Indoor Real-Time Positioning Using Ultra-Wideband Technology

by

Mohammadreza Yavari

TR15-236, August 2015

This is an unaltered version of the author's MCS thesis
Supervisor: Bradford G. Nickerson

Faculty of Computer Science
University of New Brunswick
Fredericton, N.B. E3B 5A3
Canada

Phone: (506) 453-4566
Fax: (506) 453-3566
E-mail: fcs@unb.ca
http://www.cs.unb.ca

Abstract

This thesis investigates the capability of Ultra-Wide Band (UWB) communication technology to be used for indoor real-time positioning. The integration of an inertial measurement unit (IMU) to increase positioning accuracy was also evaluated. We designed and implemented a novel evaluation method for positioning systems that compares each of the position estimates with the ground truth position at the same moment.

We experimentally verified the accuracy of an UWB positioning system in line-of-sight (LOS) conditions to be, on average, 10.97 cm for a robot moving at around 0.16 m/s on a track of length 6.7 m. In the case of non-line-of-sight (NLOS) conditions, accuracy on a track of length 9.7 m for the same moving robot was, on average, 58.47 cm. Data fusion of IMU and UWB measurements using an extended Kalman filter increased average positioning accuracy to 51.96 cm, an 18.8% increase over UWB measurements alone, in NLOS conditions. In LOS conditions, we observed that data fusion of IMU and UWB measurements decreases the average positioning accuracy (from 10.97 cm to 14.17 cm) compared to use of UWB only position estimation.

Dedication

To those striving for peace for the sake of God.

"The Believers are but a single Brotherhood: So make peace and reconciliation between your brothers and be careful of Allah that mercy may be had on you." The Holy Quran, Chapter 49, Verse 10

Acknowledgements

I would like to thank my supervisor Dr. Bradford Nickerson whose expertise, understanding, and patience, added considerably to my graduate experience.

Also, in full gratitude, I would like to acknowledge the New Brunswick Innovation Foundation and the UNB Faculty of Computer Science for all of their financial and technical support during my graduate program.

Table of Contents

Abstract ii

Dedication iii

Acknowledgments iv

Table of Contents vii

List of Tables viii

List of Figures x

Abbreviations xi

1 Introduction 1

1.1 Background 1

1.2 Literature Review 2

1.3 Research Objectives 4

1.4 Thesis outline 5

2 Range Estimation Using Wireless Radio Signals - A Review 6

2.1 Wireless Range Estimation Parameters 6

2.1.1 Time Of Arrival (TOA) 6

2.1.2 Time Difference Of Arrival (TDOA) 8

2.2 IEEE 802.15.4-2011 Range Estimation 9

2.2.1 Ultra Wide-Band Communication 10
2.2.1.1 Definition of Ultra Wide-Band system 10
2.2.1.2 Important features and applications of UWB 10
2.2.1.3 Relationship of bandwidth with data rate and power consumption 12
2.2.1.4 Impulse Radio (IR), one method of using UWB . . . 12
2.2.1.5 Regulation of UWB 13
2.2.1.6 IEEE 802.15.4-2011 packet structure 15
2.2.2 Ranging protocols 16
2.2.2.1 TWR protocol 16
2.2.2.2 SDS-TWR protocol 20
2.2.2.3 Private ranging protocol 23

3 Position Estimation Using UWB Radios **24**
3.1 Statistical Positioning Algorithms 24
3.1.1 Positioning under LOS conditions 26
3.1.2 Positioning under NLOS conditions 28
3.2 Least Squares Algorithm Linearized by Taylor's Series 29
3.3 Enhanced Linearized Least Squares Algorithm 30
3.4 Kalman Filter 34
3.4.1 Process model 34
3.4.2 Kalman Filter Equations 35

4 Positioning with IMU **39**
4.1 IMU Sensors 39
4.1.1 Accelerometer 39
4.1.2 Gyroscope 40
4.2 Reference Frames and Rotation 41

4.3 Data Fusion Using Extended Kalman Filter 44
4.3.1 Extended Kalman Filter 45
4.3.2 Variable Covariance Matrices 47

5 Experimental Setup **49**
5.1 Hardware 51
5.1.1 Decawave 51
5.1.2 ADXL345 52
5.1.3 ITG-3200 52
5.1.4 TelosB 53
5.1.5 Pololu 3pi Robot 55
5.2 Evaluation approaches 56
5.2.1 Off-line Evaluation 56
5.2.2 Real-Time Evaluation 59
5.3 Positioning Module 62
5.4 Test Procedure 65

6 Experimental Results **68**
6.1 LOS Experiments 68
6.2 NLOS Experiments 72

7 Summary and Conclusion **75**

Bibliography **82**

A Real-time evaluation source code **83**

B Off-line evaluation source code **85**

Vita

List of Tables

2.1 Limitation of indoor UWB systems radiating between 960MHz and 10,600MHz (from [13]). 14

2.2 Different choices for the symbols comprising the preamble. 15

2.3 Frequency offset error using the TWR protocol (from [4]). 22

2.4 Frequency offset error using the SDS protocol (from [4]). 23

5.1 Standard Deviation of IMU measurements. 64

6.1 Position of source nodes (in cm) for LOS experiments. 70

6.2 Results for LOS experiments. All results shown in cm. 71

6.3 Position of source nodes in NLOS experiments. All values are given in cm. 73

6.4 Average error for NLOS experiments. All results shown in cm. 73

List of Figures

2.1 Minimum standard deviation of TOA versus SNR for various pulse widths (from [60]). 7

2.2 The hyperbola indicated by TDOA (from [60]). 8

2.3 Low, central and high frequencies of an UWB system (from [60]). . . 11

2.4 Relationship between capacity, bandwidth and SNR (from [60]). . . . 13

2.5 IR UWB signals (from [60]). 13

2.6 FCC regulation for UWB systems (adapted from [43]). 14

2.7 IEEE 802.15.4-2011 packet (from [60]). 15

2.8 TWR ranging protocol (from [60]). 18

2.9 More accurate TWR (from [60]). 19

2.10 SDS ranging protocol (from [60]). 21

3.1 NLOS and LOS propagations (from [60]). 28

3.2 Geometric representation of the linearization method described by (3.23). 32

3.3 Algorithm architecture for LLS-RS with MLE solution (LRM) (modified from [20]). 33

3.4 Kalman filter example (from [60]). 38

4.1 An example of accelerometer dimensions (from [3]). 40

4.2 Mechanical gyroscope (from Wikipedia). 40

4.3 Navigation and body reference frames. 42

5.1 Experimental positioning and evaluation system architecture. 50
5.2 Decawave real time location technology (from [1]). (a) DW1000, (b) DWM1000, (c) EVB1000. 51
5.3 The 6DOF IMU model SEN-10121 produced by SparkFun Electronics and used in our experimental evaluation. 53
5.4 TelosB low-power sensor module. 54
5.5 Pololu 3pi with EVB1000 and IMU sitting next to the line following route marked in black electrical tape. 55
5.6 Potential error in off-line evaluation of the error in the estimated position. 56
5.7 Illustration of off-line evaluation of the difference between an estimated position and a reference trajectory. 58
5.8 Real-time evaluation system. 59
5.9 A picture of the real-time evaluation system for determining the ground truth position of the moving 3pi robot. 60
5.10 Real-time evaluation example. 61
5.11 Ranging implementation by Decawave (from [32]). 63

6.1 Detailed LOS experimental setup in the IB214 lab. 69
6.2 Positioning algorithms output in LOS conditions. (a) LRM, (b) LLS, (c) ground-truth position and (d) IMU-UWB. Red dots indicate the position of the source nodes. 70
6.3 Detailed layout for NLOS conditions experiments in the ITB214 lab. 72
6.4 Positioning algorithms output in NLOS conditions. (a) LRM, (b) LLS, (c) ground-truth position and (d) IMU-UWB. Red dots indicate the position of the source nodes. 74

List of Symbols, Nomenclature and Abbreviations

$\mathbf{a_n}$	Acceleration measurement in the navigation frame
$\bar{\mathbf{a}}_\mathbf{n}$	Output of the low-pass filter over acceleration measurement in the navigation frame
α	Low-pass filter coefficient in section 4.2
$\mathbf{a_b}$	Acceleration measurement in the body frame
α	Channel coefficient used in section 2.2.1
AOA	Angle Of Arrival
a_n	Acceleration measurements vector in the navigation frame
$\mathbf{A}$	Jacobian matrix used for linearization in extended Kalman filter
$\mathbf{a_b}$	Acceleration measurement in the body frame
a_n	Acceleration measurements vector in the navigation frame
B	Bandwidth
β	Effective signal bandwidth
B_{frac}	Fractional bandwidth
BPSK	Binary phase shift keying
CSS	Chirp spread spectrum
c	Speed of light
CRLB	Cramer-Rao Lower Band
C	Channel capacity used in section 2.2.1.3
c_i	Route's corner number i
CW	Clockwise direction
CCW	Counter clockwise direction
d	Distance between two consecutive light sensors
d_i	Observed distance between the target node and ith source node
$\hat{d}_i$	Distance between an estimated position and source node i
δ	Distance between the ground truth position and the last corner on the route
Δ_{TN}	Time offset between target node's (TN) clock and UTC time
Δ_{BS}	Time offset between BaseStation's clock (synchronized with light sensors) and UTC time
$\mathbf{\Delta x}$	Correction vector in non-linear least squares algorithm
DPS	Dynamic preamble selection
EVB1000	UWB ranging board made by Decawave
e	Error of the positioning estimate

e_x	frequency offset of node x (section 2.2.2.2)
η	Measurement noise
E	Signal energy
f_c	Central frequency
$\mathbf{f_i(x)}$	Function describing the relation between the ith distance measurement and target node's position
FPPOWER	Total power of the signal received, measured by EVB1000
FCC	Federal Communication Commission
FIFO	First first out queue
FOM	Figure of merit
FIFO	First first out queue
G_i	Light sensor number i
g_z	Angular velocity around Z axis measured by the gyroscope
GPS	Global Positioning System
IMU	Inertial Measurement Unit
IMU-UWB	Proposed positioning algorithm using IMU and UWB measurements
IF	Intermediate frequency
IR	Impulse Radio
I2C	Inter-Integrated Circuit
LOS	Line-Of-Sight
LLS	Linearized Least Squares algorithm
LLS-RS	Linearized Least Squares algorithm with Reference Selection
LRM	LLS-RS algorithm with MLE solution
MLE	Maximum Likelihood Estimator
N	Average noise power
n(t)	Noise
$\mathcal{N}$	Normal distribution
NLS	Non-linear Least Squares
NLOS	Non-lin-of-sight
Nf_x	Nominal clock frequency of the node x
Φ	Cross correlation
$\mathbf{\hat{P}}$	A posteriori estimate of covariance matrix by Kalman filter
$\mathbf{Q}$	Kalman filter process noise covariance
RSS	Received Signal Strength
r(t)	Received signal
$\mathbf{R}$	Measurements noise covariance
RXPOWER	Total received power measured by EVB1000
SFD	Start of frame delimiter
Rf_x	Real clock frequency of the node x
σ^2	Standard deviation of thermal noise
S	Average received signal power
SNR	Signal to Noise Ratio
SN	Source Node
s(t)	Transmitted signal

SDS	Symmetric Double Sided
TWR	Two way time ranging protocol
t_{G_n}	Time-stamp reported by light sensor number i
T_f	Frame duration
TH	Time hopping
T_r	round-trip time
T_{TW}	TOF computed by two way ranging protocol
θ	Angle between the body frame and the navigation frame (section 4.2)
$\boldsymbol{\theta}$	Unknown parameters vector (equation 3.5)
$\hat{\boldsymbol{\theta}}_{\boldsymbol{MLE}}$	Estimate of θ computed by MLE
T_c	Chip interval
T_1	Departure time of frame (section 2.2.2.1)
T_2	Arrival time of frame (section 2.2.2.1)
TOA	Time Of Arrival
T_{ta}	Turnaround time
T_{SDS}	TOF computed using SDS algorithm
$\hat{T_{SDS}}$	TOF estimated using SDS algorithm including error terms
$\hat{T_{TW}}$	TOF estimated using TWR algorithm including error terms
TN	Target Node
$\mathbf{T_b^n}$	Rotation matrix from the body frame to the navigation frame
τ	Time of arrival
$\tilde{\mathbf{P}}$	A priori estimate of covariance matrix by Kalman filter
τ_{TDOA}	Time difference of arrival estimation
$\tilde{\mathbf{x}}$	Kalman filter a priori estimate
t_{G_n}	Time-stamp reported by light sensor number n
t_{c_n}	Time-stamp indicating the moment TN passed corner number n
t_{UTC}	UTC time-stamp
t_{BS}	BaseStation's clock time-stamp
t_{TN}	TN's clock time-stamp
TDOA	Time Difference Of Arrival
TOF	Time Of Flight
$\mathbf{u}$	Kalman filter input
UWB	Ultra Wide Band
$\mathbf{v}$	Measurement noise
$\mathbf{w}$	Kalman filter process noise
$\mathbf{w}$	Measurement closure vector (section 3.2)
(x'_{t_i}, y'_{t_i})	Ground truth position at time t_i
$(\hat{x}, \hat{y})$	Estimated position of the target node
(x_{c_i}, y_{c_i})	Position of the corners of the route
$(x_{s,i}, y_{s,i})$	Source node$i$ position
(x_{cs}, y_{cs})	Common source node position
$\mathbf{x}$	Position of the target node
$\hat{\mathbf{x}}$	Estimated position of the target node

$\mathbf{z_k}$	Measurement vector for the kth time interval

Chapter 1

Introduction

1.1 Background

Indoor positioning and navigation of moving objects, also called localization in the communication networks community, is an extensive research area with many applications. Applications based on object position, from advertising to security, are widening rapidly. The Global Positioning System (GPS) and other satellite positioning systems work well with a direct line of sight to their satellites. Due to the very low level of energy of received satellite signals, satellite positioning systems can't normally be used inside buildings and indoor places. Consequently, there is significant research aimed at finding a substitute for satellite positioning which satisfies demands of indoor positioning applications. Wireless technologies which are offered for indoor positioning include infrared used in the active badges by AT&T Cambridge [54], ultrasonic energy used in active bats by AT&T Cambridge [21] and optical used in CLIPS by Tilch [49]. A wide variety of wireless solutions for indoor positioning are discussed in e.g. [6] [29] [31] [17]. Ultra wide-band (UWB) wireless communication has some characteristics which makes UWB a good candidate for indoor wireless positioning. UWB technology is described in more detail in sec-

tion 2.2.1. In this thesis, we investigate an indoor positioning system using UWB technology, and evaluate its performance as a stand-alone system compared to its performance aided by an inertial measurement unit (IMU).

1.2 Literature Review

As mentioned in the previous section, UWB wireless communication is a good choice for indoor positioning systems. Some commercial UWB positioning systems offer $\pm 10cm$ positioning accuracy which is more than sufficient for many applications. This accuracy is, however, only in line of sight (LOS) conditions. In LOS conditions there is a direct unblocked path of communication between the target node (TN), the position of which is unknown, and all source nodes (SNs), which have a known position. Indoor positioning systems are normally used in a factory, office or urban area where LOS conditions are rare; the communication path is usually blocked by an obstacle. These conditions are called non-line-of-sight (NLOS). In NLOS conditions the accuracy of UWB positioning systems decrease significantly. Our experiments discussed in chapter 6 demonstrate this significant change in accuracy due to NLOS conditions. There are a variety of approaches suggested for handling NLOS error in the literature (see e.g. [27]). These approaches can be categorized in two groups; the first group [8, 19, 22, 50, 48, 35, 25] try to detect a NLOS condition and simply don't use a measurement coming from the NLOS communication path. This approach works only if the number of LOS SNs available equals or exceeds the minimum number needed for positioning. The second group (see e.g. [22, 11, 57, 24, 59, 51, 53, 42, 39, 30]) are those that try to identify the NLOS condition, and then mitigate the error of positioning due to NLOS conditions. We use Kalman filtering for mitigating the NLOS error in this thesis. Some of the NLOS error mitigation algorithms need a priori knowledge of the statistical distribution of the NLOS errors [22], and some

others are computationally intensive [11] or are based on the assumption of having more than enough measurements [24]. Others suggest more applicable approaches such as applying a filter to the output of the trilateration algorithm (see e.g. [53, 42]). With all of these mitigation approaches in hand, NLOS error is still the major source of error for wireless positioning systems including those using UWB communication. Another solution is to use a second positioning system which is not affected by NLOS conditions to provide additional location information. The common choices for a secondary positioning system are computer vision, laser or ultrasound ranging, GPS and inertial measurement units (IMUs). IMU is a good match for indoor low-cost applications. We reviewed a significant number of papers on UWB positioning aided by IMU (see e.g. [44, 41, 7, 14, 16, 64, 23, 15, 63, 36, 26, 9, 61]), and found the most important characteristics of this type of positioning system are as follows:

- Loose vs tight coupling: There are two groups of data fusion algorithms, loosely coupled and tightly coupled. In the loosely coupled approach, each positioning system is first used to make an estimate of the position, and then the two estimates are coupled to reduce the error. In contrast, tightly coupled methods use the output from both UWB and IMU sensors, and make the final estimate directly from all information. One advantage of tight coupling is that we are able to estimate the position even when it is not possible for one of the positioning systems to do it alone (e.g. just two range measurements available for positioning). Most of the investigated researches use tightly coupled methods. We use an extended Kalman filter to make a tightly coupled system.

- On-board vs stand-alone processor: To the best of our knowledge, all research on IMU-UWB positioning integration available in the literature use a stand-alone processor carried by the target node (TN) to process the measurements and execute the positioning algorithm. While this approach gives a good evaluation of the positioning system, it may not touch details such as time syn-

chronization and processor, space and power restrictions. We performed all implementations on the processor of the UWB ranging board, with the IMU sensors connected to the same processor via an Inter-Integrated Circuit (I2C) interface [45]. So, all of the processes are in real-time and on-board to be as close as possible to an industrial experience.

- Real-time vs off-line evaluation: Most of the research in the literature evaluates the accuracy of the positioning system by comparing the predefined path with the path estimated by the positioning algorithm. We proposed a new real-time evaluation approach to measure the accuracy of our positioning system using precisely positioned ground truth nodes placed along the path. A more detailed description of our evaluation system is in section 5.2.

- IEEE 802.15.4-2011 standard: We use UWB modules produced by the Decawave company which are compliant with the IEEE 802.15.4-2011 standard.

1.3 Research Objectives

The main objective of this thesis is to implement a real-time, indoor positioning system using UWB technology. The main research questions addressed by this thesis are as follows:

- What is the accuracy of Decawave UWB positioning in LOS and NLOS conditions?

- How much can NLOS error be mitigated by methods proposed in the literature?

- How much can an inertial measurement unit (IMU) and Kalman filter increase the LOS and NLOS accuracy of UWB systems?

- Is it possible to have a communication channel operating simultaneously with the positioning system?

1.4 Thesis outline

We continue this thesis by discussing different range estimation approaches and protocols in chapter 2. Chapter 3 covers positioning algorithms based on range measurements by UWB radios. Then, in chapter 4, we describe how to use an IMU to increase the positioning accuracy. Experimental setup and results are discussed in chapters 5 and 6, respectively. Chapter 7, finally, is devoted to the summary and conclusion with an eye towards future work.

Chapter 2

Range Estimation Using Wireless Radio Signals - A Review

2.1 Wireless Range Estimation Parameters

In this section a discussion of time-based signal parameters used for ranging is presented. A description of using received signal strength (RSS) and angle of arrival (AOA) parameters is given in [58].

2.1.1 Time Of Arrival (TOA)

Time of arrival (TOA) gives us information about the distance between the TN and a SN for which the position is known. The TN's position is on a circle of radius $d = c\tau$, for c = speed of light and τ = time of arrival. The prerequisite of this information is synchronization between the TN and SN. The received signal at the TN is represented by

$$r(t) = \alpha s(t - \tau) + n(t), \tag{2.1}$$

where τ is the TOA, α is the channel coefficient and $n(t)$ is white Gaussian noise with zero mean and a spectral density of $\mathcal{N}_0/2$ watts per hertz for $\mathcal{N}$ =normal

distribution. In order to extract TOA from the received signal, we search for the maximum correlation between shifted versions of the template signal ($s(t-\hat{\tau})$) and the received signal. The $\hat{\tau}$ which gives the peak correlation provides an estimate of the TOA. For signal model (2.1), the Cramer-Rao lower bound (CRLB) is

$$\sqrt{Var\{\hat{\tau}\}} \geq \frac{1}{2\sqrt{2}\pi\sqrt{SNR}\beta}, \tag{2.2}$$

where $\hat{\tau}$ is the estimated TOA, SNR $= \alpha^2 E/\mathcal{N}_0$ is the signal to noise ratio, β is the effective signal bandwidth and E is the signal energy. One important property of TOA is that, unlike the received signal strength (RSS), its accuracy is heavily dependent on the bandwidth of the signal. Consequently, UWB systems can reach very precise ranging on the order of a few centimeters. Figure 2.1 shows the effect of signal to noise ratio (SNR) and bandwidth on the accuracy of TOA estimation.

