UNIVERSITY OF SOUTHAMPTON

A Real-Time Target Tracking System for Wireless Embedded Nodes using Ranging Measurements

by

Evangelos Basilios Mazomenos

A thesis submitted in partial fulfillment for the
degree of Doctor of Philosophy

in the
Faculty of Physical and Applied Sciences
Department of Electronics and Computer Science

February 2012

UNIVERSITY OF SOUTHAMPTON

ABSTRACT

FACULTY OF PHYSICAL AND APPLIED SCIENCES
DEPARTMENT OF ELECTRONICS AND COMPUTER SCIENCE

Doctor of Philosophy

by Evangelos Basilios Mazomenos

The area of wireless embedded nodes has attracted significant research interest, primarily with respect to the utilisation of this technology in a number of applications domains. Under this context, the main topic of this thesis pertains to the design of a framework for real-time, range-only target tracking utilizing low power wireless embedded nodes. The proposed tracking system is designed to operate solely on range measurements which are obtained without the need for additional hardware incorporated on the embedded nodes. The core objective of this research was to present a target tracking system that can be applied to real-world applications, incorporating support for effectively tracking manoeuvring targets facilitated by the ability to obtain accurate range readings from low-power embedded nodes and finally the ability to achieve real-time system operation.

The contribution of the work presented in this thesis is threefold. The tracking problem is theoretically formulated as a dynamical system with the objective being, the real-time estimation of the target's kinematic variables based on range observations. To address the need for effective tracking of manoeuvring targets an adaptive multiple-model approach was developed. The resulting system is non-linear, due to the non-linearity between the range observations and the kinematic variables. To solve this system, a novel adaptive multiple-model Particle Filter tracking algorithm is proposed. Secondly, to achieve accurate enough ranging between embedded nodes a Time-of-Flight ranging scheme is adopted as part of the proposed tracking system. The final contribution of this work pertains to the real-time operation of the tracking system.

The tracking algorithms were evaluated on a simulation environment under realistic experimental conditions. The ranging method was implemented on embedded nodes and tested in terms of accuracy in various environments. Ultimately, the entire system was implemented on hardware and tested in outdoor experiments. In the experiments carried out one mobile wireless node was used as the target and a set of anchor nodes attempted to infer the target's kinematic variables. A total of 25 experiments are presented in this thesis. An average accuracy of approximately 2.6m for position and 1.9m/s for velocity was attained in a 15m x 15m square area. Such performance, which is confirmed from the simulation results reveal the potential of the proposed range-only system in application domains where real-time tracking of mobile targets is a demand.

Contents

List of Figures

List of Tables

List of Algorithms

Nomenclature

AoA	Angle of Arrival
ASIC	Application Specific Integrated Circuit
cc	Clock Cycles
CV	Constant Velocity
CRLB	Cramer-Rao Lower Bound
DSP	Digital Signal Processor
DAC	Digital to Analog Converter
DoA	Direction of Arrival
FFT	Fast Fourier Transform
FPGA	Field Programmable Gate Array
HMM	Hidden Markov Model
i.i.d	independent and identically distributed
KF	Kalman Filter(s)
LoS	Line of Sight
MCMC	Markov Chain Monte Carlo
MAC	Medium Access Control
MEMS	Micro Electro-Mechanical Systems
MCU	Microcontroller
MMSE	Minimum Mean Square Error
MSK	Minimum Shift Keying
PF	Particle Filter(s)
PIR	Passive Infrared
pdf	Probability Density Function
QoS	Quality of Service
RF	Radio Frequency
RFID	Radio Frequency Identification
ROT-MMPF	Range-Only Tracking - Multiple Model Particle Filter Algorithm
ROT-PF	Range-Only Tracking - Particle Filter Algorithm
RSSI	Received Signal Strength Indication
RMSE	Root Mean Square Error
SNR	Signal-to-Noise Ratio
c	speed of light = 299792458 m/s

std	standard deviation
ToA	Time of Arrival
ToF	Time of Flight
UWB	Ultra Wide-Band
UART	Universal Asyncronous Receiver/Transmitter
WSN	Wireless Sensor Networks

Declaration of Authorship

I, *Evangelos Basilios Mazomenos*, declare that this thesis entitled

"A Real-Time Target Tracking System for Wireless Embedded Nodes using Ranging Measurements"

and the work presented in it are my own and has been generated by me as the result of my own original research. I confirm that:

- This work was done wholly or mainly while in candidature for a research degree at this University;
- Where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated;
- Where I have consulted the published work of others, this is always clearly attributed;
- Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work;
- I have acknowledged all main sources of help;
- Where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself;
- Either none of this work has been published before submission, or parts of this work have been published as listed on Page 7

Signed:

Date:

Acknowledgements

I wish to acknowledge the contribution of all the people who helped me directly or indirectly to complete this research work.

First and foremost, I am thankful to my supervisors Dr. Jeffrey S. Reeve and Professor Neil M. White for their continuous encouragement, academic supervision and guidance throughout this research project. Moreover, I wish to thank them for providing me with the opportunity to pursue my PhD studies in the School of Electronics and Computer Science here at the University of Southampton. I also thank Dr. Nick Harris for his valuable feedback during my MPhil transfer and Dr. Koushik Maharatna for many conversations on various research subjects.

I would also like to thank Dirk De Jager for his friendship and collaboration on several projects during these years. His assistance in the execution of the experiments reported in this thesis was unparallel and to that I owe a great deal of gratitude to him.

Many thanks also to my fellow PhD students and researchers in the Electronic Systems and Devices group for sharing ideas and discussing ideas and topics both with regard to this research as well as in general academic directions. Some of them moved on to another position and some are still here in the same lab. To them I own an amazing research atmosphere that was inspiring on an every-day basis. Special thanks must be addressed to Amit Acharrya, Alex Weddell, Pedro Barbosa, Geoff Merrett, Saqib Khursheed, Mustafa Imran Ali, Eduardo Mangieri, Taihai Chen and Sheng Yang

I would also like to thank the Institute of Engineering and Technology (IET) for their financial support through the Leslie H. Paddle Scholarship which I had the distinct honor of receiving in 2009.

A significant part of the always necessary support to complete this research work stemmed from the people close to me. I wish to thank my girlfriend Katerina for her love and support and my dear friends here in Southampton and in Greece for their friendship and encouragement during my PhD years.

Finally I wish to thank my parents Basilios and Anastasia for their love, support and inspiration especially in these years of my PhD research. Their stance during some very difficult moments, provided me with enormous determination and gave me two great role models to look up in my life. The work presented here is dedicated to them

To my parents ...

Chapter 1

Introduction

This dissertation investigates the problem of tracking mobile targets with the use of cooperating embedded nodes. The goal is to demonstrate a full-scale tracking system which is capable of estimating the position and velocity of a moving target over time with acceptable accuracy, a few meters. To achieve this, we employ a small number of cooperating nodes wirelessly connected one to another forming a Wireless Sensor Network (WSN).

1.1 Wireless Sensor Networks

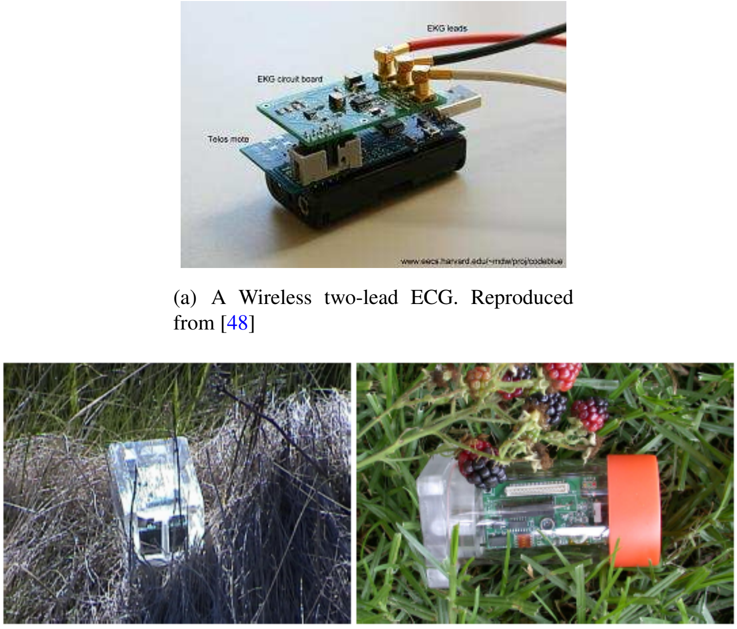
The term *" Wireless Sensor Networks"* has been extensively used in the literature over the past decade. Within the purposes of this thesis, a Wireless Sensor Network is considered as:

" A network constituted of low-power, wireless communicating, embedded nodes equipped with various sensing modalities that operate autonomously but in coordination to accomplish a specific task"

Important advancements in microelectronics, wireless communications and sensor technologies enabled the development of miniature sized platforms equipped with sensing modalities and wireless communication. These embedded devices have the ability to acquire information from the surrounding environment, process this information locally and communicate the data in a wireless manner [66,67]. This new type of wireless distributing sensing provides a novel means of interacting with the physical world and also a prominent background for a wide range of applications [43–45].

During the previous decade WSNs experienced rapid development and received a significant amount of research interest. The commercial availability of a number of hardware platforms like the well-known Mica family of motes provided a flexible, rich in sensors, testbed which in combination with the TinyOS programming framework, facilitated researchers in investigating the potential utilisation of WSN for a vast number of application scenarios [49,95].

Examples of possible application domains include, environmental monitoring, smart structures, habitat monitoring, military defense applications, surveillance and motion detection, mobile robotics, health care and medical applications, agriculture and asset management [4,9]. Environmental monitoring involves the deployment of a sensor network intended to monitor certain physical parameters over a geographical region [115, 116]. As an example, the “FIREBUG” project exploits WSN in order to collect realtime data, like humidity and barometric pressure from wildfires, for safety considerations and predictive analysis of an evolving fire. Data collected from the motes are routed to a central base station which further communicates the data to a database server enabling remote access [35]. The “Envisence GLACSWEB” project developed a prototype mote platform to monitor temperature, pressure, stress and sub glacial movement in glaciers,thus contributing to glaciology and in research on sensors networks deployments under extreme environments [88]. The“CODEBLUE” project, investigates the incorporation of various medical instruments like ECG, EMG and pulse oximeter sensors to a mote platform, for patient monitoring [48]. The majority of WSN applications encounter challenges related to ad-hoc deployment, dynamic environmental conditions, robust and unattended operation which requires the network to be capable of self-configuring and efficiently performing in terms of power consumption.



(a) A Wireless two-lead ECG. Reproduced from [48]

(b) Nodes for environmental monitoring. Adapted from [116]

FIGURE 1.1: Examples of WSN applications

1.2 Motivation

Locationing and tracking objects of interest is considered to be a pivotal functionality for a number of application domains. Under this context, the topic of the research presented in the thesis lies in the application area of target tracking with low-power embedded nodes. The basic concept for target tracking with WSNs is to deploy a number of cooperating embedded nodes to monitor a specific region of interest. Whenever a target is present, these nodes interact with the target and collect necessary information to perform the tracking operation. Generally in tracking systems, the target's dynamics are inferred by processing specific information associated with the target's kinematic variables (position, velocity, direction of movement). For example, from various sensor readings (e.g. acoustic energy) the relative distance between the source (target) and a number of infrastructure nodes, positioned in known locations can be derived. The collected data is then fed into the "tracking algorithm" which produces an estimation of the target's desired variables.

Previous decades witnessed an increasing demand in the use of the Global Positioning System (GPS) for numerous application domains. Subsequently, the GPS in nowadays considered as the golden standard technology for navigation and tracking purposes. Without doubt, there are clear and unequivocal advantages in the use of GPS. First of all, it provides worldwide coverage, since it is a satellite-based system. GPS receivers are small and relatively cost-effective. Finally the utilisation of the GPS service is free-of-charge. On the other hand the GPS does have significant shortcomings, rendering it not suitable for a number of application scenarios. GPS accuracy depends heavily on the availability and geometry of the satellites, as well as the surrounding environment resulting in accuracies between a few meters (5-10m) up to tens of meters (80m). Due to the fact that GPS requires an unobstructed line of sight to four or more GPS satellites, it is known to perform poorly indoors. GPS accuracy can be enhanced and reach even submeter levels with the use of more sophisticated GPS devices that in general are extremely costly. In addition, typical GPS receivers, if working continuously can only last for a brief period of time (2-3h) without recharging. These documented shortcomings favored the development of alternative solutions for tracking and locationing systems.

