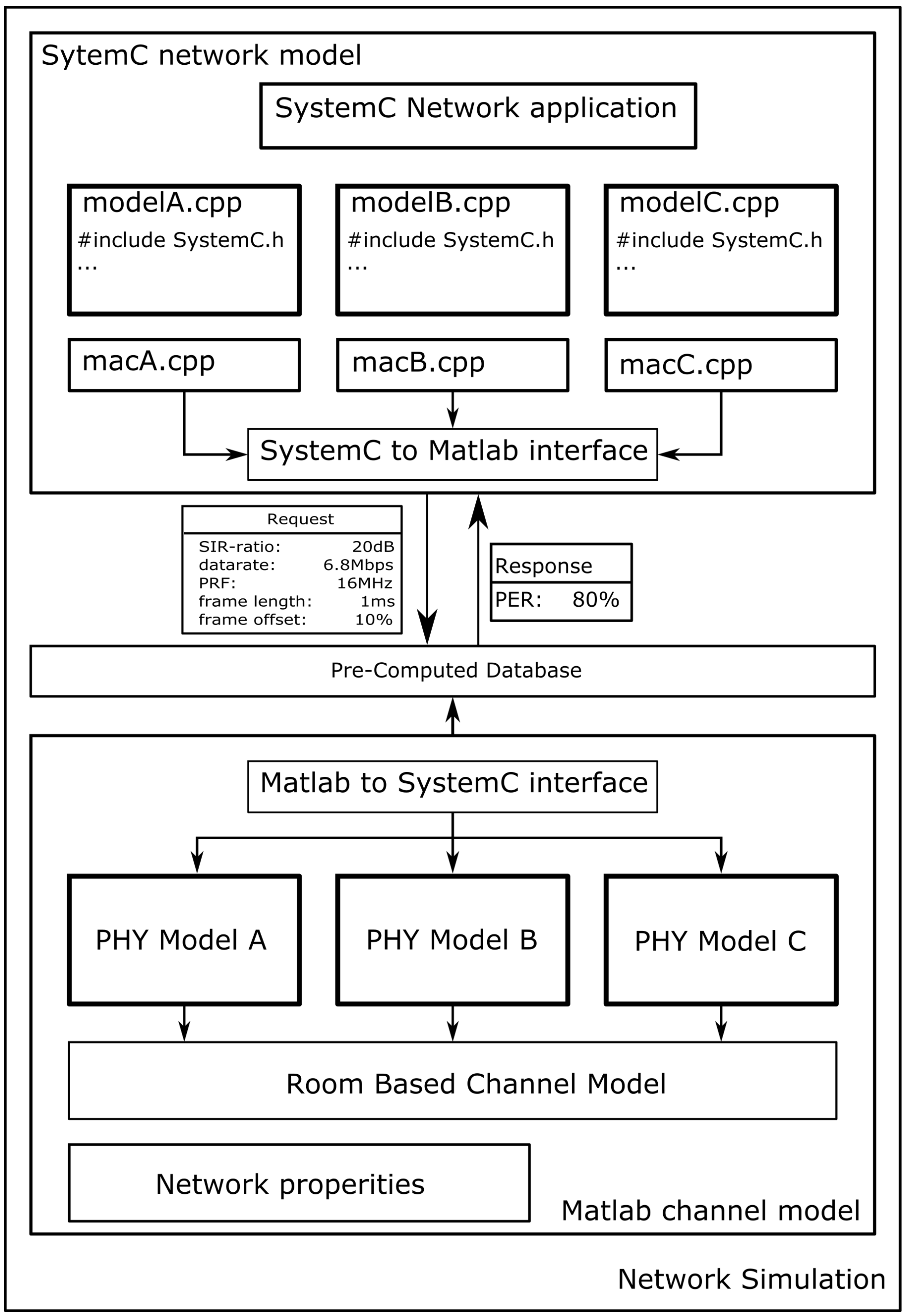

Figure 3.11: Network simulation based on SystemC and Matlab.

sists of 4 layers:

- The Network application
- The device models definitions
- Device MAC implementation
- The interface to the Matlab channel model

The SystemC network application sets up the environment. It defines the dimensions of the room, the number of anchors within the room and their positions. Also, the number and positions of the nodes that shall be localized. The physical parameters of the SystemC network application need to be handed over to the Matlab based SystemC model so that the Matlab channel can be adapted according to the environment. The different device roles are defined in C++ files e.g. name "modelA.cpp". A device can consist out of multiple sub systems that are for example needed for having an additional RF interface for data exchange. As described in section 3.5 multiple device classes are possible within a system e.g. describing the anchor device and the node device. Also, the roles in the MAC layer can differ from device to device, which means different mac implementations are required. The protocol that is used from a device is also implemented in the device model itself. The SystemC to Matlab interface is needed for having an interface to the channel model. Always when a frame shall be send based on the data collected by the mac, this interface shall report the package to the Matlab simulation. The Matlab simulation consists of:

- The interface to the SystemC simulation
- The PHY models of the devices
- The room based channel model
- Physical network properties

The SystemC interface of the channel simulation is needed for receiving the simulation parameters from the Matlab interface. Every time when a ranging is triggered in the SystemC model, the Matlab interface receives a request for evaluating the success probabilities based on the devices that participate on the ranging, the UWB-frame length and the network properties. Also, other interferences need to be considered in this channel model. Interferences can be either caused by devices that are operating within the same channel or external devices. For a UWB network that uses the ALOHA protocol, an in-channel interference could be caused by two devices communicating at the same time. Based on communication slots of the devices, the channel model needs to determine which messages are received by which device. An external device could be e.g. a Wireless Local Area Network (WLAN) according to the IEEE-802.11 [20], that is communicating in a frequency band that is close to the UWB carrier. Even if the frequency bands don't overlap, because of the high output power of the WLAN, the UWB signals could be disturbed which needs to be considered in the simulation. The PHY models in the Matlab simulation are required for having a detailed device description like it is explained in 3.2. This has the advantage of combining the internal behaviour models of the SystemC simulation

with the physical model of the device which leads to a realistic simulation of the device behaviour. The room-based channel model calculates the channel based on the devices communicating, the environment and the interferences. This model requires input data from PHY models and the network properties. Network properties are e.g. the positions of the nodes and the persons they are attached to. These properties need to be updated always when the anchor or the node positions change because of the running SystemC network application.
For combining the SystemC model with a already exiting Matlab interference model [11], the following set of inputs is selected:

- The length of the UWB-frames
- The communication datarate
- The pulse repetition frequency
- The relative position of the colliding UWB-frames
- The signal to interference ratio

These input parameters can be used for making a lookup in a database with the pre-computed interference simulation results. Based on the input parameters, the database returns a average package error rate (PER) which can be used in the high-level SystemC model. Because of a limited number of input combinations, the database can be pre-computed and reused. This leads to a significant speed-up of the simulation platform by still using realistic channel models.

3.6.1 Comparison to Other Simulation Models

A SystemC based Network model combined with the Matlab based channel model is the most promising simulation approach in comparison to the single platform solutions mentioned in 3.2, 3.3 and 3.5. Out of that reason such a model is selected for the implementation and evaluation of a UWB based network simulation.

Chapter 4

Implementation

This chapter describes the implementation of the UWB Network simulation, by defining the modules that are used for simulating the timing behaviour and the measurement accuracy of a Network setup for realizing given use cases. There will be a more detailed look at how to simulate the real behaviour of devices and what effect it has on the System.

4.1 Defining the Simulation Scenarios

A Simulation Scenario is defined by the following parameters:

- Physical network size
- Number of nodes in the network
- Number and position of the anchors in the network
- Used Network protocol
- Used ranging method

The Physical network size describes the environment and the size of the network. An environment can have the size of a small office up to a big economy hall with potentially thousands of nodes. The number of nodes in the network correlates with the environment in real-world applications. Nevertheless, for benchmarking a system setup, the number of nodes is one of the most important inputs because it can be used for determining the stability of the setup under given conditions. The number and positions of anchors needs to be evaluated based on the environment. The anchor placement has a big impact on the accuracy of the used Localization algorithms e.g. if all anchors are placed at the same positions, all the measurements will have the same result which doesn't add any meaningful data to the localization algorithm of the upper layer. Also, the channel between an anchor and a node depends on the position of the anchor. This has also an impact on the package error rate and thus also on the network throughput.
The used network protocol is required for evaluating the maximum accuracy and measurement throughput based on the protocols supported by a given hardware combined with a given ranging method. Some very simple protocols will be sufficient for a network with fewer nodes and anchors, but won't scale for bigger networks.

Figure 4.1 shows an example of how a simulation setup could look like. Listing 4.1 shows the source code for setting up the environment.

Listing 4.1: Example for setting up the simulation environment.

```
1  //seting up simulation with room dimensions Room_size_X*Room_size_Y
2  MovementSimulation simulation(Room_X,Room_Y);
3  //adding anchor to the room
4  simulation.addAnchor(x_coordinate_A1, Y_coordinate_A1, Z_coordinate_A1);
5  simulation.addAnchor(x_coordinate_A2, Y_coordinate_A2, Z_coordinate_A2);
6  simulation.addAnchor(x_coordinate_A3, Y_coordinate_A3, Z_coordinate_A3);
7  simulation.addAnchor(x_coordinate_A4, Y_coordinate_A4, Z_coordinate_A4);
8
9  //adding randomly 8 nodes to the room
10 simulation.addNodes(8);
11 //adding a node at a given possition to the room
12 //simulation.addNode(x_coordinate, Y_coordinate, Z_coordinate);
```

"MovementSimulation"is the class name of the simulation framework. The input parameters are the horizontal and vertical room size, the height of the room is not considered. From line 4 to line 7 anchors are added to the simulation, the input parameters for adding an anchor are the X,Y and Z coordinates of the anchor in the room. In line 10, 8 nodes are added at a random position in the room. For adding a node at a given position the commented function in line 12 needs to be called. This function can be used e.g. for testing a special constellation of devices. The "addNode"and "addNodes"function can return an error, if the room can't handle more nodes because of its physical size, or if a node shall be added at a position that is already occupied.

Each node and anchor has a generic protocol and ranging class which is used as a base class for a protocol definition and a ranging method definition. The ranging method and the network protocol can be set in the protocol and ranging class as member of the nodes and anchors.

4.2 Simulation of the movement of the Nodes

Every movement simulation class has got a member called "Room"which is used as a map for storing the current position. The positions of a node are oriented on a grid with a defined granularity that can be defined in the "Room.h"file. Every node occupies a field of the grid, and no field can be occupied by two nodes. An anchor can be placed anywhere in the room because anchors are small in size in comparison to a node that is attached to a person. Based on the grid, the node can move from one field to another field that is connected to the current field of the node. Figure 4.2 shows an extract of a room including its grid, two nodes named "N1"and "N2"are placed in the grid and each node occupies a field. The define named "RASTER_SIZE"defines the size of the grid. N1 wants to make a step within that field, the possible fields N1 can move to are named "P0"- "P8"were P4 means the node doesn't move at all. The field P2 is occupied by N2 so N1 can't move there. Listing 4.2 shows how a node can be moved. The function "simulation.movePersonStep(direction,nodeID)"is used in line 2 and 8 moves a node attached to a person in the given direction. The node that is moved is specified by the nodeID that is also a member of the node class. The directions of a node map with the relative field position shown in Figure 4.2. For example, code line 2 moves N1 to its relative position P1. The code in line 8 would return with an error code because the field

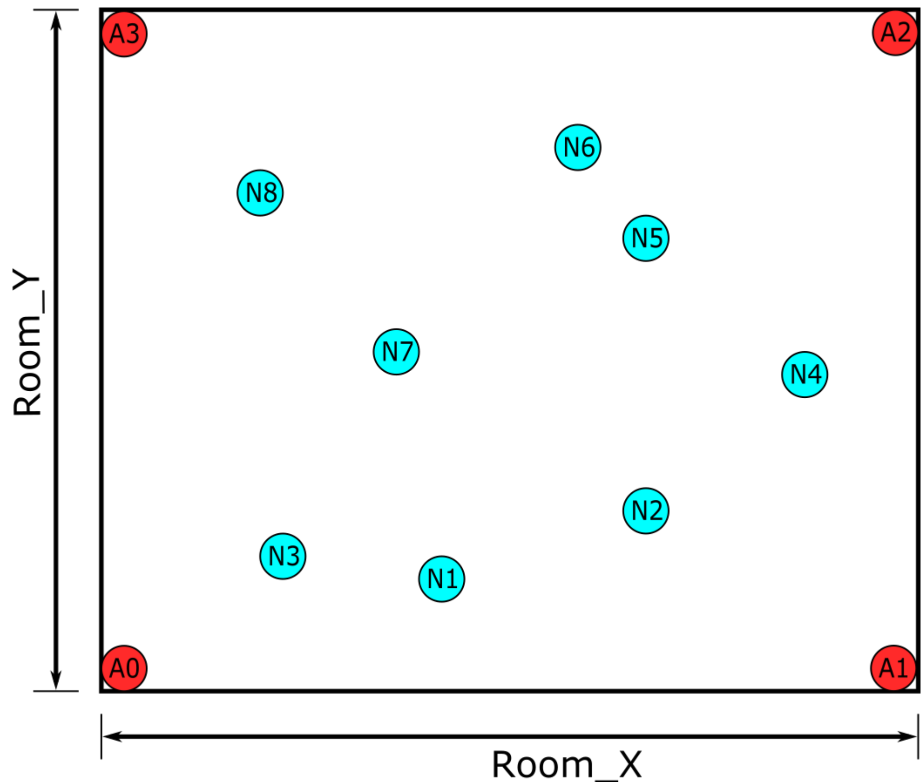

Figure 4.1: A example of a simulated network.

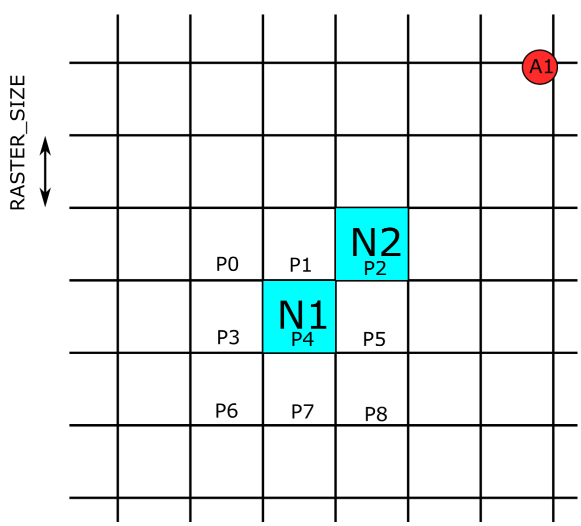

Figure 4.2: Scheme of a movement in the grid of a room.

P2 is already occupied by N2. For moving every node in a random direction, the function shown in line 5 can be called. This function can be used for simulating the movement of many persons within a room which is needed for simulating a big system where touching every node is not practicable. Since an anchor has a static position within the room, it can't be moved. Also, the position of an anchor doesn't matter for the movement of a node because anchors are placed at the ceiling of a room which means it can't collide with a person. In the simulation environment each node that is tracked is also attached to a person for simplicity reasons.

Listing 4.2: Example of moving persons.

```
//function to move a person in a direction
simulation.movePersonStep(1, 1);

//moves every person in the room
simulation.makeMovement();

//returns error
simulation.moveperson(2,1);
```

4.2.1 Combining the Node Positions with the Channel Model

Since every Node is attached to a person, the positions of the nodes need to be considered in the channel model. In the simulation, the height of a person is twice the z-coordinate of a node. If multiple nodes are added in the room, the z-coordinate is randomly deviated at values between 1.95 m and 1.35 . If a node is ranging to an Anchor the Persons between the Node and the Anchor are estimated by a ray-tracing algorithm. Figure 4.3 shows an

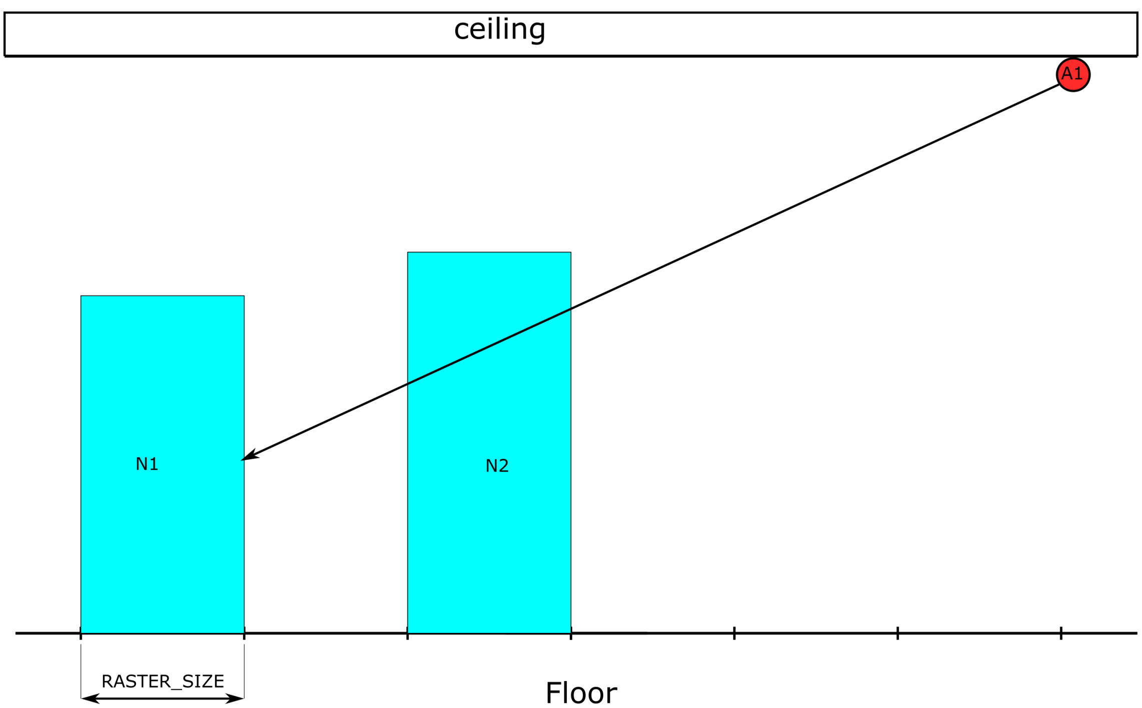

Figure 4.3: Example of a NLOS signal.

extract of a network setup where an anchor"A1"sends a UWB-frame to the node "N1". The signal is an NLOS signal because it needs to travel through the Person "N2"to reach the node N1. The ray-tracing algorithm needs to follow the LOS signal in a given step size and has to determine if it collides with a node on its way.