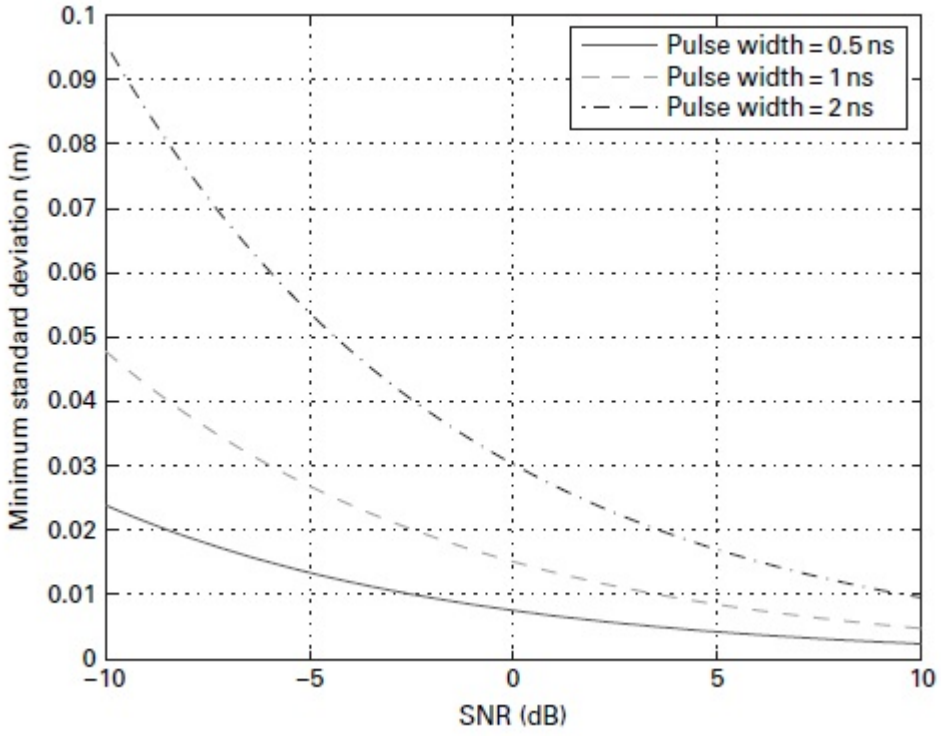

Figure 2.1: Minimum standard deviation of TOA versus SNR for various pulse widths (from [60]).

2.1.2 Time Difference Of Arrival (TDOA)

In this approach, the extracted parameter is the difference between arrival time of transmitted signal (from the TN) to two SNs. This parameter, by multiplying TDOA by the speed of light, gives us an uncertainty of the TN's position in the shape of a hyperbola as shown in Figure 2.2.

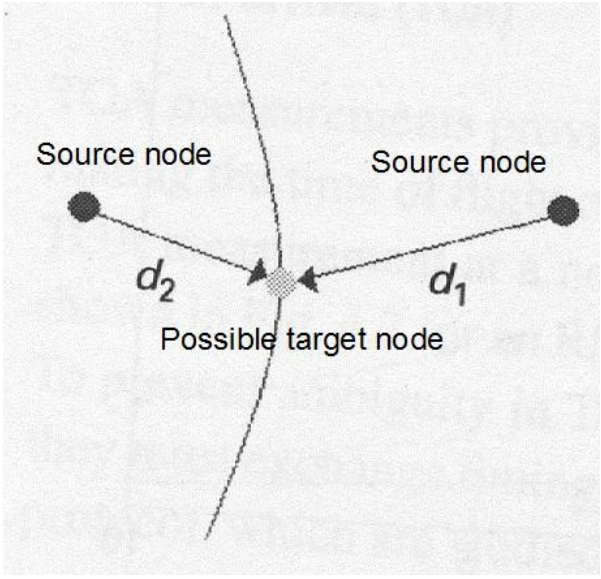

Figure 2.2: The hyperbola indicated by TDOA (from [60]).

The merit of TDOA in comparison with TOA is that there is no need for synchronization between SNs and the TN. SNs however, do need to be synchronized. In the TOA-based approach for measuring TDOA, TOA is measured at two SNs which we call τ_1 and τ_2. As SNs and TN are not synchronized, there is a time offset in τ_1 and τ_2. Since sources are synchronized with themselves, this offset is equal in both measurements. Consequently, we can measure the TDOA as

$$\tau_{TDOA} = \tau_1 - \tau_2, \tag{2.3}$$

where τ_{TDOA} is the estimate of TDOA. In this approach there is the same effect of bandwidth as in TOA measurements.

The second approach to measure the TDOA is using cross-correlation between two

received signals. We know that there is some amount of offset between received signals so the cross-correlation will reach a maximum when one of the signals is shifted with correct offset. The cross-correlation equation is

$$\phi_{1,2}(\tau) = \frac{1}{T}\int_0^T r_1(t)r_2(t+\tau)dt \tag{2.4}$$

where $r_1(t)$ and $r_2(t)$ are the received signals and T is the observation interval. Then we estimate TDOA, $\hat{\tau}_{TDOA}$ by

$$\hat{\tau}_{TDOA} = \arg\max_{\tau} |\phi_{1,2}(\tau)| \tag{2.5}$$

where

$$\arg\max_{x} f(x) := \{x | \forall y : f(y) \leq f(x)\} \tag{2.6}$$

The cross-correlation approach works well for white noise and single path channels but in the case of multi-path channel or colored noise its performance decreases significantly [60].

2.2 IEEE 802.15.4-2011 Range Estimation

Range estimation gives an estimate of the distance between two nodes. Ranging protocols are based on TOA which was discussed in section 2.1.1. TOA gives us the time of flight (TOF) of the signal at the TN. To compute the range between the TN and the SN we need to know the departure time of the signal. Then the distance is computed by multiplying TOF by the speed of light. Ranging protocols extract the TOF by different methods. The IEEE 802.15.4-2011 is the first international standard that provides a specific physical layer capable of wireless ranging. IEEE 802.15.4-2011 standard has two formats of communication signal; the first one is

impulse radio ultra wide-band (IR-UWB) signals and the second one is chirp spread spectrum (CSS) signals. The second one, CSS, is suitable for data communication purposes while the IR-UWB has the capability of precise ranging. In this chapter we consider the IR signal format.

2.2.1 Ultra Wide-Band Communication

2.2.1.1 Definition of Ultra Wide-Band system

The U.S. Federal Communication Commission (FCC) [13] has defined UWB systems as those which have an absolute bandwidth larger than 500 MHz and central frequency (f_c) larger than 2.5 GHz, or have a fractional bandwith (B_{frac}) larger than 0.2 for systems with f_c lower than 2.5 GHz. The f_c is the frequency in which the system has the maximum power density (shown in Figure 2.3) and the frequencies f_H and f_L determine the location where the power spectral density is 10 dB below the f_c. B_{frac} is defined as

$$B_{frac} = \frac{B}{f_c} \tag{2.7}$$

where B is the bandwidth of the system.

In terms of high and low frequencies, we have

$$f_c = \frac{f_H + f_L}{2} \tag{2.8}$$

so

$$B_{frac} = \frac{2(f_H - f_L)}{f_H + f_L}. \tag{2.9}$$

2.2.1.2 Important features and applications of UWB

The following are some important features of ultra wide-band systems:

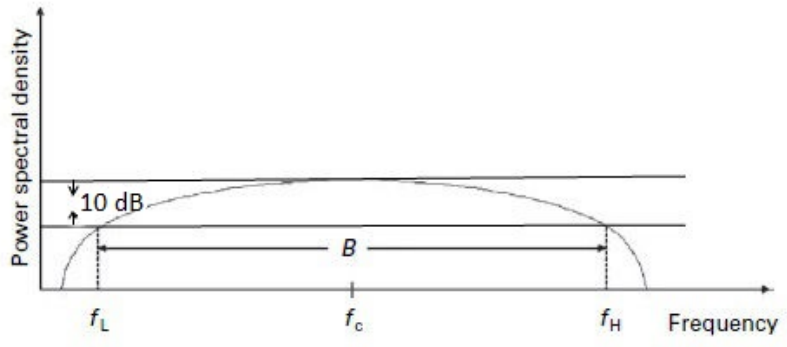

Figure 2.3: Low, central and high frequencies of an UWB system (from [60]).

- The most important characteristic of UWB is large bandwidth in comparison with prevalent narrow-band systems.

- One result of the large bandwidth of UWB is that due to the inverse relationship of time and frequency, the life-time of UWB signals is very short. Consequently, the time resolution of UWB signals is high and UWB is a good candidate for positioning systems.

- UWB systems are suitable for high speed communication due to their high bandwidth.

- Another useful property of UWB is that it is permitted to occupy low carrier frequencies, where signals can more easily pass through obstacles.

- UWB signals can be transmitted in base band so there is no need for Intermediate Frequency (IF) multipliers in transceivers. This property can lead to less expensive simpler hardware.

- The high time resolution and short wavelength of UWB signals strengthen it against multi-path interference and fading.

- UWB signals' shape is similar to noise so there is a lower chance of eavesdropping.

2.2.1.3 Relationship of bandwidth with data rate and power consumption

As indicated by the Shannon-Hartley theorem [47], there is a direct relationship between capacity and bandwidth and an inverse relationship between bandwidth and power consumption. Their theorem states

$$C = B \log_2(1 + \frac{S}{N}) \tag{2.10}$$

where C is the capacity (bits/second), B is the bandwidth, S is the average received signal power over B and N is the average noise over B. We observe that for a specific capacity we consume less power with a larger bandwidth. Secondly, because $\frac{S}{N}$ is under a logarithm, it is easier to increase the capacity by increasing of the bandwidth instead of $\frac{S}{N}$. It is common to refer to $\frac{S}{N}$ as SNR, the signal to noise ratio. Figure 2.4 shows the relationship between the bandwidth and the capacity for five different SNRs.

2.2.1.4 Impulse Radio (IR), one method of using UWB

In this method, data is transmitted by low duty UWB signals and information of the symbol is conveyed by position and/or polarity of the signals. Each symbol corresponds to one or more signals. In the following example (Figure 2.5), two consecutive IR signals represent one symbol. The IR signal can occupy one of the chip-intervals (T_c) within a frame (T_f). A time-hopping (TH) code is used for determining the accurate position of a signal in dedicated time frame to decrease the chance of interference between UWB systems. In the following example, the TH codes for the symbols are $\{2, 1\}$, $\{2, 3\}$ and $\{1, 0\}$ respectively, so the first and second signals are shifted by two and one chip-intervals respectively and so on. In this example the information corresponds to the polarity of signals, so the IR stream represents the

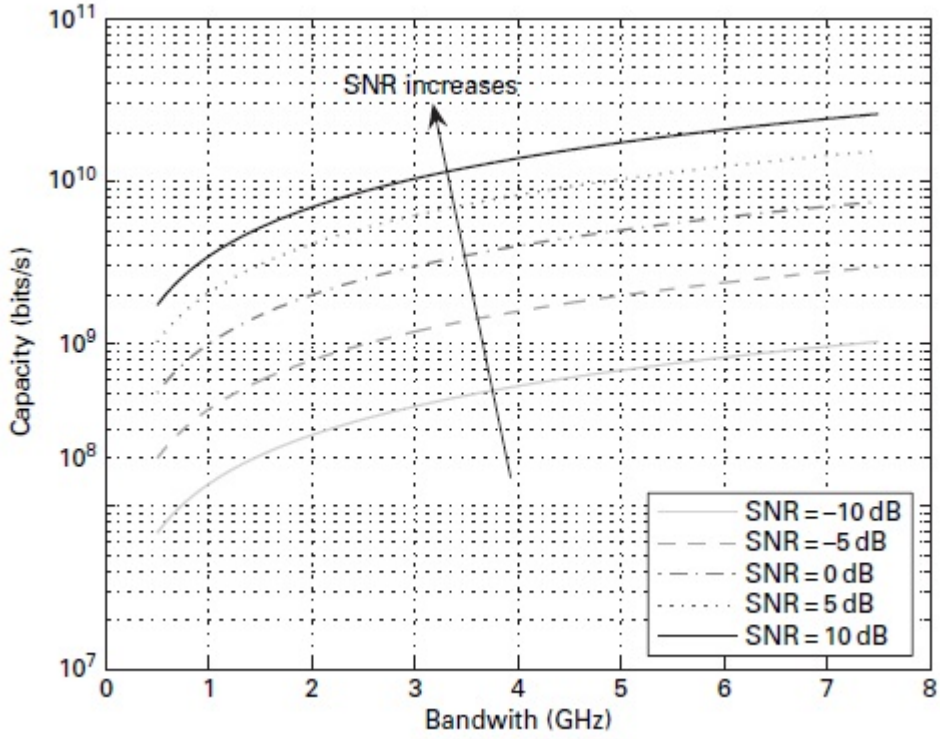

Figure 2.4: Relationship between capacity, bandwidth and SNR (from [60]).

binary data "101". This technique is commonly called Binary Phase Shift Keying (BPSK).

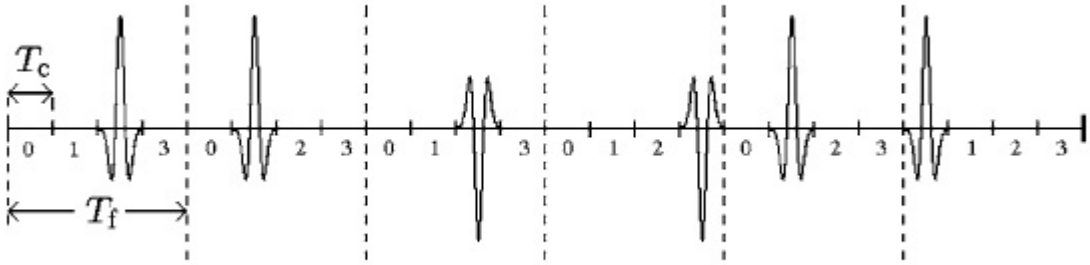

Figure 2.5: IR UWB signals (from [60]).

2.2.1.5 Regulation of UWB

Although it is a very useful property of UWB that it has a large bandwidth, this large bandwidth can lead to interference with narrow band systems. In order to

solve this problem some regulations are enforced by several organizations. One of the pioneer organizations in this field is the FCC. Figure 2.6 demonstrates the FCC's regulation [13] for UWB systems. For UWB radio communication indoors, the FCC part 15.517 requires following equivalent isotropically radiated power (EIRP) limits (2.1). There are many other restrictions for operating of UWB systems in the USA (see [13]). The general approach of all regulations is to restrict the power of signals to avoid collision with other systems. In spite of this restriction, it is an important advantage that UWB is license-free and benefits from co-existence which means that anyone can implement UWB communications without any license in a dedicated range of power at a variety of frequencies.

Table 2.1: Limitation of indoor UWB systems radiating between 960MHz and 10,600MHz (from [13]).

Frequency in MHz	EIRP in dBm
960-1610	-75.3
1610-1990	-53.3
1990-3100	-51.3
3100-10600	-41.3
Above 10600	-51.3

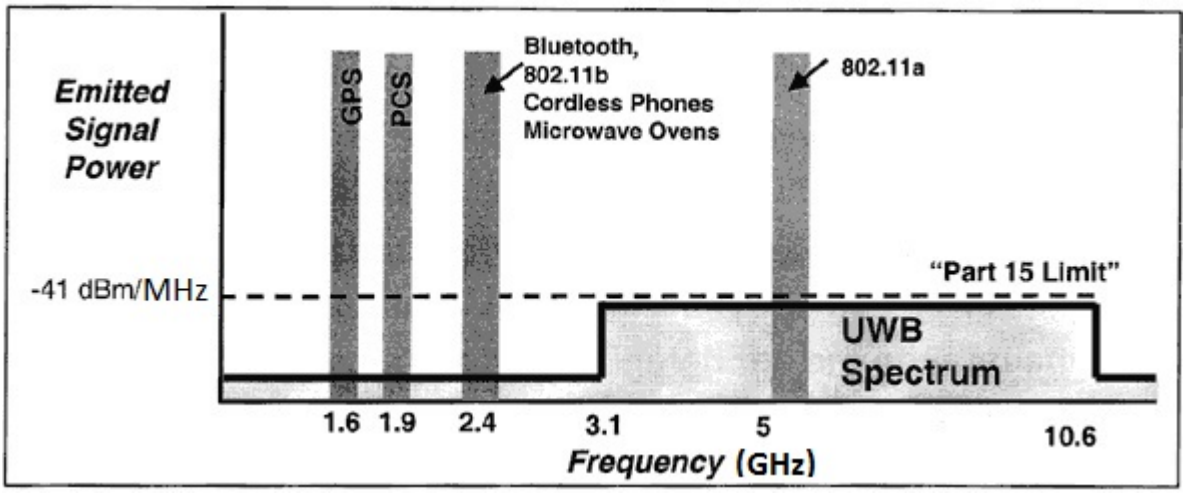

Figure 2.6: FCC regulation for UWB systems (adapted from [43]).

2.2.1.6 IEEE 802.15.4-2011 packet structure

In this section we give a brief overview of packet structure of the IEEE 802.15.4-2011 standard. Figure 2.7 depicts different parts of the packet which is explained in next subsections.

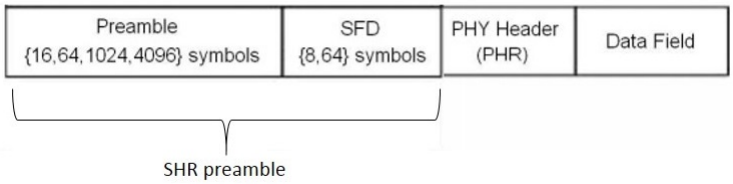

Figure 2.7: IEEE 802.15.4-2011 packet (from [60]).

Preamble: The network protocols preamble is used to synchronize entities with informing arrival of a packet. In IEEE 802.15.4-2011, the length of the preamble is on of 16, 64, 1024 or 4096 symbols. The symbols used in the preamble part of a packet is one of the eight symbols cited in table 2.2.

Table 2.2: Different choices for the symbols comprising the preamble.

Index	Symbol
1	-1000010-1011101-10001-111100-110-100
2	0101-10101000-1110-11-1-1-10010011000
3	-11011000-11-11100110100-10000-1010-1
4	00001-100-100-1111101-1100010-10110-1
5	-101-100111-11000-1101110-1010000-00
6	1100100-1-1-11-1011-10001010-11010000
7	100001-1010100100011011-1-1-10-1100-11
8	0100-10-10110000-1-1100-11011-1110100

The symbols in Table 2.2 all have an important property, called perfect periodic auto-correlation, which reduce the error in ranging caused by multi-path propagation. The details of this property are out of the scope of this report and are given in [60]. The length of the preamble is chosen regarding the positioning system demands and performance. For example, a larger packet size helps low quality receivers to

gain higher SNRs while a smaller packet size reduces the channel occupancy. Lower channel occupancy leads to more efficient energy consumption, and capability of more devices using the same channel. The IEEE 802.15.4-2011 standard includes a parameter, called figure of merit (FOM), which represents the accuracy of the range measurement. It is suggested by the standard [40] for positioning systems to start with the length of 1024 and then adjust the length of the preamble by keeping track of the FOMs.

start of frame delimiter (SFD): The SFD is a short sequence with 8 or 64 symbols which signals the end of the preamble and start the of the physical layer header. In the ranging protocols the arrival time of the signal and the process time between arrival and sending back an ACK packet should be measured precisely. The SFD is a narrow signal to trigger starting and stopping of time counting which is necessary for a precise timing.

2.2.2 Ranging protocols

IEEE 802.15.4-2011 has three different ranging protocols. The basic mandatory protocol is two way ranging (TWR). The second one, which is more precise and optional, is the symmetric double sided (SDS) TWR. The third protocol, called private ranging, is designed for systems in which the position information is sensitive and should be kept private. These three ranging protocols are discussed in the next subsections respectively. In this thesis we used the SDS-TWR protocol (discussed in section 2.2.2.2) implemented by Decawave Ltd. on EVB1000 boards.

2.2.2.1 TWR protocol

In the pure TOA system, the SN reports the departure time of the signal; hence the TN is able to compute the TOF of the signal. This TOA approach dedicates a

necessary synchronization between the TN and the SN; since the departure time is measured by the SN and is used by the TN. The main advantage of TWR over the TOA approach is its independence from the synchronization. Figure 2.8 depicts the TWR protocol. The TWR protocol consists of the following steps:

1. The TN sends a ranging request, $RFRAME_{req}$, to the SN, and records the departure time of the frame, T_1.
2. The SN replies to the ranging request after arrival with the $RFRAME_{rep}$.
3. The TN records the arrival time of the $RFRAME_{rep}$, T_2.

The TN computes the round-trip time T_r as

$$T_r = T_2 - T_1 \tag{2.11}$$

and then the TOF between the TN and the SN, called T_{TW}, is given by

$$T_{TW} = T_r/2 \tag{2.12}$$

and the distance between two nodes is given by

$$d = cT_{TW} \tag{2.13}$$

where c is the speed of light. With three distances from three SNs, the TN can determine its position based on the known positions of the SNs. In practice, as is depicted in Figure 2.8, there is a delay on the SN side between receiving of the $RFRAME_{req}$ and sending $RFRAME_{rep}$ called the turn around time T_{ta}^B. Due to the high speed of the light, a T_{ta}^B of nano-seconds causes a ranging error of tens of centimeters. Consequently, it is important to have an accurate estimate of the T_{ta}^B.

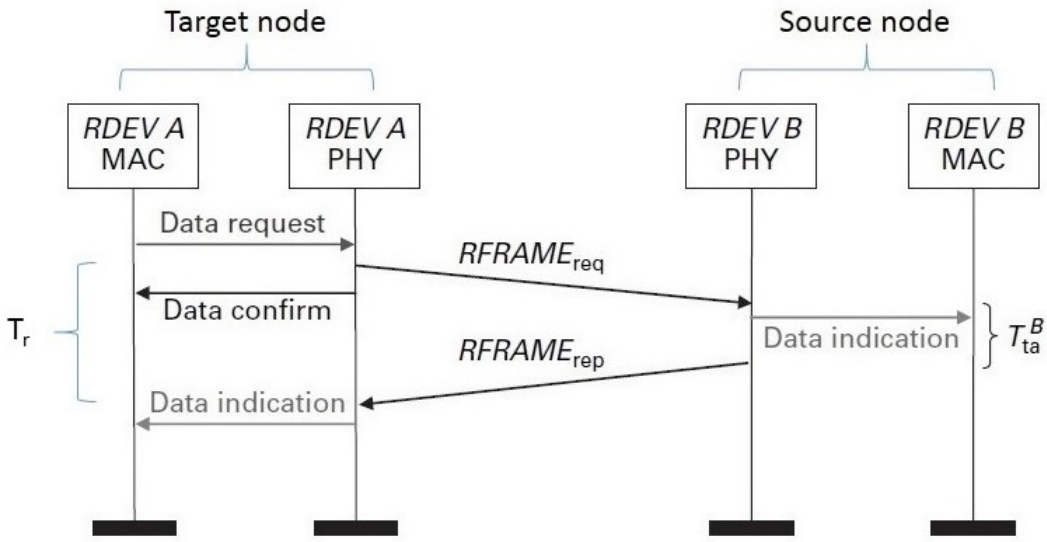

Figure 2.8: TWR ranging protocol (from [60]).