WSNs are considered to be a technology able to provide innovative solutions for locationing and tracking applications. Unlike traditional approaches, WSNs offer the possibility of employing a large number of observers (WSN nodes), tasked with monitoring the same phenomena, an approach that enables decentralised sensing, distributed computing and collaborative signal processing [131, 132]. An abundant amount of information can be accumulated from the network with high spatial and temporal resolution. For locationing and tracking, this is of particular interest, facilitating the development of more robust, flexible and cost-effective tracking systems.

An important factor that distinguishes tracking systems is the type of measurements that the system employs to achieve tracking. The objective of this research is to exclusively utilize range estimates, to estimate the trajectory and the rest of the target's kinematic variables. As a dynamical system, tracking with Range-Only measurements is a hard problem, because of the lack of global system observability when a single observer is considered. This problem stems from the basic geometric principles which dictate that the distance from a single point in the 2D space is not sufficient to determine the coordinates of another single point in that space. In situation with a single observer to satisfy the observability demand the observer itself must be non-static and in addition move with a specified motion pattern in order for the dynamical system to be solvable [114]. The criterion of observability for dynamical systems was introduced by R.E. Kalman and in simple terms provides a measure of how well can the the system internal states be inferred using the observed outputs. It has been reported that in order to satisfy the observability criterion for range-only tracking, multiple static observers deployed in different positions must provide range observations [27]. WSNs are ideal for applications where the same type of information must be collected from multiple sources, positioned in different locations, thus providing the necessary spatial resolution and system observability for range-only tracking.

Ultimately, novel tracking systems based on WSNs can then be applied to a wide range of application scenarios, including but not limited to:

- Defense Systems: Accurate target tracking is required in military surveillance systems that monitor hostile environments and are tasked with providing countermeasures for incoming targets

- Wildlife Monitoring: The study of wild animals in their natural environment can be benefitted from tracking systems allowing researchers to be aware of the animals position as well as the route they follow in their everyday activities

- People or asset tracking: Tracking systems are also required for monitoring the position of personnel in hazardous environments. For example a tracking system can be deployed for tracking the position of firemen during firefighting. In industrial infrastructures tracking systems can be used to continuously track the position of valuable assets

- Unmanned vehicle navigation: Nowadays, unmanned vehicles (UVs) are being used extensively in exploration missions particularly at inaccessible terrains (e.g. space, underwater). Autonomous functionality is an important challenge in this area because it would allow the UV to operate without the need of human intervention. To achieve this, the device must be aware of its current position as well as plan its future trajectory with precision

It is expected that for the previously mentioned application scenarios, WSNs-based tracking systems offer several advantages over standard tracking systems. GPS is considered to be susceptible to jamming making it vulnerable in military applications. Wildlife monitoring and people/asset tracking may include tracking of under dense foliage or indoors, environments where GPS performs inadequately. For the aforementioned applications a tracking accuracy of approximately 10%-15% of the area size is believed to be adequate.

1.3 Research Objective and Contributions

This project's aim is to provide a framework that allows the deployment of a network of embedded nodes to monitor a specific area of interest and has the ability to continuously track the trajectory of a mobile target in real-time for as long as the target is present within the network's coverage area. The research work is motivated by the following scenario, illustrated in Figure 1.2.

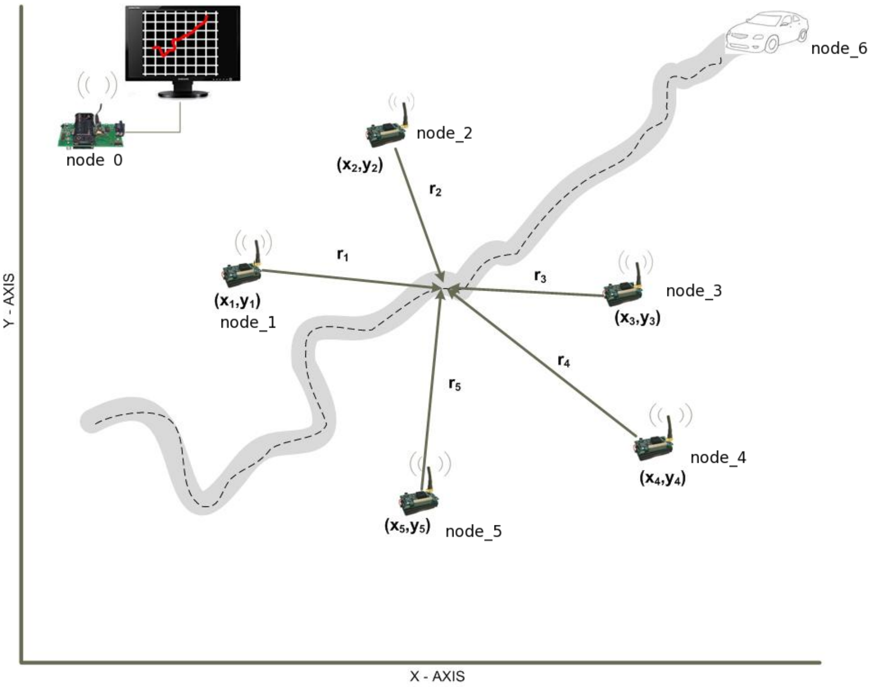


FIGURE 1.2: Tracking System Overview

The tracking system proposed in this research comprises of a number of anchor nodes (node_1 - node_5) positioned in known locations to obtain point-to-point range estimates, at a certain sampling period, between them and the target node (node_6). The range estimates are fused through wireless communication on a central node (node_0) equipped with the necessary energy supply and processing power to execute the tracking algorithms and estimate the target's kinematics variables. Finally the results produced by node_0 can be visualized at a front-end user interface and utilized according to the demands of the particular application scenario.

Under this context the primary objective of the research work presented in this thesis is the design and implementation of a complete tracking system based on wireless embedded nodes. Hardware implementation is deemed of particular importance as numerous approaches presented in the literature are based only on the theoretical formulation of the system and on simulation results. Against this, the work that was carried out in this dissertation focused on providing a real-world working system validated under realistic scenarios. This approach allowed us to consider every major and minor factor in the deployment of the tracking system which is not possible in simulation-based evaluations. The final system was implemented on

Commercially available Off-The-Self (COTS) hardware and it is the author's belief that it can be implemented in various similar platforms with minor modifications. The development of the proposed tracking system involves four major design decisions.

- The mathematical formulation of the range-only tracking problem
- The development of the tracking algorithm which solves the formulated tracking problem
- The technique to acquire the necessary ranging between the wireless nodes
- The incorporation of all the above into a full-scale system which meets certain quality demands.

The proposed tracking system is designed to operate solely on range measurements. This approach was inspired by the ability to obtain the range between a pair of embedded nodes without the need for additional hardware. In contrast, different approaches like bearing-based tracking which have been investigated for WSNs are considered to be less adaptable to low-power nodes since the acquisition of bearing information typically requires specific additional hardware (micro-radars, directional antennas) to be attached on the nodes. Unlike such approaches, to acquire the required point-to-point range between two embedded nodes a Time-Of-Flight (ToF) method is utilised. Specifically tailored for the needs of real-time operation the ToF scheme that is employed provides good accuracy (1m-3m) in the estimation of the distance between the nodes enhancing the accuracy of the tracking operation.

The mathematical formulation of the tracking problem was carried out in discrete time state-space domain. In contrast to the single-model method used in the majority of the reported systems, this research proposes the use of a multiple-model approach in representing the target's dynamics. Such modeling is required in order to provide adequate support in expressing the dynamics of manoeuvring targets. It is believed that for real-world applications, this modeling describes the motion pattern of targets more accurately.

The choice of a multiple-model approach coupled with range-only measurements resulted in a nonlinear dynamical system. Thus, alternatives to the traditional approaches for its solution were investigated. Particle Filters (PF) provide an ideal algorithmic background, within the framework of Bayesian Estimation, for nonlinear filtering. The significant advantage over traditional estimation algorithms like Kalman Filters lies on the ability to efficiently solve non-linear filtering problems. Although, PF are known to be resource intensive algorithms, the approach this research takes, by employing multiple anchors to provide range estimates, lowers the amount of particles that the algorithms needs to produce compared to other systems proposed in the literature.

The tracking algorithms and the ToF ranging method were implemented on T.I EZ430-RF2500 hardware. In the scenario depicted in Figure 1.2 three different types of nodes are considered. The anchor nodes, the target node and the central node. Each type of node was programmed with different software. Ultimately, the prototype system was deployed and the results obtained are compared to those produced from simulations. The prominent outcomes justify the choice of range-only tracking and also reveal that the system satisfies the three main objectives of accuracy, real-time operation and tracking of manoeuvring targets. Summarizing, the key contributions introduced in this dissertation are:

- Design and development of a range-only tracking system for WSN. Hardware implementation on COTS platform and evaluation under real-world scenarios.
- Utilization and implementation of a two-way ToF ranging scheme for low-power embedded nodes for the acquisition of the range information between the target and the anchors.
- Both single and adaptive multiple-model dynamical systems were formulated. The multiple-model approach is targeted to be employed in situations where manoeuvring targets are considered for tracking.
- Two Particle Filter based tracking algorithms were developed in this research, to operate on the accumulated range readings and estimate the target's kinematic variables.

1.4 Publications

The following publications have been produced as a result of the work that was carried out in this thesis.

1. E.B. Mazomenos, J. S. Reeve, N. M. White, A. D. Brown. "A Tracking System for Wireless Embedded Nodes using ToF Ranging". In: IEEE Transactions in Mobile Computing (under preparation for submission).
2. E.B. Mazomenos, D. De Jager, J. S. Reeve, N. M. White. "A Two-Way Time of Flight Ranging Scheme for Wireless Sensor Networks". In: Proc of the 8th European Conference in Wireless Sensor Networks(EWSN 2011), 23-25 February 2011, Bohn, Germany.
3. E.B. Mazomenos, J. S. Reeve, N. M. White. "Tracking Manoeuvring Mobile Nodes in Wireless Sensor Networks". In: Proc of the 7th IEEE International Conference on Networking, Sensing and Control (IEEE ICNSC), 10-13 April 2010, Chicago, IL, USA.
4. E. B. Mazomenos, J. S. Reeve, N. M. White. "Tracking with Range-Only Measurements Using a Network of Wireless Sensors". In Proc. of 6th IEEE International Conference on Broadband Communications, Networks and Systems (IEEE BROADNETS), 14-17 September 2009, Madrid, Spain
5. E. B. Mazomenos, J. S. Reeve, N. M. White. "An Accurate Range-Only Tracking System using Wireless Sensor Networks". In Proc. of Eurosensors XXII Conference, 6-9 September 2009, Lausanne, Switzerland

6. E. B. Mazomenos, J. S. Reeve, N. M. White. “Accurate Range-Only Tracking in Wireless Sensor Networks”. In 7th ACM Annual International Conference on Mobile Systems, Applications and Services (ACM MobiSys) (poster abstract), 22-25 June 2009, Krakow, Poland

7. E. B. Mazomenos, J. S. Reeve, N. M. White. “A Range-Only Tracking Algorithm for Wireless Sensor Networks”. In Proc. of the 23rd IEEE International Conference on Advanced Information Networking and Applications (AINA) Workshops, 26-29 May 2009, Bradford, U.K.

1.5 Dissertation Structure

The remainder of the thesis is structured in the following way.

- Chapter 2 introduces the reader to the fundamentals of the tracking problem. It also performs a literature review in the area of target tracking with Wireless Sensor Networks by presenting the most prominent outcomes.
- Chapter 3 presents the design and implementation of the ToF ranging method for low-power embedded nodes which was developed as part of this research. Experiments from outdoor and indoor deployments are used to quantify the achieved accuracy.
- Chapter 4 provides the theoretical foundation of the proposed tracking system. The tracking problem is mathematically formulated as a dynamical estimation problem. The two PF tracking algorithms (ROT-PF, ROT-MMPF) developed as part of this research are also presented in this Chapter
- In Chapter 5 a simulation-based analysis of the formulated tracking system is provided. The simulation environment tries to capture the real-world conditions and under these the performance of the proposed tracking system is evaluated. The simulation environment made possible to identify the factors that affect the system’s performance and quantify their effect.
- In Chapter 6 the development of the complete tracking system is analysed. The tracking system is devised by incorporating the tracking algorithms presented in Chapter 4 with the ToF ranging technique discussed in Chapter 3. A detailed analysis alongside experimental results is provided.
- Chapter 7 The final chapter summarizes the contributions that this thesis achieved and discusses possible directions for future research based on the outcomes and findings of this research project. It is believed that these directions can broaden the scope of this work and enhance the practical applicability of the proposed tracking system.