Listing 4.3 shows the pseudo-code of the ray-tracing algorithm. The ray-tracing algorithm checks how big the detuning of the signal between an anchor and a node is. The anchor and the node are both identified by its device ID. First, a trace needs to be defined by determining its start and end position, in this example, the start position is the position of the anchor and the end position is the position of the node. The algorithm travels down the trace and checks based on the room map, if a field is occupied by a Node. If this is the case, the height of the field needs to be checked. If the height of the point on the trace, is smaller than the height of the Person attached to the node, the trace is colliding with the person and the NLOS counter is increased. Afterwards, the next step on the traces is checked. This procedure is continued until the point that needs to be checked is beyond the range of the trace. At the end of the function the NLOS counter is returned. Based on the NLOS counter value the damping of the signal can be estimated.

Listing 4.3: Ray-Tracing pseudo-code.

```
simulation.isNLOS(anchor_id, node_id)
{
  start_postion = getAnchorPositions(anchor_id);
  end_position = getNodePosition(node_id);
  Trace trace(start_postion, end_position);
  travel_distance = trace.getTravelDistance();
  while(trace.distance_checked() < travel_distance)
  {
    if(trace.checkIfFieldIsOccupied())
    {
      if(trace.getHeight() < trace.getOccupiedFieldHeight())
      {
        NLOS_Cnt++;
      }
    }
    trace.make_step();
  }
  return NLOS_Cnt;
}
```

4.3 Protocol Simulation

Evaluating different network protocols is one of the central parts of the network simulation. Based on the different network properties some protocols could fit better for given requirements. Especially by using external interfaces the overhead of some protocols can be reduced a lot which leads to a big advantage from channel consumption perspective.

4.3.1 Pure ALOHA

The Pure ALOHA is the simplest protocol of all protocols. As described in section 2.2, the Aloha protocol is based on randomly transmitted messages without any time reference or collision avoidance mechanisms like listening on the channel.

Since every simulation setup is used for simulating a given use case, the throughput of the protocol needs to satisfy some requirements e.g. from localization update rate perspective. Especially by using a random-access protocol with a given update rate the randomness becomes less random from message transmission time point of view.

Listing 4.4 shows the pseudo-code for the calculation of message transmission timeslots. The slot size is depends on the number of anchors, the number of nodes the anchors shall be ranging to and the time consumption per ranging frame. In theory the ranging could also take place in smaller slots with the size of just one ranging frame. But depending on the localization update rate, the time between the rangings could become so big, that the position changes from one ranging to the other which would lead to an additional error. Also, based on the throughput calculations of the pure Aloha protocol in 2.2, the slot size has no impact on the throughput, at least not if all devices use the same slot size.

Listing 4.4: Pure ALOHA pseudo-code.

```
getPureAlohaTXTimeStamp(number_of_anchors, localization_update_time,
ranging_frame_time_consumption, number_of_slots_to_generate, start_time)
{
  Time time_consumption = ranging_frame_time_consumption * number_of_anchors;

  Time start_time = getLocalSystemTime();
  TimeSlotArray time_slot_array[number_of_slots_to_generate];
  for(i = 0; i< number_of_slots_to_generate, i++)
  {
    time_slot_array[i].setStart(start_time + localization_update_time*i
    + getRandomTimeStamp()%(localization_update_time - localization_update_time);
    time_slot_array[i].setEnd(time_slot_array[i].getStart() + time_consumption);
  }
  return time_slot_array;
}
```

The time slots are not repeated with a constant period, because if this would be the case, a collision would occur always between the same devices until one device generates a new set of time slots. Out of that reason, every slot is within a maximum start and end time which is defined by the maximum localization update rate. This means, the average period of the timeslots will be constant but the period of a sequence of two slots can be between zero and the twice the localization update rate. At the end, the time slot generation function returns an array of ranging slots with a start and end time. This start and end times are later needed for being able to detect a collision of two different messages on the same channel.

4.3.2 Slotted ALOHA

Slotted ALOHA as it is described in 2.2, is also based on a random-access scheme, where only a list of given timeslots are available. This means, the frame transmission timestamps within a given period are predefined by the slot size and the localization update rate. The pseudo-code for the timeslot generation of the Slotted Aloha protocol is shown in Listing 4.5. The big difference between the Slotted ALOHA and Pure ALOHA timestamp generation is that the Slotted ALOHA timestamp generation needs a reference time for the starting points of the time slots. Such a reference time can be achieved by the anchors by sending a broadcast at the beginning of every Broadcast period. So, the nodes can record the broadcast with the corresponding timestamp to calibrate their local time basis. Since the internal device clocks of the devices are drifting, the Time needs to be updated every Broadcast period. During the transmission of a broadcast every device must be in receive mode and must not transmit because a collision would have an impact on the whole system. Depending on the localization update frequency, many transmission slots can be generated during a broadcast period. Analogous to the Pure ALOHA time slot generation, every slot within a localization period needs to be generated randomly to avoid repetitive deterministic collisions because of matching localization periods. The slot size is pre-defined and can be transmitted in the PSDU data of the broadcast where also the broadcast period is defined. Depending on the implementation, a slot can be either used for a full ranging session between the node and all the anchors or between just one node and one anchor, but for increasing the localization accuracy all ranging sessions should be made during one slot. The time slot generation function returns the generated start and end times of the ranging slots.

Listing 4.5: Slotted ALOHA pseudo-code.

```
getPureAlohaTXTimeStamp(number_of_anchors, localization_update_time,
ranging_frame_time_consumption, number_of_slots_to_generate, start_time)
{
  Time time_consumption = ranging_frame_time_consumption * number_of_anchors;

  Time start_time = getLocalSystemTime();
  TimeSlotArray time_slot_array[number_of_slots_to_generate];
  for(i = 0; i< number_of_slots_to_generate, i++)
  {
    time_slot_array[i].setStart(start_time + localization_update_time*i
    + getRandomTimeStamp()%(localization_update_time - localization_update_time);
    time_slot_array[i].setEnd(time_slot_array[i].getStart() + time_consumption);
  }
  return time_slot_array;
}
```

4.3.2.1 UWB based Slotted ALOHA improvements

The UWB technology has in comparison to narrowband technologies the capability to distinguish between the first path of a signal and its reflections. This property can be also used to distinguish between the first signal received and an interfering signal that has been transmitted later. Only if the first signal and an interfering signal overlap from the beginning with high accuracy, a collision occurs and none of the signals can be received properly. To avoid this scenario, the Superframe protocol divides its eight synchronization slots into sub-slots with a shift of several symbols. So the likelihood of a full overlap of two signals is decreased. The same principle can be applied to the Slotted ALOHA Protocol. When a broadcast is received, every device chooses one or several transmissions slots, if two devices have selected the same slot, a collision will occur because of the same starting point of the transmissions. Even if the clocks of the devices are drifting apart, a collision is very likely because a clock frequency difference of $10\,ppm$ would only cause a $5\,\mu s$ difference at the starting point of the transmission assuming a broadcast period of $500\,ms$. Out of that reason every device should select one of the 4 slots proposed in 2.1 to decrease the collision probability by a factor of 4 if two devices are transmitting in the same slot.

4.3.3 Advantages of the ALOHA Protocol

The Pure ALOHA and Slotted ALOHA Protocol have the advantage of being very simple, the only requirement of these protocols is the UWB interface which makes them very cheap from infrastructure perspective. The protocols are also very efficient for small size networks. The fewer nodes need to be localised in the network, the lower is the collision likelihood which results in a high relative throughput rate per message. Since there is no additional UWB-frame exchange needed than the messages required for the ranging sessions, the Pure ALOHA and Slotted ALOHA are also very auspicious from a power consumption perspective.

4.3.4 Superframe Protocol

The UWB Superframe Protocol is described in 2.3.2.2 and is a mixture out of a random-access and Time Division Multiple Access (TDMA) protocol. In the random-access phase the nodes can login for a ranging session and during the TDMA phase the ranging takes place. The time slot synchronization is achieved by synchronization frames that are transmitted by the anchors during the synchronization period [6]. Listing 4.6 shows the device internal Superframe pseudo-code for requesting a ranging session by the Central Anchor system.

Listing 4.6: Device internal Superframe pseudo-code.

```
1  //generate a login slot
2  getSuperframeLoginSlot()
3  {
4    sync_time = waitForSYNCframe();
5    login_time = getRandomtime() % RANDOM_ACCESS_PERIOD_SIZE
6      + sync_time+ SYNC_TIME_SIZE;
7    TimeSlot login_slot(login_time, login_time + LOGIN_SLOT_SIZE);
8    return LoginSlot;
9  }
10
11 //execution of the Superframe ranging
12 LoginSlot = getSuperframeLoginSlot();
13 RangingSlot = requestRangingSlot(LoginSlot);
14 if(RangingSlot == OK)
15 {
16   DoRangingAt(RangingSlot);
17 }
```

For requesting a ranging slot, the synchronization frame transmitted by the anchors needs to be received. When the frame is received a synchronization timestamp for the later message exchange can be generated, this takes place in line 4. The login timestamp and the login slot are derived based on this synchronization timestamp. This time slot is returned and used for the ranging slot request. The ranging slot is requested in line 12, if the request was successful a ranging can take place at the slot that is returned by the anchors, else no ranging will take place in the TDMA phase. The advantage of using a TDMA phase is, that after a slot has been assigned to a node, the ranging can take place without any in-channel interferences. Depending on the localization update frequency, a slot can be requested more or less frequently during the Cyclic Superframes. The disadvantage of this protocol is its high protocol overhead because a ranging slot needs to be requested during every Superframe where a ranging shall take place. This leads to a big disadvantage from a power consumption perspective. Also the maximum amount of nodes that can be served is very limited because just a small part of the channel capacity is used for ranging and the rest is either needed for synchronization or for requesting a time slot.

4.3.5 TDMA Based Protocol

A TDMA based Protocol is a time slot based protocol. Every device has got a dedicated time slot for the message transmission. During this time slot no other device on the same channel is allowed to transmit with the same preamble code. This makes the protocol very collision resistant because from the protocol point of view a collision is impossible during the TDMA phase if there is no external interfering device. For the assignment of the time slot, either an additional RF interface or a login phase similar to the Superframe protocol is needed. The difference between a TDMA based protocol and the Superframe protocol is that the Superframe protocol is meant for connecting dynamically small WPAN networks. The TDMA based protocol is a new proprietary protocol that has been developed based on an analysis of the advantages and disadvantages of other protocols for indoor localization use cases. The Focus is on a reduction of the protocol overhead, having a small power consumption and high flexibility. Additionally, the channel consumption for a given number of devices should be as small as possible.

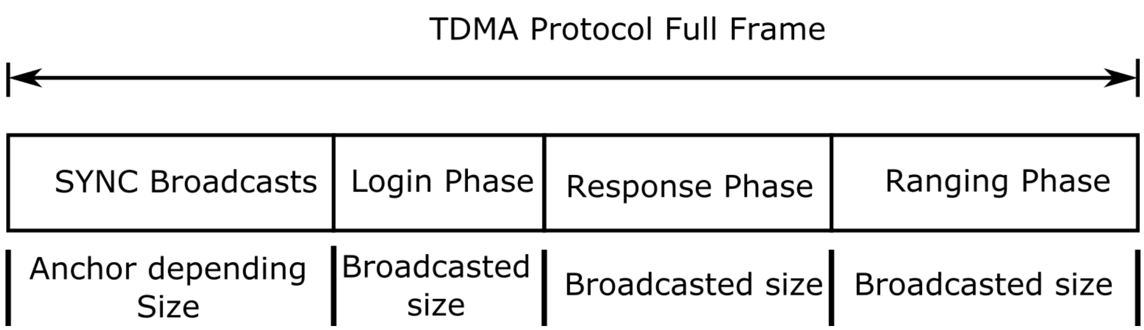

Figure 4.4: A Frame of the TDMA Protocol.

Figure 4.4 shows a full frame of the TDMA Protocol for an UWB only system. This variation of the protocol was designed for very simple low-cost devices, therefore no additional interfaces except the UWB are required for localization, login or logout. The first frame of the protocol called "SYNC Broadcasts"is used for the synchronization and transmission of timing data which is needed for not disturbing later phases of the protocol. The timing data defines:

- Length of the Login Phase
- Slot size of the Login Phase
- Start of the Response Phase
- Start and end Ranging Phase
- Time of the next SYNC Broadcast

Additionally, the broadcast contains the number of anchors in the network. The "Login Phase"starts immediately after the broadcasts and is needed for requesting a slot for the ranging. A request consists of a simple UWB Frame according to the IEEE 802.15.8 MAC protocol [6]. This frame contains the number of rangings that shall be made during a

TDMA full frame, the number of full frames the node wants to range, the duration of a frame for making a localization and the ranging method.

The "Response Phase"is used for responding and acknowledging the logins that have been received. All anchors are responding with the same response during this phase, because the system can't know which anchor was heard during the SYNC Broadcasts phase. The response contains the start and end time of the ranging slots with a corresponding node address. Since a node can request multiple localizations during the ranging phase, the response can also contain multiple ranging slots for a node address.

The "Ranging Phase"consists of the slots defined during the response Phase, Figure 4.5 shows an example of a TDMA Ranging Phase. During the ranging phase every Node

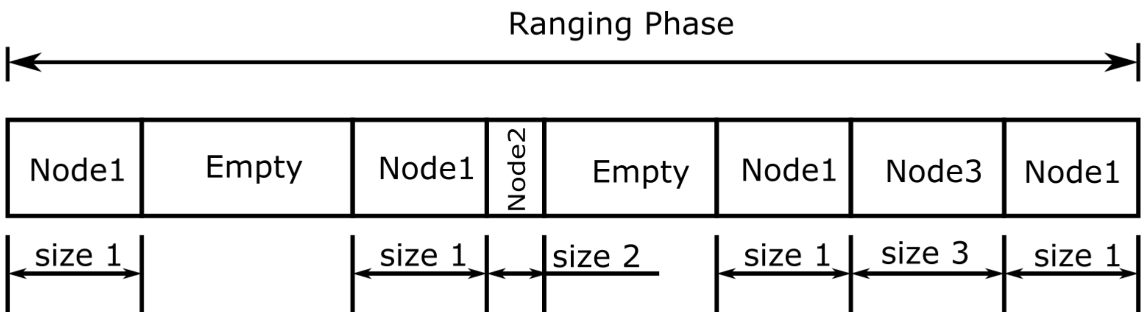

Figure 4.5: An example of a TDMA Ranging Phase.

that wants to be localized has got one or several slots with a given size. This size is known to the node and anchor network. Out of that reason every node only needs to turn on its transceiver during its own ranging slot, the slots of other devices are not of interest to the node which means it also doesn't need to receive during this phase. This has an advantage from power consumption point of view because for the rest of the time, a node can go into a power saving state because it knows that its attention is not needed during the other slots. In the example shown in Figure 4.5 Node1 has got 4 slots during a Ranging phase, Node3 has got just one slot that is a bit longer than the slot of Node1 and Node2 has got a very short slot. This means, the nodes are using different ranging techniques, or are ranging to a different amount of devices in the network. The reason for this could be that a node could hear e.g. only 5 of 8 devices during the SYNC Phase and out of that reason it only requested a slot size large enough for ranging with 5 devices. This dynamic slot size is an advantage from a channel capacity point of view, because a node can request the needed amount of transmission time. If a node would be forced to use a constant size of transmissions slot, the slot might be to small or to big which is inefficient from a channel capacity perspective. For a small network size with a low localization update rate, the most time of the Ranging Phase will be empty which means that no communication is going on during this slot. In the empty phase the anchors can also turn of their receivers which decreases the power consumption in comparison to a ALOHA protocol where the anchors need to be receiving all the time.

The access scheme during the Login Phase is based on the Slotted ALOHA protocol with the UWB specific modifications described in 4.3.2 for improving the throughput rate. The SYNC Broadcasts Phase gives the time reference and the slot size that shall be used during the Login Phase, the random preamble symbol shift shall be made according to table 2.1.

During the SYNC Broadcasts and the Response Phase, the access scheme is based on a TDMA scheme with a pre-defined slot order since the Anchor network is static.