IEEE 802.15.4-2011 uses a more advanced TWR approach which gives an estimate of the T_{ta} to the TN for more accurate ranging. Figure 2.9 depicts this approach. A counter in the SN starts when detecting the first symbol of the SFD of $RFRAME_{req}$ and stops when the first symbol of SFD in $RFRAME_{rep}$ is sent. Then, after sending $RFRAME_{rep}$, the SN sends a time stamp report including the stop and start value of the counter. Finally, the TN sends an ACK back to the SN. Using this approach, the TOF is computed as

$$T_{TW} = \frac{T_r - T_{ta}^B}{2} \tag{2.14}$$

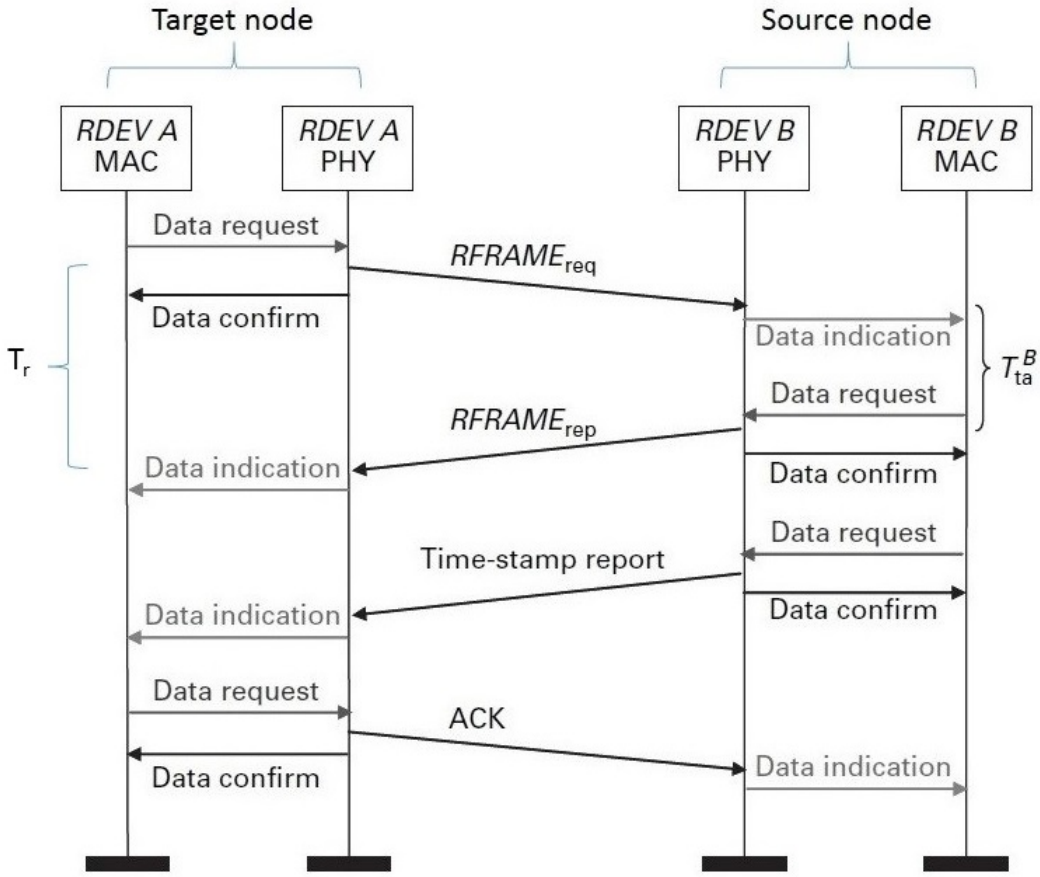

Figure 2.9: More accurate TWR (from [60]).

2.2.2.2 SDS-TWR protocol

One of the error sources in the TWR approach is the clock offset. The crystal oscillators used in sensor devices (TN or SN) are not working exactly with the nominal frequency, so there is a small positive or negative offset in the time measurements. With the high speed of light, this small offset may cause a significant error in ranging. The SDS protocol is designed to mitigate the clock offset error. In the SDS protocol, depicted in Figure 2.10, after the TN receives the $RFRAME_{rep}$, it sends a second $RFRAME_{req}$ to the SN. Consequently, each of the nodes has an estimate of the round trip time, T_r, and turnaround time, T_{ta}. Finally, the SN sends a time stamp including measured T_r and T_{ta} to the TN.

Then the TN estimates the TOF as

$$T_{SDS} = \frac{(T_r^A - T_{ta}^A) + (T_r^B - T_{ta}^B)}{4} \tag{2.15}$$

To observe the merit of the SDS protocol over the TWR protocol, we define the frequency offsets of the TN and SN e_A and e_B as

$$e_A = \frac{Rf_A - Nf_A}{Nf_A} \tag{2.16}$$

$$e_B = \frac{Rf_B - Nf_B}{Nf_B} \tag{2.17}$$

where Rf_x and Nf_x are the real frequency and nominal frequency of the node x. Then estimates of the TOF given by the TWR protocol and the SDS protocol are represented as

$$\hat{T}_{TW} = \frac{T_r^A(1 + e_A) - T_{ta}^B(1 + e_B)}{2} \tag{2.18}$$

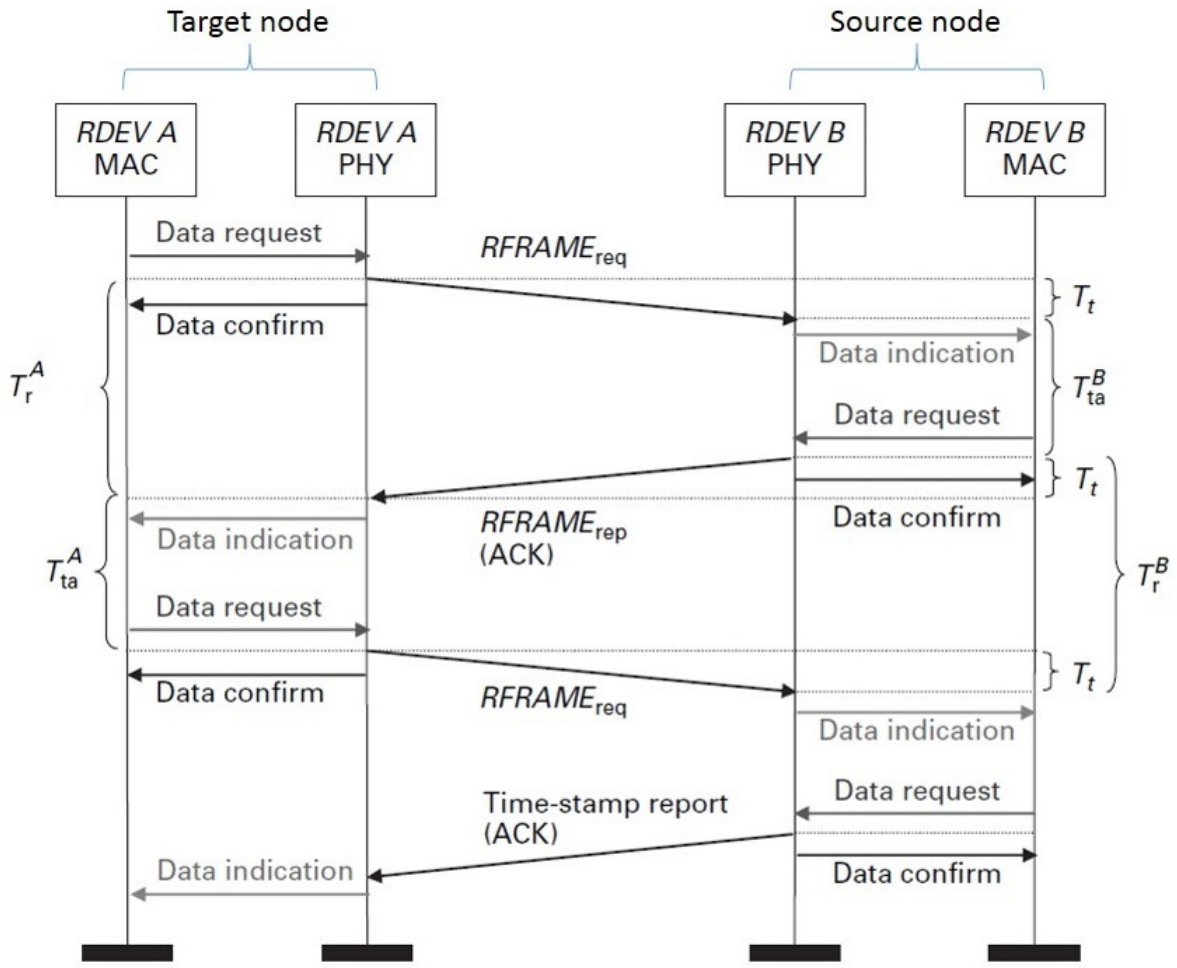

Figure 2.10: SDS ranging protocol (from [60]).

$$\hat{T}_{SDS} = \frac{(T_r^A - T_{ta}^A)(1 + e_A) + (T_r^B - T_{ta}^B)(1 + e_B)}{4} \tag{2.19}$$

Note that in indoor positioning systems, measured distances are not much more than 30 meters, therefore the maximum of T_{TW} and T_{SDS}, which are the time it takes for light to travel about 30 meters, are in the order of $0.1\mu s$. Another point is that the T_{ta} is not just the response time of the device but also includes the duration of packet, and is on the order of milliseconds. Consequently, T_{TW} and T_{SDS} are much smaller than T_{ta}. The error of each estimation is represented as

$$\hat{T}_{TW} - T_{TW} = T_{TW} e_A + \frac{e_A - e_B}{2} T_{ta}^B \tag{2.20}$$

$$T_{TW} \ll T_{ta}^B \Rightarrow \hat{T}_{TW} - T_{TW} \approx \frac{e_A - e_B}{2} T_{ta}^B \tag{2.21}$$

$$\hat{T}_{SDS} - T_{SDS} = \frac{T_{SDS}}{2}(e_A + e_B) + \frac{T_{ta}^B - T_{ta}^A}{4}(e_A - e_B) \tag{2.22}$$

$$T_{SDS} \ll T_{ta}^B - T_{ta}^A \Rightarrow \hat{T}_{SDS} - T_{SDS} = \frac{T_{ta}^B - T_{ta}^A}{4}(e_A - e_B) \tag{2.23}$$

We observe that the clock offset error is mitigated in the SDS protocol since the T_{ta}^B is significantly larger than $T_{ta}^B - T_{ta}^A$. Tables 2.3 and 2.4 show some typical results for the frequency offset error computed by equation (2.21) and (2.23). We see that the SDS protocol mitigates the frequency offset error significantly.

Table 2.3: Frequency offset error using the TWR protocol (from [4]).

	$e_A - e_B$			
T_{ta}^B	2ppm	20ppm	40ppm	80ppm
$100\mu s$	0.1 ns	1 ns	2 ns	4 ns
5ms	5 ns	50 ns	100 ns	200 ns

Table 2.4: Frequency offset error using the SDS protocol (from [4]).

	$e_A - e_B$			
$T_{ta}^B - T_{ta}^A$	2ppm	20ppm	40ppm	80ppm
$1\mu s$	0.0005 ns	0.005 ns	0.01 ns	0.02 ns
$10\mu s$	0.005 ns	0.05 ns	0.1 ns	0.2 ns
$100\mu s$	0.05 ns	0.5 ns	1 ns	2 ns

2.2.2.3 Private ranging protocol

In some cases the ranging information is sensitive. A hacker may perform different attacks. He may just eavesdrop to determine the range and consequently position information. He may also try to disturb the ranging protocol by sending a fake response. The IEEE802.15.4-2011 standard has an optional private ranging protocol which provides some security services. The first technique used in the private ranging protocol is encrypting the time stamps before sending them. The rationale of this technique is that having all information but the time stamp the hostile is not able to compute the range between two nodes. Note that this encryption is performed before sending the time stamp and after time measurements so it is not time sensitive. The second method used in the private ranging protocol is dynamic preamble selection (DPS). Through this approach, ranging nodes use a longer preamble with 127 symbols. The TN and the SN agree on the preamble sequence, among eight different choices, by passing encrypted messages in the beginning of the ranging protocol. The preambles must change for each ranging process to protect the system against replay attacks.

Chapter 3

Position Estimation Using UWB Radios

Using SDS-TWR ranging protocol, equation (2.19) gives us an estimation of the TOF of the signal between a SN and the TN. We compute the distance between these two nodes as follows:

$$d = c\hat{T}_{SDS} \tag{3.1}$$

where c is the speed of light. We use the measured distance in positioning algorithms to find the position of the TN.

3.1 Statistical Positioning Algorithms

In this section we discuss methods of position estimation based on the parameters we cited in the last section. There are two main categories of positioning methods.The mapping category uses a calibration table which includes any of the above measurements corresponding to known positions. This calibration table is produced prior to the positioning in a training phase. One important consideration in mapping category methods is updating the calibration table of known positions which is

challenging, especially in outdoor positioning systems. Consequently, mapping category methods are not prevalent in outdoor applications. In this report, our focus is on the non-mapping category which uses parameters extracted directly from the signal instead of a pre-computed calibration table. The non-mapping category of positioning approaches has two sub-categories; geometric approaches and statistical approaches. Statistical approaches are more practical and discussed in this section. Please refer to [58] for discussion on the geometric approach.

Since geometric approaches cannot cope with practical noisy environments, usually statistical approaches are used in practice. In this noisy framework, we define a model (adapted from [60]) for noisy measurements as

$$z = f(x, y) + \eta, \tag{3.2}$$

where z is the result of a noisy measurement, $f(x, y)$ is the true value of this measurement which is a function of TN's position and η is the noise of this measurement. For the techniques discussed in this section, $f(x, y)$ is as follows (adapted from [60]) :

$$f(x,y) = \begin{cases} \sqrt{(x-x_s)^2 + (y-y_s)^2} & TOA/RSS \\ \arctan((y-y_s)/(x-y_s)) & AOA \\ \sqrt{(x-x_s)^2 + (y-y_s)^2} - \sqrt{(x-x_{cs})^2 + (y-y_{cs})^2} & TDOA \end{cases} \tag{3.3}$$

where (x_s, y_s) is the known position of the SN, and (x_{cs}, y_{cs}) is the position of the common SN for the TDOA technique. In vector-space notation, the cited model is changed to (from [60])

$$\mathbf{z} = \mathbf{f}(x, y) + \boldsymbol{\eta}, \tag{3.4}$$

where $\mathbf{z} = [z_1 ... z_{N_m}]^T$, $\mathbf{f}(x, y) = [f_1(x, y) .... f_{N_m}(x, y)]^T$ and $\boldsymbol{\eta} = [\eta_1 ... \eta_{N_m}]^T$. N_m is

equal to the number of SNs in TOA, RSS and AOA approaches, and one less than the number of SNs in the TDOA approach. Assume that the noise which affects our measurement is known except for a set of parameters, $\boldsymbol{\lambda}$. So, we have a vector of unknown parameters, $\boldsymbol{\theta}$, as (from [60])

$$\boldsymbol{\theta} = [x, y, \boldsymbol{\lambda}]^T \tag{3.5}$$

where (x, y) is the position of the TN. In such problems, we can use parametric approaches to estimate the true value of $\boldsymbol{\theta}$. Two prevalent parametric approaches are Bayesian and Maximum-Likelihood Estimator (MLE) [5]. The Bayesian approach is useful in the case that some a prior information about $\boldsymbol{\theta}$ is available. In this thesis, we assume no such a prior information, so, we focus on the MLE approach. The MLE approach finds $\boldsymbol{\theta}$ which gives the maximum probability for the observations. Formally, we can define $\boldsymbol{\theta}$ estimated by the MLE, $\hat{\boldsymbol{\theta}}_{MLE}$, as

$$\hat{\boldsymbol{\theta}}_{MLE} = \arg\max_{\boldsymbol{\theta}} p(\mathbf{z}|\boldsymbol{\theta}). \tag{3.6}$$

Since the function $f(x, y)$ is deterministic, we can express the likelihood function, $p(\mathbf{z}|\boldsymbol{\theta})$ as

$$p(\mathbf{z}|\boldsymbol{\theta}) = p_{\boldsymbol{\eta}}(\mathbf{z} - \mathbf{f}(x, y)|\boldsymbol{\theta}), \tag{3.7}$$

where $p(.|\boldsymbol{\theta})$ is the conditional probability density function of the noise for a given parameter vector $\boldsymbol{\theta}$.

3.1.1 Positioning under LOS conditions

The statistical solution for positioning differs if we have independent noise in measurements or dependent noise. In this section we discuss the case of independent noise; [58] gives a description of statistical positioning in the presence of dependent

noise. In the case of independent noise, we can express equation (3.7) as

$$p(\mathbf{z}|\boldsymbol{\theta}) = \prod_{i=1}^{N_m} p_{\eta_i}(z_i - f_i(x,y)|\boldsymbol{\theta}) \tag{3.8}$$

where z_i is the ith measurement, $f_i(x,y)$ is the true value of the ith measurement and the p_{η_i} is the conditional probability density function of the ith measurement. The independent noise assumption is reasonable for the AOA, TOA and RSS approaches. However, in the case of TDOA, we have correlated noise for several SNs' measurements due to the presence of the common SN. For systems working under LOS conditions, the majority of the noise is thermal noise. We can model the noise of these environments as a Gaussian zero mean random variable as

$$p_{\eta_i}(u) = \frac{1}{\sqrt{2\pi}\sigma_i} exp(-\frac{u^2}{2\sigma_i^2}). \tag{3.9}$$

The unknown variables vector $\boldsymbol{\theta}$, reduces to $[x,y]^T$. Then the likelihood function in (3.8) is expressed as

$$p(\mathbf{z}|\boldsymbol{\theta}) = \frac{1}{2\pi^{N_m/2}\prod_{i=1}^{N_m}\sigma_i} exp\left(-\sum_{i=1}^{N_m}\frac{(z_i - f_i(x,y))^2}{2\sigma_i^2}\right). \tag{3.10}$$

With this expression of the $p(\mathbf{z}|\boldsymbol{\theta})$, the MLE for (3.6) is calculated by (from [60])

$$\hat{\boldsymbol{\theta}}_{MLE} = \underset{[x,y]^T}{\arg\min}\left(\sum_{i=1}^{N_m}\frac{(z_i - f_i(x,y))^2}{\sigma_i^2}\right) \tag{3.11}$$

where

$$\underset{(x,y)}{\arg\min}\, f(x,y) := \{(x,y)|\forall(w,z) : f(w,z) \geq f(x,y)\}. \tag{3.12}$$

Equation (3.11) is a commonly used non-linear least squares (NLS) estimator [52]. In the case of sufficiently large effective bandwidth and SNR the standard deviation

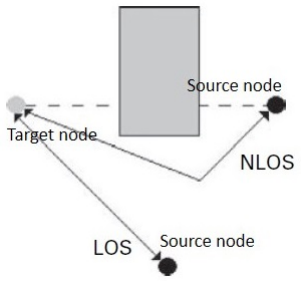

Figure 3.1: NLOS and LOS propagations (from [60]).

of thermal noise, σ_i, is given by $\frac{1}{8\pi^2\beta^2 SNR}$, where β is the effective bandwidth of the signal and SNR is the signal to noise ratio [37]. We observe that the variance of the noise in each measurement weights the measurement in an inverse manner, which matches our intuition that stronger noise implies less accurate measurement. Several approaches are cited in literature to solve (3.11) equation including gradient descent algorithms and linearization techniques using Taylor expansion [10][28].

3.1.2 Positioning under NLOS conditions

In most practical systems there are some obstacles between the TN and SNs. Therefore, signals transmitted by the TN have to travel a longer indirect path to reach the SN. Figure 3.1 depicts the difference between NLOS propagation and LOS propagation.

NLOS propagation acts as a noise source for signal parameters such as TOA, RSS and AOA; we call this noise NLOS noise. For example, the TOA of the signal increases since it has to follow a longer path and the time of arrival is affected by a positive noise. Due to NLOS propagation (commonly called multipath), the noise model in equation (3.9) is not accurate for NLOS cases. The NLOS noise is commonly modeled by a Gamma probability distribution [38]. NLOS noise dominates the background noise modeled by equation (3.9). In section 3.4 we discuss an approach for mitigating NLOS noise by using a Kalman filter.

3.2 Least Squares Algorithm Linearized by Taylor's Series

As described in section 3.1.1, equation (3.11), the statistical approach for trilateration leads to solving a non-linear least squares (NLS) problem. One of the common approaches to face this problem is linearization of the non-linear function, $\mathbf{f}(\mathbf{x})$, by using Taylor's series [18] [62] as follows:

$$\mathbf{f_i}(\mathbf{x}) \approx \mathbf{f_i}(\hat{\mathbf{x}}_{\mathbf{j-1}}) + \mathbf{A}(\mathbf{x} - \hat{\mathbf{x}}_{\mathbf{j-1}}) \tag{3.13}$$

where state vector, $\hat{\mathbf{x}}_{\mathbf{j-1}}$, is the last estimated position:

$$\hat{\mathbf{x}}_{\mathbf{j-1}} = \begin{bmatrix} \hat{x}_{j-1} \\ \hat{y}_{j-1} \end{bmatrix} \tag{3.14}$$

and

$$\mathbf{A} = \begin{bmatrix} \frac{\partial f_1(x)}{\partial x}|_{\hat{x}_{j-1}} & \frac{\partial f_1(x)}{\partial y}|_{\hat{x}_{j-1}} \\ \frac{\partial f_2(x)}{\partial x}|_{\hat{x}_{j-1}} & \frac{\partial f_2(x)}{\partial y}|_{\hat{x}_{j-1}} \\ \frac{\partial f_3(x)}{\partial x}|_{\hat{x}_{j-1}} & \frac{\partial f_3(x)}{\partial y}|_{\hat{x}_{j-1}} \end{bmatrix} = \begin{bmatrix} \frac{\hat{x}_{j-1}-x_{s,1}}{d_1} & \frac{\hat{y}_{j-1}-y_{s,1}}{d_1} \\ \frac{\hat{x}_{j-1}-x_{s,2}}{d_2} & \frac{\hat{y}_{j-1}-y_{s,2}}{d_2} \\ \frac{\hat{x}_{j-1}-x_{s,3}}{d_3} & \frac{\hat{y}_{j-1}-y_{s,3}}{d_3} \end{bmatrix} \tag{3.15}$$

To find the best estimate of the position from this linear equation, we compute a correction vector, $\mathbf{\Delta x}$, as described in equation (3.16),

$$\mathbf{\Delta x} = (\mathbf{A^T P A})^{-1}\mathbf{A^T P w} \tag{3.16}$$

where the measurement closure vector, $\mathbf{w}$, is defined as follows:

$$\mathbf{w} = \begin{bmatrix} \hat{d}_1 - d_1 \\ \hat{d}_2 - d_2 \\ \hat{d}_3 - d_3 \end{bmatrix} \tag{3.17}$$

$\hat{d}_i$ is the estimated distance from the SNi computed as:

$$\hat{d}_i = \sqrt{(\hat{x}_{j-1} - x_{s,i})^2 + (\hat{y}_{j-1} - y_{s,i})^2} \tag{3.18}$$

and the measurement weight matrix, $\mathbf{P}$, is defined as follows:

$$\mathbf{P} = \begin{bmatrix} \frac{1}{\sigma_1^2} & 0 & 0 \\ 0 & \frac{1}{\sigma_2^2} & 0 \\ 0 & 0 & \frac{1}{\sigma_3^2} \end{bmatrix} \tag{3.19}$$

where σ_i^2 is the variance of the measurement from SN i.