Chapter 2

Background and Related Work

The generic purpose of a tracking system is to determine the location and/or direction of a moving object (usually refereed to as "the target") on a near-continuous basis for a specific time of interest [124]. In the twentieth century several technological advancements facilitated the development of positioning and tracking systems. Many of these are used extensively in numerous everyday applications associated with navigation and tracking. Examples include the GPS and RADAR positioning systems [50, 103].

This chapter performs an introduction in the target tracking application and also surveys related work in the area of tracking with WSNs. Section 2.1 presents the core phases and operations of a tracking system. The different types of measurements that are utilized for tracking are summarized in Section 2.2. Background details on WSNs technology are provided in Section 2.3 and the specific hardware that was used in this research is presented in Section 2.4. In Section 2.5 results from the most prominent WSNs tracking systems reported in the relevant literature are presented and a discussion follows analysing the limitations of these systems. Section 2.6 summarizes the Chapter.

2.1 Fundamental Principles of detection and tracking

Tracking of a moving object is achieved either using a predefined set of landmarks (anchor positions) or with dead-reckoning. In the first approach, the target's kinematic variables are estimated with respect to the anchors with the use of specific observations (range, bearings) which become available to the system. For example, by knowing the distance of an object from at least three anchor points, its position can be calculated using trilateration (Figure 2.1). Trilateration is is used in GPS to calculate the position of the receiver on the earth's surface.

In dead-reckoning, the target's kinematics are estimated with the use of motion dynamics laws given a known starting point. If an object starts from point $A(a_x, a_y)$ moving with a constant speed v and heading ϕ then at time t its coordinates will be given as:

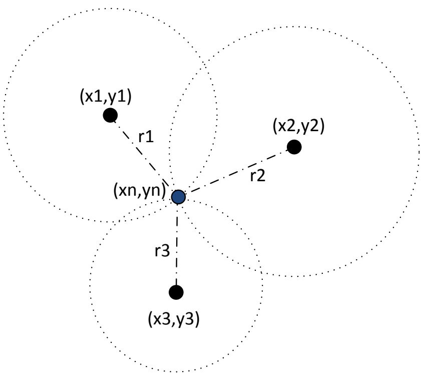


FIGURE 2.1: Trilateration: The unknown coordinates x_n, y_n can be calculated as the intersection of the three circles

$$x = a_x + vcos(\phi)$$
$$y = a_y + vsin(\phi)$$

Both tracking with landmarks and dead-reckoning have inherent limitations that can potentially lead to diminishing results. In tracking with landmarks, a major issue is the fact that the error in the observations is directly projected in the final result. For example when using trilateration to estimate the position of a target based on range from three anchors, any error in the range data is directly inserted into the position estimation. On the other hand, an important drawback of dead-reckoning is that this method suffers from an accumulation of the measurement error, since it relies upon the target's ability to measure its own dynamics. Consequently, the majority of tracking systems employ a combination of dead-reckoning and information obtained with respect to a number of anchors. Such an approach which is also followed in this research, exploits the potentials of both methodologies. Dead-reckoning is used to model the target's motion pattern and provide an initial estimate of the target's position. This is further refined by exploiting the observations that are received with respect to a number of landmarks. By combining these methodologies in a dynamical system, better tracking accuracy is achieved than simply trilateration or dead-reckoning.

As discussed previously, tracking systems are tasked with the estimation of the target's dynamics. However, every scheme that provides some kind of information regarding the kinematic attributes of a moving object can be considered to some extend as a tracking system. Tracking systems are closely related or sometimes conflated with motion detection systems. Motion detection usually refers to the process or the technique used to determine the physical presence or absence of a moving object (target) from an area of interest. There is obviously a fine line between motion detection and target tracking. In order for tracking to commence, the presence of the target must have been established. In some occasions a target tracking system may also integrate detection as a separate part of the complete system.

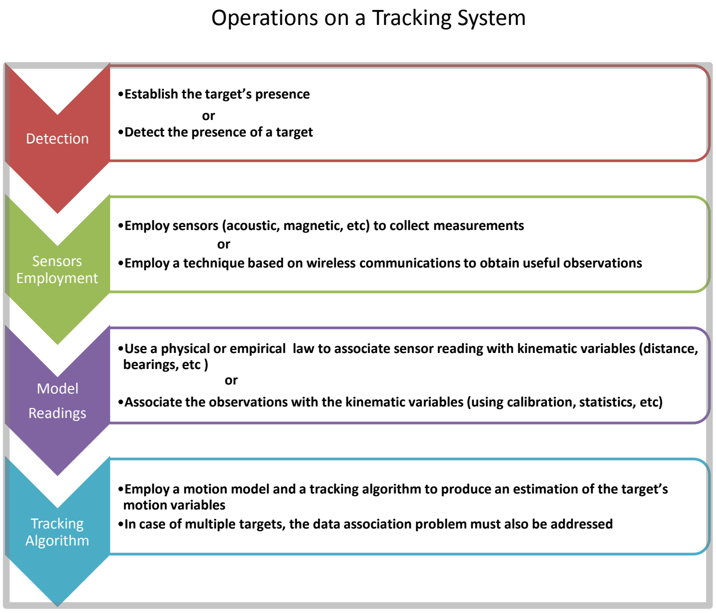
Another operation which is closely related to target tracking, is usually termed in academic research as “locationing” and in many situations the term is used with tracking interchangeably. Occasionally, “locationing” may refer to techniques that try to determine the location (in terms of coordinates, or the relative distance to a known location) of static non-mobile objects (e.g. nodes). A distinction between “locationing” and “tracking” can be attempted by pointing out that locationing may not involve the presence of a target, a specified object of interest that the system is tasked with monitoring its location.

Normally the particular tracking scenario that every system is developed upon, dictates which of the above operations must be implemented or not. For example a tracking system which is used for tracking assets or personnel in an industrial environment may not require the need for detection or locationing because every asset or member of the personnel may carry a device with a unique identifier whenever tracking commences. Moreover the anchor nodes that are used may be placed in pre-defined locations so there is no need for a self-locationing technique to be applied prior to tracking. On a different situation, a tracking system that intends to track enemies in a hostile environment like a battlefield might include ad-hoc deployment (airborne dropped) of the anchor nodes. After the anchors have reached the ground, the system may employ a self-locationing technique to assign relative coordinates to the anchor nodes. It may also include a technique to detect when a target is present in the area of interest and finally execute the tracking operation to track the target. Military surveillance scenarios is one application domain where motion detection, anchor node locationing and target tracking must be incorporated on a single system.

2.1.1 Phases of a Tracking System

The operational phases of a tracking system are described in this section. These refer to operations that are normally executed on a sequential way in tracking systems [126]. These parts are presented here from an information flow (datapath) perspective. Figure 2.2 illustrates the major phases of a target tracking system .

The initial phase pertains to the system becoming aware of the presence of an object that must be tracked. This phase may include the operation of detecting the presence of a moving target using a motion detection scheme. As pointed out previously this is the case in military tracking system where the aim is to track incoming intruders. Alternatively, in different application scenarios (e.g. industrial security systems) the target may voluntarily register itself as being present in order for the tracking operation to commence. In this case the system is not tasked with detecting the presence of the target. The second phase involves the initialisation of a number of anchor devices (e.g. sensors, wireless nodes) to interact with the target and obtain the necessary data for tracking. Finally the third and fourth phases include the mathematical relationship of this data with the target’s kinematic variables (e.g. position, velocity) and the execution of the tracking algorithm for the estimation of these variables. Also in the event that multiple targets need to be simultaneously tracked an extra step is introduced, in the algorithmic procedure, which is tasked with associating the accumulated observations with individuals targets. An estimation then is



•Establish the target's presence
or
•Detect the presence of a target

•Employ sensors (acoustic, magnetic, etc) to collect measurements
or
•Employ a technique based on wireless communications to obtain useful observations

•Use a physical or empirical law to associate sensor reading with kinematic variables (distance, bearings, etc)
or
•Associate the observations with the kinematic variables (using calibration, statistics, etc)

•Employ a motion model and a tracking algorithm to produce an estimation of the target's motion variables
•In case of multiple targets, the data association problem must also be addressed

FIGURE 2.2: The phases of a target tracking system

produced for every target separately based on its individual set of data [17].

2.2 Information Acquisition and Measurements Categories

The majority of tracking systems are based on two well-known types of observations which are obtained relatively to a number of known locations and are utilized as the basis for the formulation of tracking algorithms. The following sections analyse these two types of data and presents methods for their acquisition in wireless embedded nodes.

1. Range-Distance observations
2. Bearing Observations

The aforementioned types of observations have been used so extensively, that tracking systems are categorized based on the type of data they employ as, range-only tracking systems or bearing-only tracking systems or tracking systems that utilize both types of measurements.

2.2.1 Range Measurements

Range measurements in tracking systems pertains to the ability of calculating the Euclidean distance between the target and a reference point at a specific time instance. Considering tracking of mobile embedded nodes, a number of techniques have been developed for estimating the distance between two wirelessly communicating nodes. A number of these techniques are presented in the following section, with details regarding their implementation.

2.2.1.1 Received Signal Strength Indication (RSSI)

The Received Signal Strength Indication is the measurement of the power of a radio signal received at a certain receiver (e.g. the power of the electromagnetic signal received by an antenna). The RSSI is utilized in IEEE 802.11 networks to indicate the amount of energy present in the channel. When this amount is lower than a predefined threshold, the transmitter can transmit a certain packet. Usually RSSI is expressed in dbm units. The RSS value is known to attenuate with increasing distance. However, an accurate relationship between the RSSI and distance depends on the specific transceiver considered, the specification of the wireless network as well as the occupancy in the wireless channel at the time of the measurement. Due to the documented association of the RSS observation to the distance between communication nodes, RSSI has been investigated in locationing schemes for wireless nodes in a number of different network technologies [16, 65, 89, 100, 106, 129].

In an attempt to describe the association of the RSSI value against the distance in wireless embedded nodes various models have been proposed in the literature, based on experimental results. The SpotOn system investigates the localisation of mobile nodes (RFID) from RSSI values obtained between the nodes and a number of base stations [59] in an indoor environment. The RSSI value is expressed as a function of the distance between the two devices as:

$$RSSI(r) = 0.0236r^2 + 0.629r + 4.781 \tag{2.1}$$

The empirically derived relationship, obtained from indoors deployments, suffers from the need for constant recalibration once another deployment environment is considered. Savvides *et al.* perform a similar investigation of modeling the RSS value as a function of distance on WINS sensor nodes. After performing experiments in various environments (both indoors and outdoors) with nodes placed at the ground as well as being elevated from it the authors deduce that a single consistent model that relates the RSSI with distance could not be obtained, due to multipath fading and shadowing effects. Subsequently, an analytical relationship was produced only in an idealised setting (a football field) with nodes positioned at ground level. Considering the following inverse power model in 2.2.1.1 the two parameters (X,r_n) are calculated from experimental datasets obtained for two transmission power levels with the use of least squares fit [106]. Details are given in Figure 2.3.

$$P_{RSSI} = \frac{X}{r_n} \tag{2.2}$$

where X is the transmitting power of the transmitter and r_n an attenuation coefficient.