4.3.5.1 TDMA with external Interfaces

More complex UWB applications will also require an additional data channel that can be used for exchanging bigger files like floor plans needed for indoor navigation. Such additional data channel has the advantage of having a much higher channel capacity from a data transmission rate perspective. Also, narrowband data channels are more power efficient than UWB-channels which is an advantage for battery driven devices. If a second data interface is available, the communication going on in the Login and Response Phase can be pushed to that interface for decreasing the UWB-channel consumption and saving power. This is possible because in the Login and Response Phase only data communication is going on without a need for an accurate timestamp generation. Figure 4.6 shows the proprietary TDMA protocol with adoptions for a system that has got a second narrowband interface. The Login Phase and the Response Phase are now no more a part of a TDMA

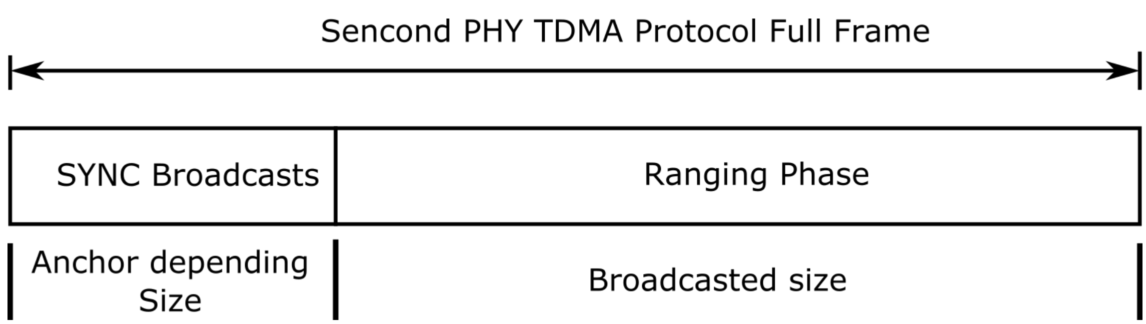

Figure 4.6: The TDMA Protocol with adoptions for a second data interface.

Frame. This has the advantage that the size of the Ranging Phase can be increased by the size of the Login and Response Phase. The SYNC Broadcasts phase stays the same and also broadcasts the same data.

4.3.5.2 UWB only TDMA vs Second PHY TMDA

The advantages of a Second PHY TDMA protocol in comparison to a UWB only TDMA are:

- Increased UWB-channel capacity
- Decrease power consumption
- Higher data rates
- Secure Login

Especially the possibility of having a secure login based on an encrypted data channel e.g. BLE or GSM, leads to higher overall system security which is a requirement of many secure access scenario.

Nevertheless, a mixture of second PHY and UWB only nodes needs to be possible for non-secure low power applications. This can be achieved by reusing the UWB only Protocol for systems that also use offer a second data channel. Devices with the second data channel can login securely and can be also tracked by a secure pre-established ranging sequence. Devices that offer only an UWB interface can still be added by making a request during the UWB Login Phase. By adoption the length of the UWB Login Phase and the UWB Response phase, also the channel capacity of the ranging can be increased. This enables a localization system for tracking non-secured devices e.g. lab tools with an UWB only interface and tracking people securely with a second PHY UWB device. By empowering only second PHY UWB device to open a door of a secured area, also the secure access requirements can be fulfilled with the advantage of having e.g. a non-secure lab inventory localization. For the simulation of a secondary PHY the SytemC channels can be used because these PHY is only needed for pure data transmission.

4.4 Simulation of Different Ranging Techniques

One of the most critical factors for achieving a good channel capacity is the ranging technology that is used. Especially for networks with a big number of anchors and nodes, ranging technologies with a squared complexity will exceed the network limitations very quickly. This section will be about the implementation of the different ranging technologies that shall be evaluated in the network simulation. Also, the clock-drift based measurement error will be explained in more details and how it can be considered during the network simulation with only a little overhead.

4.4.1 Simulation of Device Internal Clock Drift

Figure 4.7 shows the internal setup of a device class. The Device application is the application required for fulfilling the use case that needs to be evaluated. The device properties store the timing behaviour and the physical position of the devices. The Local clock derivation function is required for adding an error to the global clock the simulation is running with. The Ranging method is triggered by the Device application whenever a new ranging is required, based on the application input and the used ranging method, the UWB-PHY is activated for making a message exchange. Listing 4.7 shows the pseudo-code for generating timestamps from a device by considering the measurement errors and the internal clock-drifts. When a new device is created, the internal clock-drift can be set in the device properties. Additional to the clock-drift also the timestamp measurement accuracy needs to be configured. The measurement accuracy adds an error to the timestamps that are generated by the UWB-PHY for considering the device internal errors e.g. caused by the PLL or sampling Jitter. Whenever a UWB package is sent, it is received a signal propagation time later by the other devices in the network. For considering this signal propagation time in the global system time, the global clock needs to be increased according to the distance the signal travels. The global system time is not influenced during the time a message travels because the global clock has an ideal behaviour in the simulation. Nevertheless, whenever a timestamp is generated when a device receives a message, the clock-drift and the random measurement error needs to be considered on timestamp generation. This device specific timestamp derivation is based on the global clock and is shown in line 10. If a device transmits a message, the time that is consumed for the transmission depends on the ideal frame size and the clock the frame is generated with. Out of that reason, the function shown in line 18 considers the internal clock-drift for the data processing and the frame generation

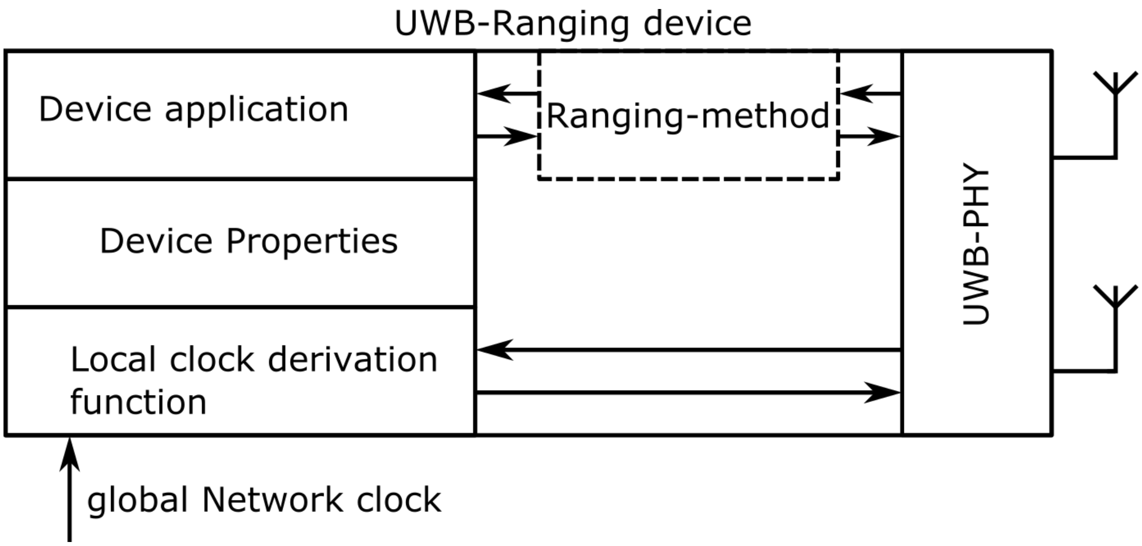

Figure 4.7: Internal setup of a UWB device.

Listing 4.7: Functions for device internal time measurement.

```
1  setupDevice(clock_drift, measurement_accuracy)
2  {
3    this->internal_clock_drift_ = clock_drift;
4    this->measurement_accuracy_= measurement_accuracy;
5  }
6  getRandomError()
7  {
8    return ((rand()%(2*TIME_RESOLUTION) - TIME_RESOLUTION)*
9      measurement_accuracy /TIME_RESOLUTION);
10 }
11 getTimeStamp()
12 {
13   return (getGlobalSystemTime()*(1+this->internal_clock_drift_)+getRandomError());
14 }
15 increaseGlobalClock(time)
16 {
17   global_time += time;
18 }
19 increaseLocalClock(time)
20 {
21   global_time += time/(1+internal_clock_drift_);
22 }
```

Of course, a static device clock-drift is not a realistic scenario since also the environment e.g. temperature has an impact on the internal clock of a device. Nevertheless, being able to set the internal clock-drift to a static maximum will give the worst-case scenario for the system simulation regarding the measurement error of different ranging technologies.

4.4.2 Single Sided - Two-Way Ranging

The SS-TWR is the most simple ranging method and is described in section 2.1.1, it is based on a simple two message exchange and the method itself doesn't offer any measurement data for compensating the clock-drift depending clock-drift error.

Listing 4.8: SS-TWR pseudo-code for evaluating the clock-drift based measurement error.

```
1  doSS_TWR(device1, device2, clock_drift_compensation)
2  {
3    device1.waitForRangingSlot();
4    device1.sendMessageTo(device2);
5    device1.increaseLocalClock(UWB_FRAME_LENGHT);
6    send_Poll = device1.getTimeStamp();
7    device1.increaseGlobalClock(getSignalPropagationTime(device1,device2));
8    rec_Poll = device2.getTimeStamp();
9    device2.increaseLocalClock(PROCESSING_TIME);
10   device2.sendMessageTo(device1);
11   device2.increaseLocalClock(UWB_FRAME_LENGHT);
12   send_Resp = device2.getTimeStamp();
13   device2.increaseGlobalClock(getSignalPropagationTime(device2,device1));
14   rec_Resp = device1.getTimeStamp();
15
16   if(clock_drift_compensation == 0)
17     TOF = ((rec_Resp - send_Poll) - (send_Resp - rec_Poll))/2;
18   else
19     TOF = ((rec_Resp - send_Poll) - (send_Resp - rec_Poll)*
20       (1 + (device2.getClockDrift()-device1.getClockDrift()) +
21       getRandomValue(clock_drift_compensation)))/2;
22   return TOF;
23 }
```

Listing 4.8 shows the pseudo-code for simulating the high-level SS-TWR message exchange. The code in line 3 is required for starting the message exchange according to the ranging slot defined by the higher protocol layer. When the time for transmitting the poll message is reached, the device starts transmitting a message to the responding device. Based on the functions defined in Listing 4.7 the timestamps are created according to a real message exchange of two UWB devices. Since UWB transceivers are detecting signals based on the output of the signal correlation unit of the receiver, many UWB devices offer a reference value for the required PLL tuning. This value is generated based on the timing behaviour of the preamble which means it can be used for estimating the relative clock difference. Based on this value, that can be read out of the PHY layer of a device, the clock-drift based measurement error can be compensated with a given accuracy. This kind of error compensation is simulated with the code shown in line 19. For a realistic simulation of the PHY based clock-drift value estimation a PHY error needs to be considered which is done in line 20 of listing 4.8.

4.4.3 Double Sided - Two-Way Ranging

The DS-TWR works like the SS-TWR the only difference is that an additional message is added to the sequences. The whole timestamp generation and consideration of the clock-drifts works analogous to the SS-TWR pseudo-code shown in 4.8. Listing 4.9 shows the pseudo-code for the timestamp generation of a DS-TWR sequence and the clock compensated TOF measurement according to section 2.1.2. The advantage of the DS-TWR is that the compensation of the clock-drift is based on the comparison of the round-trip and response times of two SS-TWR sequences.

Listing 4.9: Time stamp generation of DS-TWR.

```
doDS_TWR(device1, device2)
{
  device1.waitForRangingSlot();
  device1.increaseLocalClock(UWB_FRAME_LENGHT);
  send_Poll = device1.getTimeStamp();
  device1.increaseGlobalClock(getSignalPropagationTime(device1,device2));
  rec_Poll = device2.getTimeStamp();
  device2.increaseLocalClock(PROCESSING_TIME1);
  device2.increaseLocalClock(UWB_FRAME_LENGHT);
  send_Resp = device2.getTimeStamp();
  device2.increaseGlobalClock(getSignalPropagationTime(device2,device1));
  rec_Resp = device1.getTimeStamp();
  device1.increaseLocalClock(PROCESSING_TIME2);
  device1.increaseLocalClock(UWB_FRAME_LENGHT);
  send_Fin = device1.getTimeStamp();
  device1.increaseGlobalClock(getSignalPropagationTime(device1,device2));
  rec_Fin = device2.getTimeStamp();
  TOF = (rec_Resp - send_Poll)*(rec_Fin - send_Resp) -
    (send_Resp -rec_Poll)*(send_Fin - rec_Resp);
  TOF /= (rec_Resp - send_Poll)+(rec_Fin - send_Resp) +
    (send_Resp -rec_Poll)+(send_Fin - rec_Resp);
  return TOF;
}
```

Since both devices have different properties, also the required processing time can differ from one device to the other. Out of that reason the two SS-TWR methods that are compared don't have the same size. This asymmetry can have an impact on the DS-TWR measurement accuracy that was initially only limited by the inaccuracy of the Time stamp generation.

4.4.4 Asymmetric Ranging

The Asymmetric Ranging consists of multiple DS-TWR sequences, where the response messages of the different devices are time multiplexed in a way that all the responding devices can respond without causing a collision with other devices. The Asymmetric Ranging is explained in section 3.1.3.1, the high-level pseudo-code for the timestamp generation is shown in Listing 4.10. For multi-device ranging technologies it's important to consider the different signal propagation times between the devices. For a proper simulation of the TOF measurement every timestamp needs to be created according to the message arrival times of a real system. This means, the device that is the closest to the transmitter of a message receives the message first, followed by the other devices that are further away. Out of that reason the devices need to be sorted according to their distance to the devices and the system time needs to be increased from device to device that is receiving e.g. the poll message. For simplicity reasons, the timestamp generation is not shown in details in Listing 4.10. Also, the calculation of the time of flights isn't shown because it's done in a analogous way to the DS-TWR and like it is explained in section 3.1.3.1. A big disadvantage of this ranging method is its highly asymmetric timing because of the big number of nodes that are responding to the poll message. The higher number of responders the bigger the processing time of the initiator becomes. This means, the clock-drift can be compensated with a lower accuracy because the maximum achievable clock-drift compensation accuracy correlates with the ratio of the two processing times and the timestamp measurement accuracy.

Listing 4.10: Asymmetric Ranging timestamp generation.

```
doAsymRanging(device1, anchors[])
{
  device_list_sorted = sortByDistance(anchors, device1);
  device1.waitForRangingSlot();
  device1.increaseLocalClock(UWB_FRAME_LENGHT);
  send_Poll = device1.getTimeStamp();
  anchors_Rx_time_stamps[] = getRxTimeStamps(device_list_sorted,device1);
  anchors_Tx_time_stamps[] = getTxTimeStamps(device_list_sorted,device1);
  device1_Rx_time_stamps[] = getRxTimeStamps(device1,device_list_sorted);
  device1.increaseLocalClock(UWB_FRAME_LENGHT + PROCESSING_TIME);
  send_Poll = device1.getTimeStamp();
  anchors_Fin_Rx_time_stamps[] = getRxTimeStamps(device_list_sorted,device1);
  TOFs[] = getTOF(send_Poll,anchors_Rx_time_stamps,anchors_Tx_time_stamps,
                  device1_Rx_time_stamps,send_Poll,anchors_Fin_Rx_time_stamps);
  return TOFs;
}
```

4.4.5 Spy Ranging

Spy Ranging is a SS-TWR or DS-TWR based TOF measurement method that is explained in section 3.1.3.2. Depending on the PHY layer based clock-drift compensation that is available, a SS-TWR based Spy Ranging can lead to a good measurement accuracy by increasing the channel capacity because of the smaller amount of messages that need to be sent.

Listing 4.11: Spy Ranging timestamp generation.

```
doSpyRanging(devices[])
{
  device_list_sorted = sortByDistance(devices);
  devices[0].waitForRangingSlot();
  device[0].increaseLocalClock(UWB_FRAME_LENGHT);
  send_Poll = device[0].getTimeStamp();
  anchors_Rx_Poll_time_stamps[] = getRxTimeStamps(devices);
  devices[1].increaseLocalClock(PROCESSING_TIME+UWB_FRAME_LENGHT);
  send_Response = devices[1].getTimeStamp();
  anchors_Rx_Response_time_stamps[] = getRxTimeStamps(devices);
  rec_Resp = device1.getTimeStamp;
  TOFs[] = getTOFSpyRanging(send_Poll,anchors_Rx_Poll_time_stamps send_Response,
           anchors_Rx_Response_time_stamps, rec_Resp);
  return TOFs;
}
```

Listing 4.11 shows the pseudo-code of a SS-TWR based Spy Ranging. Similar to the Asymmetric Ranging, in the Spy Ranging the devices need to be sorted according to the signal propagation time. A Spy-Ranging network always consists of two active devices, in the listing these devices are called "device[0]"and "device[1]". The two active devices perform a SS-TWR exchange and derive their timestamps accordingly. All the other devices are listening and do only create timestamps of received poll messages on their own clocks, also for the timestamp creation the signal propagation time needs to be considered for getting realistic TOF values. The advantage of this method is that the ranging frames are very short which leads to a low collision probability. Also, the clock-drift based error should be less than e.g. for the Asymmetric Ranging. This is because for the DS-TWR based Ranging the TWR frames should be nearly symmetric and for SS-TWR the message exchange should be short enough for compensating the clock-drift based error by using the PHY indicator.

4.5 Localization Algorithms

The main application of UWB Systems is providing an accurate localization of the devices within the network. The localization accuracy basically depends on two factors, the accuracy of the UWB based distance measurement and the position of the anchors within the system. The more accurate the measurement data is, the smaller is the error of the localization algorithms. Anyhow, if the anchors are very close together the measurement data doesn't really add any meaningful data to the localization because all the anchors are just creating nearly the same measurement values that can't be considered for the localisation because of the error.
Since the anchor location has also an impact on the UWB-channel, there is most likely a tradeoff between the channel and the localization accuracy. This tradeoff needs to be evaluated, as a two-dimensional device tracking algorithm needs to be implemented for evaluating the different anchor setups.