The correction vector $\mathbf{\Delta x}$ is the error of our estimation so we add it to the current estimation and try to minimize it in an iterative manner as follows:

$$\mathbf{\hat{x}_j} = \mathbf{\hat{x}_{j-1}} + \mathbf{\Delta x} \tag{3.20}$$

We stop the iteration when $|\mathbf{\Delta x}| \leq \epsilon$.

The advantage of this algorithm is its low mathematical complexity. On the other hand, this algorithm needs a good estimate of the start position of the TN and takes some iterations to converge to an accurate estimate. We implemented this positioning algorithm on the Decawave EVB1000 board. The results and comparison with the other positioning algorithms are discussed in chapter 6.

3.3 Enhanced Linearized Least Squares Algorithm

Cheung et al. [12] proposed a matrix representation of the non-linear least squares (NLS) problem in equation (3.11). With three SNs the matrix representation is as follows:

$$\mathbf{Ax} = \mathbf{b} \tag{3.21}$$

$$\text{where, } \mathbf{A} = \begin{bmatrix} x_{s,1} & y_{s,1} & -0.5 \\ x_{s,2} & y_{s,2} & -0.5 \\ x_{s,3} & y_{s,3} & -0.5 \end{bmatrix}, \mathbf{b} = \frac{1}{2} \begin{bmatrix} x_{s,1}^2 + y_{s,1}^2 - d_1^2 \\ x_{s,2}^2 + y_{s,2}^2 - d_2^2 \\ x_{s,3}^2 + y_{s,3}^2 - d_3^2 \end{bmatrix}, \text{ and}$$

$$\mathbf{x} = \begin{bmatrix} x \\ y \\ s \end{bmatrix} \text{ with s} = x^2 + y^2.$$

The solution for this equation is [12]:

$$\hat{\mathbf{x}} = (\mathbf{A^T A})^{-1} \mathbf{A^T b} \tag{3.22}$$

which includes the non-linear term, s. Guvenc et al. [20] presented a new linearized version of the equation (3.21), called LLS, by choosing a reference SN and subtracting the corresponding terms from other rows of the equation's matrices. After rearranging the terms, we have the following linear model:

$$\mathbf{A}_2 x = \mathbf{b}_2 \tag{3.23}$$

where

$$\mathbf{A}_2 = \begin{bmatrix} x_{s,i} - x_{s,r} & y_{s,i} - y_{s,r} \\ x_{s,j} - x_{s,r} & y_{s,j} - y_{s,r} \end{bmatrix}, \mathbf{b}_2 = \frac{1}{2} \begin{bmatrix} (d_r^2 - d_i^2) - k_i \\ (d_r^2 - d_j^2) - k_j \end{bmatrix}$$

$$k_n = (x_{s,r}^2 + y_{s,r}^2) - (x_{s,n}^2 + y_{s,n}^2)$$

r is the index of the reference SN and i and j are indexes of the other two SNs. The geometric representation of LLS is depicted in Figure 3.2. The range measurements by the SNs provide the circles as the potential position of the TN and so the non-linear least squares problem is to find the intersection of the three circles. The LLS algorithm chooses one of the SNs as the reference, and connects intersections of the reference node's circle with each of two other circles by a line. Therefore, instead of

the circles, we have two lines as the potential position of the TN. The linear equation (3.23) is used to find the intersection of these two lines as follows:

$$\mathbf{x} = \mathbf{A}_2^{-1}\mathbf{b}_2 \tag{3.24}$$

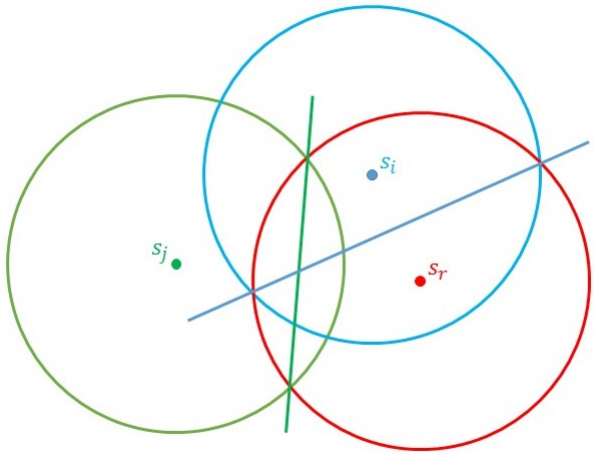

Figure 3.2: Geometric representation of the linearization method described by (3.23).

It is proven [18] that the distance of the TN from the reference SN has a significant impact on the error of the LLS algorithm. To increase the accuracy of LLS it is suggested to choose the closest SN as the reference node; we call this algorithm LLS-RS. By subtracting the reference node's term from the others we make the noise of the system correlated. It is shown in [46] that the optimum estimator in the presence of correlated noise is the Maximum Likelihood Estimator (MLE). The MLE solution for LLS-RS is given in [46] as follows:

$$\hat{\mathbf{x}} = (\mathbf{A_2^T C^{-1} A_2})^{-1}\mathbf{A_2^T C^{-1} b_2} \tag{3.25}$$

where covariance matrix $\mathbf{C}$ is defined as follows:

$$\mathbf{C} = \begin{bmatrix} 4d_r^2\sigma^2 + 4\sigma^4 + 4\sigma^2 d_1^2 & 4d_r^2\sigma^2 + 2\sigma^4 & 4d_r^2\sigma^2 + 2\sigma^4 \\ 4d_r^2\sigma^2 + 2\sigma^4 & 4d_r^2\sigma^2 + 4\sigma^4 + 4\sigma^2 d_2^2 & 4d_r^2\sigma^2 + 2\sigma^4 \\ 4d_r^2\sigma^2 + 2\sigma^4 & 4d_r^2\sigma^2 + 2\sigma^4 & 4d_r^2\sigma^2 + 4\sigma^4 + 4\sigma^2 d_3^2 \end{bmatrix} \quad (3.26)$$

and σ is the standard deviation of the range measurements.

We implemented the LLS-RS algorithm using the MLE solution on the EVB1000 board for trilateration. In the next chapters we call the LLS-RS algorithm using MLE the LRM algorithm. Figure 3.3 depicts the architecture of this positioning algorithm. The results and comparison with other positioning algorithms are discussed in chapter 6.

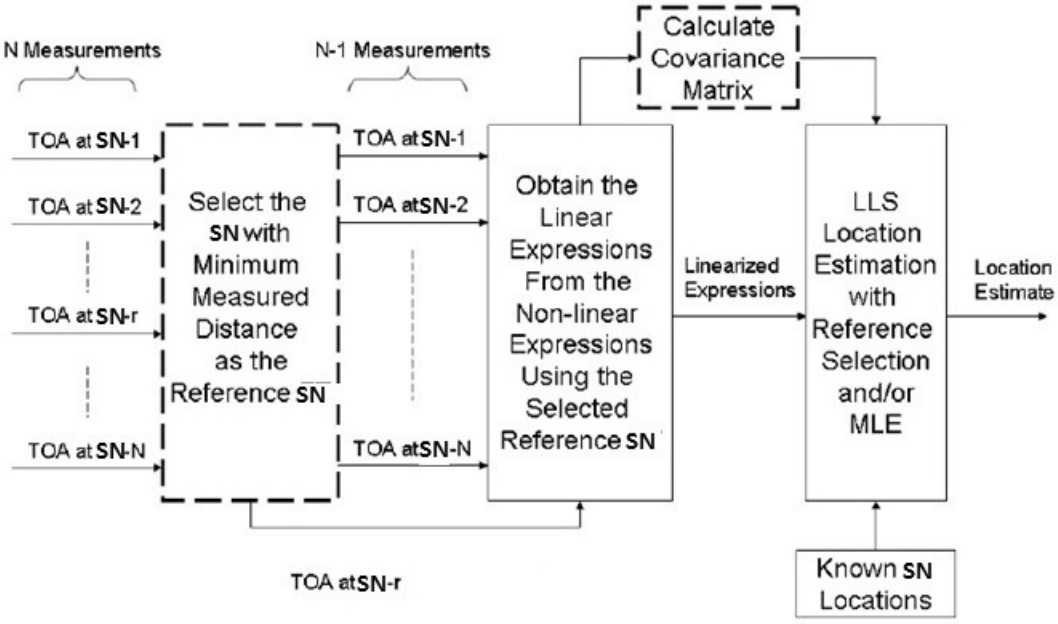

Figure 3.3: Algorithm architecture for LLS-RS with MLE solution (LRM) (modified from [20]).

3.4 Kalman Filter

A Kalman filter estimates the new state of a process in two steps. In the first step, prediction, the new state of the process is estimated by previous information in the absence of a new measurement. The second step, correction, uses new noisy measurements as feedback to correct estimated values at the prediction step.

3.4.1 Process model

A Kalman filter describes the process using two linear models. The first model, called the transition model, represents the relation between the new state and the previous state in the presence of optional control commands and transition noise. The transition model is expressed as

$$\mathbf{x_k} = \mathbf{A}\mathbf{x_{k-1}} + \mathbf{B}\mathbf{u_k} + \mathbf{w_k} \tag{3.27}$$

where $\mathbf{x_k}$ is the state vector of the process in the kth time interval. For example, the state matrix may include the position of a TN in two dimensions. Matrix $\mathbf{A}$ describes the relation of two successive states in absence of the control command and transition noise. $\mathbf{u_k}$ is the optional control command of the system in the kth time interval and $\mathbf{B}$ describes the effect of control commands (e.g. steering or throttle changes) on the state. Finally, $\mathbf{w_K}$ is a zero mean white Gaussian noise with covariance $\mathbf{Q}$. The second model, called the observation model, describes the relation between the measurement and the new state as

$$\mathbf{z_k} = \mathbf{H}\mathbf{x_k} + \mathbf{v_k} \tag{3.28}$$

where $\mathbf{z_k}$ is the noisy measurement vector in the kth time interval, $\mathbf{H}$ describes the relation between state and measurement in the absence of noise, and $\mathbf{v_k}$ is the

measurement noise which is assumed to be zero mean white Gaussian noise with covariance $\mathbf{R}$.

There are two essential assumptions of the Kalman filter: (1) two linear relations describe the state transitions and the observation vectors, and (2) the transition noise and measurement noise are zero mean white Gaussian.

3.4.2 Kalman Filter Equations

Kalman filter has two steps: prediction and correction. The prediction step gives an a priori estimate of the new state, $\tilde{\mathbf{x}}_\mathbf{k}$, based on information prior to the kth measurement. The prediction step is represented as (adapted from [55]):

$$\tilde{\mathbf{x}}_\mathbf{k} = \mathbf{A}\hat{\mathbf{x}}_{\mathbf{k-1}} + \mathbf{B}\mathbf{u}_\mathbf{k} \tag{3.29}$$

where $\hat{\mathbf{x}}_{\mathbf{k-1}}$ is the final estimate (a posteriori estimate) at the $(k-1)$th time interval. In the correction step, the a posteriori estimate is computed as

$$\hat{\mathbf{x}}_\mathbf{k} = \tilde{\mathbf{x}}_\mathbf{k} + \mathbf{M}_\mathbf{k}(\mathbf{z}_\mathbf{k} - \mathbf{H}\tilde{\mathbf{x}}_\mathbf{k}) \tag{3.30}$$

where

$$\mathbf{M}_\mathbf{k} = \tilde{\mathbf{P}}_\mathbf{k}\mathbf{H}^\mathbf{T}(\mathbf{H}\tilde{\mathbf{P}}_\mathbf{K}\mathbf{H}^\mathbf{T} + \mathbf{R})^{-\mathbf{1}} \tag{3.31}$$

$$\tilde{\mathbf{P}}_\mathbf{k} = \mathbf{A}\hat{\mathbf{P}}_{\mathbf{k-1}}\mathbf{A}^\mathbf{T} + \mathbf{Q} \tag{3.32}$$

$$\hat{\mathbf{P}}_\mathbf{k} = (\mathbf{1} - \mathbf{M}_\mathbf{k}\mathbf{H})\tilde{\mathbf{P}}_\mathbf{k} \tag{3.33}$$

A good introduction to the Kalman filter including more details is given in [55]. A useful observation from equation (3.30) is that the a posteriori estimate is the addition of the a priori estimate and a weighted difference between the measurement and predicted measurement. Note that in equation (3.31) for $\mathbf{M}_\mathbf{k}$ we observe that if the

measurement error $\mathbf{R}$ approaches zero, $\mathbf{M_k}$ becomes $\mathbf{H^{-1}}$, and the a posteriori estimate, $\hat{\mathbf{X}}_\mathbf{k}$, is $\mathbf{H^{-1}z_k}$. In other words, due to the high accuracy of the measurement, we do not use the a priori estimate. On the other hand, if $\mathbf{R}$ becomes significantly larger, $\mathbf{M_k}$ approaches zero and the a posteriori estimate is equal to the a priori estimate. In this case, due to large measurement noise, we do not trust the new measurement and just use a prior information. We give an example (from [60]) to illustrate the usability of the Kalman filter. Assume that there are four SNs measuring the TOA of a TN which moves with a constant velocity of 1m/s . Each SN makes four measurements per second. The noise of measurement is modeled by a zero mean normal distribution with variance of 0.5m^2. At each measurement, the position of the TN is estimated by the MLE algorithm represented by equation (3.11), and then the Kalman filter is applied to the result of the MLE algorithm. In this example, the state vector of the Kalman filter is

$$\mathbf{x_k} = [x_k, y_k, \dot{x_k}, \dot{y_k}]^T \tag{3.34}$$

where (x_k, y_k) and $(\dot{x_k}, \dot{y_k})$ are the position of the TN and its velocity at the kth measurement. Also the measurement vector is the output of the MLE estimator represented as:

$$\mathbf{z_k} = [x_{z_k}, y_{z_k}]^T \tag{3.35}$$

The matrix $\mathbf{A}$ in equation (3.27) is

$$\mathbf{A} = \begin{bmatrix} 1 & 0 & \Delta t & 0 \\ 0 & 1 & 0 & \Delta t \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{3.36}$$

where Δt is the time interval between successive measurements. This definition for

matrix $\mathbf{A}$ is based on the assumption that the TN is moving with a constant speed; so, for example, our estimate for the new position of the TN and its velocity in the x dimension are

$$\hat{\mathbf{x}}_{\mathbf{k}} = \mathbf{x}_{\mathbf{k-1}} + \Delta t \dot{\mathbf{x}}_{\mathbf{k-1}} \tag{3.37}$$

and

$$\hat{\dot{x}}_k = \dot{x}_k \tag{3.38}$$

respectively. The noise matrix in the transition model, equation (3.30) is

$$\mathbf{W}_{\mathbf{k}} = \begin{bmatrix} \mathcal{N}(0,0) \\ \mathcal{N}(0,0) \\ \mathcal{N}(0,(\Delta t)^2) \\ \mathcal{N}(0,(\Delta t)^2) \end{bmatrix} \tag{3.39}$$

which applies a random acceleration in order to compensate for possible changes of the TN's speed in practice.

Matrix $\mathbf{H}$ in equation (3.30) is

$$\mathbf{H} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \tag{3.40}$$

Note that the measurement includes just the position but not the velocity of the TN. Matrix $\mathbf{v_k}$, noise of the measurement, is

$$\mathbf{v}_{\mathbf{k}} = \begin{bmatrix} \mathcal{N}(0,0.5) \\ \mathcal{N}(0,0.5) \end{bmatrix} \tag{3.41}$$

which applies a zero mean normal random noise with 0.5m^2 variance to the position measurement.

Figure 3.4 depicts the result of this example. The black line is the true path of the

TN, crosses are positions estimated by the MLE algorithm, and the heavy solid line is the result of applying the Kalman filter.

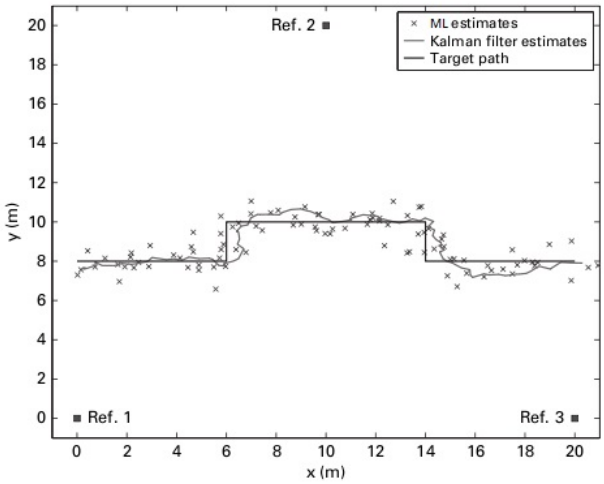

Figure 3.4: Kalman filter example (from [60]).

Chapter 4

Positioning with IMU

4.1 IMU Sensors

4.1.1 Accelerometer

An accelerometer is an electromechanical device measuring both static and dynamic acceleration in the three orthogonal axes of a reference frame mounted on the sensor. An example of static acceleration is gravity. An accelerometer measures approximately $9.8\frac{\mathrm{m}}{\mathrm{s}^2}$ in a stationary position when no external force is applied and measures zero if it is in a free-fall condition. Dynamic acceleration is made by external forces such as moving or tapping the accelerometer. The direction of the gravity vector $\mathbf{g}$ measured by the accelerometer can be used to determine the orientation of the sensor. Also, the dynamic acceleration measurements can be used to compute the velocity of a moving sensor, and subsequently its position change over time. Nowadays, microelectromechanical system (MEMS) accelerometers are widely used in gaming consoles and smart-phones to provide a movement-based interface to users. MEMS accelerometers' dimensions are as small as a few millimeters as depicted in Figure 4.1. In addition, these types of accelerometers are very power efficient, as low as tens of μA in measurement mode and less than 5 μA in standby mode at $3 - 5V$.

4.1.2 Gyroscope

A gyroscope is a device measuring the rotational velocity around three axes of a reference frame. The output of gyroscope is a vector composed of the rotational speed around three orthogonal axes in degrees per second ($\frac{\text{deg}}{s}$). The gyroscope was invented in 1852 [2]; at the beginning it was a purely mechanical gyroscope with three gimbals, a frame, a spin axis and rotor as depicted in Figure (4.2).

Modern gyroscopes are usually based on MEMS (microelectromechanical system).

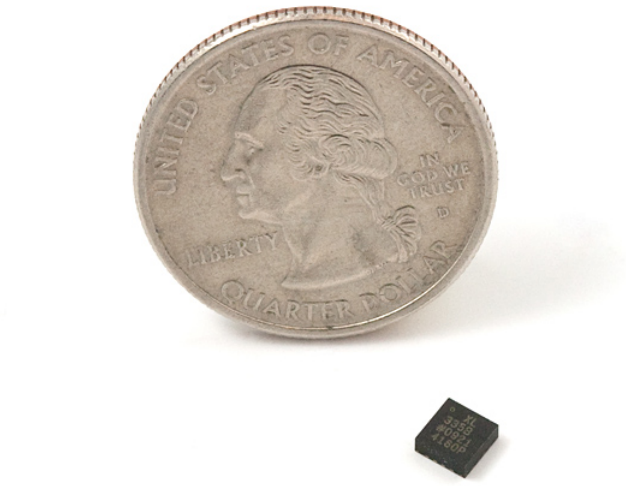

Figure 4.1: An example of accelerometer dimensions (from [3]).

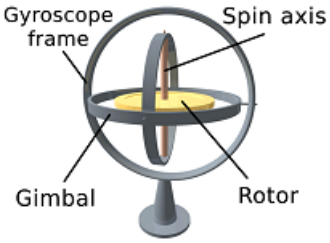

Figure 4.2: Mechanical gyroscope (from Wikipedia).

Two common applications of gyroscopes are development of inertial navigation systems and stabilization control for aerial vehicles or boats. A gyroscope combined with an accelerometer is used for direction and motion detection in many modern smart-phones and gaming consoles.

In this thesis, a gyroscope is used to measure the rotational velocity and then compute the orientation of the moving robot in the navigation frame by integrating the angular velocities.

4.2 Reference Frames and Rotation

Each navigation system needs a reference frame to represent position, velocity and orientation. The frame used to represent the final results of positioning is called "Navigation Frame". The measurements reported by the accelerometer and gyroscope sensors are represented in frames mounted on each sensor. The data sheet of the sensors describes the orientation of the frame axes. For simplicity of the mathematical equations, we assume the reference frames of the two sensors are aligned together. In addition, we assume the sensor boards are mounted in the center of the robot frame. So, the measurement from both sensors are represented in a reference frame mounted on the center of the robot frame called the "body frame". Figure (4.3) depicts the relation between the navigation frame and the body frame.

The gyroscope is used to compute the orientation of the body frame corresponding to the navigation frame, θ, as follows:

$$\theta = g_z \Delta t \tag{4.1}$$

where Δt is the time interval for one position update and g_z is the angular velocity measured by the gyroscope around the z axis.