In general RSSI is considered to be an unreliable indicator and the resolution that offers depends on a number of external factors [42, 128]. The absence of a generic model between the signal strength indication and the distance between communicating nodes results in calibration methods to be used extensively in tracking or locationing systems that select to use the RSSI to

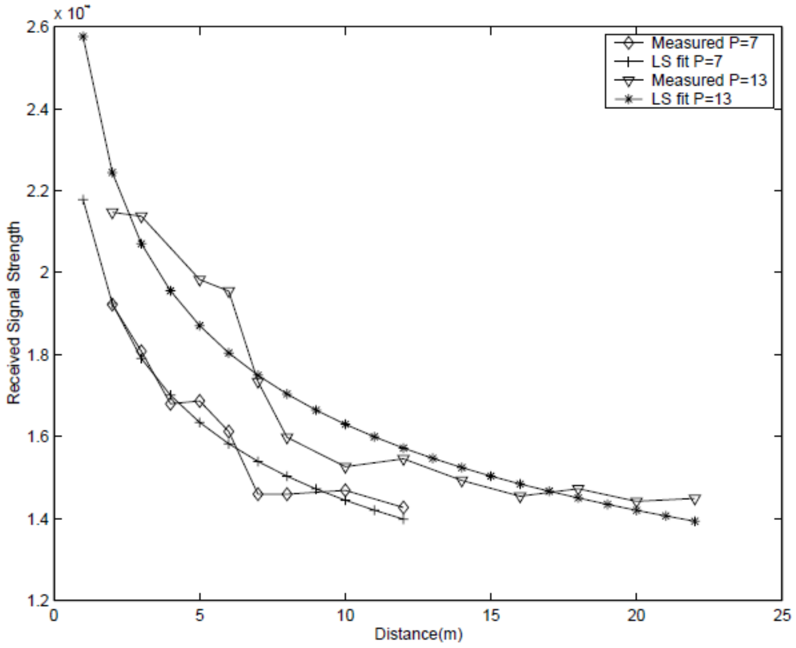


FIGURE 2.3: The relationship between RSSI and distance in WINS node. Reproduced from [106]

estimate ranging. In addition, obstacles in the surrounding environment can cause unexpected attenuation of the RSSI making difficult the association with the distance between the participating nodes. The authors in [106] deduce that RSSI can provide an accuracy of a few meters when nodes are placed on a flat plane. In all other cases, their experiments revealed that the use of RSSI for distance estimation can lead to unpredictable results. To conclude with, RSSI can be used as a ranging technique but with extreme caution taking into account that significant errors may be introduced.

2.2.1.2 Time of Flight - Time (Difference) of Arrival

Another technique that can be employed in order to estimate the distance between two wireless communicating nodes is the Time of Flight or Time (Difference) of Arrival (ToF - TDoA). ToF ranging systems attempt to estimate the point-to-point distance between two communicating devices by capturing the time that a signal requires to travel from one device to the other. Since the speed of the signal is known and constant (e.g., the speed of light for electromagnetic signals), the distance can then be calculated. McCrady *et.al* are among the first to propose a ToF ranging system for WSNs [83]. However their work lacks implementation. Signal bandwidth is know to be one important parameter that affects the ranging accuracy in ToF systems. In simple words the higher the signal bandwidth is, the better the timing accuracy a ToF system can achieve. Subsequently, Ultra Wideband (UWB) transceivers have the ability to yield fine-grained resolution in measuring the ToF due to the high bandwidth occupancy (500 MHz) they employ. Thus, a handful of ToF ranging systems are based on Ultra Wideband (UWB) technology [30,47,78]. However, low-power WSNs nodes are normally not equipped with UWB transceivers and their incorporation on embedded nodes presents a number of challenges. Lanzisera *et. al.* propose a ToA locationing scheme for low-power ASIC WSN nodes [77]. In the prototype hardware, an FPGA board is attached to the WSN node to carry out the necessary calculations. FPGA boards alongside WSN nodes are also used in [70], where a RF-ToF ranging system is presented and the ToF is extracted by the channel impulse which is produced after converting the received signal from the time to the frequency domain by applying FFT.

An intriguing approach for ToF ranging in WSNs is the one that employs acoustic signals instead of electromagnetic ones. It is known that acoustic signals travel in a much slower speed than electromagnetic signals thus making them easier to utilize in ToF scenarios. Both ultrasonic and audible sound signals have been utilized in ToF ranging systems. Occasionally, acoustic and RF signals can be combined in a time difference of arrival method (TDoA). The two signals are emitted simultaneously and the RF signal is used to synchronize the receiver. The TDoA value is considered to be the ToF of the acoustic signal. A ranging system based on this approach is implemented on the Mica2 mote in [105]. A simple tone which is produced by the mote's sounder is the acoustic signal that it is timed. The "Calamari" localization system follows a similar approach but employs the tone detector of the Mica mote instead of the sounder and requires all participating nodes to be pre-calibrated to achieve good accuracy [127]. Another example where acoustic and RF signals are used on the same system is the "Cricket" locationing system developed at MIT [92]. One disadvantage of acoustic ranging, is the limited effective range of acoustic-based ranging systems. The TDoA systems presented previously are capable of producing accurate ranging of a few centimeters but within a limited range.

2.2.1.3 Acoustic Energy

The attenuation of the energy that an acoustic signal carries can be modeled as a function of source-to-sensor distance [105]. This attenuation model in conjunction with acoustic sensors can be employed to achieve locationing of stationary or tracking of mobile targets. In the presence of N sensors the acoustic energy emitted by the k^{th} target and received in the i^{th} sensor can be expressed according to Sheng *et al.* as [108, 109]:

$$s_i(n) = \gamma_i \sum_{k=1}^{K} \frac{\alpha_k(n - t_{ki})}{||\rho(n - t_{ki}) - r_i||} \tag{2.3}$$

where: $\alpha_k(n - t_{ki})$ is the intensity of the k^{th} acoustic source measured at one meter from that source, while t_{ki} is the propagation delay from the k^{th} source to the i^{th} sensor. ρ_k is the unknown position vector of the k^{th} target and r_i is the position vector of the stationary i^{th} acoustic sensor. Finally n denotes time. This model is formed for multiple acoustic sources but it can easily be simplified for a single source in the presence of multiple acoustic sensors.

The acoustic attenuation model has been extensively investigated and proposed in a number of locationing and tracking approaches [2, 22]. The major drawback of acoustic localization systems is that can provide very good accuracy but for limited distances. As a result, tracking systems that employ acoustic locationing tend to use a very dense network deployment even for small areas.

2.2.1.4 Other types of sensors that can be utilized

Similar to acoustic sensors, other type of sensors can be utilized to provide range measurements. The measured value of a number of physical modalities depends on the distance between the source and the sensing device that performs the measurement. Generally the value of this modality will attenuate as distance increases. Examples of sensor types include light intensity sensors and magnetic sensors. For example, Zhang *et al.* present an algorithm for navigating mobile sensors towards a stationary point. To evaluate their algorithm they use a light intensity sensors (photoresistor) in order to navigate a number of mobile sensors [130].

Binary proximity sensors (PIR sensors) are another class of sensors that can provide tracking-efficient information. Binary detection sensors are deployed normally in large numbers forming a dense deployment. PIR sensors detect an intruding event whenever a moving object is physically present within their detection range. A priori knowledge of the sensor coordinates enables tracking by assuming that the target's position is estimated adequately by the detecting sensor known coordinates. Such an approach offers the advantage of modeling the system using a linear approach, hence a Kalman Filter is used to infer to the target's dynamics [69].

Another approach in estimating the distance between two wireless nodes is proposed by Maroti *et al.* and termed as "Radio Interferometric Positioning". The concept idea of this technique is that the distance between two wireless nodes can be estimated by programming the two nodes (transmitters) to concurrently emit unmodulated high frequency sine waves, at very close frequencies and capturing the composite interference signal which is produced, by two other nodes (receivers). The relative phase offset of the composite signal at the receivers side depends on the in-node distances between each receiver and the two transmitters. The formulation of a "q-range" equation is illustrated in Figure 2.4. By positioning at least three of the four nodes in known coordinates the coordinates of the fourth node (receiver node) can be extracted using multiple "q-range equations". This method reports errors in the range of centimeters on outdoor deployments where 16 stationary Mica2 nodes are deployed in a 18m x 18m area [74, 81].

2.2.2 Bearing Measurements

Bearing is defined as the direction of movement of a mobile object in relation to another object (mobile or stationary). It is expressed as the angle between a line connecting two points and a north-south line (meridian). It is used extensively for navigation (marine, aviation, land) purposes. In land navigation, bearing is traditionally defined as a fixed number line which gives the smallest arc (never to exceed 90 degrees). More specifically, a bearing is measured both east and

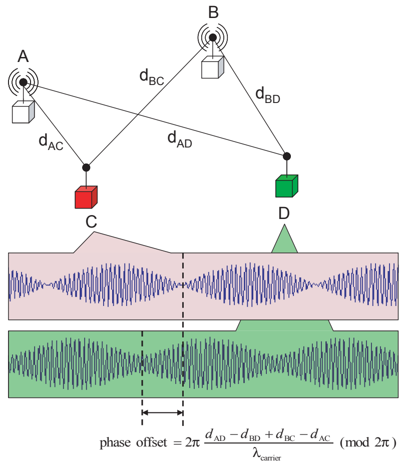

B
A
d_BC
d_BD
d_AC
d_AD
C
D
phase offset = 2π (d_AD − d_BD + d_BC − d_AC) / λ_carrier (mod 2π)

FIGURE 2.4: Radio Interferometry ranging technique. Reprinted from [81]

west from north and south, divided into four quadrants. In contrast, an azimuth is a clockwise measurement of a circle from a zero point at a fixed horizontal plane of reference (such as a north meridian), expressed in degrees, mils, or other units of angular measurement. For example, a given azimuth in degrees would be expressed as 60°, while the equivalent bearing would be expressed as N 60° E [34].

Bearing measurements are referred also as angle of arrival (AoA) measurements or direction of arrival (DoA). The coordinates of a point in the place can be calculated if multiple angle (bearing) measurements with respect to known reference points are available. Thus, DoA observations have been extensively used in locationing and tracking systems [28, 82, 84, 96, 102, 117, 118].

To acquire bearings measurements in WSNs nodes, the use of directional antennas has been investigated in [13]. Amudson *et al* propose the use of antenna arrays and radio interferometry to acquire AoA readings [8]. The “Cricket Compass” is a prototype compass platform equipped with two ultrasound receivers. By measuring the phase difference of ultrasonic signals emitted from mounted-ceiling beacons positioned in known locations, the bearing of the receiver with respect to the beacons can be derived [94].

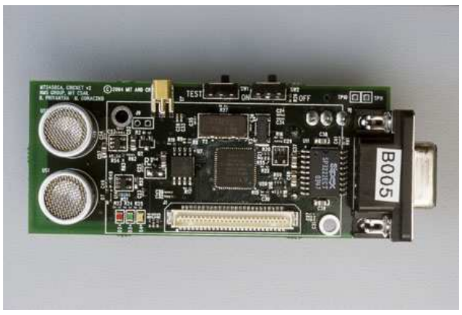


FIGURE 2.5: The Cricket node. Reproduced from [92]

2.2.3 ToF vs RSSI

Albeit being a difficult task, a number of research efforts tried to compare the prevalent ranging techniques used for estimating the distance between WSNs nodes. A performance comparison between ToF and RSSI techniques specifically for sensor networks was performed by Shi *et al.* [110]. The ToF technique was implemented on UWB devices while the RSSI on 802.15.4 nodes. The authors claim that in multihop environments, where the distance between two nodes (source node - destination node) is calculated by calculating the distances of multiple intermediate nodes, RSSI yields better results than ToF. RSSI proves to be more accurate than the ToF. The authors reach this conclusion by theoretically deriving the variance of the results that the two methods yield. For the RSSI the authors consider a log-normal model to associate RSSI to distance. In the ToF case the authors consider that the distance estimation is normally distributed around the true distance. By taking into account multiple intermediate distances that are estimated with these two models, the authors conclude that for RSSI an evenly distributed number of hops (intermediate distances from source to destination) results in the best possible accuracy and also that the accuracy improves with higher number of hops. On the other hand, for ToF based estimation, accuracy is solely based on the number of hops and deteriorates with increasing number of hops. Savvides *et al.* also combine these two ranging methods and conclude that ToA is more accurate than RSSI. However the ToA technique they considered is based on two signals an RF and an ultrasonic one and ranging is achieved by measuring the time difference of arrival between the two signals. Another comparison between these two techniques is provided by Patwari *et. al* [91] where the two techniques demonstrate equal accuracy in localizing stationary nodes. Following this pattern of comparing ToF to RSSI, we attempt our own comparison for these two techniques given the specific hardware that we chose to implement our ToF method (Section 3.9).