4.5.1 Tracking in Two Dimensional Space

For most of the indoor localization systems a tracking in the two dimensional space is sufficient because another device sensor e.g. the barometer can give the height of a device. Also, in most scenarios, a Human will carry the UWB device which means it has a more or less constant height within the same room of a building. Additionally based on the anchors in LOS and the tracking of the movement, the floor a person is located at, can be easily estimated. Out of that reason, for indoor localization it's a good choice to focus on a 2D device tracking anchor setup. Figure 4.8 shows a localization system in a 2D plane. The system consists of three anchors and one Node. The node measures the distance to every anchor in a very short time, so the movement of the node during the measurement can be neglected in the distance calculation. Every anchor has got an X and a Y coordinate named Ai_X and Ai_Y where the i stands for the index of the Anchor. Based on the distances and the known anchor positions, the generic equations 4.1, 4.2 and 4.3 can be formed. N_X and N_Y are the X and Y coordinates of the node.

$$d1^2 = (N_X - A1_X)^2 + (N_Y - A1_Y)^2 \tag{4.1}$$

$$d2^2 = (N_X - A2_X)^2 + (N_Y - A2_Y)^2 \tag{4.2}$$

$$d3^2 = (N_X - A3_X)^2 + (N_Y - A3_Y)^2 \tag{4.3}$$

For simplifying the equations the anchors are placed on the X and Y axis of the coordinate system and anchor "A1"is placed in the center of the coordinate system. This leads

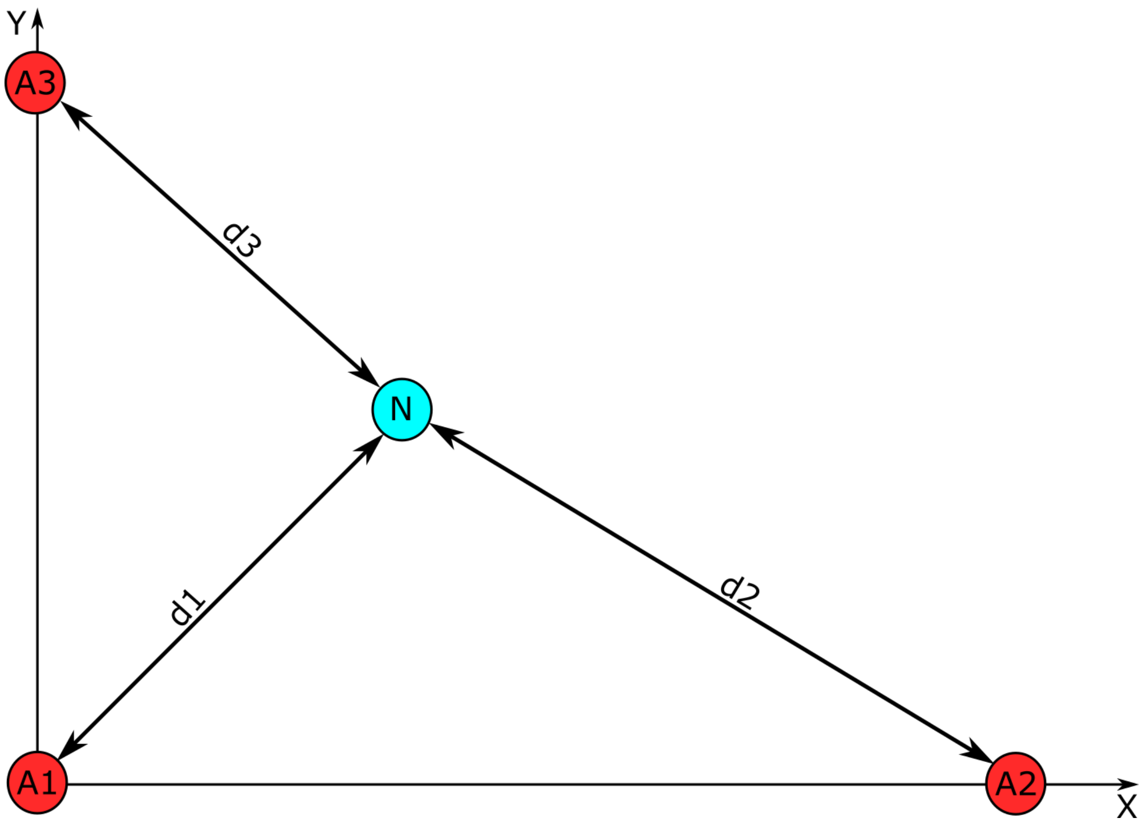

Figure 4.8: Example of a 2D localization setup with 3 anchors.

to equations 4.4,4.5 and 4.6. This equation system can be simplified more easily for calculating the X and Y coordinate of the node.

$$d1^2 = N_X^2 + N_Y^2 \tag{4.4}$$

$$d2^2 = (N_X - A2_X)^2 + N_Y^2 \tag{4.5}$$

$$d3^2 = N_X^2 + (N_Y - A3_Y)^2 \tag{4.6}$$

By combining the equations 4.4 and 4.5, equation 4.7 can be formed.

$$N_X = \frac{d2^2 - A2_X^2 - d1^2}{2 \cdot A2_X} \tag{4.7}$$

By combining the equation 4.4 and 4.6, equation 4.8 can be formed.

$$N_Y = \frac{d3^2 - A2_Y^2 - d1^2}{2 \cdot A2_Y} \tag{4.8}$$

In a two-dimensional coordinate system it would be sufficient to have only two reference distances to two different anchors for calculating the X and Y coordinate of the node. Combining equation 4.4 and 4.7 can be easily formed to equation 4.9.

$$N_Y = \sqrt{d1^2 - N_X^2} = \sqrt{d1^2 - (\frac{d2^2 - A2_X^2 - d1^2}{2 \cdot A2_X})^2} \tag{4.9}$$

Also, other anchor pairs can be combined in an analogous way for calculating the X and Y coordinate of the node. Nevertheless, such a two-dimensional example doesn't exist in a real live application since every device has got a 3D position.

4.5.2 Three dimensional Tracking

Since most of the indoor localization systems have their anchors placed on the ceiling, at least the assumption that all anchors are placed in a plane holds. This has the advantage that some simplification regarding the 3D system can be made. Equation 4.7 still holds if the anchors A1 and A2 are placed on the X axis of the coordinate system. The height of the node doesn't matter for solving the equation system, e.g. the Y position of the node could be rotated around the X axis and the X position would still be the same. This means, combining the Y coordinate with a rotation around the X axis leads to a polar coordinate system that can reach any Y and Z coordinate by a variation of the angle of the rotation and the distance of the node to the X axis. Out of that reason an anchor pair placed along an axis can be always used for the determination of the position of a node on the axis the anchors are placed along. The same principle can be applied to any anchor pair on any axis in a room since two anchors always have a common anchor axis. Especially if the anchors are placed on two orthogonal axes, like the example shown in Figure 4.8, the Anchor pair placed on the Y axis can be combined with the anchor pair on the X axis for estimating the X and Y position of the node without requiring any additional measurement. If the anchor axes aren't orthogonal, the position of the node could be still estimated but the distance measurement error would have a bigger impact on the calculated position of the node. Out of that reason, anchors should always be placed on a vertical or horizontal axis for getting good localization results. The anchor Setup shown in Figure 4.8 can also be used for 3D positioning if the calculated X and Y coordinate is combined with a distance measurement. Equation 4.10 can be formed based on the device position in Figure 4.8.

$$d1^2 = N_X^2 + N_Y^2 + N_Z^2 \tag{4.10}$$

The equation 4.10 can be reformed to 4.11, since the X and Y coordinate are known because of the equations 4.7 and 4.8, the Z coordinate of the node can be calculated.

$$N_Z = \sqrt{d1^2 - N_X^2 - N_Y^2} \tag{4.11}$$

The only disadvantage of this equation is that there is an ambiguity of the Z coordinate, it can be either negative or positive. Since the anchors are placed on the ceiling it's very unlikely that the node is placed above the anchors because of the wavelength of the UWB signal and its property of propagating through objects.

4.5.3 Combining the Data of Multiple Anchors

Since at least three anchors are required for doing a 3D positioning, most indoor localization systems consist of multiple anchors because it's very likely that one of the anchors won't be in line of sight and will only measure a reflection. Even if an anchor measures just a reflection, the measurement data seems to be valid at least from the UWB PHY and MAC point of view. Out of that reason, for filtering erroneous measurements, the measurement data of multiple UWB devices need to be combined for detecting if a NLOS signal was measured. Figure 4.9 shows the example of 4.8 with an additional anchor in the system. The anchors A1, A2 and A3 do have a LOS signal to the node, but the LOS of anchor A4 is blocked by a obstacle. This means, the first signal that is received by A4 from the node is a reflection caused by a wall.

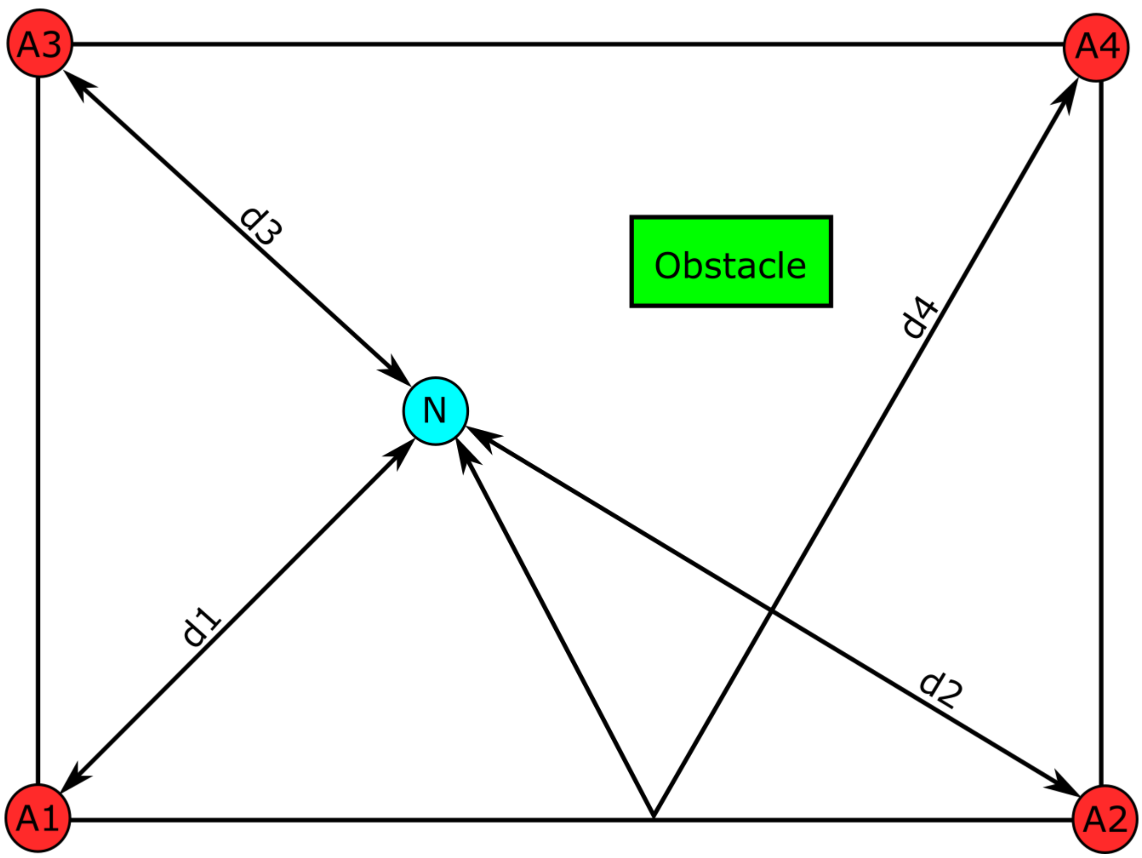

Figure 4.9: A example of a localization system with a NLOS signal.

For determining the NLOS anchors, the properties of a LOS signal in comparison to a NLOS signal can be used. Assuming a node is measuring an NLOS signal to a node, the measured distance will be always bigger than the real LOS distance. Since the UWB TOF measurement has a given accuracy, this property can be used for determining if a signal is in LOS or not. For example, Figure 4.9 consists of 4 anchors, this Group of 4 anchors can be split up into 4 sub-groups of 3 anchors, each group differs to other groups by at least one anchor. Based on the position of the anchors in the sub-groups and the measured distances to the anchors in the sub-groups, 4 different node positions can be calculated. Each node position belongs to one sub-group of anchors. Based on the calculated node positions, the ideal distance to the 4th anchor can be calculated and thus the NLOS Anchor can be estimated. If this calculated distance fits to the measured distance within given borders, none of the measured signals was NLOS. If the calculated distance is bigger than the actual measured distance, one of the members of the sub-group must have measured a NLOS signal because that's the only possibility to get such a scenario. For the checking of the distance a given measurement accuracy needs to be considered. If the measured distance of the fourth anchor is bigger than the calculated distance, the 4th anchor has measured a NLOS signal and it's very likely that the estimated position of the node is correct. The property that the UWB technology always gives the upper bound of the distance to a device, is especially an advantage if multiple distance measurements can be combined because every additional LOS measurement increases the probability of estimating the NLOS signals.

4.5.4 Combining Multiple LOS Measurements

If the LOS signals are estimated, the NLOS signals can be thrown away because the added value to the node localization is either meaningless or decreases the localization accuracy of the node. Also, the LOS signals need to be separated in more and less meaningful ones. Figure 4.10 shows an example consisting of four anchors placed on the same axis and a node that shall be localized. All the anchors have a unblock LOS to the node, even if the measurement error is the same for all the anchors the localization error differs from anchor to anchor depending on the position. Including a measurement error on equation

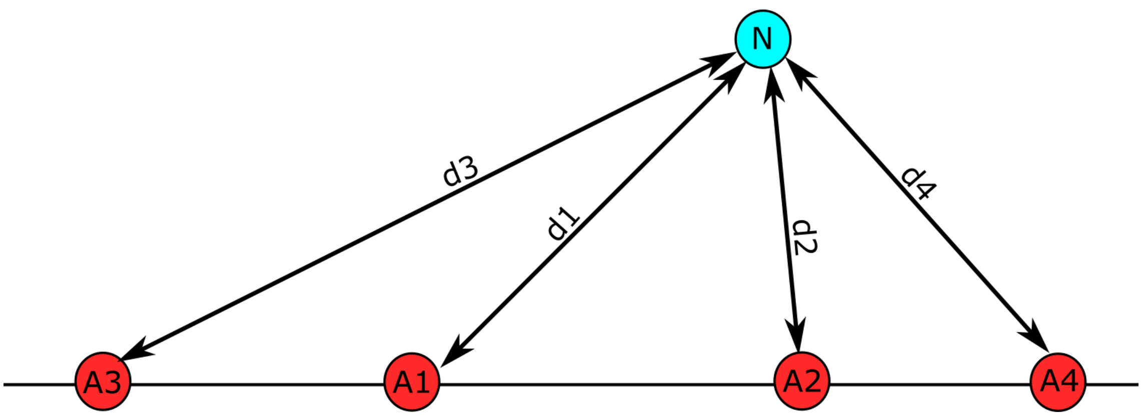

Figure 4.10: A example localization system with multiple LOS anchors placed on the same axis.

4.7 leads to equation 4.12.

$$N_X = \frac{(d2 + \Delta e)^2 - A2_X^2 - (d1 + \Delta e)^2}{2 \cdot A2_X} \tag{4.12}$$

Based on equation 4.12 the localization error along an axis is indirectly proportional to the distance between the anchors and directly proportional to the distance between the anchors and the node. This means, for a good accuracy of the node localization, the sum of the measured distances of an anchor pair divided by the distance between the anchors can be used as indicator for the accuracy of the device localization, since the real measurement error can't be estimated. If the stochastic error between two devices is known, this error can also be considered by multiplying it with the accuracy indicator.

$$I_{accruacy} = \frac{(d2 \cdot \Delta e_1) + (d1 \cdot \Delta e_2)}{2 \cdot A2_X} \tag{4.13}$$

Equation 4.13 shows how the accuracy indicator can be calculated. Basically it is the sum of the measured distance multiplied with the average measurement accuracy of the device, divided by the distance between the anchors. By comparing the accuracy indicators, the anchor pair that will lead to the best measurement accuracy can be estimated. For the position calculation the vertical and horizontal pair with the smallest indicator values shall be used.