Acceleration measurement by the accelerometer is represented as follows:

$$\mathbf{a_b} = [a_x, a_y]^T \tag{4.2}$$

where a_x and a_y are the measured accelerations in the X and Y axis directions. The $\mathbf{a_b}$ vector is in the body frame, while $\mathbf{a_n}$ is the equivalent representation of $\mathbf{a_b}$ in the navigation frame. We use a rotation matrix $\mathbf{T_b^n}$ to rotate $\mathbf{a_b}$ and compute $\mathbf{a_n}$, as follows:

$$\mathbf{a_n} = \mathbf{a_b}\mathbf{T_b^n} \tag{4.3}$$

where

$$\mathbf{T_b^n} = \begin{bmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{bmatrix} \tag{4.4}$$

We use a low-pass filter to reduce the noise of accelerometer measurements as follows:

$$\bar{\mathbf{a}}_\mathbf{n} = \mathbf{a_n}\alpha + (\bar{\mathbf{a}}_{\mathbf{n-1}}(1.0 - \alpha)) \tag{4.5}$$

where $\bar{\mathbf{a}}_\mathbf{n}$ is the output of the filter for the nth measurement and α is the constant value of 0.2. The value of 0.2 determined experimentally to give reasonably smooth

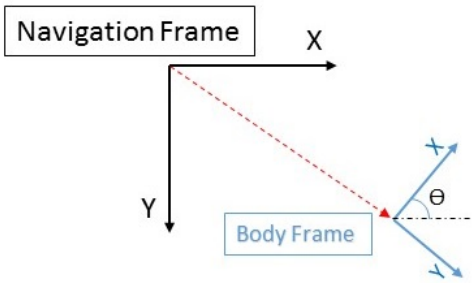

Figure 4.3: Navigation and body reference frames.

values for the accelerometer used in our experiments (see Chapter 5).

4.3 Data Fusion Using Extended Kalman Filter

A prerequisite for the standard Kalman Filter described in section 3.4 is linear transition and observation models as shown in equations (3.27) and (3.28), respectively. In the positioning system the state and measurement vectors are:

$$\mathbf{x} = \begin{bmatrix} x \\ y \\ v_x \\ v_y \end{bmatrix} \tag{4.6}$$

and

$$\mathbf{z} = \begin{bmatrix} d_1 \\ d_2 \\ d_3 \end{bmatrix} \tag{4.7}$$

where $[v_x, v_y]^T$ is the velocity vector, $[x, y]^T$ is the position vector and d_1, d_2 and d_3 are distances measured by the UWB nodes. The observation equation for this system is defined as follows:

$$\mathbf{z_k} = f(\mathbf{x_k}) + \mathbf{w} \tag{4.8}$$

$$\mathbf{f(x_k)} = \begin{bmatrix} \sqrt{(x_k - s_{x,1})^2 + (y_k - s_{y,1})^2} \\ \sqrt{(x_k - s_{x,2})^2 + (y_k - s_{y,2})^2} \\ \sqrt{(x_k - s_{x,3})^2 + (y_k - s_{y,3})^2} \end{bmatrix} \tag{4.9}$$

where $\mathbf{s_i} = [s_{x,i}, s_{y,i}]^T$ is the position of the UWB stationary node with ID i. This observation model is not linear for the positioning system, so the standard Kalman filter can't be used. The extended Kalman filter is a modified version of the Kalman filter which works for non-linear models.

4.3.1 Extended Kalman Filter

The main idea of the extended Kalman filter is to linearize the nonlinear models around the current value of the state vector. The general equations for nonlinear process and observation models are as follows [56]:

$$\mathbf{x_k} = \mathbf{f}(\mathbf{x_{k-1}}, \mathbf{u_k}, \mathbf{w}) \tag{4.10}$$

$$\mathbf{z_k} = \mathbf{h}(\mathbf{x_k}, \mathbf{v}) \tag{4.11}$$

where $\mathbf{w}$ is the process noise with covariance matrix $\mathbf{Q}$, $\mathbf{v}$ is the measurement noise with covariance matrix $\mathbf{R}$ and $\mathbf{u_k}$ is the input to the system. The extended Kalman filter uses the Jacobian matrix of each nonlinear function to linearize the model at a specific point as follows:

$$\mathbf{x_k} \approx \tilde{\mathbf{x}}_\mathbf{k} + \mathbf{A}(\mathbf{x_{k-1}} - \hat{\mathbf{x}}_\mathbf{k-1}) + \mathbf{w} \tag{4.12}$$

$$\mathbf{z_k} = \tilde{\mathbf{z}}_\mathbf{k} + \mathbf{H}(\mathbf{x_k} - \tilde{\mathbf{x}}_\mathbf{k}) + \mathbf{v} \tag{4.13}$$

$$\tilde{\mathbf{x}}_\mathbf{k} = \mathbf{f}(\hat{\mathbf{x}}_\mathbf{k-1}, \mathbf{u_k}, \mathbf{0}) \tag{4.14}$$

$$\tilde{\mathbf{z}}_\mathbf{k} = \mathbf{h}(\tilde{\mathbf{x}}_\mathbf{k}, \mathbf{0}) \tag{4.15}$$

where $\hat{\mathbf{x}}_\mathbf{k-1}$ is the a posteriori estimate of the state vector computed in the last iteration, $\tilde{\mathbf{x}}_\mathbf{k}$ is the a priori estimate of the state vector in the current iteration, $\mathbf{H}$ and $\mathbf{A}$ are the Jacobian matrices of $\mathbf{h}$ and $\mathbf{f}$ functions as follows:

$$\mathbf{A}_{[\mathbf{i},\mathbf{j}]} = \frac{\partial \mathbf{f}_{[\mathbf{i}]}}{\partial \mathbf{x}_{[\mathbf{j}]}}(\hat{\mathbf{x}}_\mathbf{k-1}, \mathbf{u_k}, \mathbf{0}) \tag{4.16}$$

$$\mathbf{H}_{[\mathbf{i},\mathbf{j}]} = \frac{\partial \mathbf{h}_{[\mathbf{i}]}}{\partial \mathbf{x}_{[\mathbf{j}]}}\mathbf{h}(\tilde{\mathbf{x}}_\mathbf{k}, \mathbf{0}) \tag{4.17}$$

In the positioning system the observation model is not linear as was shown in equation (4.9). Therefore we compute matrix $\mathbf{H}$ as follows:

$$\mathbf{H_k} = \frac{\partial \mathbf{h}(\tilde{\mathbf{x}}_\mathbf{k})}{\partial \mathbf{x_k}} = \begin{bmatrix} \frac{\tilde{x}_k - s_{x,1}}{\sqrt{(\tilde{x}_k - s_{x,1})^2 + (\tilde{y}_k - s_{y,1})^2}} & \frac{\tilde{y}_k - s_{y,1}}{\sqrt{(\tilde{x}_k - s_{x,1})^2 + (\tilde{y}_k - s_{y,1})^2}} \\ \frac{\tilde{x}_k - s_{x,2}}{\sqrt{(\tilde{x}_k - s_{x,2})^2 + (\tilde{y}_k - s_{y,2})^2}} & \frac{\tilde{y}_k - s_{y,2}}{\sqrt{(\tilde{x}_k - s_{x,2})^2 + (\tilde{y}_k - s_{y,2})^2}} \\ \frac{\tilde{x}_k - s_{x,3}}{\sqrt{(\tilde{x}_k - s_{x,3})^2 + (\tilde{y}_k - s_{y,3})^2}} & \frac{\tilde{y}_k - s_{y,3}}{\sqrt{(\tilde{x}_k - s_{x,3})^2 + (\tilde{y}_k - s_{y,3})^2}} \end{bmatrix} \tag{4.18}$$

where

$$\tilde{\mathbf{x}}_\mathbf{k} = [\tilde{x}_k, \tilde{y}_k]^T \tag{4.19}$$

The process model for the positioning system is linear. Matrices $\mathbf{A}$ and $\mathbf{B}$ are defined as follows:

$$\mathbf{A} = \begin{bmatrix} 1 & 0 & \Delta t & 0 \\ 0 & 1 & 0 & \Delta t \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{4.20}$$

$$\mathbf{B} = \begin{bmatrix} \frac{\Delta t^2}{2} & 0 \\ 0 & \frac{\Delta t^2}{2} \\ \Delta t & 0 \\ 0 & \Delta t \end{bmatrix} \tag{4.21}$$

The $\mathbf{B}$ matrix describes the effect of the TN's acceleration on the state vector (see equations (3.29) and (4.6)). The input to the positioning system are the acceleration measurements made by the accelerometer. The input matrix, $\mathbf{u}$, is defined as follows:

$$\mathbf{u} = \begin{bmatrix} a_x \\ a_y \end{bmatrix} \tag{4.22}$$

where a_x and a_y are the acceleration measurements along the X and Y axes by

the accelerometer in the navigation frame, and after applying the low pass filter $\bar{\mathbf{a}}$ explained by equation (4.5).

The a posteriori estimation of the state vector is computed as follows:

$$\hat{\mathbf{x}}_\mathbf{k} = \tilde{\mathbf{x}}_\mathbf{k} + \mathbf{K}_\mathbf{k}(\mathbf{z}_\mathbf{k} - \mathbf{f}(\tilde{\mathbf{x}}_\mathbf{k}, \mathbf{0})) \tag{4.23}$$

where the Kalman gain matrix, $\mathbf{K}_\mathbf{k}$, is computed as follows:

$$\mathbf{K}_\mathbf{k} = \tilde{\mathbf{P}}_\mathbf{k}\mathbf{H}_\mathbf{k}^\mathbf{T}(\mathbf{H}_\mathbf{k}\tilde{\mathbf{P}}_\mathbf{k}\mathbf{H}_\mathbf{k}^\mathbf{T} + \mathbf{R})^{-\mathbf{1}} \tag{4.24}$$

the error covariance matrix, $\mathbf{P}$ is defined as follows:

$$\tilde{\mathbf{P}}_\mathbf{k} = \mathbf{A}\hat{\mathbf{P}}_{\mathbf{k-1}}\mathbf{A}^\mathbf{T} + \mathbf{B}\mathbf{Q}\mathbf{B}^\mathbf{T} \tag{4.25}$$

and

$$\hat{\mathbf{P}}_\mathbf{k} = (\mathbf{I} - \mathbf{K}_\mathbf{k}\mathbf{H}_\mathbf{k})\tilde{\mathbf{P}}_\mathbf{k} \tag{4.26}$$

The process noise covariance matrix $\mathbf{Q}$ in the positioning system is determined by the error variance of the accelerometer measurements in each axis. The measurement noise covariance matrix $\mathbf{R}$ is computed by the error variance of the UWB range measurements. Error variances were measured for both the UWB and IMU sensors in experiments as discussed in the next section.

4.3.2 Variable Covariance Matrices

Our observations show high accuracy of the UWB range measurements in LOS conditions while acceleration measurements are relatively noisy. Therefore, it is wise to choose a smaller value for the measurement noise covariance matrix $\mathbf{R}$ in relation to the process noise covariance $\mathbf{Q}$ in LOS conditions. On the other hand, in NLOS conditions, UWB measurement accuracy is decreased significantly by errors arising from

NLOS conditions. In NLOS conditions it is wise to trust IMU measurements more by increasing values in the measurement noise covariance matrix **R**. This change increases the accuracy of positioning in NLOS conditions. To handle these two different situations we use a variable covariance approach. Decawave has suggested a method to distinguish between LOS and NLOS conditions using the total received power, called RXPOWER, and the first path received power, called FPPOWER as follows:

Algorithm 1 NLOS condition detection algorithm

```
1: input: i, source node number
2: procedure IsNlos
3:     RXPOWER = GetRXPOWER(i)
4:     FPPOWER = GetFPPOWER(i)
5:     d = RXPOWER - FPPOWER
6:     if d > 6 then
7:         return True
8:     else
9:         return False
```

Refer to [33] section 4.6 for a detailed description of this method. If a NLOS condition is detected by Algorithm 1, **R** is increased 30 times to make the UWB measurements less trusted. Experiments were run with different coefficient values instead of 30, including 2, 5, 10 and 50 to reach the best result. Finally, we chose 30 as it gave the best positioning accuracy. To the best of our knowledge, there is no suggestion in the literature about the relative weight of UWB vs IMU measurements.

Chapter 5

Experimental Setup

The architecture of our positioning and evaluation system is depicted in Figure 5.1.

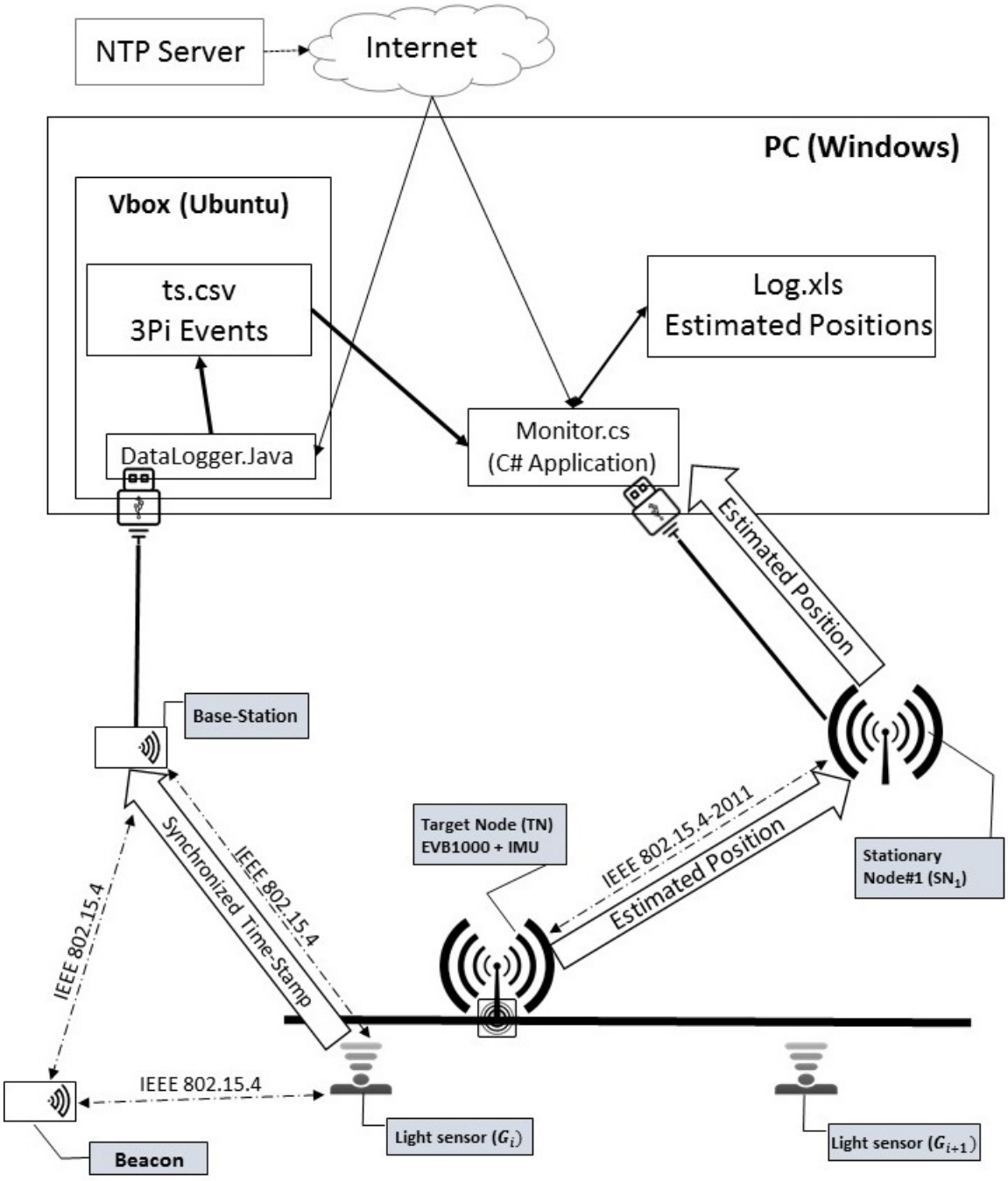

Figure 5.1: Experimental positioning and evaluation system architecture.

5.1 Hardware

5.1.1 Decawave

The Decawave company has an IEEE802.15.4-2011 compliant sensor which is able to perform positioning by TOA or TDOA approaches. The producer claims that the accuracy of this sensor working with 1.3 GHz bandwidth is ±10 cm. According to the producer website, the key benefits of this sensor, called DW1000, are precise ranging, long LOS and NLOS communication range (up to 290 m), high data rate (up to 6.8 Mbit/s) and low power consumption. The EVB1000 Evaluation Board is a complete device including the DW1000 IC, ARM Cortex M3 programmable microprocessor, LCD, USB connection and antenna. The ranging protocol SDS-TWR discussed in section (2.2.2.2) is implemented on the EVB1000, but not on the DW1000. The dimensions of the EVB1000 are 7×7 cm excluding the off-board antenna and the range of the center frequency of six available UWB channels is 3.5 to 6.5 GHz. The Decawave EVB1000 device is suitable for research and development because it is programmable and smaller and it has an embedded LCD and USB connections. The price of a kit of two EVB1000 boards is approximately CAD$850.

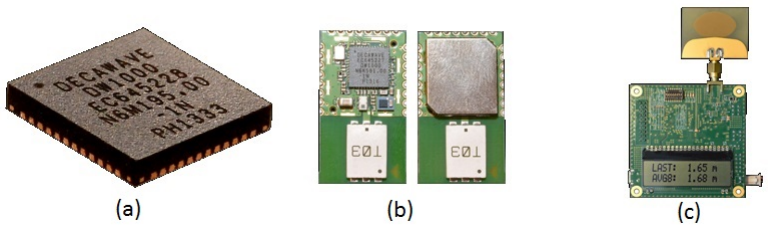

Figure 5.2: Decawave real time location technology (from [1]). (a) DW1000, (b) DWM1000, (c) EVB1000.

We used four EVB1000 boards in this thesis, one for the moving node and three for the stationary nodes.

5.1.2 ADXL345

The ADXL345 is a MEMS digital accelerometer produced by the Analog Devices company. The major features of this sensor are as follows:

- Ultra-low power consumption, as low as 40 μA in measurement mode and 0.1 μA in standby mode.
- Up to 13 bits measurement resolution in ± 16 **g** range of acceleration.
- Embedded FIFO queue to store up to 32 measurements.
- tap and double-tap detection options.

We connected the IMU sensor containing the accelerometer to the ARM processor of the EVB1000 board through an I2C connection. The embedded FIFO queue of this sensor is valuable for our positioning system as the highest acceleration sample frequency we can achieve is 7 Hz. A maximum 7 Hz sampling frequency is due to the time required to perform UWB range estimation and due to the lack of threading capability on the ARM microprocessor without an operating system.

5.1.3 ITG-3200

The ITG-3200 is a MEMS 3-axis gyro produced by InvenSense Inc. This gyro offers 16-bit digital measurement of angular velocity around each of the three axes. The operating current consumption of this gyro is 6.5 mA and it needs a voltage supply in range of 2.1 V to 3.6 V. The ITG-3200 features an embedded, configurable low-pass filter as well as an I2C interface.

We use this gyro to compute the orientation of the robot in the navigation frame; then this angle is used to compute the rotation matrix described in equation (4.4).

In this thesis we use the six degrees of freedom IMU produced by SparkFun Electronics which is composed of an ITG-3200 gyro and ADXL345 accelerometer, along

with the circuits needed for driving the sensors and I2C connection. The breakout board is depicted in Figure 5.3.

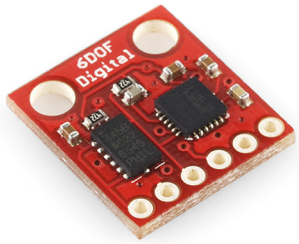

Figure 5.3: The 6DOF IMU model SEN-10121 produced by SparkFun Electronics and used in our experimental evaluation.

5.1.4 TelosB

The TelosB is a low power wireless sensor module developed by UC Berkeley. This module features 250 kbps data communication compliant with the IEEE802.15.4 standard. We use the embedded light sensor of the TelosB module in our real-time evaluation system, as discussed in section 5.2. We programmed this module using the TinyOS operating system. Figure 5.4 depicts the TelosB module.

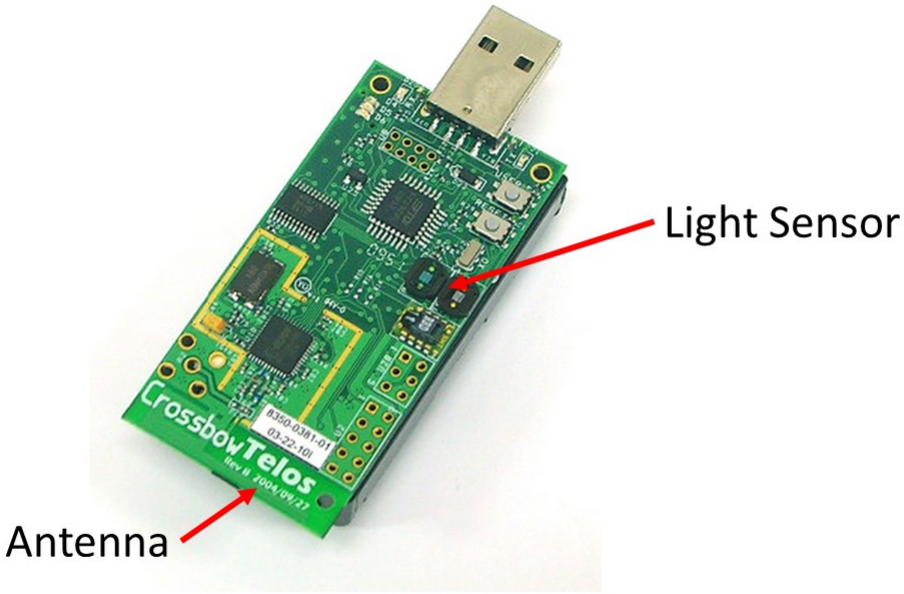

Figure 5.4: TelosB low-power sensor module.