Gustaffson *et al.* present a thorough investigation of available ranging methods (RSSI, TDOA, AOA) regardless of hardware demands, concluding that AOA can provide the most accurate locationing information. For the tracking operation, the authors distinguish between dynamic approaches and geometric approaches where the position estimation problem is interpreted as a problem of solving nonlinear equations. The authors compare the most studied motion models for four different types of targets (stationary, indoor, walking, vehicles) and calculate the lower error bounds for each one of these model [52].

2.3 Wireless Sensor Nodes

The hardware architecture of a typical WSN node includes four major subsystems. These subsystems are listed in the following and their interconnection is illustrated in Figure 2.6

- The intelligent sensing subsystem

- The wireless communication subsystem (predominantly radio frequency (RF) wireless communication is used)
- The core processor subsystem
- The energy supply subsystem (typically consists of batteries, optionally may include energy harvesting)

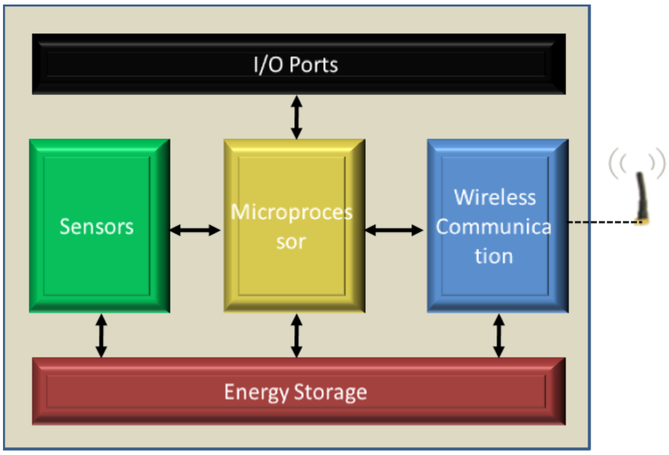


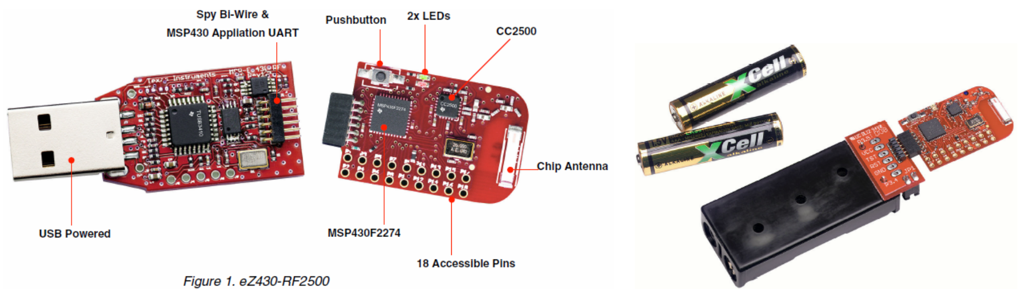
FIGURE 2.6: A typical wireless sensor node

WSNs typically communicate with wireless Radio-Frequency (RF) communication on the 868 / 915MHz or the 2.4GHz range under the IEEE 802.15.4 standard for wireless personal area networks. The 802.15.4 supports bit rates upto to 250kbps [14, 15]. The 802.15.4 standard enabled the creation of a number of protocols specifications for the upper layers of the communications stack. The most prominent of these as it pertains to the WSNs technology is the Zigbee specification. The Zigbee specification builds on the top of the physical layer and the medium access control defined on the 802.15.4 defining the network and application layers to complete the network stack [5].

WSNs are intended to operate autonomously and usually without the ability of recharging or replacing their energy supply, typically batteries. For this, significant research is devoted in energy harvesting solutions (solar, vibration) for embedded nodes that provide the ability of replenishing the node's energy supply [6, 64, 79].

2.4 Hardware Platform used in this research

The Texas Instruments (T.I.) EZ430-RF2500 is a complete wireless development platform, combining the MSP430 microcontroller and the CC2500 low-power radio module. The EZ430-RF2500T target board connects via a UART serial port to the TUSB3410 dongle which bridges the UART to a USB PC port for programming, debugging and serial communication purposes [60]. The EZ430-RF2500 battery board employs two AAA batteries as the platform's power supply. In Figure 2.7 the EZ430-RF2500 development kit is illustrated.



(a) The ez430-rf2500 and TUSB3410

(b) The ez430-rf2500 battery board

FIGURE 2.7: The EZ430-RF2500 development kit. Reprinted from [60]

2.4.1 Microcontroller and Radio

The EZ430-RF2500 has an MSP430F2274 microcontroller (mcu). The MSP430F2274 is a 16-bit micro-cpu with a clock speed up to 16MHz. It employs 32kB of flash memory and only 1 kB of RAM. The MSP430F2274 has 21 development pins and operates on the 1.8 - 3.6V voltage range. The MSP430F2274 offers the ability to be put to a number of power-preserving states ensuring low-power consumption. The MSP430F2274 microcontroller contains the following clock sources. A low-power 12KHz crystal oscillator (VLO) and a more accurate and energy demanding digital controlled oscillator (DCO). The VLO is sourced to the auxiliary clock signal (ACLK) while the DCO is used for the cpu's main clock (MCLK) and the submain clock (SMCLK). The MSP430F2274 also offers an additional clock source (oscillator) which can be driven from external crystals and is factory calibrated (at 32.768 KHz). Two timers/counters (named Timer A (16bit) and Timer B (variable bit-length)) are present and can be linked to any of the available clock sources. The timers have multiple capture/compare registers that can be used to initiate interrupts.

The CC2500 is a low-cost radio transceiver on the 2.4GHz RF band, designed for low-power embedded applications. Various modulation formats (OOK, 2-FSK, MSK) and data rates (2.4 - 500 kBaud) are supported. It employs a 26-27 MHz crystal oscillator for timing purposes (setting frequency). The configuration of CC2500 is done by programming a number of 8-bit registers. A number of these registers are associated to general purpose output digital pins (GDO0-2) that can be used for signaling interrupts related to the radio operation. In the proposed tracking system one of this pins is programmed to go “hi” whenever a wireless packet has been transmitted or received [62].

The CC2500 radio does not directly support the IEEE 802.15.4 frame format. It uses a format provided by T.I., very similar to the one described in the 802.15.4 protocol, and offers the option of manipulating that format to the user's convenience. The CC2500 consists of a variable length preamble sequence (PRE), a synchronisation word (SYNC WORD), a length byte (LEN), an address byte (ADD), the data payload (PAY) and finally an optional two-byte cycle redundancy check field (CRC). The complete packet frame is illustrated in the following Figure 2.8. The packet's length is 256 bytes although the cc2500 has an option for larger packet formats.

Depending on the application's requirements, optimum settings can be selected.

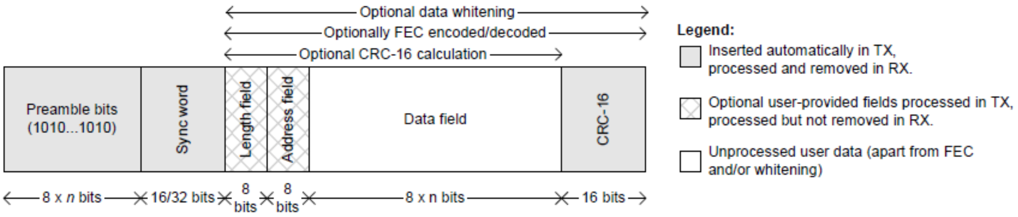

Optional data whitening
Optionally FEC encoded/decoded
Optional CRC-16 calculation
Preamble bits (1010...1010)
Sync word
Length field
Address field
Data field
CRC-16
8 x n bits
16/32 bits
8 bits
8 bits
8 x n bits
16 bits
Legend:
Inserted automatically in TX, processed and removed in RX.
Optional user-provided fields processed in TX, processed but not removed in RX.
Unprocessed user data (apart from FEC and/or whitening)

FIGURE 2.8: The CC2500 packet format. Reprinted from [62]

2.4.2 Programming Environment

T.I. provides support for the ez430-rf2500 through the Code Composer Studio (CCS) environment that offers advance program composing, code editing and debugging. The programming language is an extended version of the "C" language with various device specific routines. T.I also provides "Simplicity", a relatively simple low-power RF network protocol stack tailored for small scale RF networks, which allows the inexperienced user to operate the ez430-rf2500 hardware out of the box [61]. Even though, "Simplicity" offers various abstraction layers, facilitating the development of new code, for the purposes of this project the basic operations of the node were developed from the beginning. The main reason for following this direction was the attempt to minimise the code size. Hence, only the most basic components were used from T.I. libraries and on the top of these all the necessary node operations were built. In general we found the EZ430-RF2500 relatively easy to be programmed.

2.5 Locationing and Tracking in WSNs

This section reviews a number of projects related to position estimation developed with the use of WSNs. As mentioned earlier, locationing and tracking are considered suitable application for WSNs due to the fact that WSNs can provide information with high spatial resolution because of their ability to be deployed in large numbers over a specific area. A plethora of approaches regarding tracking/locationing in WSNs has been presented over the previous years. We will restrain ourselves to the tracking systems that were implemented and demonstrated at full scale.

A WSN deployment intended for tracking will have to address a number of issues apart from the tracking operation. Latency is of significant importance for tracking systems. Cao *et al.* studies the problem of delays in detecting targets in a surveillance scenario. The authors consider that a WSN which is deployed for surveillance purposes will include a duty-cycle scheme, for power management purposes, where nodes sleep and wake-up periodically to investigate for possible targets being present. Several networks parameters are considered that can affect the system's performance [26]. The importance of the routing protocol, particularly when a larger-scale deployment is attempted, is investigated by Tran *et al.* in [123]. The authors provide a comparison of some basic routing protocols in respect to tracking application demands. Their approach ameliorates existing routing approaches in WSN by minimizing the network's energy consumption. Finally, the problem of low accuracy in data gathered due to the fact that inexpensive sensor circuits are used with WSN must also be considered. Roosta *et al.* investigate the effect of sensors that provide faulty data or "liars" in a network of binary sensors that performs detection of moving targets. The authors consider various types of faulty data and propose methods to amend the effects caused by the "liars" on the robustness of the system [101].

2.5.1 Military Surveillance Systems

The basic concept in military scenarios is to deploy a WSN to form a security perimeter. The network's primary mission is to detect any intruders that breach that perimeter using various sensors (RADARS, PIR, magnetic etc). In addition, such systems should be able to effectively classify the intruder to one of the available categories (unarmed people, armed troops or military vehicles) [24].

Large-scale military systems based on dense WSNs deployments are presented under various projects in the literature. "A line in the sand" describes a military surveillance scenario where the system must be able to detect, classify and then loosely track incoming targets. A dense network of WSN nodes equipped with micro-RADARS is deployed for that purpose [10]. A similar military scenario is also considered in the "Vigilnet" project where an integrated sensor network system is deployed for the detection of hostile targets (vehicle, personnel). The system employs Mica2 and XSM motes equipped with magnetic, acoustic and PIR sensors and capable of classifying the incoming targets. The "Vigilnet" software architecture is a complex multiple-layered structure that also includes power management, clock synchronisation, group formation and distributed processing [51,54,55]. Duarte et al. describe another military classification system based on WSN hardware. The hardware used is the WINS NG 2.0 node, a rather powerful platform compared to the platforms used in the previous projects. Each node in this system is equipped with seismic (geophone), acoustic (microphone) and infrared (PIR) sensors which provide data in order to classify five different types of military vehicles. For that purpose a number of classification algorithms are presented and compared [40].

On a slightly different approach, the system presented by Oh *et al.* describe a dense WSN formed of PIR sensors that intends to detect incoming targets (personnel) and employ pursuers as countermeasures [29,85]. The PIR sensors act as binary detectors, signaling a detection event whenever a moving object is within their coverage area. Tracking is accomplished by considering the a-priori known locations of the PIR sensors as the target's location. The tracking algorithm employd in this system is based on a Markov Chain Monte Carlo Data Association algorithm which is focused on the data association problem in order to enable tracking of multiple present targets [86,87]. The final system was implemented on 557 TRIO motes [41].

These systems although very prominent are tasked with a different purpose than the one we target. Detection and classification is deemed to be more important than real-time target tracking.