4.6 Simulation of the Dynamic Channel Properties

Since a person will be moving continuously in a room, also the environmental influence will change. E.g. the closer a person comes to a signal blocking obstacle, the bigger becomes the impact of that obstacle on the UWB-channel. As mentioned in section 3.1.4 one of the biggest influences on the channel causes the person itself that carries the device. Out of that reason the person that carries the device has to be considered in the channel model. Figure 4.11 shows the impact of the Human body on the UWB signal. For the

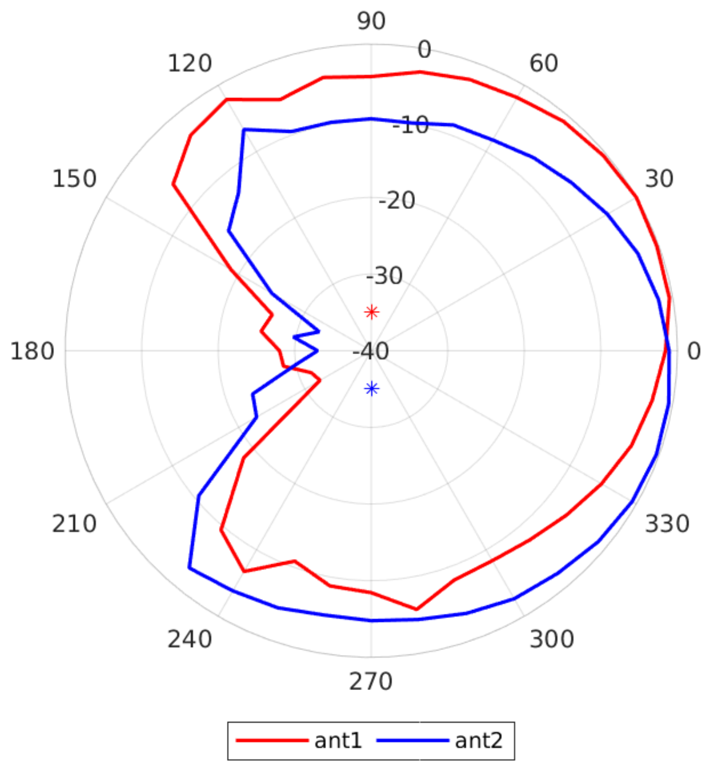

Figure 4.11: Influence of the human body on the antenna pattern.

measurement, a two antenna system was used with a distance of about $2.3\,cm$ which is roughly the $\lambda/2$ of the $6.5\,GHz$ bandwidth. A two-antenna setup was used for the measurement, for showing also the influences of a surrounding antenna on a receiving

antenna. A two antenna setup is very likely to be used if the costs for RF switches should be saved. The measurement result shows that the impact of one antenna on the other antenna causes a maximum antenna gain difference of about $5\,dB$ for the angles of $90\,^\circ$ and $270\,^\circ$. This gain difference depends on the location of the influencing antenna, e.g. if the antenna is located at the position of $270\,^\circ$, the received signal will be the weakest at $270\,^\circ$.
The influence of one antenna on the other antenna is negligible small in comparison to the impact of the human body on the UWB signal. The body of the person was placed at the position of $180\,^\circ$ degrees for the measurements. Since the antennas where just a half wavelength away from each other, the human body has had a roughly equal influence on both antennas. The minimum antenna gain that could be achieved is about $-35\,dB$ less than the antenna gain at the position of $0\,^\circ$ which can be considered as reference gain because of the point symmetrical antenna pattern that was used. Since there was a space between the device and the person, the weakening of the antenna gain stops at an angle of about $+-150\,^\circ$. If the UWB device is carried by the person the half of the antenna gain pattern will be damped by $-35\,dB$ which needs to be considered. These assumptions hold for persons carrying their device in the front pocket, if the device is carried in the back pocket the gain behaviour is rotated by $180\,^\circ$.

4.6.1 Combining the Movement Simulation with the Channel

Since the movement simulation has got a parameter for the direction a person is moving at, this parameter can be used for the creation of the dynamic antenna gain pattern.

Listing 4.12: Pseudo-code for determining the influence of the human body on the antenna gain.

```
getAntennaGain(node, anchor)
{
  LOSchannel = getRay(node,anchor);
  if(node.checkOrientationMatch(LOSchannel))
  {
    return node.antennaGain(LOSchannel);
  }
  else
  {
    return node.antennaGain(LOSchannel) - HUMAN_BODY_ATTENUATION;
  }
}
doRanging(node,anchor)
{
  attennaGain = getAntennaGain(node,anchor);
  startRanging(node,anchor,antennaGain);
  ...
}
```

Listing 4.12 shows the pseudo-code for combining the orientation of a person with the channel between two devices. A channel is defined by the starting and the end point the LOS signal, this means the node and anchor position specifies the channel. Since the device internal TX and RX antennas are close to each other's or even shared, the channel is the same for the transmission and the reception of a signal. This means, for the poll and response message the same channel can be assumed as static since the nodes are moving only a very little distance during the message transmission. If the orientation is pointing to the same direction as the LOS signal the Human body has no meaningful impact on the signal. If the orientation of a person is pointing in the opposite direction, the signal is damped by the measured human body attenuation of about $35\,dB$. This damping can be passed over to the PHY and the channel model for estimating the changes of the package error rate (PER) based on the signal strength.

4.6.2 Visualization and debugging

For the implementation and evaluation of the of different scenarios a graphical output is required for checking if the scenario was implemented properly. Since the graphical output is only needed for debugging, to keep the implementation effort as small as possible, the visualisation of the room including the node and anchors was implemented in the command line. Figure 4.12 shows an example of a room with one anchor and 50 nodes. The Room has the size of $1000\,cm \cdot 1000\,cm$ and the raster resolution is set to $50\,cm$.Every node gets a number for the identification and a channel quality indicator. For simplicity, the quality of the channel was split into three groups:

- Green, high channel quality, good signal strength
- Yellow, medium channel quality, still receivable LOS signal
- Red, low channel quality, no more receivable LOS signal

If a node is symbolized as green, the orientation of the node is pointing to the anchor and there are no or only very small objects in the line of sight to the anchor, e.g. the signal is travelling above the other persons or is only colliding for a very short distance. A node is marked as yellow if the signal direction and the orientation of the node doesn't fit or a bigger obstacle e.g. a person is in the path of the signal. The yellow zone has the same margin from a signal strength point of view as the green zone which is about the size of a human body attenuation. If a node is marked red, it is either orientated wrong and at least a person is in the signal path or two or more persons are in the signal path and the node is orientated correctly. At the borders of a room are indicators placed for the field a node is located at, the anchor is symbolized as a vertical and horizontal crossing line.

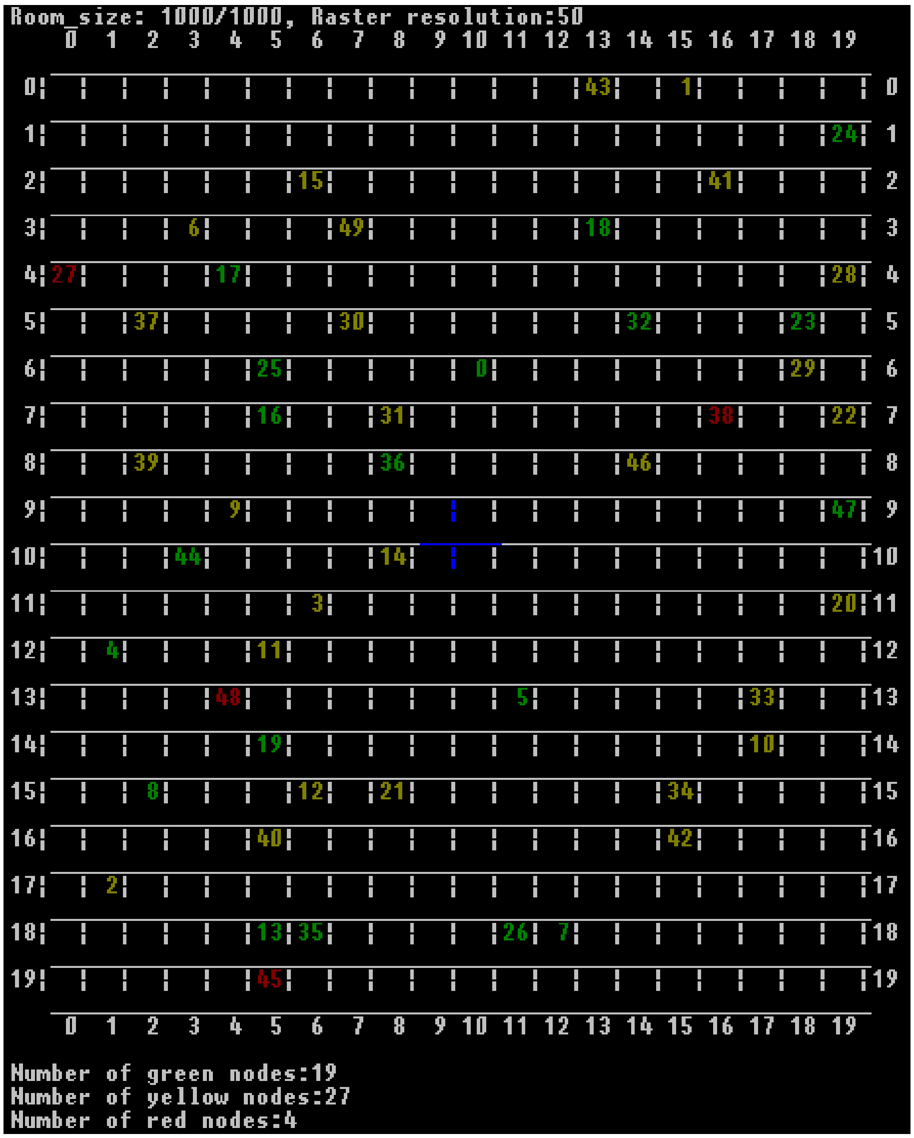

Figure 4.12: The vizualization of a room with the anchors and the nodes.

Chapter 5

Evaluation

This chapter describes the evaluating of the different explained methods from an accuracy and throughput perspective. The anchor density that is required for localizing every node is estimated by simulating use cases including simulating the nodes' movement within the room. Furthermore, the throughput of the different protocols is evaluated and compared between the protocols. Based on this evaluation the best protocol for a given use case is indicated.

5.1 Ranging Method comparison

The comparison of the ranging methods is graded based on two criterias, the measurement accuracy that can be achieved and the number of messages that is required for localizing a node.

5.1.1 SS-TWR

Figure 5.1 shows the error distribution of the SS-TWR for a constant processing and UWB-frame time. The maximum drift of the internal clock was the input parameter that has been changed from graph to graph. The graphs have been created with a clock-drift difference of $1\,ppm$. The possible clock-drift values were equally distributed between $0\,ppm$ and the maximum clock-drift. Based on multiple measurements the average, minimum and maximum error have been evaluated. Using that data, the error distribution is shown. The inaccuracy based on the timestamp generation unit can be neglected because it has a triangle shaped error distribution with a maximum error of $25\,mm$. This error is meaningless in comparison to the clock drift based error. Thus, the error distributions are represented as equal distributions. The plots show that a clock-drift of $3\,ppm$ can cause a maximum error of $25\,cm$. This means, the measurement inaccuracy is increased by a factor of 10 by having a maximum clock-drift of just $3\,ppm$. The probability of a measurement accuracy can be calculated by integrating over the accuracy borders. E.g. a integration from $0\,mm$ to $125\,mm$ of the $3\,ppm$ distribution leads to a probability of $50\,\%$. Figure 5.2 shows how the average error correlates with the clock accuracy limitations of the UWB devices. For higher clock-drift values, the correlation between the clock-drift and the average error is linear. The timestamp generation unit only creates a meaningful accuracy impact for clock-drift values between $0\,ppm$ and $1\,ppm$ which is very hard to achieve for a

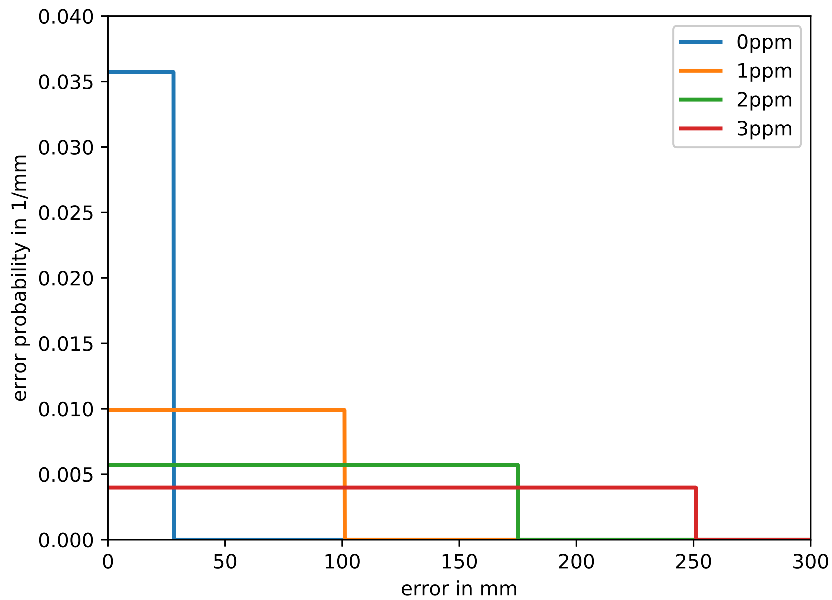

Figure 5.1: The measurement error distribution of the SS-TWR.

non-temperature compensated system [21]. The simulations show that the average error for a system with an maximum drift of 10 *ppm* is already about 40 *cm*. This is an average error increase by the factor 40 in comparison to a 0 *ppm* measurement. For the simulations, the UWB-frame length has been configured to 200 μs and the processing time was 300 μs. Depending on these values, the impact of the clock-drift on the measurement increases or decreases.

5.1.2 DS-TWR

As mentioned in section 2.1.2, the DS-TWR has a much weaker correlation between the clock-drift and the measurement accuracy. Figure 5.3 shows the error distribution of the DS-TWR in dependency to the relative clock-drift of two devices. The clock of one device was constantly set to 0 *ppm*, the clock-drift of the other device was adjusted for determining the impact of the clock-drift on the measurement accuracy. The physical distances between the devices has been set to 100 *m*. This is the worst-case scenario that can happen with common UWB devices because of the maximum receiver sensitivity [22]. The simulation results show that a clock-drift of 100 *ppm* causes an error of just a few millimetres which is less than the error caused by the timestamp generation unit. For a clock-drift of 1000 *ppm* the clock-drift error is about 5 *cm* which does have an impact on the total measurement error. Since UWB devices cannot communicate if their clocks are off by more than 100 *ppm*, the impact of the clock-drift based error is negligible for a DS-TWR based distance measurement.

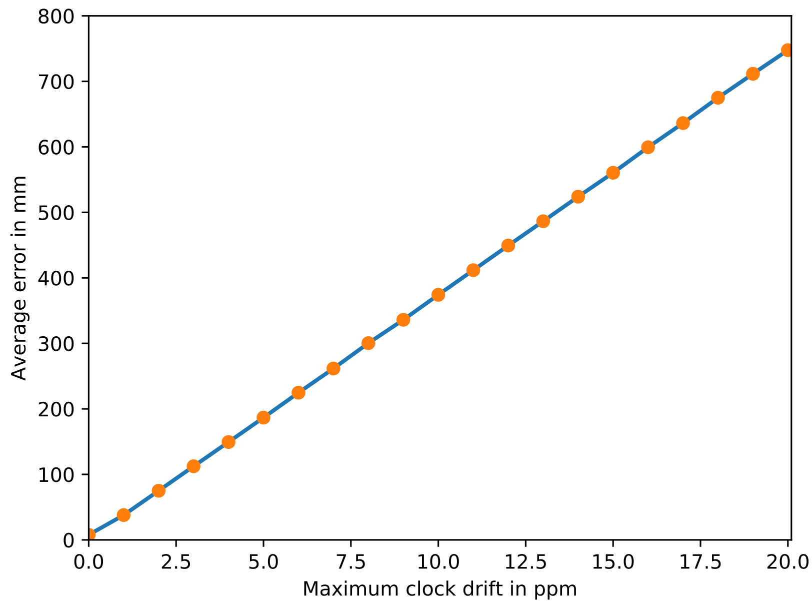

Figure 5.2: The average measurement error as function of the clock-drift.

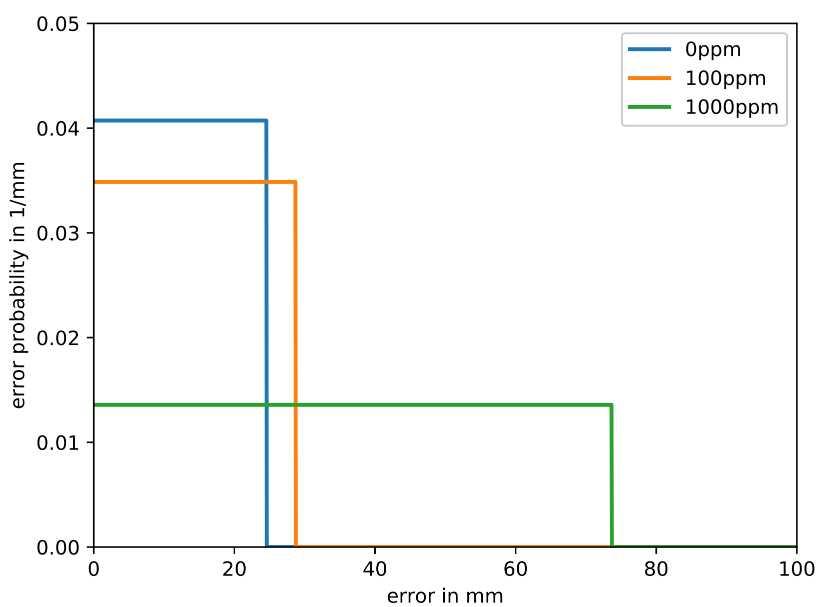

Figure 5.3: The measurement error distribution of the DS-TWR.