5.1.5 Pololu 3pi Robot

The Pololu 3pi is a robot produced by the Pololu company that is optimized for line following. This robot is equipped with two micro metal gear-motors, five reflectance sensors, an LCD, a buzzer and push buttons connected to an ATmega328 micorcontroller. We used this robot to carry the target node EVB1000 board as well as the IMU board over the defined experimental route. We used black electrical tape to mark the route on the floor tiles of the IB214 wireless sensor network lab at UNB. Figure 5.5 shows the Pololu 3pi with EVB1000 and IMU.

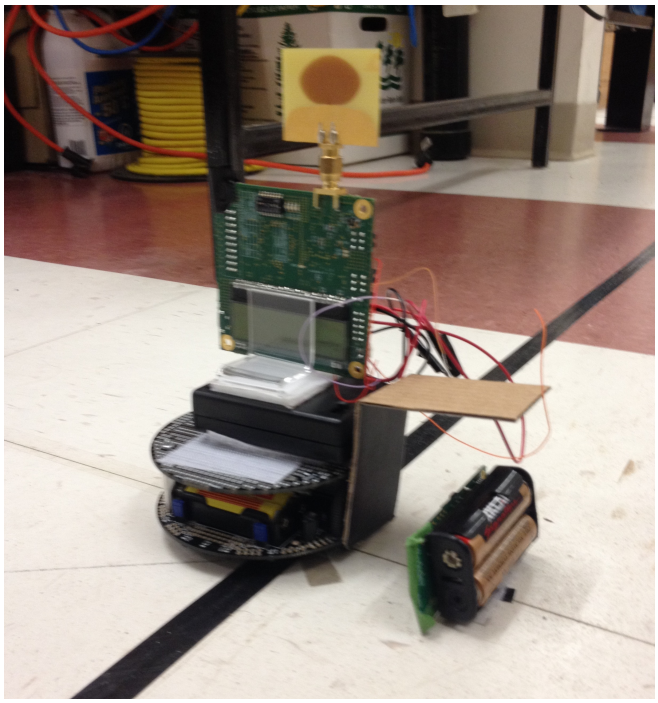

Figure 5.5: Pololu 3pi with EVB1000 and IMU sitting next to the line following route marked in black electrical tape.

5.2 Evaluation approaches

5.2.1 Off-line Evaluation

The most common approach, suggested in the literature, for evaluating positioning systems is using a predefined reference trajectory. In this method, the robot or human carrying the positioning system moves along the reference trajectory; then the computed positions by the positioning system provide an estimated trajectory. Accuracy of the positioning system is determined by the difference between the reference trajectory and the estimated trajectory. This approach can underestimate the error of the estimated positions compared to the reference trajectory. Figure 5.6 illustrates how this inaccurate estimate of the actual error can occur.

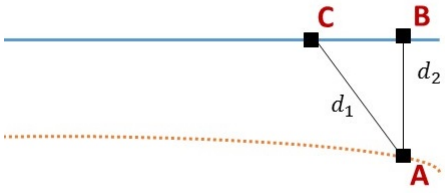

Figure 5.6: Potential error in off-line evaluation of the error in the estimated position.

Point A is reported by the positioning system when point C was the actual position of the moving node at time t. The off-line evaluation approach is not aware of the real-time actual position at time t, and computes the minimum distance between the estimated and the reference trajectory. Therefore, d_2 is the estimated error while the real error is d_1. The off-line evaluation method thus computes the minimum possible error of the system.

We implemented the off-line evaluation using `C#` language program called "Monitor.cs". Figure 5.7 depicts two possible cases. If the estimated position is in the gray

area, such as point B, the positioning error is the distance between the estimated position and the closest vertex as follows:

$$e = \min_{i \in \{1,3,5,7\}} \sqrt{(\hat{x} - x_{c_i})^2 + (\hat{y} - y_{c_i})^2} \tag{5.1}$$

where $(\hat{x}, \hat{y})$ is the estimated position and (x_{c_i}, y_{c_i}) is position of the ith vertex of the route. If the estimated position is in the white area, such as point A, the positioning error is the distance between the estimated position and the closest edge as follows:

$$e = \min_{i,1...4} \begin{cases} \hat{x} - r_i & i = 1, 3 \\ \hat{y} - r_i & i = 2, 4 \end{cases} \tag{5.2}$$

where r_1, r_2, r_3 and r_4 are the route's left vertices' x coordinate, top vertices' y coordinate, right vertices' x coordinate and bottom vertices' y coordinate, respectively. Appendix B contains the source code used for the off-line evaluation algorithm.

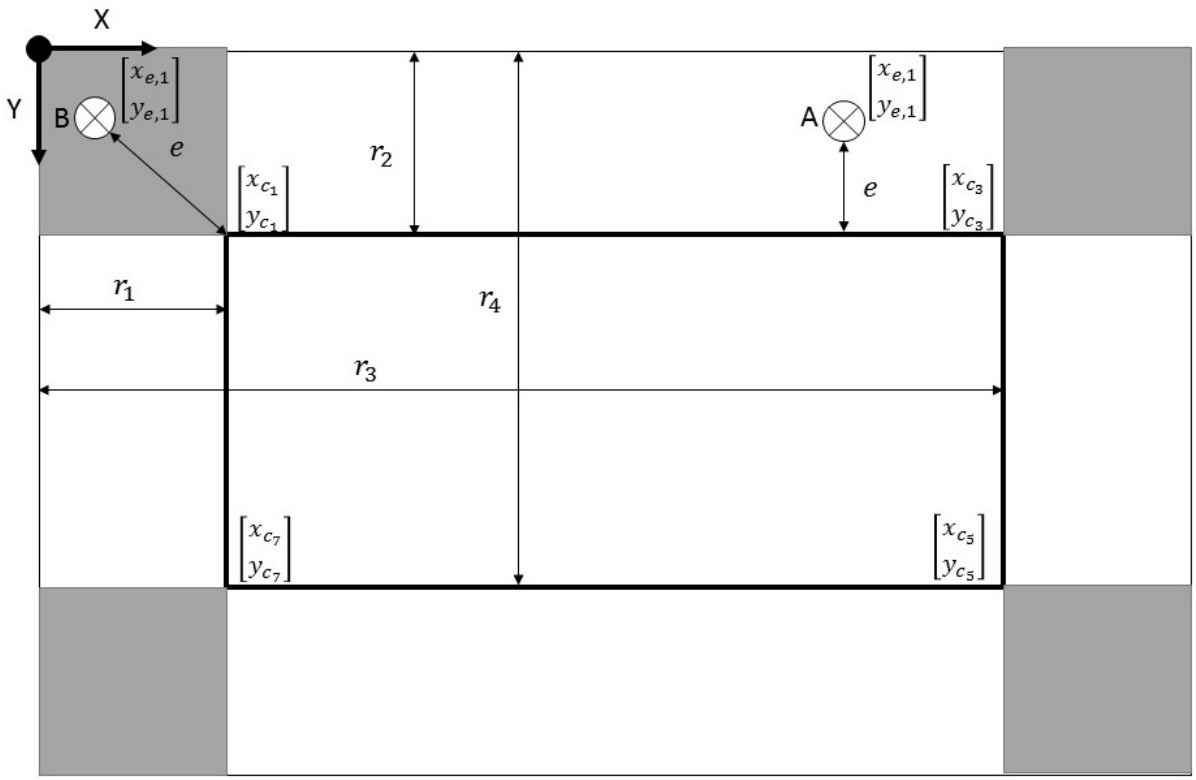

Figure 5.7: Illustration of off-line evaluation of the difference between an estimated position and a reference trajectory.

5.2.2 Real-Time Evaluation

We designed and implemented a second positioning system to determine the ground-truth position of the moving node in real time. This second positioning system is independent of the position estimation system, and takes advantage of our rectangular track layout. Figure 5.8 depicts the innovative real-time evaluation system used in this thesis.

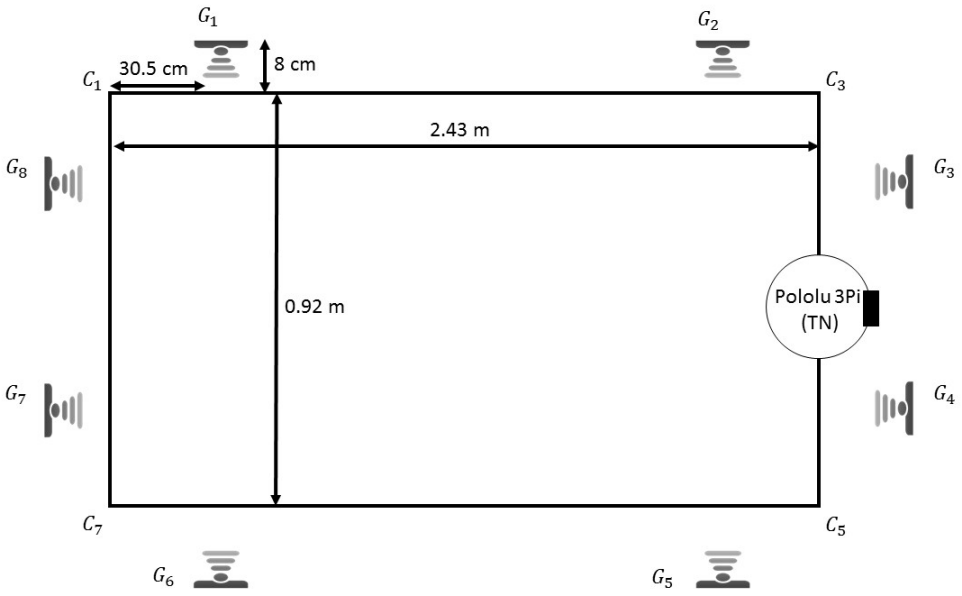

Figure 5.8: Real-time evaluation system.

We placed two TelosBs near each end of the route's edges (G1,...,G8 in Figure 5.8). The output of the light sensor embedded in the TelosB mote was used to determine when the 3pi robot passed by each TelosB along the route. Figure 5.9 shows a photo taken from the experimental setup.

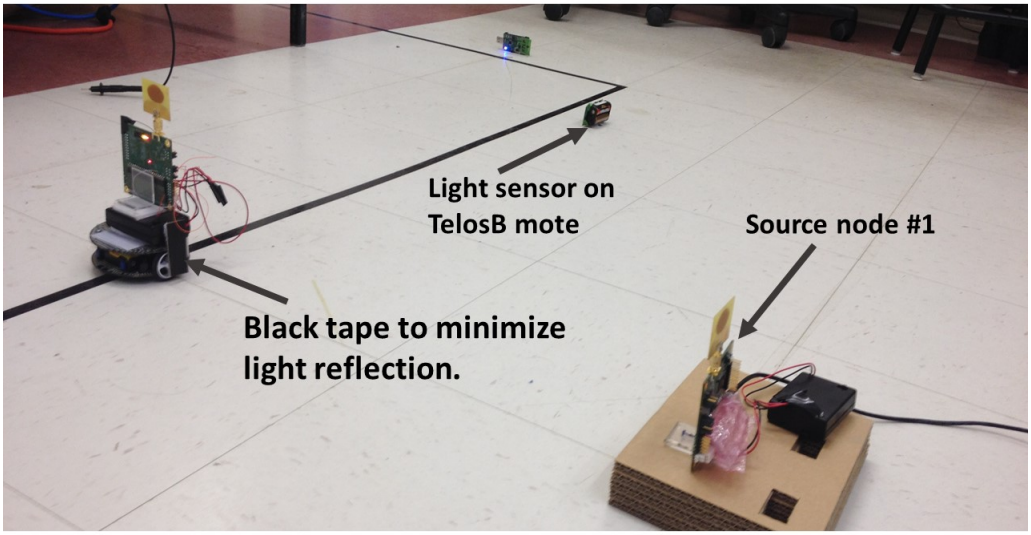

Figure 5.9: A picture of the real-time evaluation system for determining the ground truth position of the moving 3pi robot.

When the 3pi robot passes in front of the TelosB mote, it blocks the light reaching the light sensor on the TelosB mote G_n, and the light sensor output drops significantly. At this instant of time t_i, we create a synchronized time stamp event that is sent to the DataLogger.java program on the base station. Using two time-stamps from two consecutive ground truth motes on an edge, t_{G_n} and $t_{G_{n+1}}$, and the distance between two sensors, d, we compute the constant speed v of the 3pi robot on this edge. Using the speed of the robot and the time of passing a specific point on the edge where a ground truth mote is located, we are able to compute the position of the robot at any moment it was on this edge, using the following equations:

$$v = d/(t_{G_{n+1}} - t_{G_n}) \tag{5.3}$$

$$t_{c_n} = t_{G_n} - (0.30/v)\ ,\ t_{c_{n+2}} = t_{G_{n+1}} + (0.30/v) \tag{5.4}$$

$$\delta = v(t_i - t_{c_n}) \tag{5.5}$$

$$\text{Horizontal edge: if}(n \in \{1,5\}) : x'_{t_i} = x_{c_n} + \delta,\ y'_{t_i} = y_{c_n} \tag{5.6}$$

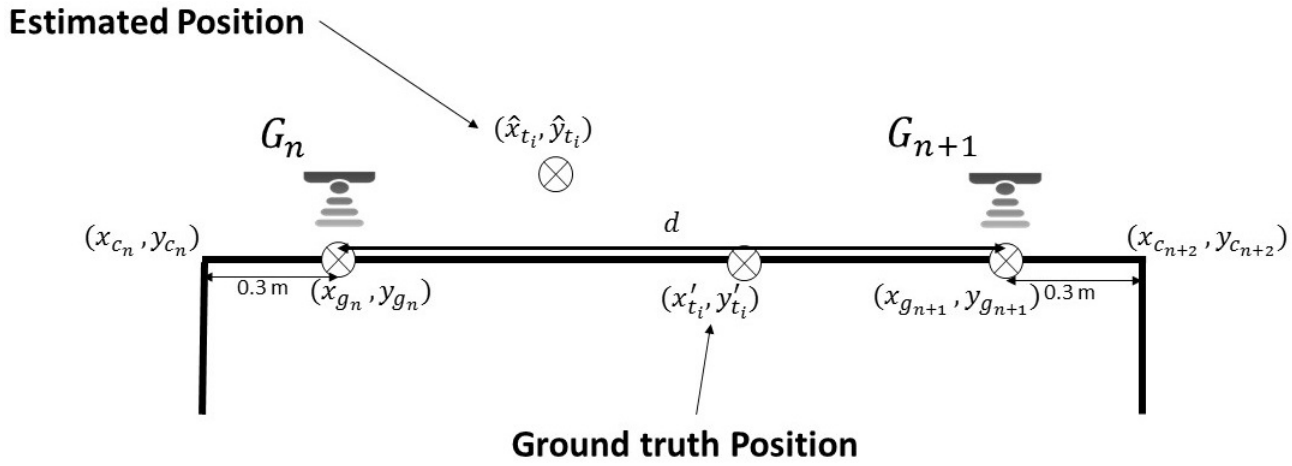

Figure 5.10: Real-time evaluation example.

$$\text{Vertical edge: if}(n \in \{3, 7\}) : y'_{t_i} = y_{c_n} + \delta,\ x'_{t_i} = x_{c_n} \tag{5.7}$$

The position error e is then computed as follows:

$$e = \sqrt{(\hat{x}_{t_i} - x'_{t_i})^2 + (\hat{y}_{t_i} - y'_{t_i})^2} \tag{5.8}$$

Figure 5.10 depicts an example of real-time evaluation. We repeat this procedure for all positions of the robot for the entire duration of the experiment using the Monitor.cs program, as illustrated in Figure 5.1. Appendix A contains the source code for the program used to compute the errors each second as the 3pi traversed the line following route.

The time synchronization protocol used to guarantee that all eight ground truth motes use the same time base is the Flooding Time Synchronization Protocol (FTSP). An implementation of FTSP is provided by TinyOS, and is described in more detail in [34].

5.3 Positioning Module

The positioning algorithm is implemented on the EVB1000. Decawave ships this module with sample source code for ranging between two EVB1000 modules. This range measurement was used as the input to the positioning algorithm. In the ranging implementation there are two roles, "Anchor" and "Tag". According to the terminology used here, all stationary nodes are anchors and the moving node is a tag. Figure 5.11 depicts the architecture of Decawave UWB ranging implementation. The second phase, labeled "Ranging Phase" in the figure, is the SDS-TWR protocol of IEEE 802.15.4-2011 standard discussed in section 2.10. The first phase of the implementation, called "Discovery Phase", is designed by Decawave to pair tags and anchors for ranging. In the case of positioning, we need the moving node to measure its range from three stationary nodes. We observed that the moving node is likely to pair with closer stationary nodes as they send the "Ranging init" message sooner. It is a problem for positioning since we need range measurements from all stationary nodes with the same frequency. We modified the implementation of the discovery phase (see the source code in Appendix C) so that the moving node (tag) pairs with the stationary nodes (anchors) sequentially using their unique ID.

We found calibration of the IMU sensor an essential part of the positioning system. As a result of our observations, we skipped the first 2000 measurements and used the next 1000 for calibration. Furthermore, we found calibration using the real positioning iterations more effective than running a stand-alone calibration phase without other procedures. So, we run the positioning algorithm and use the first IMU measurement for calibration discarding the positioning results, then start the real positioning phase using calibrated IMU measurements. Another implementation issue for the integrated IMU-UWB positioning system is that the sampling frequency of the IMU sensors is 100 Hz while the frequency of UWB ranging (with three stationary nodes) is 7 Hz. According to our experiments, on average, we have ranging

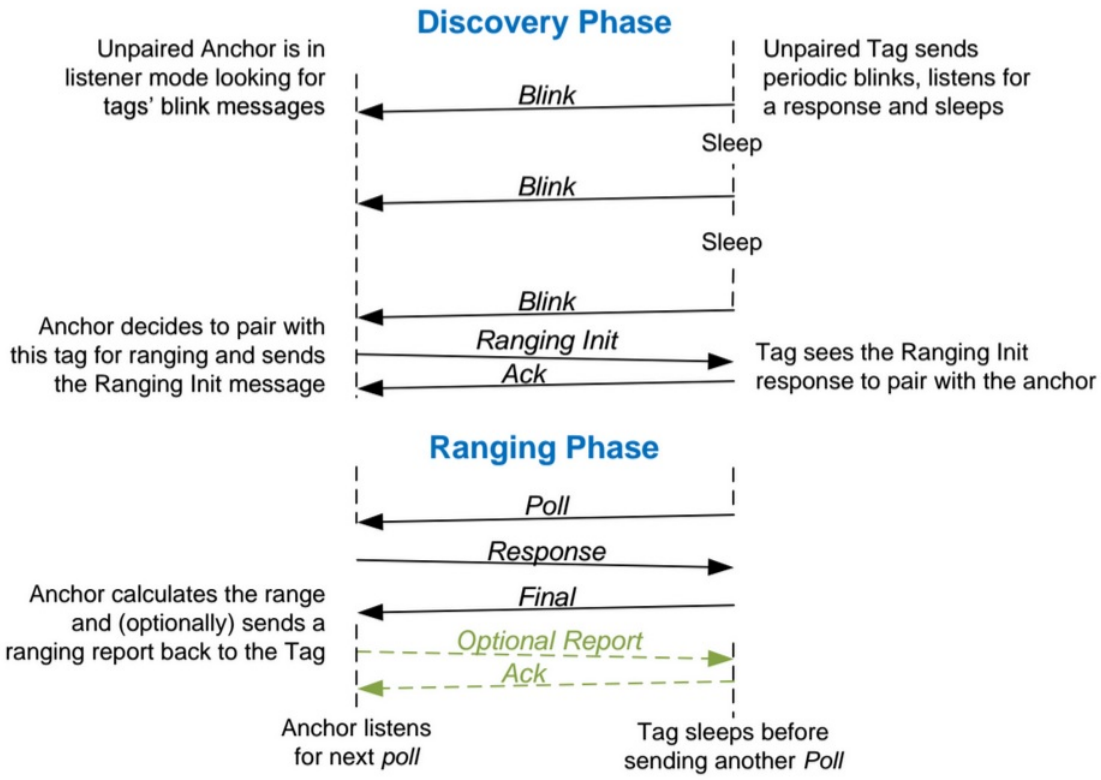

Figure 5.11: Ranging implementation by Decawave (from [32]).

measurements every 0.143 s, and there are 14 IMU samples available between two range measurements. There are three options to handle this problem:

- Reduce the frequency of the sensors' sampling to 7 Hz. We performed experiments applying this approach for both gyroscope and accelerometer. The dynamic noise in the gyroscope measurements is not high so the samples are pretty steady for 7 Hz with standard deviation $\sigma < 0.054 \deg/s$. However, the acceleration measurements have significant variations at the same 7 Hz rate, and decreasing the acceleration sampling frequency makes significant errors. Table 5.1 shows the standard deviation of gyroscope and accelerometer measurements over the first 6000 samples, in $\deg/s$ and cm/s^2 respectively.

Table 5.1: Standard Deviation of IMU measurements.

a_x	a_y	g_z
6.98 cm/s^2	6.4 cm/s^2	0.054 $\deg/s$

- Compute the process model of Kalman-filter using all 14 acceleration and angular velocity values available between two range measurements, and then use the last output as the estimated position. In experiments, we found this approach increased the positioning error. The IMU is not a good positioning system by itself even for 14 iterations. It should be coupled with another positioning system continuously.

- The third option, which gives the best results, is to make an average over 14 acceleration values and use the result as the input to the EKF. We used the embedded FIFO buffer of the ADXL345 for collecting the acceleration measurements between two range measurements.

5.4 Test Procedure

Running an experiment is started by computing the time-offset between the target node's time stamps, made by the EVB1000 internal clock, and the UTC time. For this purpose, we put the target node in synchronization mode using a toggle switch, and connected it to the workstation using a USB cable. When the TN is in synchronization mode, it transmits time-stamps via the USB port every 1s. On the workstation we have `Monitor.cs` listening to the serial ports. Once a time-stamp is received from the TN, the UTC time is retrieved from the global time server (`tick.usno.navy.mil`) on the Internet, in milliseconds. The offset between these two timestamps, called Δ_{TN}, is computed as:

$$\Delta_{TN} = t_{UTC} - t_{TN} \tag{5.9}$$

and recorded on the workstation's hard-drive. Then, the TN is returned to positioning mode to proceed with the main procedure described in Algorithm 2.