2.5.2 WSNs tracking systems

Coates *et al.* consider a hierarchical clustered WSN comprising of class-B sensor nodes that measure, either the range or the AoA and class-A cluster heads that aggregate the data gathered from the class-B nodes. Each of the class-A cluster heads runs its own local PF tracking algorithm based from the data acquired from the class-B nodes in his neighborhood (cluster). The weights for each particle are then calculated based on information from all the cluster heads. Finally each cluster head represents a particle with a certain weight associated to it and a global estimation can be extracted. The incentive behind this approach is to distribute the processing burden across a number of nodes. The drawback of such an approach is that there is a need for a large number of cluster heads-particles (> 200) and subsequently even larger for class-A nodes to achieve good accuracy, resulting to a network that involves an excessive number of nodes [31,32].

The “CRICKET” locationing system [92] developed at MIT, focuses on developing an effective and energy-efficient indoor locationing system. Cricket consists of location beacons that are attached to the ceiling of a building, and receivers, called listeners, attached to devices that require locationing. Each beacon periodically transmits its location information in an RF message. At the same time, the beacon also transmits an ultrasonic pulse. The listeners listen to beacon transmissions and compute their own locations by calculating the TDoA of the two signals emitted from nearby beacons. The user’s location is determined in relation to the already known location of the mounted nodes [93].

The same locationing system is used in an indoor tracking system which employs the same TDoA ranging technique and two signal processing stages to track slow speed users. These two stages correspond to an Extended Kalman Filter (EKF) tracking algorithm and a Least Squares Minimization [113]. The achieved accuracy was 15cm in a 3x1.5m area. The CRICKET ranging system was also used in order to provide measurements for a centralised localization and tracking Bayesian algorithm named “LaSLAT”. The algorithm employs Laplace’s method to approximate the probability distribution of the state vector. The “LaSLAT” algorithm reported a few centimeters of error in an 7m x 7m indoor area and approximately 0.5m error in an 27m x 32m dense outdoor deployment [119].

RADAR is an indoor locationing system which is based on low-power WSNs. The RADAR system implements a localisation service utilizing a map of RF signal strength built during the deployment phase. A number of infrastructure nodes, positioned in known locations is used to generate RSSI values for different positions on the coverage area and build the signal strength database. During the normal operation, the RF signal strength of a node which needs to be localised is measured by the closest infrastructure nodes. Subsequently the observations are fused to a central computer, which examines the signal strength map to obtain the best fit for the current transmitter position. The achieved accuracy is between 2 to 3 meters [16]. Another methodology utilizes RF signal propagation models to construct the signal strength map [63, 107].

Ahmed *et al.* address the combined problem of target detection and tracking. Target presence or absence is modeled by a probability function. As a result the Bayesian Estimation algorithm that is used attempts to estimate apart from the target's state vector an extra binary variable which indicates the presence of the target. The prototype system was integrated on Crossbow MicaZ motes which used the intensity of acoustic signals emanated by the target. A dense network of MicaZ nodes provided range readings (through acoustic intensity). A Particle Filter algorithm, which employs a large number of particles (5000) is developed to estimate the target's position. The reported results are in the area of 0.1 - 0.25m in an indoors 1 x 3m area for target's moving in slow speeds of 0.2 - 0.35m/s with the use of at least 8 anchor nodes [1–3, 97].

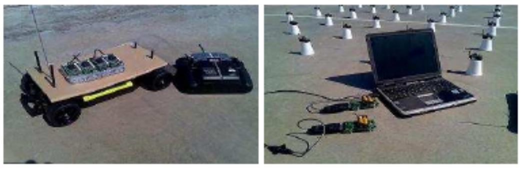


FIGURE 2.9: Dense network set up for acoustic based tracking. Reprinted from [1]

Radio Interferometry ranging for WSNs nodes was analysed in Section 2.2. In [72] Kusy *et al.* employ Radio Interferometry to obtain ranging observations for a tracking system named "inTrack. In "inTrack", the target node and one infrastructure node (positioned in known coordinates) simultaneously transmit the required sine-waves and the resulting composite signal is captured by two or more other infrastructure nodes (receivers). By combining multiple "q-ranges" (see Fig 2.4) from multiple infrastructure nodes and employing a geometric algorithm to compute the intersection of hyperbolaes (defined by the q-ranges) the target's position is inferred.

An extension of "inTrack", is presented in [75], where additionally to the "q-ranges" the Doppler shift of the sine-wave transmitted from the target node is also measured at the receivers (infrastructure nodes) side and used to estimate the target's velocity. A tracking algorithm combining an Extended Kalman Filter and a Constrained Non-linear Least Squares (CNLS) optimisation is used to infer the target's position and velocity. The reported accuracy of the system increases with increasing number of participating infrastracture nodes [73]. A deployment of 8 infrastructure nodes in 50mx30m area reported results of 1.3 - 2.2m for position and 0.13 - 0.35 m/s for velocity [7].

Table 2.1 lists the reported performance of the previously discussed WSNs tracking/locationing systems. The systems that are based on acoustic ranging not only require a dense deployment of anchor nodes, even in small areas, but also have the need for additional hardware like ultrasound transceivers to be attached on the WSNs nodes. Moreover in order to apply acoustic ranging the target must produce certain acoustic signals. This is demonstrated in experiments for "LaSLat"

(49 anchors in a 10m x 17m area indoors, 42 in a 27mx32m area outdoors) and in Ahmed *et al.* ($>$ 8 anchors in a 1m x 3m area).

InTrack and its extension mTrack are the two systems which are based on the Radio Interferometry ranging scheme. This method produced high fidelity ranging (5cm) for WSNs nodes over extended areas (18m x 18m). Consequently the tracking results of inTrack and mTrack are very accurate. However the interferometric ranging method requires the target node and an another node to transmit sine waves simultaneously. This translates to the two nodes being constantly synchronised, a relatively burdensome requirement for WSNs devices. Different to this, the proposed ToF ranging scheme only has a calibration requirement.

Although these systems produced seminal results, limitations are still present. Previous research considered a linear model in order to represent the target's motion dynamics. However this approach can not effectively cope with alterations in the position and velocity vectors of a manoeuvring target. To achieve tracking of manoeuvring targets, adaptive estimation algorithms and a multiple-model approach to describe the development of the target's dynamics in time are investigated in this thesis.

Tracking System	Position Error (m)	Velocity Error (m/s)
LaSLat [119]	5cm (indoors), 0.5 (outdoors)	n/a
RADAR [16]	2-3 (indoors)	n/a
inTrack [72]	0.6	n/a
mTrack [75]	1.3 - 2.2	0.13-0.35
Ahmed *et al.* [3]	0.1-0.25m	n/a

TABLE 2.1: Reported Performance of WSNs tracking/locationing systems

2.6 Chapter Summary

This chapter provided the general principles underlying the design of target tracking systems. It presented the various types of measurements that are used for tracking as well as methods to acquire them in wireless nodes. Section 2.5 highlighted some of the most prominent attempts in target tracking with WSNs. The majority of these attempts provided significant results. Nevertheless limitation are still in place. The research presented in this thesis aims at addressing these open challenges.

Chapter 3

Time of Flight Ranging for Wireless Embedded Nodes

This chapter discusses the theoretical aspects as well as the development of a two-way time of flight (ToF) ranging technique targeted to be used in the range-only tracking system introduced in Chapter 1. In Section 2.2 a number of existing methods to acquire the range in embedded nodes, was analysed. In this chapter the reasons for opting to use an RF-ToF ranging technique are initially discussed with some background information regarding ToF ranging (Sections 3.1-3.4. In sequel the specific details of the two-way ToF ranging method that is proposed as part of this research are provided in Section 3.5. A thorough analysis, on the development on the T.I. EZ430-RF2500 hardware platform, follows in Section 3.6. A number of experiments are presented in Section 3.7 to reveal the performance achieved and quantify the accuracy of the ranging technique. In Sections 3.8 and 3.9 we perform a comparison on the achieved accuracy of the two-way ToF method against the theoretically expected performance and RSSI respectively. The conclusions of this Chapter are discussed in Section 3.10.

3.1 Preliminaries of Time of Flight Ranging

The ability to estimate the relative distance between low-power wireless embedded nodes is paramount for a number of applications that require location-awareness [56, 90]. In the general case, two or more nodes will engage in some kind of interaction, typically transmit and/or receive signals, and will be tasked with measuring a property of that interaction that can be appropriately processed in order to extract the relative distance between the two interacting nodes.

Important advancements in microelectronics technology over the past decade, resulted in the production of very accurate clocks (ns accuracy) in electronic devices. As a result, a handful of ToF methods has been utilized in a number of established navigational and positioning systems (e.g. GPS). Consequently, as node localization became a necessity in WSNs, ToF techniques for low-power sensor nodes have been investigated. One major area of concern, has been the fact that low-power embedded devices are not equipped with high frequency clocks and time synchronization in these devices is an inherently difficult task, which has also attracted significant research interest itself [77, 80, 112].

Time of Flight (ToF) or Time of Arrival (ToA) methods are based on calculating the transit time of a signal and use it to estimate the point-to-point range of two nodes. The amount of time that a signal requires to reach the receiver is measured with the use of on-node clocks. The a-priori knowledge of the signal's velocity enables the approximation of the desired distance.

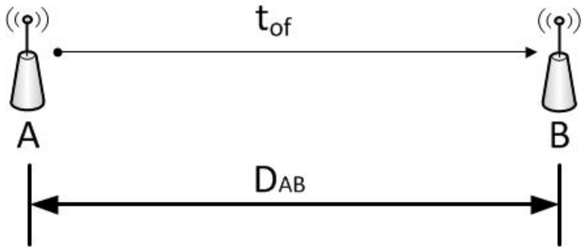


FIGURE 3.1: Basic Time of Flight Principle

$$\mathbf{D}_{AB} = \mathbf{c} \times \mathbf{t}_{tof} \tag{3.1}$$

3.2 System Requirements

Accuracy and latency are the desired requirements for ranging systems. For the real-time tracking system that we consider, where the system obtains the measurements, runs the tracking algorithms and produces the estimates in real-time, the amount of time that is spent in obtaining the observations is of particular importance. It is critical to ensure that the ranging observations are produced and fused, to the tracking algorithm, within an amount of time that allows for an acceptable sampling interval to be maintained. In other words the ranging scheme should not cause the overall sampling interval to exceed certain limits. Simulations in Chapter 5 justify that the system's accuracy deteriorates significantly when the sampling interval increases. Thus the ranging data from the anchor nodes must be collated and fused in a timely manner that will not jeopardize the overall performance of the system. Point-to-point ranging accuracy is defined, in terms of the difference between the estimated distance between two nodes and the true distance between them. Different locationing applications pose different specifications in terms of the tracking accuracy required.

For the tracking system that we consider, where we intend to track target's moving within the boundaries of human gaiting speed, an accuracy between 2-3m and and a sampling interval of no more than 2s are deemed to be sufficient.

3.3 Methods of performing time of flight ranging

Following the analysis provided in [76], this section reviews the different methodologies that have been proposed in order to obtain the ToF period between a pair of wireless communicating nodes.

3.3.1 One way time of arrival

This method requires the two nodes to be synchronised under a common clock system. As shown in Figure 3.2, node A sends a message at time $\mathbf{t} = 0$ and at the same time node B starts a clock which stops when B receives the message from node A. The value captured by the timer in node B equals to the $\mathbf{t}_{tof}$.

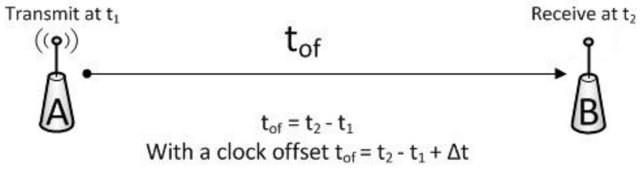


FIGURE 3.2: One way time of flight with synchronised nodes

3.3.2 Full duplex two way time of flight

In this case, nodes A and B are equipped with full-duplex radios, meaning that they are capable of receiving and transmitting at the same time. Hence, as shown in Figure 3.3 node A initiates the ranging transaction by transmitting a ranging signal at frequency $\mathbf{f}_{c1}$ and also captures the value of a timer (time $\mathbf{t}_1$). Node B receives the signal and immediately replies by transmitting a similar ranging signal at frequency $\mathbf{f}_{c2}$ while it continues to receive the primary signal. Node A, will receive the signal sent by node B at time $\mathbf{t}_2$. The $\mathbf{t}_{tof}$ is derived by subtracting the two values and dividing the result by two, $\mathbf{t}_{tof} = \frac{\mathbf{t}_2 - \mathbf{t}_1}{2}$.