5.1.3 Asymmetric Ranging

From a measurement accuracy perspective, the asymmetric ranging shows the same behaviour as the DS-TWR. The simulation results show that the clock-drift can be compensated also for very asymmetric DS-TWR frames. If the first SS-TWR sequence has 100 times the size of the second SS-TWR, the impact of the clock-drift is still the same as for the a one-to-one frame ratio. This behaviour allows the Asymmetric Ranging to be used for big networks consisting of dozens of anchors. For the simulations the distance between the devices has been set to the worst-case scenario of $100\,m$ because of the given receiver sensitivity limitation [22]. The clock-drift of the node was set to $100\,ppm$ and the anchors were set to minus $100\,ppm$. In the simulation, a UWB-frame has the length of $200\,\mu s$. Between the first frame and the first response, a processing time of $300\,\mu s$ is considered. For the later UWB-frames, no processing time is required because all the other devices can use the time of the

first data processing and frame exchange for preparing their messages.

5.1.4 Spy Ranging

For the evaluation of the Spy Ranging, the clock-drift was compensated by the clock-drift indicator generated by the correlation unit as described in section 4.4.2. Since a Spy Ranging System according to section 3.1.3.2 consists of active and passive devices, the measurement accuracy of both devices has to be considered during the simulations. Between the active anchor and the node, a clock-drift compensated SS-TWR takes place. Figure 5.4 shows the error distribution of the SS-TWR including the error distribution of the timestamp generation units on both devices. Also, for the passive distance measurements, the distribution of the timestamp error is considered. The simulation results show that distance measurements between the active anchor and the node have a better accuracy than the distance measurement between the node and the passive anchor. The decreased accuracy can be explained by the fact that the passive distance measurement is based on the measurement of the active anchor. This means, the error of generating two receive timestamps needs to be added to the SS-TWR error. In contrast to the error distribution shown in Figure 5.1 and 5.3, Figure 5.4 shows error distribution under consideration of the signed error. For the simulation of the Spy Ranging based error, the UWB-frame length is configured to $200\,\mu s$ and the processing time has been set to $300\,\mu s$. The distance between all the devices is set to $100\,m$. The clock-drift is fully compensated by the PHY indicator.

5.2 Comparison of the Channel Consumption

The distinct ranging methods show different properties from an accuracy but also from the message afford perspective. Figure 5.5 shows the required number of messages for ranging between one node and several anchors by using different distance measurement techniques. The Spy Ranging method has been configured to use one active anchor instead of many for keeping the channel consumption as low as possible. The simulation results show that the number of required messages is linear to number of anchors for all the ranging methods except the Spy Ranging. Spy Ranging has a constant message count for

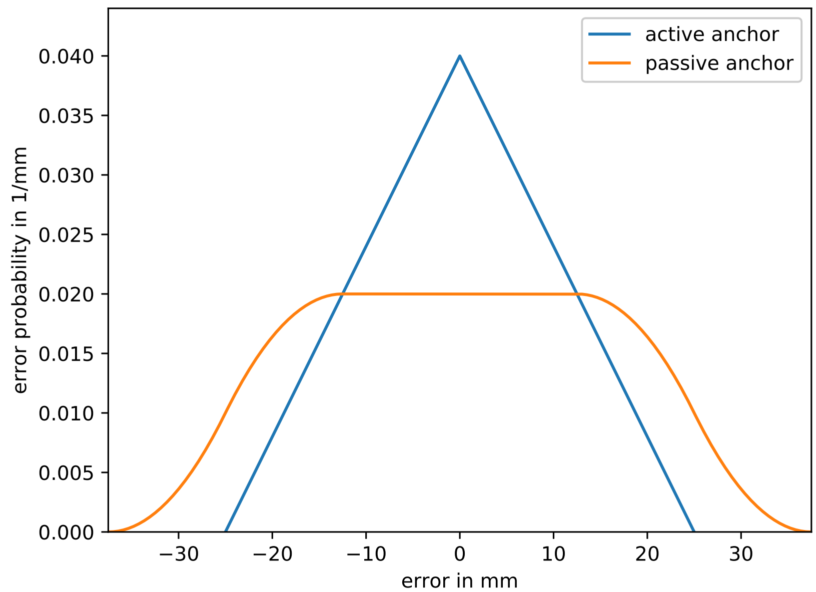

Figure 5.4: The number of messages required for localizing a node.

a constant number of nodes which makes it a good choice for networks with a big number of anchors. If the Spy Ranging method can't be applied because of given device limitations, the Asymmetric Ranging should be used because of the small number of messages that are required and the ranging accuracy. Figure 5.6 shows the number of messages that are required for localizing a given number of nodes by using different ranging methods. For being able to localize every node, the number of anchors was selected in a way that for every 20 nodes, an additional anchor needs to be added to the network. The figure shows a linear correlation between the Spy Ranging and the increasing number of nodes. The number of anchors has no impact on the message count if the number of active anchors is constant. For the simulation, only one active anchor has been used to keep the channel capacity as low as possible. The other ranging methods show a correlation of the message count and the number of anchors. Every time when an additional anchor is added to the system a significant increase of the number of messages can be observed. The maximum number localizations, correlates with the given channel capacity and the required number of messages for localizing the nodes. Thus, ranging methods like the DS-TWR are not suitable for localizing a big number of nodes.

5.3 Simulating Dynamic Channels influences

This section deals with evaluating the dynamic channel influences of a system consisting of multiple nodes and anchors. Two systems are evaluated. The first represents an office or a laboratory use case and the second system represents a large tracking system responsible for localizing hundreds of nodes by multiple anchors.

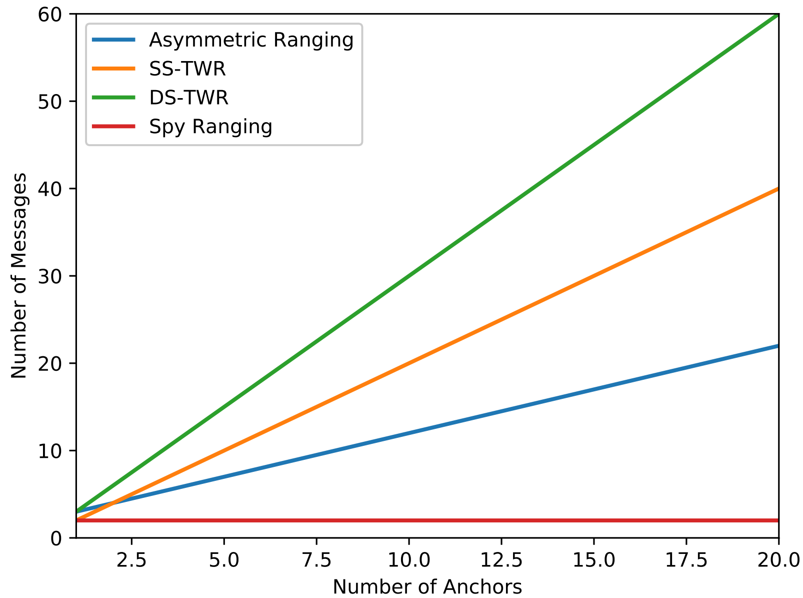

Figure 5.5: The number of messages required for localizing nodes by one anchor.

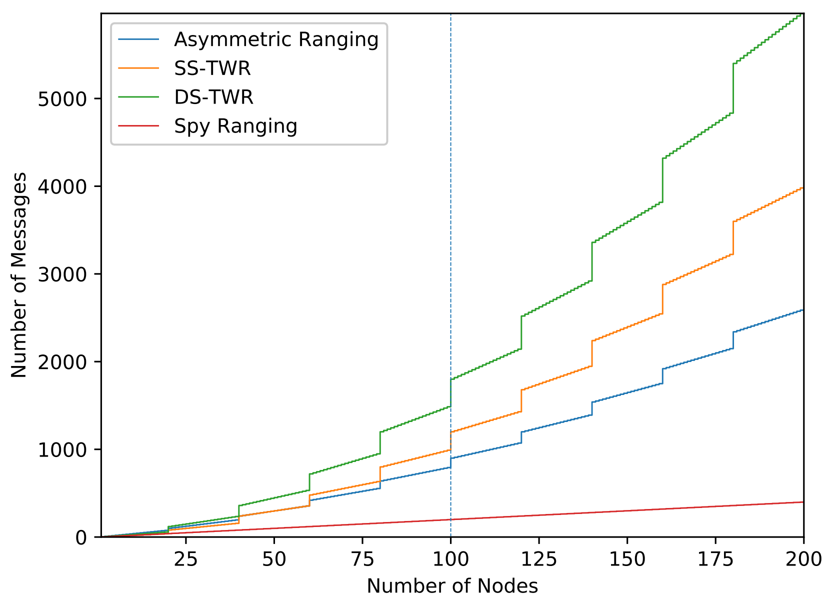

Figure 5.6: The number of messages required for localizing multiple nodes by multiple anchors.

5.4 Single-Anchor Simulation

A single-anchor system is the most simple system from localization algorithm and protocol perspective. The anchor in this system consists of three orthogonal orientated antennas that are used for measuring the phase difference of the incoming signal [23]. Based on the phase difference of the received signal, the incoming angle of the signal can be estimated around the antenna axis. By combining this data with the measured distance, the 3D position of a node can be estimated. A single-anchor system induces lower infrastructure costs in comparison to a multi-anchor system. Furthermore, for localizing a node, there is no need for complex post processing of the data generated by multiple anchors. Figure 5.7 shows the simulation results of a single-anchor system. The simulated room had the area of $10\,m \cdot 10\,m$ and a height of $3\,m$. The anchor is placed in the centre of the ceiling. The nodes

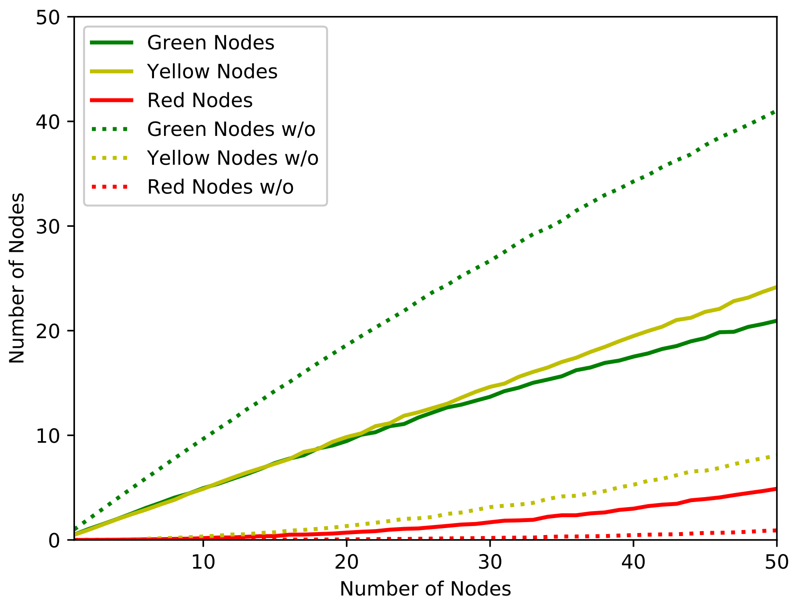

Figure 5.7: The distribution of channels for multiple nodes in a room.

are separated in green, yellow and red nodes according to the channel quality definition in section 4.6.1. The nodes in the simulated room are always added on a random position that is not occupied by any other device. For every number of nodes within the room, the simulation has been repeated 1000 times and for every rerun of the simulation, the nodes are added randomly again. The continuous lines show the simulation results considering the influence of the person carrying a node. The dotted lines show the simulation results considering the person that carries a node is not considered. All the simulation consider the persons that are moving within the room and their impact on the UWB channel. The simulation results show the impact of the person that carries the device on the UWB-channel. Most of the devices do have a good channel if the orientation/body of the person carrying the node is not considered. If the person carrying the node is considered, the majority of the devices have a non-ideal channel. Also, the number of nodes that can't communicate is increased drastically if the orientation of the attached person is considered. Especially if the density of nodes within the room increases, the number of nodes that can't be localized increases drastically.

Figure 5.8 shows the ratio of nodes and their channel quality with and without considering the person that carries the device that needs to be localized. The node distributions

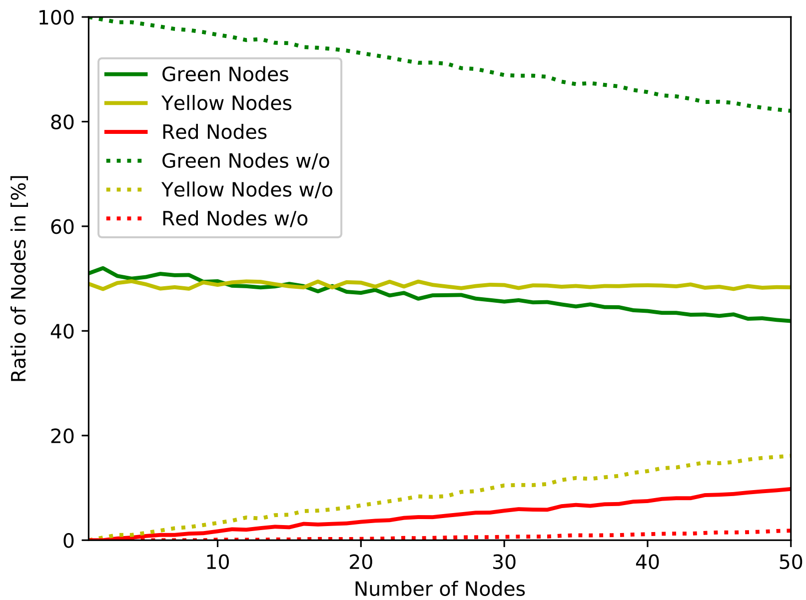

Figure 5.8: The relative distribution of nodes with their according channel, dotted lines without consideration of the person carrying the node, solid lines with consideration of the person carrying the node.

show that at the beginning all nodes do have an ideal channel if the person carrying a device is neglected. If the person carrying the device is taken into account, already 50 % of the devices have a weak channel, thus, increasing likelihood of a weak channel. The more nodes are placed in the room, the bigger is the likelihood that a node can't be localized. For example, if 50 persons with an attached node are placed in a $10\,m \cdot 10\,m$ room, approximately 10 % of the persons can't be localized. In comparison, if only 20 persons are placed in the same room merely 4 % of the person can't be localized. This behaviour scales also for bigger rooms with a comparable node and anchor density.

Figure 5.9 shows the comparison of an one-anchor based localization system with a four anchor based location system. The results of the one anchor system are reused from Figure 5.8, the orientation of the person with the attached device is considered as well. The room size of the four-anchor system is the same as for the one anchor system. The anchors are placed on the corners of a $5\,m \cdot 5\,m$ square located in the centre of the room. In the simulation results a node is green if the node has a line of sight signal to at least three anchors and the majority and two of these three anchors do have a good channel to the node. If two of the three anchors do have a weak channel, the node is yellow and if the node has two or less anchors in LOS the node is red. The simulation results show that the system consisting of four anchors has better properties from a signal quality perspective, since most of the nodes are green from the beginning. The probability of not being able to localize a node is approximately the same as for the single-anchor system. This means, a single-anchor system with three antennas has some disadvantages from a signal quality point of view compared to a four-anchor system with single antenna anchors. Nevertheless the infrastructure costs and the data processing afford is increased by using a four anchors. Additionally, depending on the ranging method, the message afford for localizing a node by a single-anchor system is drastically decreased in comparison to a four anchor based system. Depending on the available anchor technology and the environment, a single-

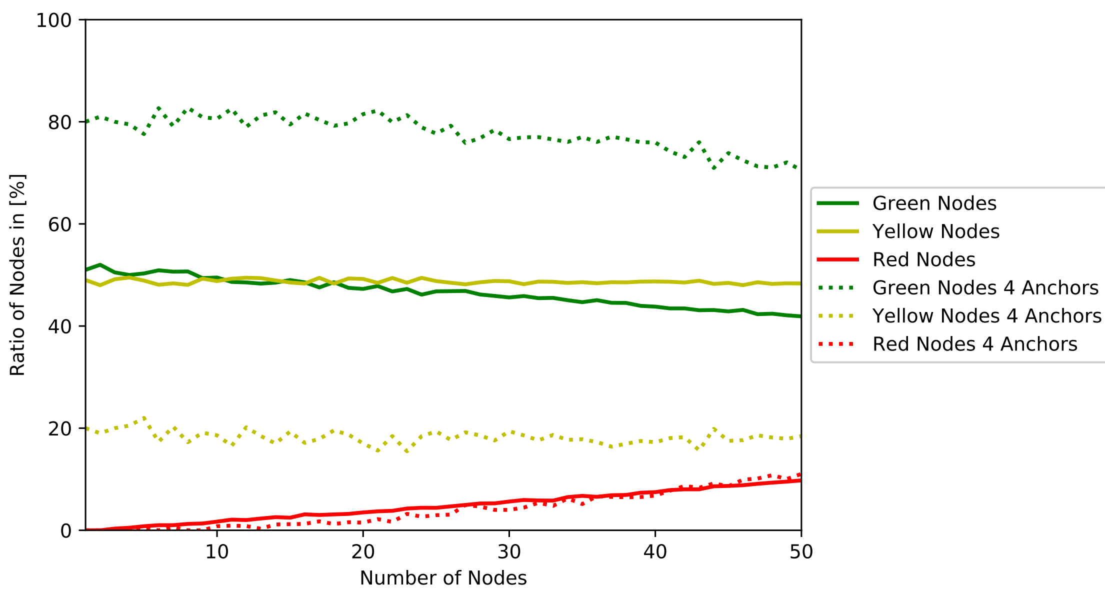

Figure 5.9: The comparison of a single-anchor system to a four-anchor system, dotted lines are for four-anchor system, solid lines for single-anchor system.

anchor system could make more sense for localizing a small number of nodes within a room.