The second step is to install the light sensors (TelosB motes) on the determined positions around the experiment's route. There are 8 TelosB motes programmed to report if the light sensor is blocked. In addition, one TelosB mote (the beacon node) is programmed with the `RadioCountToLeds` example from the Tinyos distribution as needed for the FTSP algorithm. A 10th TelosB mote is programmed with a modified BaseStation program. The modification adds the code necessary to synchronize all TelosB motes using the FTSP algorithm [34]. The FTSP algorithm makes a synchronized clock for all the light-sensor nodes and the base-station node. The BaseStation node sends time-stamps made by this clock to the workstation (Ubuntu virtual-machine), program `DataLogger.java`. This program retrieves the UTC time from the same time-server on the Internet which is used by `Monitor.cs`. The retrieved UTC time-stamp is compared to the time stamps sent by the BaseStation

node and the computed time-offset, called Δ_{BS}, is computed as:

$$\Delta_{BS} = t_{UTC} - t_{BS} \tag{5.10}$$

and stored in memory. When a light-sensor reports light blockage by the Pololu 3pi (TN) accompanied with a time-stamp, Δ_{BS} is used to convert the time-stamp to UTC time as:

$$t_{G_n} = t_{G_n} + \Delta_{BS} \tag{5.11}$$

and writes the data to the `ts.csv` file.

The main procedure of the positioning module, programmed on the TN, starts with the calibration of IMU sensors. Then, an infinite loop starts which calls the IMU-UWB algorithm after a new set of distance measurements from all SNs is retrieved. As shown in Algorithm 2, the TN sends the IMU-UWB position estimate accompanied with a time-stamp and distance measurements to SN1, and the data is then sent to `Monitor.cs` on the workstation. Time-stamps are converted to UTC time-stamps as:

$$t_i = t_i + \Delta_{TN} \tag{5.12}$$

and all the data is written to `Log.xls` file.

When TN finishes three loops moving around the route, it is manually turned off. The file `ts.csv` is moved to the Windows OS (from the Ubuntu VM). The `Monitor.cs` program includes the RLM and LLS algorithms' implementation. It produces two other estimates of the target node positions during the experiment using the recorded distances in the `Log.xls` file. In addition, the time-stamps in `ts.csv` reported by the light sensors are used to reach a very accurate estimate of the position of the target node (as explained in equations (5.6) and (5.7)), called ground-truth position, related to the time-stamps reported by stationary nodes. We calculate the real-time (RT) error using equation (5.8) for each of LRM, LLS and IMU-UWB position estimates.

The output of the RT evaluation is the average calculated error in cm. An off-line (OL) evaluation is also computed for each of three position estimates using equation (5.1) and (5.2).

Algorithm 2 Target Node Algorithm

```
1: procedure MAIN PROCEDURE
2:     Counter = 0;
3:     while True do
4:         measureDistance();
5:         if All three distances received then
6:             read-Gyroscope();
7:             read-Accelerometer();
8:             UWB-IMU(); //Computes position estimate x̂_k using equation (4.23)
9:             if Counter < 3000 then
10:                doCalib();// Calibrate IMU sensor using first 3000 samples
11:                Counter++;
12:            else
13:                sendReport(); //send distances (d_1, d_2 and d_3), x̂_k and time-stamp
    t_i to SN_1
```

Chapter 6

Experimental Results

Two positioning algorithms are discussed in chapter 3. The first one is the linearized least squares algorithm, equation (3.20), and the second one is the enhanced linearized least squares by using MLE, equation (3.25); we call these two algorithms LLS and LRM, respectively. We use these two algorithms as UWB-based approaches to compare with the IMU-aided UWB positioning approach discussed in chapter 4, denoted as IMU-UWB. To reach a more accurate comparison between these three algorithms, we apply them to the same experimental data. The IMU-UWB algorithm is implemented on the target node (EVB1000), while the LRM and LLS algorithms are implemented on the PC using `Monitor.cs`. We ran two sets of experiments, one in LOS and the one in NLOS conditions. Each of the sets is comprised of three clockwise and three counter-clockwise loops of the target node moving around a defined route. In our experiments, the speed of Pololu 3pi robot which carries the target node and IMU board, is 0.16 m/s.

6.1 LOS Experiments

In this section we discuss the results from experiments performed in LOS conditions. Figure 6.1 depicts the positions of the stationary nodes and light sensors as well as

the route (of length 6.7 m) for the target node's trajectory.

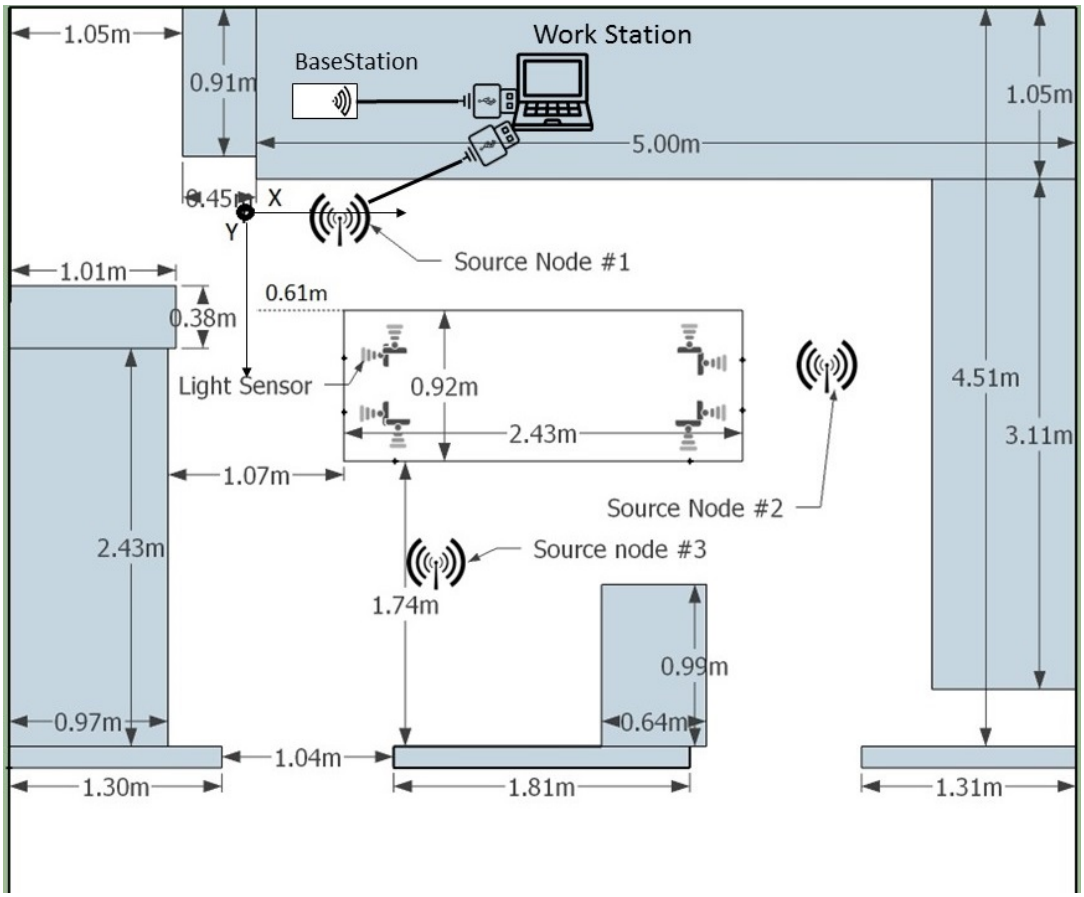

Figure 6.1: Detailed LOS experimental setup in the IB214 lab.

Table 6.1 shows the position of the source nodes for the LOS experiments.

Table 6.1: Position of source nodes (in cm) for LOS experiments.

ID	x	y
1	61	0
2	364.5	91.5
3	122	213.5

Figure 6.2 depicts the output of each algorithm for one loop (experiment number 6) in LOS conditions.

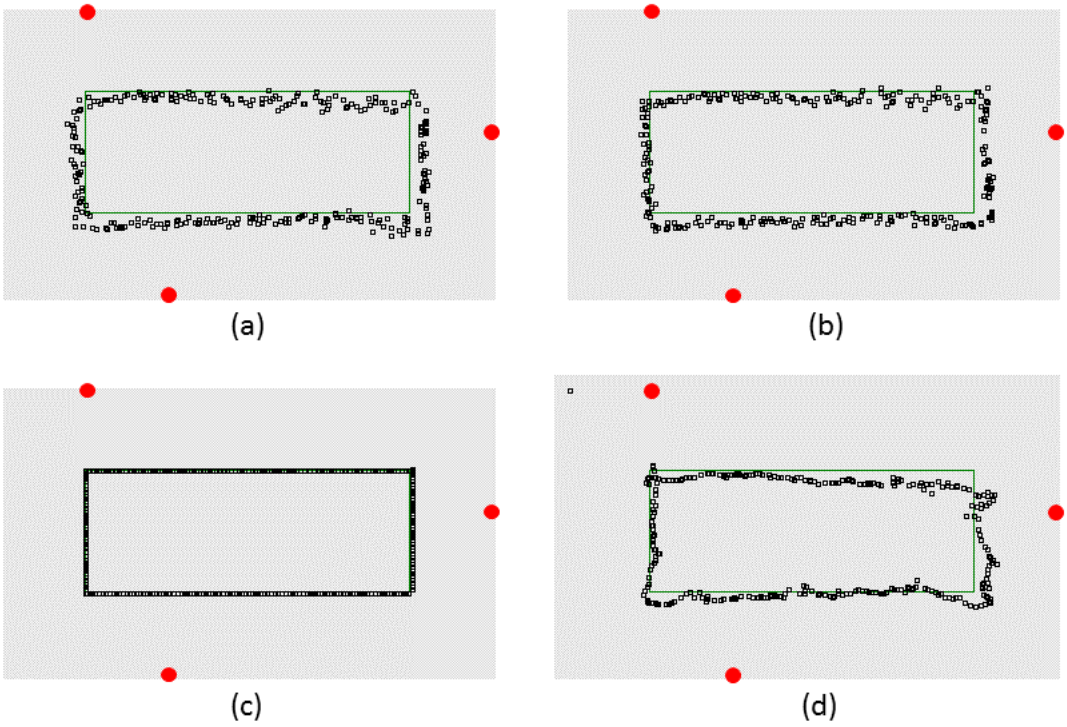

Figure 6.2: Positioning algorithms output in LOS conditions. (a) LRM, (b) LLS, (c) ground-truth position and (d) IMU-UWB. Red dots indicate the position of the source nodes.

We ran the experiments for three clockwise and three counter-clockwise loops comprising a total of 1737 points of positioning or around 386 points per loop. Table 6.2 shows the average real-time (RT) and off-line (OL) error of each positioning algorithm in LOS conditions.

The first observation is that the RT error is always more than the OL error. It is

Table 6.2: Results for LOS experiments. All results shown in cm.

ID	Points	Direction	LRM OL	LRM RT	LLS OL	LLS RT	IMU-UWB OL	IMU-UWB RT	Duration(s)
1	292	CCW	7.61	13.20	6.43	12.31	4.77	15.61	43.25
2	296	CCW	7.57	13.39	6.64	12.89	7.30	20.83	43.66
3	300	CCW	7.32	10.87	6.11	10.00	6.65	17.60	43.74
4	282	CW	5.79	9.72	4.63	8.76	6.28	10.70	45.05
5	288	CW	5.77	12.15	4.59	10.99	5.82	11.69	46.12
6	279	CW	5.79	11.83	4.50	10.85	4.40	8.60	45.19
Avg	1737	-	6.64	11.86	5.48	10.97	5.87	14.17	-

consistent with the fact that it is not accurate to choose the closest point on the route to the estimated position as ground-truth position as done for the OL evaluation. Secondly, we observe that the addition of the IMU doesn't increase the accuracy of the UWB positioning system. The results are explained due to the high accuracy of the UWB positioning system in LOS conditions and the relatively high noise of the measurements from the inexpensive IMU sensors we used. Two other investigations [15, 9] confirm the fact that inexpensive IMU systems are not able to increase the accuracy of UWB positioning system in LOS conditions. De Angelis et al. [15] suggest that IMU can improve positioning accuracy if the target node has speeds higher than 1 m/s in LOS conditions.

6.2 NLOS Experiments

To evaluate the positioning systems in NLOS conditions, we designed a second route (of length 9.7 m) for the moving robot as depicted in Figure 6.3. Similar to the LOS experiments, we ran the experiment for three clockwise (CW) and three counter-clockwise (CCW) loops. A total of 2789 points of positioning were collected and evaluated for NLOS conditions.

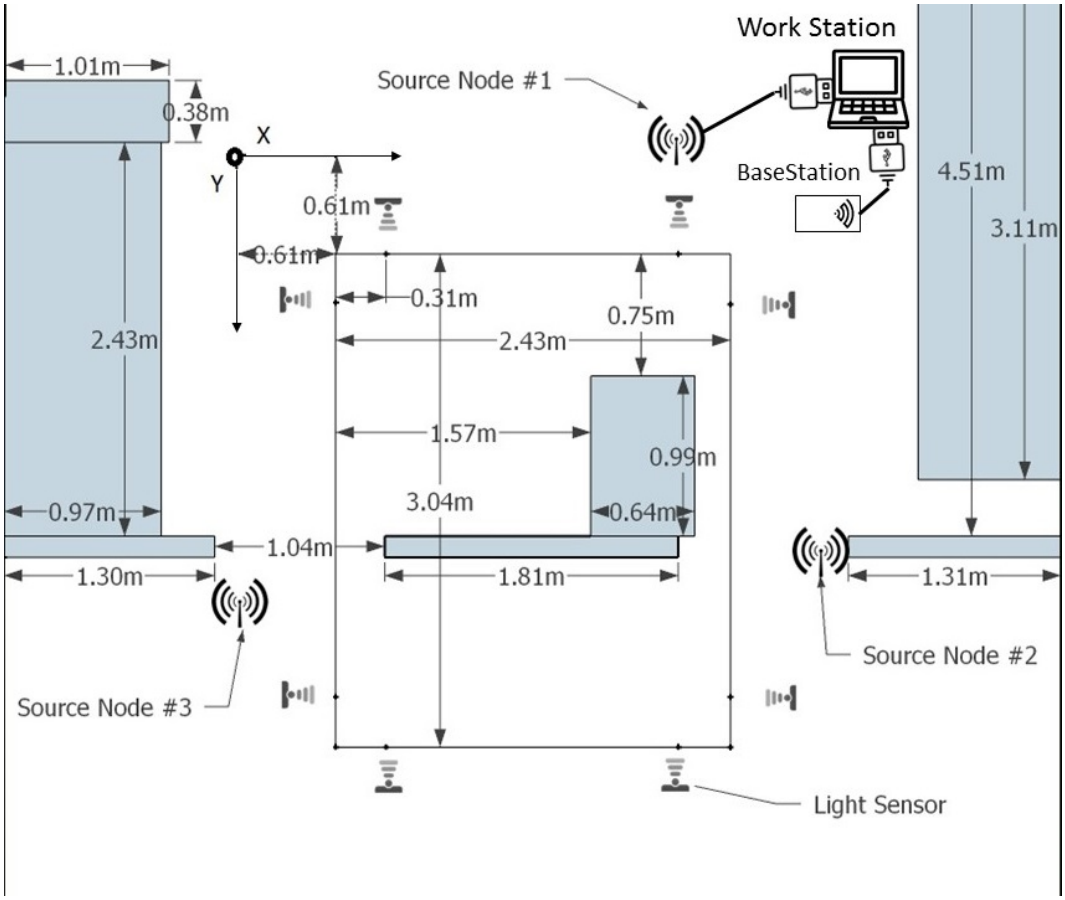

Figure 6.3: Detailed layout for NLOS conditions experiments in the ITB214 lab.

Table 6.3 shows the position of the source nodes for the NLOS experiments.

Table 6.3: Position of source nodes in NLOS experiments. All values are given in cm.

ID	x	y
1	275	0
2	364.5	244
3	0	274

Figure 6.4 depicts the output of one loop (experiment number 6) of the NLOS experiments for the four different positioning approaches. Table 6.4 shows the average real-time (RT) and off-line (OL) error for each of the positioning approaches in NLOS conditions.

Table 6.4: Average error for NLOS experiments. All results shown in cm.

ID	Points	Direction	LRM OL	LRM RT	LLS OL	LLS RT	IMU-UWB OL	IMU-UWB RT	Duration(s)
1	459	CCW	57.71	109.10	33.53	60.82	27.18	46.50	71.63
2	462	CCW	59.66	98.91	33.98	49.88	31.54	61.69	71.47
3	459	CCW	57.49	126.67	33.02	80.1	28.15	73.18	70.81
4	465	CW	60.54	105.81	36.71	53.69	29.07	43.26	68.08
5	477	CW	59.14	101.93	36.35	53.28	30.71	44.78	69.54
6	467	CW	61.84	101.77	35.96	53.07	29.4	42.36	68.70
Avg	2789	-	59.40	107.37	34.93	58.47	29.34	51.96	-

According to the RT evaluation, IMU assistance increases the average accuracy of UWB positioning by 60.24% and 18.78%, respectively, compared to the LRM and LLS algorithms. It is notable that the LLS algorithm requires a known start position $\mathbf{x_0}$ which here is always (0.0, 0.0). Also, LLS is an iterative algorithm; using 0.01 cm as the convergence threshold for $\mathbf{\Delta x}$ in equation (5.3) in our experiments, it takes 8 iterations on average for the LLS algorithm to conclude, with a maximum of 31 iterations. In our experiments when the LLS algorithm was implemented on the EVB1000 with ARM Cortex-M3 processor and an UWB measurement rate of 7 Hz, there was no slowdown detected by the iterative LLS algorithm. In order to have a known initial position, every time that the TN restarts, it must be turned on at a specific position or the initial position must be transmitted to the TN manually. This might be a drawback if the initial position $\mathbf{x}_0$ cannot be estimated from known conditions.

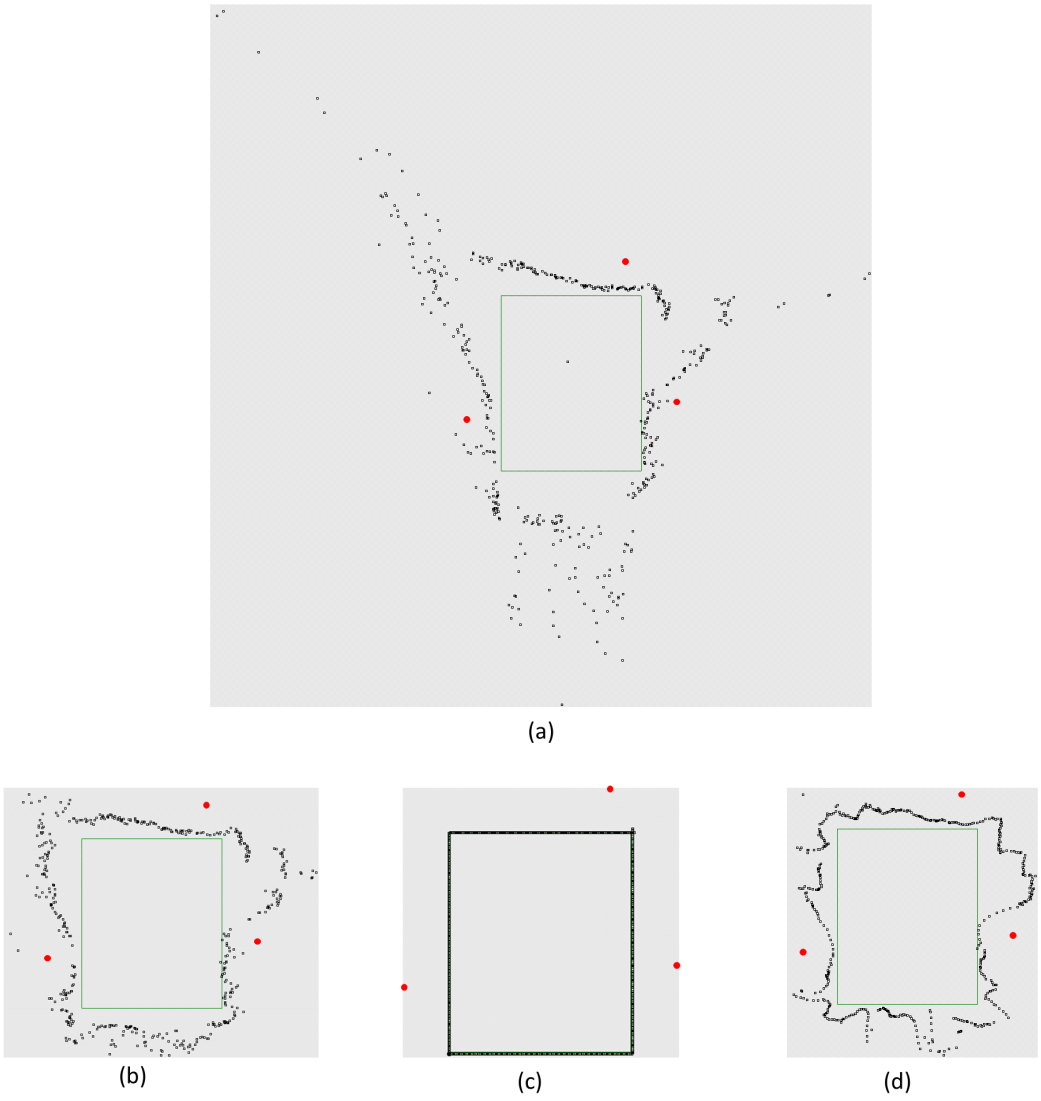

Figure 6.4: Positioning algorithms output in NLOS conditions. (a) LRM, (b) LLS, (c) ground-truth position and (d) IMU-UWB. Red dots indicate the position of the source nodes.

Chapter 7

Summary and Conclusion

To measure the accuracy of UWB indoor positioning we implemented two different statistical positioning algorithms suggested in the literature on Decawave's EVB1000 board. In addition, we used a Kalman filter to integrate an onboard IMU system with the UWB system in an attempt to increase the positioning accuracy of the UWB-only system. We performed experiments both in LOS and NLOS conditions. Experimental validation mounted the moving target node on a 3pi robot that followed two tracks at a velocity of approximately 0.16 m/s. The LOS track was 6.7 m in length, and the NLOS track was 9.7 m in length.