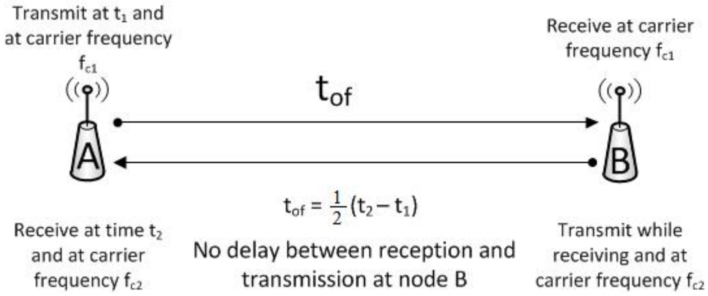

Transmit at t_1 and at carrier frequency f_{c1}
Receive at carrier frequency f_{c1}
t_{of}
A
B
$t_{of} = \frac{1}{2}(t_2 - t_1)$
Receive at time t_2 and at carrier frequency f_{c2}
No delay between reception and transmission at node B
Transmit while receiving and at carrier frequency f_{c2}

FIGURE 3.3: Full-duplex two way time of flight

3.3.3 Half duplex two way time of flight

Usually WSNs nodes do not have full duplex radios. However the two way time of flight method can be adapted for half duplex radios. This is of particular importance since a number of well known and widely used wireless system are based on half duplex communication (WiLAN, GSM). The amended two-way time of flight method is known as two-way time transfer and is depicted in Figure 3.4. While in the previous two methods only one node was tasked with running a timer and capturing time instances (node B in the first and node A in the second method), in the two way time transfer method both nodes must run their individual timers and capture the value of the timer upon receiving or transmitting a ranging signal. Since there is no full duplex support, all the communications take place on the same carrier frequency. At the beginning, node A sends the first ranging signal and captures the time of its timer ($\mathbf{t}_{tAB}$). Node B receives the signal at time ($\mathbf{t}_{rAB}$). After a period of time, that corresponds to node B swapping its state, from receiver to transmitter node B replies by sending a ranging signal back to node A. Normally, node B captures the time when the reply signal was transmitted ($\mathbf{t}_{tBA}$). Following, node A receives the reply signal and stores the time of reception ($\mathbf{t}_{rBA}$).

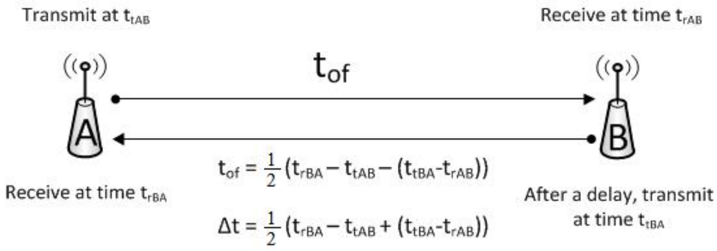

Transmit at t_{tAB}
Receive at time t_{rAB}
t_{of}
A
B
$t_{of} = \frac{1}{2}(t_{rBA} - t_{tAB} - (t_{tBA}\text{-}t_{rAB}))$
Receive at time t_{rBA}
After a delay, transmit at time t_{tBA}
$\Delta t = \frac{1}{2}(t_{rBA} - t_{tAB} + (t_{tBA}\text{-}t_{rAB}))$

FIGURE 3.4: Half-duplex two way ranging

Having in mind that the four required time instances are available and $\mathbf{t}_{tAB} < \mathbf{t}_{rAB} < \mathbf{t}_{tBA} < \mathbf{t}_{rAB}$. The timer in node A measures $\mathbf{t}_A = \mathbf{t}_{rBA} - \mathbf{t}_{tAB}$, and in node B $\mathbf{t}_B = \mathbf{t}_{tBA} - \mathbf{t}_{rAB}$, the required $\mathbf{t}_{tof}$ is derived from the following relationship:

$$\mathbf{t}_{tof} = \frac{1}{2}(\mathbf{t}_A - \mathbf{t}_B) \tag{3.2}$$

the clock offset $\mathbf{\Delta t}$ between the two nodes which is used in order to synchronise the two clocks can also be calculated from the following:

$$\mathbf{\Delta t} = \frac{1}{2}(\mathbf{t}_A + \mathbf{t}_B) \tag{3.3}$$

3.4 Sources of Ranging Error

The achievable accuracy of any RF-ToF ranging system is primarily limited by the following four factors:

1. corrupting noise and interference
2. clock synchronisation and accuracy
3. multipath channel effects
4. timing inaccuracies

The above listed factors, introduce random, temporally and spatially random errors into the range estimate. Various methods have been developed in order to mitigate the effects of these factors.

3.4.1 Corrupting Noise

Noise as well as interference, are two major factors that can cause the accuracy to degrade. For example, noise can cause the receiver to detect signals in the wrong time leading to faulty measurements. The effect of noise in RF-ranging methods can be quantified with the use of the Signal-To-Noise Ratio (SNR) on the receivers side and the occupied bandwidth (B). These measures are linked via the Cramér-Rao Lower bound (CRB). The CRB is a measure which is used extensively in estimation theory and statistics and expresses a lower bound on the variance of a parameter estimator. For one-way ToF ranging systems the CRB has been derived and is given by the following relationship [125].

$$\sigma_{tof}^2 \geq \frac{1}{(2\pi B)^2 E_s/N_0}\left(1+\frac{1}{E_s/N_0}\right) \tag{3.4}$$

The E_s/N_0 ratio is the energy per transmitted symbol divided by the noise power which is available in the communication channel. This ratio is known as the Signal-to-Noise ratio (SNR). By using this and considering that by averaging n multiple transactions, regarding them as independent identically distributed random variables with variance var , reduces the variance to var/n the CRB becomes:

$$\sigma_{tof}^2 \geq \frac{1}{(2\pi B)^2 \cdot SNR \cdot n} \tag{3.5}$$

where: $SNR = E_s/N_0$

The above derived relationship is valid for one way ToF estimation. In a two-way time transfer system the Cramer-Rao lower bound will simply be the average of the two time measurements. Hence:

$$\sigma_{tof}^2 \geq \frac{1}{2(2\pi B)^2 \cdot SNR \cdot n} \tag{3.6}$$

Subsequently the lower bound of the distance error is given as the product of $c \cdot \sigma_{tof}$

3.4.2 Clock synchronisation

Clock synchronisation is a key aspect in every ToF system. Obviously the times of transmission and reception of wireless signals must be known using a common time base in order to deduce accurate measurements. Clock synchronisation is of particular importance for ToA ranging systems where both nodes that participate in the ranging system must be accurately synchronised in order to obtain the $\mathtt{t}_{tof}$. The two-way time transfer method exhibits an advantage over the one-way method. Node A measures locally the time that the transmitted signal requires to get back (to Node A) after being retransmitted from Node B. Node B must measure the time that the signal spends on node B. This method therefore, has no real need for clock synchronisation since timing takes place locally.

3.4.3 Multipath channel effects

ToF systems can also be affected from multipath propagation. Multipath interference typically occurs, because the transmitted signal bounces off objects in the environment, and then adds to the LoS signal. Consequently, the LoS signal can be severely attenuated which may result in the signal being incorrectly received or lost completely. The error caused by multipath interference is difficult to quantify as it depends upon the deployment environment.

3.4.4 Timing Inaccuracies

Apart from the sources of ranging error that were analysed previously a number of additional uncertainties may add non-deterministic delays that will result in distorted timing of the two-way round trip timing value. A thorough analysis of these uncertainties is performed by Maróti *et. al* [80] in their work on synchronization techniques. One must also consider that an additional factor of uncertainty will be the drift over time that the clock oscillator on the embedded node will demonstrate. The output frequency of the node's clock is susceptible to drift and is affected by the surrounding temperature and the node's supply voltage. It is therefore, not uncommon to observe different latencies even on the same hardware. Additional timing uncertainties may incur from the node's radio operation during the transmission and reception of packets. These uncertainties are influenced by factors such as the message length, the interrupt handling and channel availability. It is imperative to ensure that the effect of these errors will remain as constant as possible in the implementation of the proposed ToF system in order to avoid erroneous timing of the two-way message transmission that will result in diminishing ranging accuracy.

3.5 The proposed method

3.5.1 Two way time of flight

This section provides the details of the developed ranging method for the purposes of this research project. As pointed out earlier the range-only nature of the proposed WSN tracking system demands an accurate, minimum-latency ranging method to be developed in low power embedded WSN nodes.

The proposed ranging method is inspired by the work presented by Thorbjornsen *et. al.* in [121]. Our intention is to evaluate the two-way ToF ranging technique and ultimately incorporate it in the range-only tracking system which is presented in the following chapter. The approach presented here, attempts to achieve better resolution in timing the value of the two-way message exchange by employing a different method on how the timer's value is captured. Instead of detecting a received message by sampling the receiver with a constant sampling rate, the receiver is programmed to signal an interrupt whenever a ranging message has completed a two-way path. The interrupt routine is then used to capture the value of the running timer. This approach results in better resolution of the two-way timing values, thus achieves better resolution in the resulting distance estimation by processing multiple two-way transactions between the participating nodes.

The basic concept of the proposed two-way ToF ranging system is illustrated in Figure 3.5. The objective is to estimate the distance between node A and node B. Initially node A sends the first ranging signal and captures the time of its timer ($\mathbf{t}_{tAB}$). Node B receives the signal and after a period of time, that corresponds to node B swapping its state, from receiver to transmitter (as well as a number of other delays) node B sends a ranging signal back to node A. Following, node A receives the reply signal and stores the time of reception ($\mathbf{t}_{rBA}$). The timer in node A measures $\mathbf{t}_A = \mathbf{t}_{rBA} - \mathbf{t}_{tAB}$ multiple times. Instead of using a clock at node B to measure the time that the signal spends on the node, our approach is to measure all the delays that occur during this two-way signal exchange process. This is accomplished by placing the transceivers at a minimum distance ($< 0.2m$) and executing multiple transactions that are averaged to produce the minimum time($\mathbf{t}_{min}$) that is required in order to complete a message exchange. This time corresponds to a minimal ToF period and reveals all the hardware and software delays that occur during a two-way ranging transaction. We make the assumption that the these delays remain constant and are independent of the distance between the nodes. Subsequently only the propagation delay will increase the two-way time transfer value as the nodes are placed at greater distance.

Figure 3.6 illustrates a timing diagram of a message exchange between the two nodes. Send and receive occurs on the rising edge of the nodes clocks. Assuming that for a given distance the t_{ToF} period will be the same and the delay T_{B_proc} that node B requires to process the ranging signal and submit the reply is constant, then the only ambiguity will be inserted by the delays associated to the clocks phase shift and frequency drift. Given that the two clocks are

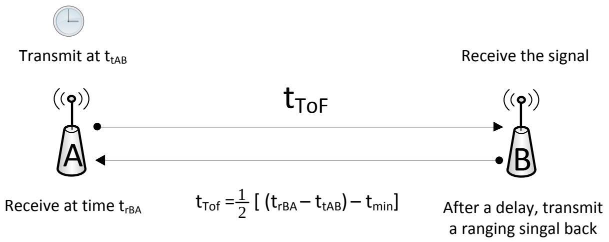


FIGURE 3.5: Proposed Two-way ToF Ranging

unsynchronized and have a small difference in frequency the phase offset between the devices will oscillate, thus the delays T_{d1} and T_{d2} will follow a similar varying pattern. By oversampling, we capture a normally distributed set of multiple timing transactions centered around the mean ToF value. Subsequently, capturing a sufficiently large number of timing values allows us to extract the mean ToF value from the Gaussian distribution which can then be, linearly associated to the distance between the nodes.