5.5 Big System Simulation

For rooms with a big size, a single-anchor system based on three antennas is not suitable. The distance from a node to the anchor has a linear impact on the AOA based localization error e.g. $(sin(1°) \cdot 1\,m) = 1/10 * (sin(1°) \cdot 10\,m)$ .The distance measurement error between a node and an anchor is nearly constant for a variable distance between the devices. So a multi-anchor, single antenna system achieves a better measurement accuracy. Out of that reason such an approach has been used for the evaluation of the big scale localization system.
Figure 5.11 shows the simulation results of a room with the dimensions of $30\,m \cdot 30\,m$ and

a height of $3\,m$. The anchor placement is shown in Figure 5.10, the blue anchor in the center of the room is only placed if a seven-anchor system is simulated.

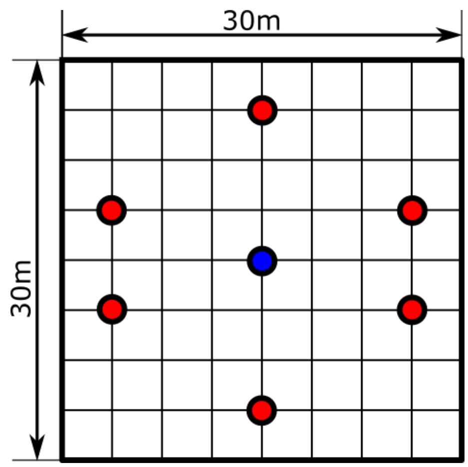

Figure 5.10: The Anchor placement of the big scale localization simulation, the blue anchor in the middle is optional.

The nodes have been added randomly in the room in groups of tens, every number of nodes placed in the room has been simulated 50 times with a different placement of the nodes. The simulation results show the distribution of the channel properties for a six and a seven-anchor system. Bot simulations show the similar results if not too many nodes are added to the room. Especially at the beginning, the probability of that a node can't be localized is nearly zero. The more nodes are added to the system, the higher becomes the probability of not being able to localize a node. Especially a good distribution of the anchors within the room is important for covering all possible node locations. The comparison of the six-anchor simulation with the seven-anchor simulation shows that adding just one anchor in the middle of the room can increase the probability of a node getting localized by more than $5\,\%$.

5.6 Evaluation of Localization Accuracy

Simulations show that a node can be localized with an accuracy of $20\,cm$ or better, if three or more anchors are able to range with the node. Since it's not possible to cover every node with a good signal quality, not every node can be localized at every moment in time. A possibility for handling this problem would be to increase the anchor density until every node can range to three anchors. This would lead to a very high anchor density and thus to high infrastructure costs. Also, depending on the ranging method that is used, adding more anchors might help for getting a good signal quality to every node, but it limits the maximum number of nodes from a channel capacity perspective. This means, the nodes may be in LOS but there is no time left for ranging to the nodes because all the ranging slots are already occupied. Additionally, more anchors lead to a higher amount of data that needs to be shared and processed. Especially more complex localization algorithms that are based on matrix inversion do have an non-linear runtime time correlation regard-

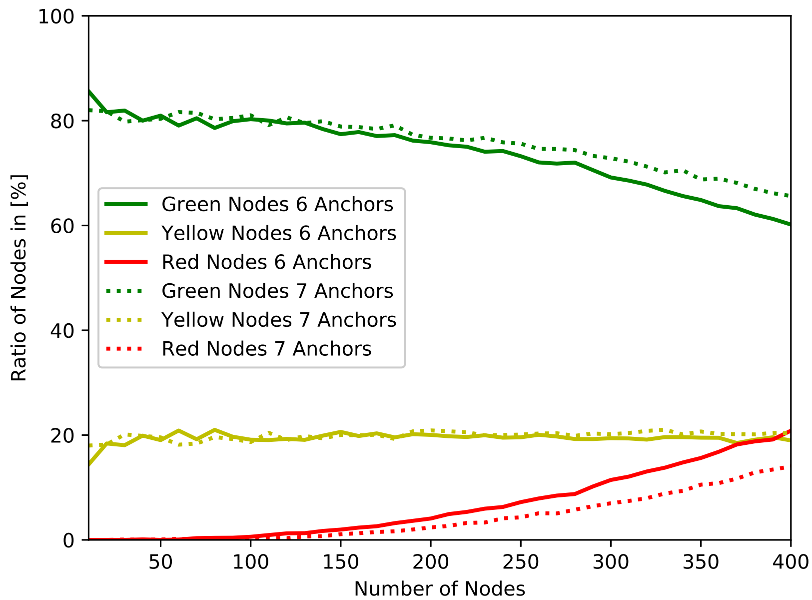

Figure 5.11: The simulation results of a big room with 6 and 7 anchors, dotted lines represent the 7 anchor system, solid lines represent the 6 anchor system.

ing the number of distance measurements per node [24].

Because of the limitations that come with extensively adding anchors to a localization system, the problem of not being able to localize a node should be solved by considering the movement of the nodes. The location of a person has a big impact on the channel of the attached node and persons need to move for changing their position. Out of that reason, it's very likely that a person that moves will cause a change from a NLOS to LOS signal. If the person is not moving, the channel of the attached node has only changed slightly because of the moving environment, but therefore also the position of the person stays the same. This means, there is no increase of the localization error since the position hasn't changed. The movement simulations show that less than one percent of the nodes is still not trackable after changing its position. After two changes the probability that a node can't be localized is nearly zero. This means, the average localization error is just slightly increased because of the small likelihood of not being able to localize a node. Depending on the use case requirements, solving the NLOS problem in time might be an cost effective solution. Also, less than one percent of the devices do have a localization error bigger than 50 cm based on the simulation results. Especially if the anchors are placed properly,the devices that can't be tracked will be located at uncritical regions. For example in secure door access scenarios it doesn't matter if a devices in the middle of the room can't be tracked. The only requirement is that the node can be localized if it is close by a door, so the door can open according to the access permissions.

5.7 Protocol Analysis

This section is deals with simulating the single- and multi-anchor systems from a protocol point of view. Based on the simulation results of the channel quality simulations, input data for the protocol simulation is generated which has a meaningful impact on the localization throughput. The Superframe protocol is not evaluated for multi anchor systems since it's not suitable for indoor localization systems. This throughput limitation is caused by the limited number of ranging slots and the short random access phase described in section 2.3.2.2. Table 5.1 shows the timing behaviour the protocol simulations are based on.

	Duration
UWB-frame	$1000\,\mu s$
Processing time	$300\,\mu s$
Guard time	$20\,\mu s$
Considered clock-drift	$100\,\mu s$
Localization update rate	$1000\,ms$

Table 5.1: The configuration of a UWB-frame.

The big system use case is based on the anchor setup described in section 5.5. The small system/office use case is based on the setup described in section 5.4. Table 5.2 shows the parameters that were used for evaluating the protocols.

	Number of anchors	Room size
Office use case	1/4	$10\,m \cdot 10\,m$
Big System	6	$30\,m \cdot 30\,m$

Table 5.2: The configuration of the simulated systems.

5.7.1 Pure ALOHA

Figure 5.12 shows simulation results of the throughput analysis of the office use case. The localization update period was set to one second. The simulation results show two different setups of the room. The dotted line shows the evaluation of the single-anchor system, the continuous line shows the simulation results of a four-anchor system. Both systems use the Pure ALOHA protocol for the message exchange. Since the Asymmetric Ranging and the Spy-Ranging require a multi-anchor system, this methods aren't evaluated for the single-anchor simulations. The SS-TWR and DS-TWR methods show a nearly optimal throughput for the single-anchor simulation. The high throughput is caused by the big localization update rate of one second, since only a maximum of 50 nodes needs to be localized, a collision is very unlikely. The UWB-channel has the biggest impact on the number of nodes that can be localized by using a single-anchor system. The four-anchor system does have a nearly equal behaviour from a channel quality perspective, but the

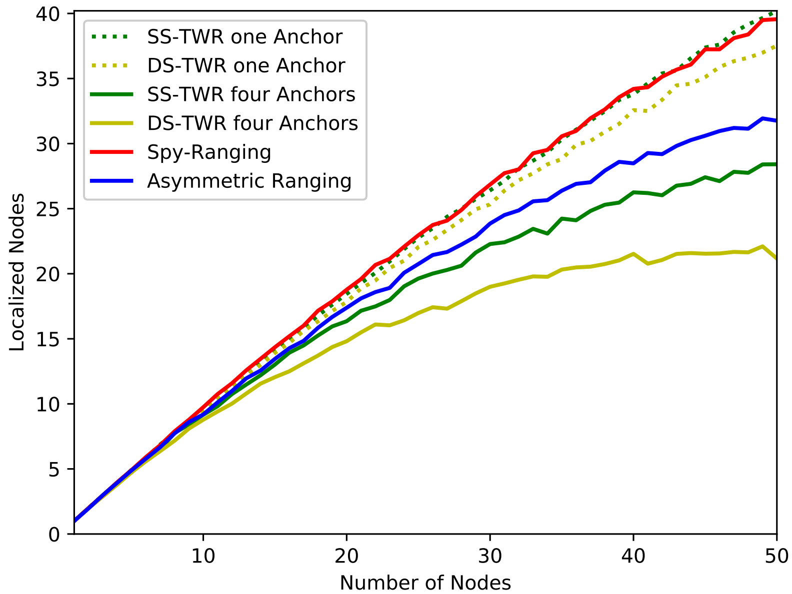

Figure 5.12: Localization throughput of the Pure ALOHA protocol, office use case simulation, the dotted line represents the single-anchor system, the solid line represents the multi-anchor system.

number of required messages is increased drastically in comparison to the single-anchor simulations. Hence, the collision likelihood is increased for the SS-TWR, the DS-TWR and the Asymmetric Ranging. Only the Spy-Ranging shows a throughput rate that is compare able to the SS-TWR. The reason for this high throughput rate is the Spy ranging's ability to range to all anchors with a single message exchange. This drastically decreases the collision likelihood.

Figure 5.13 shows the simulation results of the big system use case. The throughput results were created by evaluating the Pure ALOHA protocol. The DS-TWR shows the worst behaviour from a throughput perspective because of the time afford that is required for making a localization between the node and all the anchors. Assuming a DS-TWR ranging to one anchor takes $3\,ms$ a localization of all the nodes takes already about four seconds. Thus, there isn't enough channel capacity left for a ranging between the anchors and all the nodes. This results in a high collision likelihood and, thus, in a low throughput rate. The time consumption for a SS-TWR and Asymmetric Ranging is slightly reduced, hence, increasing the throughput. The Spy Ranging is the best ranging method from a channel consumption perspective. Its throughput rate is the higher than the throughput of all the other ranging methods. Nevertheless, the Pure ALOHA protocol only manages to localize less than half of the nodes in the room

5.7.2 Slotted ALOHA

Figure 5.14 shows the simulation results of a office use case system. The single-anchor

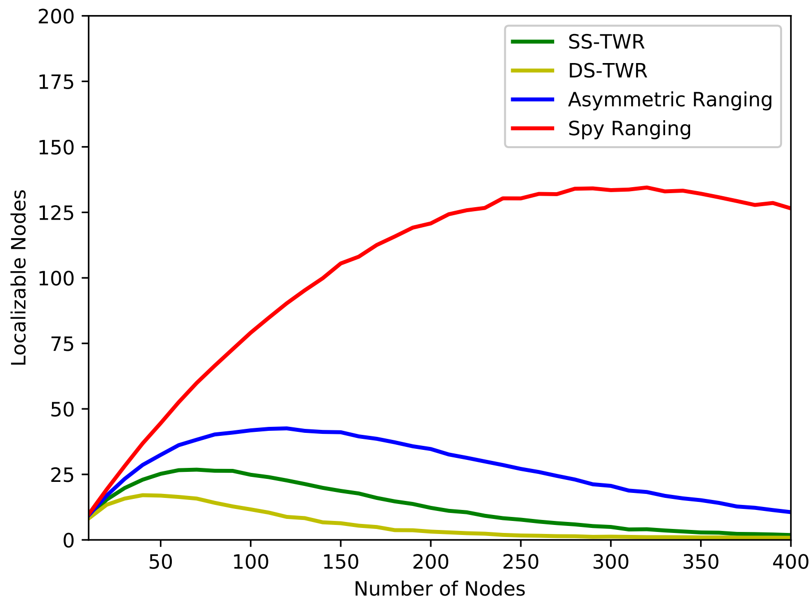

Figure 5.13: Localization throughput of the Pure ALOHA protocol, big system use case.

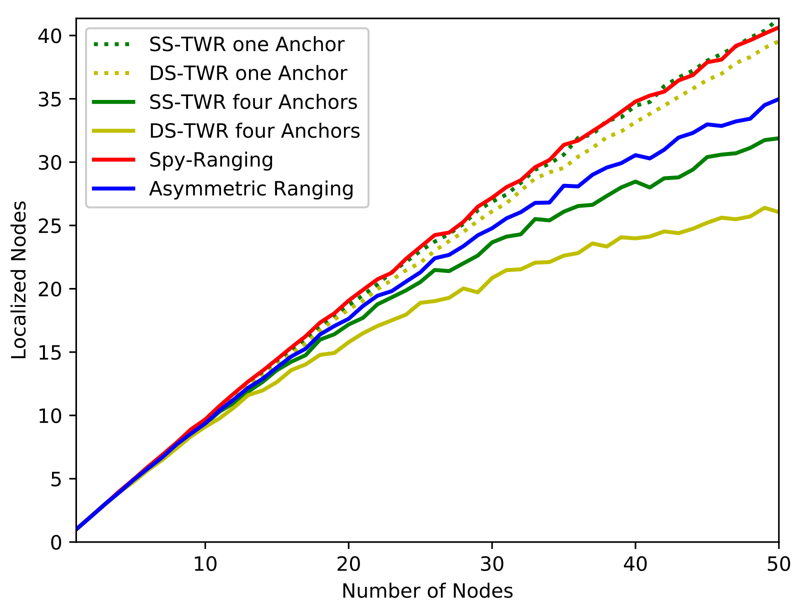

Figure 5.14: Localization throughput of the Slotted ALOHA protocol, office use case simulation.

system approach shows only a small difference from a throughput perspective to the SS-TWR and DS-TWR. The reason for the high throughput rate is that only a small part of the channel is used. Hence, for small number of anchors, the Slotted ALOHA protocol shows no significant change to the Pure ALOHA protocol from a throughput perspective.

Figure 5.15 shows the simulation results of the big system use case. The results show the throughput rate of the 6 anchors system by using the Slotted ALOHA protocol. The

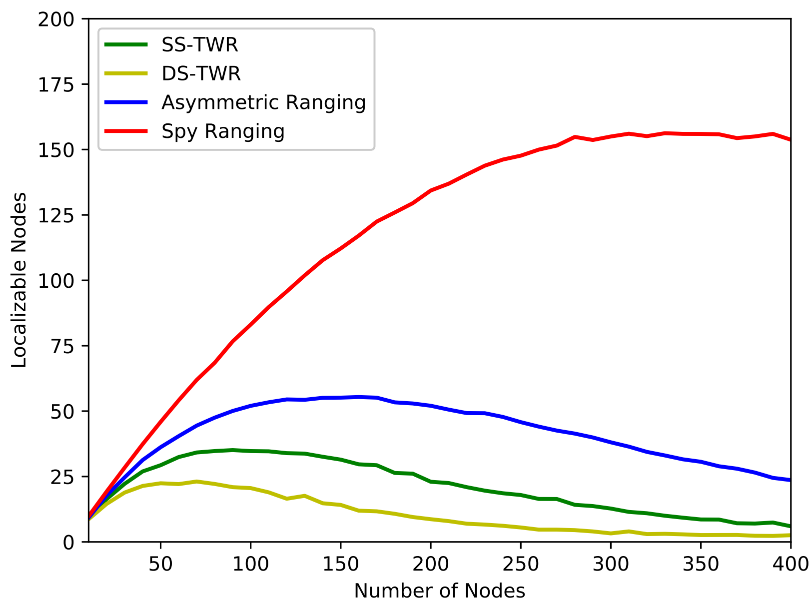

Figure 5.15: Localization throughput of the Slotted ALOHA protocol, big system use case simulation.

DS-TWR shows the worst results for localizing a big number of nodes. This behaviour was expected since it has the biggest channel consumption. The SS-TWR and the Asymmetric Ranging show an increased throughput. However, as the channel is fully occupied, the collision likelihood is still high. Also the Spy Ranging simulation results show an suboptimal throughput compared to the theoretical channel capacity. Overall the Slotted ALOHA protocol throughput rate is higher than the Pure ALOHA throughput. For localizing every node, many distance measurements need to be repeated which is not feasible for systems with a high number of nodes.