We proposed a new evaluation approach which compares the estimated position of the target node with an independent ground truth position estimation. A light sensor was used to detect the existence of the target node at specific positions with a precise time stamp that allowed accurate calculation of the target node velocity and ground truth position. Results of our experiments indicate the average accuracy of UWB positioning system to be 10.97 cm in LOS conditions and 58.47 cm in NLOS conditions using a linearized least squares (LLS) algorithm and real-time evaluation discussed in chapters 3 and 5, respectively. The IEEE.802.15.4-2011 ranging protocol permits packets to establish a communication channel simultaneously while

real-time range measurements are being observed. This feature is provided by the Decawave company's implementation of the IEEE.802.15.4-2011 standard. We used this communication channel to send the positioning algorithm results as well as raw measurements to a PC.

To evaluate the capability of IMU systems as an aid for UWB positioning systems, we used a six degrees of freedom IMU, composed of a gyroscope and an accelerometer. Extended Kalman filtering was used to fuse information retrieved from the UWB nodes and the IMU board. Our experimental results show that the IMU system is not able to increase UWB positioning accuracy in LOS conditions. In the case of NLOS conditions, however, the IMU system increased average positioning accuracy by 18.78% (from an average accuracy of 58.47 cm down to 51.96 cm) compared to a linearized least squares algorithm using only the ranges provided by the UWB system for computing real-time positions.

Future work could consider evaluation of the discussed positioning approaches with a target node moving much faster. For example, average human walking speed is 1.4 m/s, which is about 10 times faster than the speed of the 3pi robot used in our experiments. How accurate would UWB positioning be if used for even faster moving objects such as quadcopters that can move at 14 m/s or faster?

In addition, it needs to be confirmed if the proposed IMU-UWB approach works for multiple target nodes. The positioning approaches presented here are all capable of three dimensional positioning. What accuracy is achieved for the third dimension when using UWB real-time positioning?

References

[1] *Decawave products overview*, `http://www.decawave.com/products/overview,2015-07-16`.

[2] *Encyclopedia: Gyroscope*, `http://www.encyclopedia.com/topic/gyroscope.aspx,2015-06-14`.

[3] *Sparkfun: Adxl335*, `https://www.sparkfun.com/products/9265,2015-06-14`.

[4] *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 Std 802.15.4a-2007 (Amendment to IEEE Std 802.15.4-2006) (2007), 1–203.

[5] S. Al-Jazzar and J. J. Caffery, *ML and bayesian TOA location estimators for NLOS environments*, IEEE Veh. Technol. Conf. (VTC) **2** (2002).

[6] Klaithem Al Nuaimi and Hesham Kamel, *A survey of indoor positioning systems and algorithms*, Innovations in Information Technology (IIT), 2011 International Conference on, IEEE, 2011, pp. 185–190.

[7] Christian Ascher, Lukasz Zwirello, Thomas Zwick, and Gert Trommer, *Integrity monitoring for UWB/INS tightly coupled pedestrian indoor scenarios*, Second International Conference on Indoor Positioning and Indoor Navigation (IPIN), IEEE, 2011, pp. 1–6.

[8] Joan Borras, Paul Hatrack, and Narayan B Mandayam, *Decision theoretic framework for NLOS identification*, Vehicular Technology Conference, 1998. VTC 98. 48th IEEE, vol. 2, IEEE, 1998, pp. 1583–1587.

[9] José Borràs Sillero, *Sensor fusion methods for indoor navigation using UWB radio aided INS/DR*, Master's thesis, KTH Electrical Engineering, Stockholm, Sweden, 2012.

[10] James J Caffery, *Wireless location in CDMA cellular radio systems*, Springer Science & Business Media, 2000.

[11] Pi-Chun Chen, *A non-line-of-sight error mitigation algorithm in location estimation*, Wireless Communications and Networking Conference, 1999. WCNC. 1999 IEEE, IEEE, 1999, pp. 316–320.

[12] Ka Wai Cheung, Hing-Cheung So, W-K Ma, and Yiu-Tong Chan, *Least squares algorithms for time-of-arrival-based mobile location*, IEEE Transactions on Signal Processing **52** (2004), no. 4, 1121–1130.

[13] Federal Communication Commission, *First report and order 02-48, part 15 (radio frequency devices), subpart F (ultra-wideband operation)*, Available at `http://www.ecfr.gov/cgi-bin/retrieveECFR?gp=&SID=6c43fbe34e2b680697d89b883e87ebf0&n=47y1.0.1.1.16.6&r=SUBPART&ty=HTML` (last visited : 19.11.2013).

[14] Juan Antonio Corrales, FA Candelas, and Fernando Torres, *Hybrid tracking of human operators using IMU/UWB data fusion by a kalman filter*, Human-Robot Interaction (HRI), 2008 3rd ACM/IEEE International Conference on, IEEE, 2008, pp. 193–200.

[15] Alessio De Angelis, John Nilsson, Isaac Skog, Peter Händel, and Paolo Carbone, *Indoor positioning by ultrawide band radio aided inertial navigation*, Metrology and Measurement Systems **17** (2010), no. 3, 447–460.

[16] Qigao Fan, Yaheng Wu, Jing Hui, Lei Wu, Zhenzhong Yu, and Lijuan Zhou, *Integrated navigation fusion strategy of INS/UWB for indoor carrier attitude angle and position synchronous tracking*, The Scientific World Journal **2014** (2014), Article ID 215303, 13 pages, doi:10.1155/2014/215303.

[17] Yanying Gu, Anthony Lo, and Ignas Niemegeers, *A survey of indoor positioning systems for wireless personal networks*, IEEE Communications Surveys & Tutorials **11** (2009), no. 1, 13–32.

[18] Ismail Guvenc and Chia-Chin Chong, *A survey on TOA based wireless localization and NLOS mitigation techniques*, IEEE Communications Surveys & Tutorials **11** (2009), no. 3, 107–124.

[19] Ismail Guvenc, Chia-Chin Chong, and Fujio Watanabe, *NLOS identification and mitigation for UWB localization systems*, Wireless Communications and Networking Conference, 2007. WCNC 2007., IEEE, 2007, pp. 1571–1576.

[20] Ismail Guvenc, Sinan Gezici, Fujio Watanabe, and Hiroshi Inamura, *Enhancements to linear least squares localization through reference selection and ML estimation*, Wireless Communications and Networking Conference, 2008. WCNC 2008., IEEE, 2008, pp. 284–289.

[21] Mike Hazas and Andy Hopper, *Broadband ultrasonic location systems for improved indoor positioning*, IEEE Transactions on Mobile Computing **5** (2006), no. 5, 536–547.

[22] Mohammad Heidari, Ferit Ozan Akgül, and Kaveh Pahlavan, *Identification of the absence of direct path in indoor localization systems*, IEEE 18th International Symposium on Personal, Indoor and Mobile Radio Communications, PIMRC 2007, IEEE, 2007, pp. 1–6.

[23] J.D. Hol, F. Dijkstra, H. Luinge, and T.B. Schon, *Tightly coupled uwb/imu pose estimation*, IEEE International Conference on Ultra-Wideband. ICUWB 2009., Sept 2009, pp. 688–692.

[24] Lei Jiao, Jianping Xing, Xuan Zhang, Chaoli Zhao, and Jun Zhang, *LCC-Rwgh: A NLOS error mitigation algorithm for localization in wireless sensor network*, IEEE International Conference on Control and Automation. ICCA 2007, 2007, pp. 1354–1359.

[25] Yung-Hoon Jo, Joon-Yong Lee, Dong-Heon Lee, and Shin-Hoo Kang, *Accuracy enhancement for UWB indoor positioning using ray tracing*, The Journal of Korean Institute of Communications and Information Sciences **31** (2006), no. 10C, 921–926.

[26] D. Jourdan, Jr. Deyst, J.J., M.Z. Win, and N. Roy, *Monte carlo localization in dense multipath environments using uwb ranging*, IEEE International Conference on Ultra-Wideband. ICU 2005 (2005), 314–319.

[27] Jasurbek Khodjaev, Yongwan Park, and Aamir Saeed Malik, *Survey of NLOS identification and error mitigation problems in UWB-based positioning algorithms for dense environments*, Annals of telecommunications-annales des télécommunications **65** (2010), no. 5-6, 301–311.

[28] Wuk Kim, Jang-Gyu Lee, and Gyu-In Jee, *The interior-point method for an optimal treatment of bias in trilateration location*, IEEE Transactions on Vehicular Technology **55** (2006), no. 4, 1291–1301.

[29] Hakan Koyuncu and Shuang Hua Yang, *A survey of indoor positioning and object locating systems*, IJCSNS International Journal of Computer Science and Network Security **10** (2010), no. 5, 121–128.

[30] Fan Li, Wenwu Xie, Jun Wang, and Shouyin Liu, *A new two-step ranging algorithm in NLOS environment for UWB systems*, Asia-Pacific Conference on Communications. APCC'06., IEEE, 2006, pp. 1–5.

[31] Hui Liu, Houshang Darabi, Pat Banerjee, and Jing Liu, *Survey of wireless indoor positioning techniques and systems*, IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews **37** (2007), no. 6, 1067–1080.

[32] Decawave Ltd, *Decaranging (ARM) source code guide, version 1.5, 2013.*

[33] _____, *DW1000 User Manual, version 1.0, 2013.*

[34] Miklós Maróti, Branislav Kusy, Gyula Simon, and Ákos Lédeczi, *The flooding time synchronization protocol*, Proceedings of the 2nd international conference on Embedded networked sensor systems, ACM, 2004, pp. 39–49.

[35] Lorenzo Mucchi and Patrizio Marcocci, *A new UWB indoor channel identification method*, 2nd International Conference on Cognitive Radio Oriented Wireless Networks and Communications. CrownCom 2007., IEEE, 2007, pp. 58–62.

[36] Sylvain Pittet, Valerie Renaudin, Bertrand Merminod, and Michel Kasser, *UWB and MEMS based indoor navigation*, Journal of Navigation **61** (2008), no. 03, 369–384.

[37] Yihong Qi, Hisashi Kobayashi, and H. Suda, *Analysis of wireless geolocation in a non-line-of-sight environment*, IEEE Transactions on Wireless Communications, **5** (2006), no. 3, 672–681.

[38] _____, *Analysis of wireless geolocation in a non-line-of-sight environment*, IEEE Transactions on Wireless Communications **5** (2006), no. 3, 672–681.

[39] Rashid Saeed, Sabira Khatun, Borhanuddin Mohd Ali, Mohd Khazani, et al., *Ultra-wideband (UWB) geolocation in NLOS multipath fading environments*, Proc. of the joint 7th IEEE Malaysia International Conference on Communication., 2005 13th IEEE International Conference on Networks, vol. 2, IEEE, 2005, pp. 1068–1073.

[40] Z. Sahinoglu and S. Gezici, *Ranging in the IEEE 802.15.4a standard*, Wireless and Microwave Technology Conference. WAMICON '06. IEEE, 2006, pp. 1–5.

[41] Alberto Savioli, Emanuele Goldoni, Pietro Savazzi, and Paolo Gamba, *Low complexity indoor localization in wireless sensor networks by UWB and inertial data fusion*, arXiv preprint arXiv:1305.1657 (2013).

[42] Jens Schroeder, Stefan Galler, Kyandoghere Kyamakya, and Thomas Kaiser, *Three-dimensional indoor localization in non line of sight UWB channels*, IEEE International Conference on Ultra-Wideband, 2007. ICUWB, IEEE, 2007, pp. 89–93.

[43] Stuart Schwartz and Joseph Bobier, *Going beyond interruptible usage*, Available at `http://www.tmcnet.com/tmcnet/columnists/articles/xg-spectrum-sharing-schwartz-bobier.htm`.

[44] Sebastian Sczyslo, Jens Schroeder, Stefan Galler, and Thomas Kaiser, *Hybrid localization using UWB and inertial sensors*, IEEE International Conference on Ultra-Wideband, vol. 3, 2008, pp. 89–92.

[45] Philips Semiconductors, *The I2C-bus specification*, Philips Semiconductors **9397** (2000), no. 750, 00954.

[46] Sailes K Sengijpta, *Fundamentals of statistical signal processing: Estimation theory*, Technometrics **37** (1995), no. 4, 465–466.

[47] C. E. Shannon, *Communication in the presence of noise*, Proc. Institute of Radio Engineers **1** (1949), no. 37, 10–21.

[48] Hiroyuki Shimizu, Hironari Masui, Masanori Ishii, Kozo Sakawa, and Takehiko Kobayashi, *LOS and NLOS path-loss and delay characteristics at 3.35 GHz in a residential environment*, IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences **83** (2000), no. 7, 1356–1364.

[49] Sebastian Tilch and Rainer Mautz, *Current investigations at the ETH zurich in optical indoor positioning*, 7th Workshop on Positioning Navigation and Communication (WPNC), IEEE, 2010, pp. 174–178.

[50] S Venkatesh and RM Buehrer, *Non-line-of-sight identification in ultra-wideband systems based on received signal statistics*, IET Microwaves, Antennas & Propagation **1** (2007), no. 6, 1120–1130.

[51] Swaroop Venkatesh and R Michael Buehrer, *NLOS mitigation using linear programming in ultrawideband location-aware networks*, IEEE Transactions on Vehicular Technology **56** (2007), no. 5, 3182–3198.

[52] Sichun Wang, Robert Inkol, and Brad R. Jackson, *Relationship between the maximum likelihood emitter location estimators based on received signal strength (RSS) and received signal strength difference (rssd)*, 26th Biennial Symposium on Communications (QBSC), Kingston, ON, Canada, 2012, pp. 503–513.

[53] Chin-Der Wann and Chih-Sheng Hsueh, *NLOS mitigation with biased kalman filters for range estimation in UWB systems*, IEEE Region 10 Conference (TENCON), IEEE, 2007, pp. 1–4.

[54] Roy Want, Andy Hopper, Veronica Falcao, and Jonathan Gibbons, *The active badge location system*, ACM Transactions on Information Systems (TOIS) **10** (1992), no. 1, 91–102.

[55] G. Welch and G. Bishop, *An introduction to the kalman filter*, University of North Carolina at Chapel Hill, Chapel Hill, NC **7** (1995), no. 1.

[56] Greg Welch and Gary Bishop, *An introduction to the kalman filter*, Proceedings of the SIGGRAPH Course, Los Angeles (2001).

[57] Shaohua Wu, Yongkui Ma, Qinyu Zhang, and Naitong Zhang, *NLOS error mitigation for UWB ranging in dense multipath environments*, Wireless Communications and Networking Conference, WCNC 2007, IEEE, 2007, pp. 1565–1570.

[58] Mohammadreza Yavari and Bradford G. Nickerson, *Ultra wideband wireless positioning systems*, Tech. report, 2014, UNB Faculty of Computer Science Technical Report TR14-230.

[59] Kegen Yu and Y Jay Guo, *NLOS error mitigation for mobile location estimation in wireless networks*, IEEE 65th Vehicular Technology Conference, Dublin, IEEE, 2007, pp. 1071–1075.

[60] Ismail Gvenc Zafer Sahinoglu, Sinan Gezici, *Ultra-wideband positioning systems: Theoretical limits, ranging algorithms, and protocols*, Cambridge University Press, October 6, 2008.

[61] Miao Zhang, Arun Vydhyanathan, Alexander Young, and Renk Luinge, *Robust height tracking by proper accounting of nonlinearities in an integrated UWB/MEMS-based-IMU/baro system*, Position Location and Navigation Symposium (PLANS), IEEE/ION, IEEE, 2012, pp. 414–421.

[62] Lingchen Zhou, *Position estimation of nodes moving in a wireless sensor network*, Tech. report, 2014, UNB Faculty of Computer Science Technical Report TR14-229.

[63] Lukasz Zwirello, Christian Ascher, Gert F Trommer, and Thomas Zwick, *Study on UWB/INS integration techniques*, 8th Workshop on Positioning, Navigation and Communication (WPNC), IEEE, 2011, pp. 13–17.

[64] Lukasz Zwirello, Xuyang Li, Thomas Zwick, Christian Ascher, Sebastian Werling, and Gert F Trommer, *Sensor data fusion in UWB-supported inertial navigation systems for indoor navigation*, IEEE International Conference on Robotics and Automation (ICRA), IEEE, 2013, pp. 3154–3159.

Appendix A

Real-time evaluation source code

This algorithm computes ground truth position of the target node at an arbitrary moment t_i using equations (5.6) and (5.7). Function "trueParams" computes velocity of the target node on each of the edges of the route v and time-stamps for the corners of the route t_{c_n}. Function "getEdge" determines on which edge was the TN at time t_i. Finally, function "getGroundTruth" computes the ground position of the TN (x'_{t_i}, y'_{t_i}) at time t_i.

```
1
2  private void trueParams()
3  {
4  vertexv[0] = Math.Abs((lightPoint[1][0] - lightPoint[0][0])) /
       (lightTS[1] - lightTS[0]);
5  vertexv[1] = Math.Abs((lightPoint[3][1] - lightPoint[2][1])) /
       (lightTS[3] - lightTS[2]);
6  vertexv[2] = Math.Abs((lightPoint[5][0] - lightPoint[4][0])) /
       (lightTS[5] - lightTS[4]);
7  vertexv[3] = Math.Abs((lightPoint[7][1] - lightPoint[6][1])) /
       (lightTS[7] - lightTS[6]);
8
9  vertexts[0] = lightTS[0] - (30.5 / vertexv[0]);
10 vertexts[1] = (lightTS[1] + lightTS[2]) / 2;
11 vertexts[2] = (lightTS[3] + lightTS[4]) / 2;
12 vertexts[3] = (lightTS[5] + lightTS[6]) / 2;
13 vertexts[4] = lightTS[7] + (30.5 / vertexv[3]);
14
15 }
16
17 private int getEdge(double posts)
18 {
19 int e = 0;
20
21 if ((vertexts[0] < posts && posts < vertexts[1]) || (vertexts[0] > posts
       && posts > vertexts[1]))
22 e = 1;
```

```
if ((vertexts[1] < posts && posts < vertexts[2]) || (vertexts[1] > posts
    && posts > vertexts[2]))
e = 2;
if ((vertexts[2] < posts && posts < vertexts[3]) || (vertexts[2] > posts
    && posts > vertexts[3]))
e = 3;
if ((vertexts[3] < posts && posts < vertexts[4]) || (vertexts[3] > posts
    && posts > vertexts[4]))
e = 4;

if (e == 0)
throw new Exception("True position not found");

return e;
}
private double[] getGroundTruth(double posts)
{
double[] groundTruth = new double[2];
int e = getEdge(posts);
switch (e)
{
case 1:
groundTruth[1] = lightPoint[0][1];
groundTruth[0] = lightPoint[0][0] + (posts - lightTS[0]) * vertexv[0];
break;
case 2:
groundTruth[0] = lightPoint[2][0];
groundTruth[1] = lightPoint[2][1] + (posts - lightTS[2]) * vertexv[1];
break;
case 3:
groundTruth[1] = lightPoint[4][1];
groundTruth[0] = lightPoint[4][0] - (posts - lightTS[4]) * vertexv[2];
break;
case 4:
groundTruth[0] = lightPoint[6][0];
groundTruth[1] = lightPoint[6][1] - (posts - lightTS[6]) * vertexv[3];
break;
default:
throw new Exception("No true position!");
}
return groundTruth;
}
```

Appendix B

Off-line evaluation source code

This algorithm computes the minimum distance between the position estimation $\hat{\mathbf{x}}$ and the route using equations (5.1) and (5.2).

```
1   double getOLerror(int[,] pos)
2   {
3   double err = 0;
4   double d;
5   double[] d4 = new double[4];
6   double left = rout_start.X;
7   double top = rout_start.Y;
8   double right = left + rout_width;
9   double bottom = top + rout_heigth;
10  for (int i = 0; i < posNum; i++)
11  {
12  if ((pos[i, 0] >= left & pos[i, 0] <= right) || (pos[i, 1] >= top &
        pos[i, 1] <= bottom))
13  {
14  d4[0] = Math.Abs(top - pos[i, 1]);
15  d4[1] = Math.Abs(bottom - pos[i, 1]);
16  d4[2] = Math.Abs(pos[i, 0] - left);
17  d4[3] = Math.Abs(pos[i, 0] - right);
18  }
19  else
20  {
21  d4[0] = (int)Math.Sqrt(Math.Pow(pos[i, 0] - left, 2) + Math.Pow(pos[i,
        1] - top, 2));
22  d4[1] = (int)Math.Sqrt(Math.Pow(pos[i, 0] - left, 2) + Math.Pow(pos[i,
        1] - bottom, 2));
23  d4[2] = (int)Math.Sqrt(Math.Pow(pos[i, 0] - right, 2) +
        Math.Pow(pos[i, 1] - top, 2));
24  d4[3] = (int)Math.Sqrt(Math.Pow(pos[i, 0] - right, 2) +
        Math.Pow(pos[i, 1] - bottom, 2));
25  }
26
```

```
27      d = d4.Min();
28      err = err + d;
29      }
30      err = err / posNum;
31      return err;
32      }
```

Vita

Candidate's full name:
Mohammadreza Yavari

University attended:
Master of Computer Science (2013-2015): University of New Brunswick Fredericton, Canada
Electrical and Computer Engineering (2008-2013): Isfahan University of Technology Isfahan, Iran

Publications:
Zhang, Weiqi; Yavari, Mohammadreza and Nickerson, Bradford G. Performance of Mobile Wireless Sensor Network Communication with 6LoWPAN, Proc. of the 28th Canadian Conference on Electrical and Computer Engineering (CCECE 2015), Halifax, N.S., May 3-6, 2015, pp.1048-1053.

Conference Poster Presentations:
Mohammad reza Yavari and Bradford G. Nickerson. IEEE 802.15.4-2011 UWB Real Time Positioning Aided by INS. University of New Brunswick Computer Science Research Expo 2014, Fredericton, Canada

Mohammad reza Yavari and Bradford G. Nickerson. Ultra-Wide Band (UWB) real time positioning aided by IMU. University of New Brunswick Computer Science Research Expo 2015, Fredericton, Canada

Other documents:
Mohammadreza Yavari and Bradford G. Nickerson, Ultra Wideband Wireless Positioning Systems, Technical Report TR14-230, University of New Brunswick, Faculty of Computer Science, March 2014.