5.7.3 Superframe Protocol

Figure 5.16 shows the localization throughput of the Superframe protocol. Since the protocol was designed for small networks, only the one anchor network has been evaluated. The results show that the throughput rate is the same for the SS-TWR and the DS-TWR. This is caused by the random access based request of the ranging slots. If a slot is assigned, the ranging can take place without any collision, so the ranging method doesn't matter for a single-anchor system. Because of the short random access phase of the Superframe

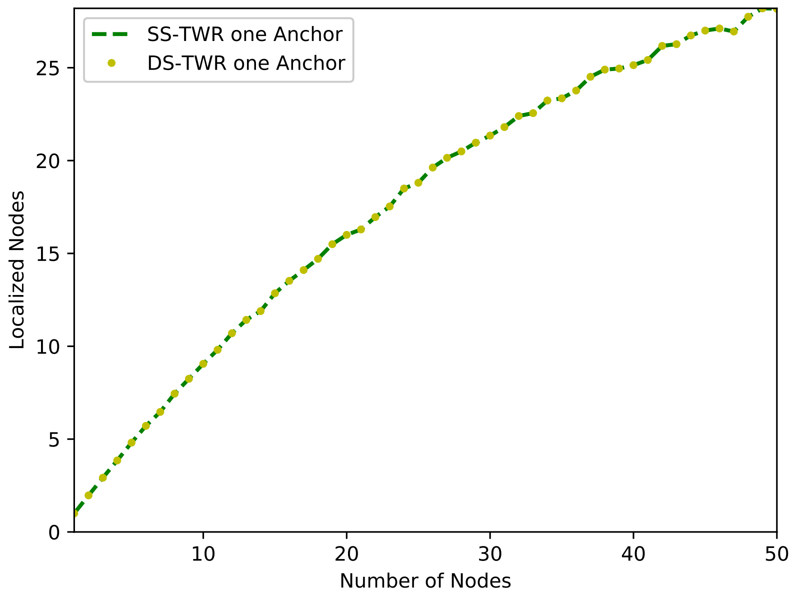

Figure 5.16: Localization throughput of the Superframe protocol, simulated office use case.

protocol the collision likelihood is increased drastically in comparison to the ALOHA protocols.

5.7.4 TDMA Protocol

Figure 5.17 shows the office use case simulation results using the TDMA protocol for the ranging slot generation. The simulation results show that all the ranging protocols for the single- and four-anchor simulations show the same behaviour. This effect is caused by the time-based slot generation including a guard time which is described in section 4.3.5. Every device gets a slot for distance measurements. Since the guard time considers the clock inaccuracy of the devices, inter-slot collisions aren't possible. Because there is also enough channel capacity left, every node can range with every anchor. The non-ideal channel is the only reason why not every device can be localised. The four-anchor system and the single-anchor system differs a bit from an UWB-channel perspective. Out of that reason the throughput simulations of both systems differ slightly. The ranging methods show no difference from a throughput perspective because of the available channel capacity. Figure 5.18 shows the simulation results of the big system use case. The simulations show the throughput of the different ranging methods. The TDMA protocol was used for scheduling the message exchange. The simulation results show that the DS-TWR ranging has the smallest throughput rate because of the high message afford. Since the TDMA protocol schedules only the available places in the network, the maximum throughput rate is limited by the channel capacity. Since the TDMA protocol doesn't allow collisions, the protocol-based throughput rate is constant when the maximum number of nodes is scheduled. Because of the impact of additional nodes on the UWB-channel, the throughput

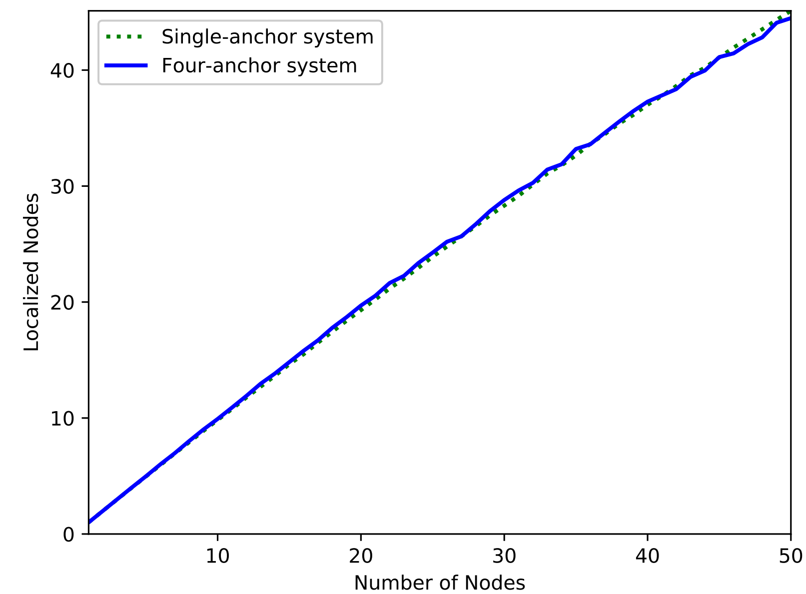

Figure 5.17: Localization throughput of the TDMA protocol, simuated office use case.

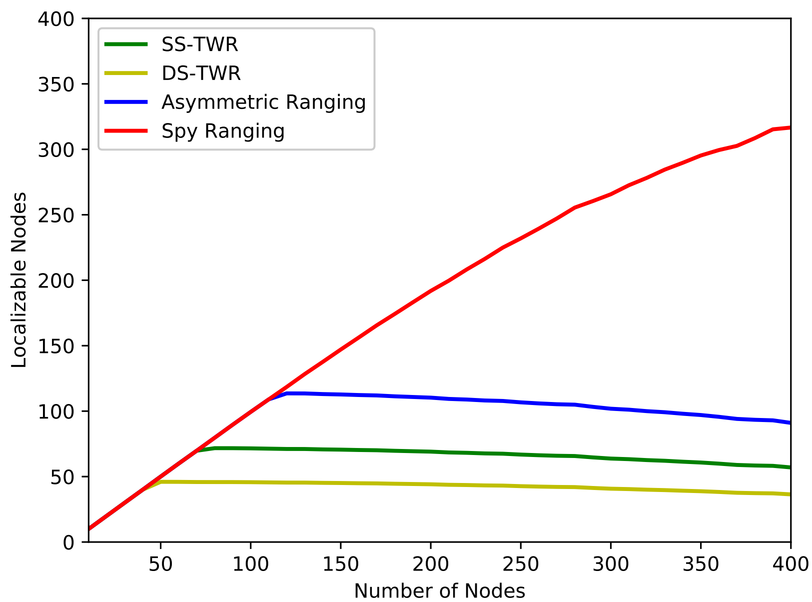

Figure 5.18: Localization throughput of the TDMA protocol, big system use case simulation.

rate decreases slightly by adding more nodes. Also, the DS-TWR and the Asymmetric Ranging require too much channel capacity for localizing every node. The available channel capacity is only sufficient for the Spy Ranging. Out of that reason, the only limiting factor of the throughput is the UWB-channel if the Spy Ranging method is used. For solving this problem a multi channel system can be used. This requires more complex anchors that can range to two devices at the same time which leads to additional infrastructure costs.

Chapter 6

Conclusion

The Matlab based interference simulations and assumptions about the UWB-channel could be validated by making reference measurements based on a real UWB-transceiver system [11], [25]. The results of the simulated use cases show, that the best combination of the ranging method and the used protocol, depends on the system and the number of nodes that need to be tracked. For small rooms, a single-anchor system is better from a infrastructure perspective. Also, to localize a small number of nodes, the Pure ALOHA or Slotted ALOHA protocol shows comparable results to the TDMA protocol by having less implementation afford. For multi-anchor systems the Spy Ranging method shows the best throughput rates because of the constant message afford. This means, the decision of the protocol that is used can be made based on the number of nodes that need to be localized if the Spy ranging method can be used. For big systems with hundreds of nodes, the a TDMA protocol shows the best results because of the time slot based collision avoidance. The decision of the ranging method can be made on the available channel capacity. In general, if all nodes are able to range with all the anchors from a channel capacity perspective, the TDMA protocol is able to schedule the message exchange without any collisions. Nevertheless, Spy Ranging has an advantage from a power consumption perspective which can be important for battery powered devices. Also, having more channel capacity left, allows also for a later increase of the number of nodes. Out of that reason, the Spy Ranging method should be used if the achieved localization accuracy is within the requirements.

6.1 Simulation Limitations

The current simulation model is based on one logical channel. This means systems consisting of thousands of nodes can't be simulated because such systems would require multiple channels. Also the collision detection only checks for collisions that happen on the selected channel. More complex multi channel system can cause inter-channel interferences which can limit the throughput. Since currently used simulation models doesn't support the inter-channel collision detection such systems can't be simulated. Another limiting factor of the currently used simulation model is the run-time of the used collision detection algorithms. The currently used algorithms do have a upper run-time complexity boarder of $O(n) = n^2$, where n represents the the number of transmitted messages. By simulating bigger systems, the runtime can increase drastically which is a disadvantage from an usability perspective. Also, the implemented ray tracing algorithms do scale linear with the room size, the number of nodes within the system and the number of anchors. In real world scenarios the required number of anchors within in the system and the room size do have a linear correlation with the number nodes. This implies that the runtime behaviour of the currently used ray tracing algorithm does have a runtime complexity boarder of $O(n) = n^3$ where n represents the number of nodes. So the Ray tracing algorithm has a even bigger impact on the over all simulation run time than the collision detection algorithm.

6.2 Future work

The current SystemC simulation can be only used for simulation indoor localization systems that are based on the UWB technology, but there are many other impulse radio based applications that can be included in the simulation model. Especially wireless body networks should be considered during the simulation form an interference perspective because of the proximity to the UWB radio. Especially pulse based body networks could use the same channel as UWB-based indoor localisation systems which make them a to a potential interferer with a big impact on the localization throughput [26]. Also, a extension of the current device models by other wireless or wired interfaces can enable the SytemC model for simulating more complex use cases. Especially Bluetooth based systems are used as common interface for smart home systems since they are available in most of the devices. Many smart homes do already have an Bluetooth based door access system [27]. Such systems can be extended by an UWB systems for offering a more secure and precise service. Bluetooth Low Energy (BLE) can be also used as RSSI based low power wakeup for triggering an UWB based localization. Such a BLE system can have a meaning full impact on the power consumption of the system because a UWB-based ranging can be triggered on demand. Also, BLE can be used for negotiating the UWB-channel parameters for later ranging sessions which increases the system flexibility from a channel parameter perspective. One of the most auspicious use cases of UWB systems are impulse based indoor radar applications [28]. Such radar applications can be used for a movement detection or a monitoring of security areas. Based on such a monitoring system a UWB based localization can be triggered or also smart bulbs could switch on in a hallway. Since such systems are also transmitting in the UWB channels, the impact on parallel running localization system needs to be considered.

Bibliography

[1] Terence Barrett, "History of Ultra Wideband Communications and Radar: Part I, UWB Communications," Jan. 2001. [Online]. Available: https://pdfs.semanticscholar.org/fdfd/ecba84442dc2e33fe12a42cfd6f337252b1a.pdf

[2] IEEE Computer Society, "802.15.4-2015 - IEEE Standard for Low-Rate Wireless Networks," IEEE, Tech. Rep., 2015. [Online]. Available: https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=7460875

[3] Decawave, "APS013 APPLICATION NOTE, The implementation of two-way ranging with the DW1000," Tech. Rep. [Online]. Available: https://www.decawave.com/wp-content/uploads/2018/08/aps013_dw1000_and_two_way_ranging_v2.2.pdf

[4] Decawave, "APS011 APPLICATION NOTE, SOURCES OF ERROR IN DW1000 BASED TWO-WAY RANGING (TWR) SCHEMES," Tech. Rep. [Online]. Available: https://www.decawave.com/wp-content/uploads/2018/08/aps011_sources_of_error_in_twr.pdf

[5] CSE IIT, Kharagpur, "Medium Access Control (MAC) Techniques," Tech. Rep. [Online]. Available: https://nptel.ac.in/courses/106105080/pdf/M5L2.pdf

[6] IEEE Computer Society, "IEEE Std 802.15.8™-2017, IEEE Standard for Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Peer Aware Communications (PAC) Sponsored by the LAN/MAN Standards Committee IEEE 3 ," IEEE, Tech. Rep., 2017. [Online]. Available: https://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=8287784

[7] Hoon Oh et al., "Design and Implementation of a MAC Protocol for Timely and Reliable Delivery of Command and Data in Dynamic Wireless Sensor Networks," Tech. Rep., 2013. [Online]. Available: https://www.researchgate.net/publication/257300165_Design_and_Implementation_of_a_MAC_Protocol_for_Timely_and_Reliable_Delivery_of_Command_and_Data_in_Dynamic_Wireless_Sensor_Networks

[8] omnetpp.org, "Omnet++ Simulation Manual," no. 355, Feb. 2018. [Online]. Available: https://doc.omnetpp.org/omnetpp/manual/#sec:introduction:what-is-omnetpp

[9] Till Knollmann, "Einführung in OMNeT++ und OverSim," Tech. Rep., May 2017. [Online]. Available: https://cs.uni-paderborn.de/fileadmin/informatik/fg/ti/Lehre/SS_2017/VAD/Einstieg_in_OMNetpp.pdf

[10] Michael Kirsche and Matti Schnurbusch, "A New IEEE 802.15.4 Simulation Model for OMNeT++ / INETv," *arXiv:1409.1177v2 [cs.NI] 13 Sep 2015*, 2015. [Online]. Available: https://arxiv.org/pdf/1409.1177.pdf

[11] David Veit et al., "A Simulation Environment for UWB Hardware Development and Protocol Design," Tech. Rep., 2019, under review.

[12] IEEE Computer Society, "802.15.4a-2007 - IEEE Standard for Information Technology - Telecommunications and Information Exchange Between Systems - Local and Metropolitan Area Networks - Specific Requirement Part 15.4: Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low-Rate Wireless Personal Area Networks (WPANs)," IEEE, Tech. Rep., 2007. [Online]. Available: https://ieeexplore.ieee.org/document/4299496

[13] Decawave, "HOW TO USE, CONFIGURE AND PROGRAM THE DW1000 UWB TRANSCEIVER," Tech. Rep. [Online]. Available: https://www.decawave.com/sites/default/files/resources/dw1000_user_manual_2.11.pdf

[14] Ivan A. Mantilla-Gaviria et al., "Localization Algorithms for Multilateration (MLAT) Systems in Airport Surface Surveillance ," Tech. Rep., May 2014. [Online]. Available: https://www.researchgate.net/publication/259849645_Localization_algorithms_for_multilateration_MLAT_systems_in_airport_surface_surveillance

[15] HK Government, "Trilateration and Global Positioning System ," Tech. Rep. [Online]. Available: https://www.edb.gov.hk/attachment/en/curriculum-development/kla/ma/res/STEM%20example_sec_trilateration%20GPS_eng.pdf

[16] Santhosh Krishna, "Ultra Wideband Channel Model for IEEE 802.15.4a and Performance Comparison of DBPSK/OQPSK Systems," Tech. Rep., 2014. [Online]. Available: https://pdfs.semanticscholar.org/7611/eef25de5b3900dca578adf9440e6db3af276.pdf

[17] Stephen Knox, "PyNSim Architecture," Tech. Rep., 2014.

[18] Hannes Muhr, "Einsatz von SystemC im Hardware/Software-Codesign," Tech. Rep., 2000. [Online]. Available: https://publik.tuwien.ac.at/files/pub-et_3966.pdf

[19] "Electronic System Design using C and SystemC," Tech. Rep. [Online]. Available: https://www.uni-ulm.de/in/mikro/lehre/vorlesungen-ws/electronic-system-design-using-c-and-systemc/

[20] IEEE Computer Society, "IEEE Std 802.11-2016, Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications," IEEE, Tech. Rep., 2016. [Online]. Available: https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=7786995

[21] Hui Zhou et al., "Frequency Accuracy and Stability Dependencies of Crystal Oscillators," Tech. Rep., 2008. [Online]. Available: http://www.headracetiming.com/resources/documents/Carleton%20University%20Crystal%20Oscillators%20Paper.pdf

[22] Ben Van Herbruggen et al., “Wi-PoS: A Low-Cost, Open Source Ultra-Wideband (UWB) Hardware Platform with Long Range Sub-GHz Backbone,” Tech. Rep., 2019. [Online]. Available: https://www.mdpi.com/1424-8220/19/7/1548/pdf

[23] Igor Dotlic et al., “Angle of Arrival Estimation Using Decawave DW1000 Integrated Circuits,” Tech. Rep. [Online]. Available: https://www.decawave.com/sites/default/files/angle_of_arrival_estimation_using_dw1000_online.pdf

[24] R.Mahfoudhi, “A Fast Triangular Matrix Inversion,” Tech. Rep., 2012. [Online]. Available: http://www.iaeng.org/publication/WCE2012/WCE2012_pp100-102.pdf

[25] Decawave, “DWM1000IEEE802.15.4-2011 UWB TransceiverModule,” Tech. Rep., 2011. [Online]. Available: https://www.decawave.com/sites/default/files/resources/DWM1000-Datasheet-V1.6.pdf

[26] Huasong Cao et al., “Enabling Technologies for Wireless BodyArea Networks: A Survey and Outlook,” Tech. Rep., 2009. [Online]. Available: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.338.1526&rep=rep1&type=pdf

[27] Lia Kamelia et al., “DOOR-AUTOMATION SYSTEM USING BLUETOOTH-BASED ANDROID FOR MOBILE PHONE,” Tech. Rep., 2014. [Online]. Available: https://www.researchgate.net/publication/292624568_Door-automation_system_using_bluetooth-based_android_for_mobile_phone

[28] SangHyun Chang et al., “Human Detection and Tracking via Ultra-Wideband(UWB) Radar ,” Tech. Rep., May 2010. [Online]. Available: https://authors.library.caltech.edu/75167/1/05509451.pdf

[29] Jerome ROUSSELOT, “Ultra Low Power Communication Protocols for UWB Impulse Radio Wireless Sensor Networks,” Tech. Rep., 2010. [Online]. Available: https://infoscience.epfl.ch/record/147987/files/EPFL_TH4720.pdf

[30] IEEE Computer Society, “THE ALOHA SYSTEM - Another alternative for computer communications,” Fall Joint Computer Conference, Tech. Rep., 1970. [Online]. Available: https://www.clear.rice.edu/comp551/papers/Abramson-Aloha.pdf