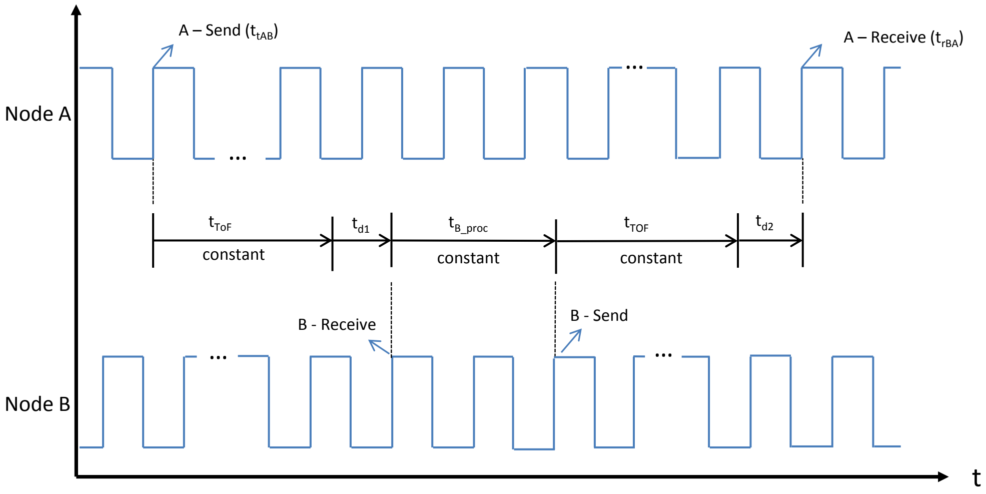


FIGURE 3.6: Timing Diagram of a two-way message exchange

Due to the timing inaccuracies analysed in Section 3.4.4, we expect the ToF values to vary for a set distance. During the calibration stage the additional delays introduced by these factors must be sufficiently captured in order to be excluded from the ToF values. To achieve this, the combined delays which are introduced by these factors must remain as constant as possible during any experimental set-up. By oversampling the ToF values sufficiently, the errors that are associated to the timing uncertainties are averaged out and do not affect the mean calculated averaged measurements given that the calibration value is removed.

3.6 Implementation of the two-way time of flight ranging method

3.6.1 Preliminaries

This section provides the details of the implementation of the ToF-ranging technique on the EZ430-RF2500 hardware.The approach that was followed was along the direction of satisfying the paramount requirements in regard to the ranging system, namely latency, accuracy and robustness.

3.6.2 Timing and radio settings

Latency and robustness are two interconnected characteristics since trying to maximize one of them may result in reducing the other. Latency refers to the amount of time that is required in order to complete a ranging estimation between the two nodes. Since multiple ranging transactions are required in order to produce a single range estimate, it must be ensured that the ranging transactions as well as all the mathematical calculations are completed within an acceptable time period. The majority of the time is spent in the exchange of ranging messages between the two nodes. Since the packet's payload is predefined and kept as simple as possible (a small number of characters that act as verification upon reception) the most important factor in this, becomes the node's communication data rate. High data rates ensure that the message exchange between the two nodes takes place in a timely manner. Conversely high data rates may affect the system's robustness, due to the fact that a high data rate may result in a system that is more susceptible to transmission errors. On the other hand, robustness affects latency in a positive way since a robust system has a high successful packet reception rate, which translates to, very few retransmissions required in order to reach the nominal number of ranging transactions.

The radio parameters were configured in order to obtain the maximum sensitivity of the radio thus reducing the transmission errors. The data rates were 250kbps and 500kbps. The transmission power was set to the highest value possible, that is +1dBm. The modulation used is minimum-shift keying (MSK), the preamble length was set to 2 bytes and the sync word to 4 bytes. Finally sync word detection was set to 30/32 bits. These settings were used following T.I suggestions.

To achieve the maximum possible accuracy in timing the two-way ToF of a signal the maximum possible clock frequency is used. The mcu's timer A is sourced at the DCO which is set at the maximum possible clock frequency at 16 MHz. Timing extraction of the ToF is accomplished with the use of a GDO pin, which is set to go high whenever a SYNC WORD is transmitted or received, at the end of the sync word the pin goes back to "low". In details, in the developed software, the GDO pin is programmed to signal an interrupt in the event that it changes state from low-to-high. By using the interrupt, timer A gets reset whenever a SYNC WORD of a ranging message is transmitted and its value is captured whenever a SYNC WORD is received (assuming that comes from the other device that takes place in the ranging procedure). A software binary variable acts as a lock in order to avoid unwanted triggering of the interrupt. This method avoids the need for sampling the pin with a predefined rate, since the GDO pin itself triggers an interrupt which results in the timer's value being captured. Moreover by employing this method we avoid any possible software delays since the timer's value is captured directly from the hardware register before the radio message is completely received.

3.6.3 Software and Ranging Algorithm

As mentioned earlier, the two-way ToF ranging is performed between a pair of EZ430-RF2500 devices programmed independently with different software. One of them is termed the *requester* and the other the *responder*. The *requester* device is the device that initiates the sequence in order for the two devices to engage in exchanging the necessary ranging messages. Practically, the "requester" device controls the initiation and termination of the ranging technique.

To begin with, in the *requester* device a slow clock (12 KHz) sourced at Timer B, triggers the initiation of the entire process. The slow clock is set to have a period which is longer than the time that is required to complete a ranging operation, meaning the exchange of the nominal number of ranging packets between the two nodes and the extraction of the average time. When Timer B fires, the *requester* device sends a "request to send" packet and waits for the *responder* to reply . This procedure is repeated twice to ensure that the communication link between requester-responder is established successfully. Following, the requester begins the transmission of the first ranging packet and also resets the value of the 16 MHz timer (after the transmission of the SYNC WORD). Immediately after the transmission is completed, the requester switches its status to receiver and waits for a return packet from the *responder*. Upon, a successful reception of a ranging packet by the *responder*, the *responder* device checks to verify that the received packet is a correct ranging packet and then (while also swaps status from receiver to transmitter) transmits back a ranging packet at the *requester*.

On reception of a ranging packet at the *requester* following previous transmission of a ranging packet, the GDO pin triggers an interrupt (when the SYNC WORD of the incoming packet is correctly detected) which captures Timer's A value which corresponds to the two-way ToF. When the full packet is received, it is checked for correctness and if is found to be ok then Timer's A captured value is stored. The ranging transaction counter is incremented and the next cycle of ranging transmissions begins. This two-way packet exchange process is repeated until the nominal ranging transactions number is reached. The *requester* device then enters the calculation phase. In the event that a false packet has triggered the SYNC WORD interrupt the captured value is disregarded.

The calculation phase involves the extraction of the ToF out of the stored timer values. The majority of the approaches in the literature that include multiple measurements usually average the ToF value of "n" two-way roundtrip transactions. Nevertheless, in the event that one or in general a small fraction of these "n" transactions has produced erroneous timing, by including it in the average it will alter the correct value to a wrong direction. To avoid this, and to assure that the ToF calculation is based on the most accurate and "true" transactions a procedure is followed, where the mean and the standard deviation of the "n" ToF values is initially calculated.

Let us assume that we obtain "n" two-way ToF values $\mathbf{t}_n$(expressed in clock cycles of the 16MHz DCO clock): The initial average mean $\widetilde{\mathbf{t}}_{tof}$:

$$\widetilde{\mathbf{t}}_{tof} = \frac{1}{n}\sum_{i=1}^{n}\mathbf{t}_i \tag{3.7}$$

while the standard deviation of the n timing values is given from the following:

$$\sigma_{tof} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(\mathbf{t}_\mathbf{i} - \widetilde{\mathbf{t}}_{tof})^2} \tag{3.8}$$

In the following step we calculate the absolute difference of each on of the "n" values from the initial mean:

$$\mathbf{t}_{n\_diff} = abs(\mathbf{t}_n - \widetilde{\mathbf{t}}_{tof}) \tag{3.9}$$

Ultimately, out of the "n" collected ToF values we exclude the ones that their absolute difference to the initial mean is greater than the standard deviation. The final ToF $\widehat{\mathbf{t}}_{tof}$ value is calculated by averaging the remaining "m" values as shown below:

$$\widehat{\mathbf{t}}_{tof} = \frac{1}{m}\sum_{i=1}^{m}\mathbf{t}_m \tag{3.10}$$

The final ToF averaged value is refined from all the values that might be erroneous.

After the final estimate is produced. The program resets all the variables and waits for Timer B to fire the next time in order for the same procedure to be repeated.

3.6.4 Conversion to distance

Obtaining the two-way ToF is the main step in estimating the range between the two nodes. That value is converted to distance by executing the following.

1. Calibrate the $\widehat{\mathbf{t}}_{tof}$ value by subtracting it from the minimum two-way ToF. The minimum two-way ToF is obtained when the two nodes are placed with zero-distance between them. That timing corresponds basically to a minimal ToF period and reveals the delays that the two nodes require in order to complete a ranging transaction

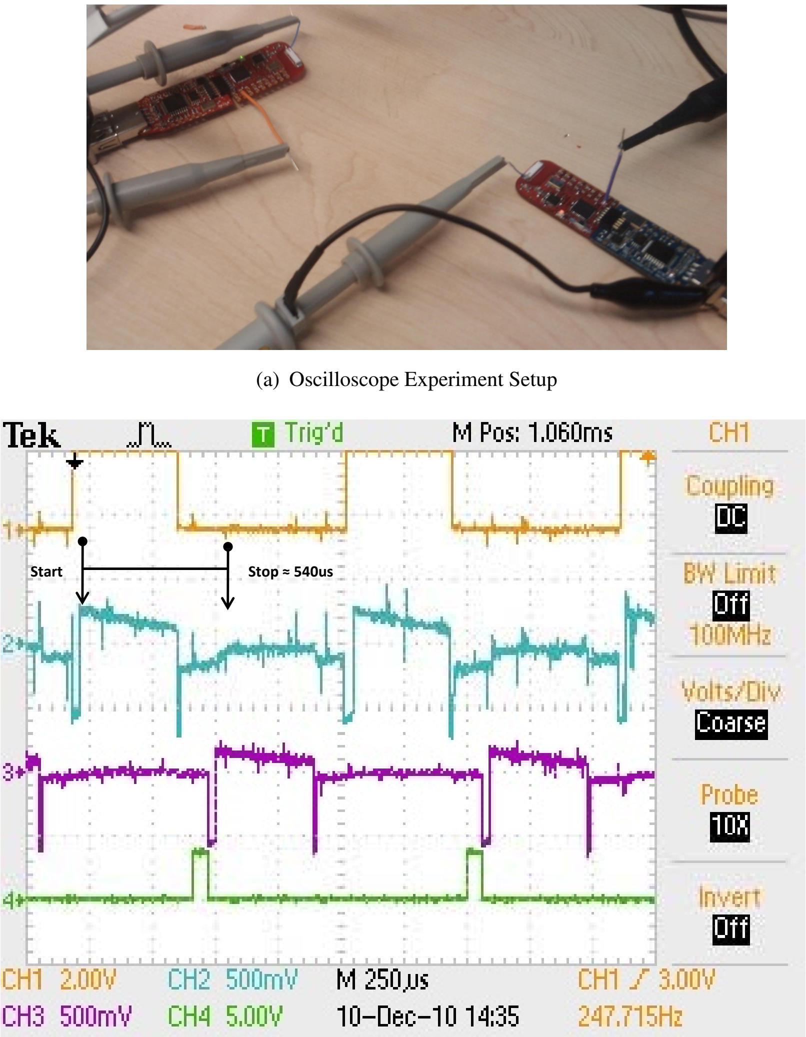
2. Divide the calibrated value by two, to get a single-way ToF time

3. Multiply the above with the speed of light to convert time to distance. The value of the speed of light used was adjusted according to the refractive index of visible light (n = 1.003), $v_{air} = c/n$

3.6.5 Mitigating the effect of timing uncertainties

Additional precautions were taken to minimize the effect of uncertainty sources during a single two-way transaction. As pointed out in [80], various factors affect the uncertainties in a message transmission. Through our implementation we tried to maintain these uncertainties as constant as possible. Constant packet length was used to avoid varying transmission/reception times. The clear channel assessment option was not used as we assumed that there was no contention in accessing the channel during the experiments. An important source of delay that we had to tackle, is the amount of time the *responder* requires in order to acknowledge a correct ranging signal and reply accordingly. To guarantee a constant response time on the *responder's* side, we used a minimal static code routine specifically for this application. All other interrupts were disabled apart from the one associated to message detection. To evaluate the *responder's* reply delay, we used the same 16MHz clock to capture the time from the moment a packet is detected at the *responder* until the reply message is transmitted back. This delay, which includes the $9.6\mu s$ that is required for the transceiver to change state from Rx to Tx, was found to be constant during the exchange of ranging signals. Nevertheless one of the latencies that we were not able to address pertains to the interrupt handling. In essence, we assume that the moment an interrupt flag is raised from the radio, to signal the reception of a ranging packet, the MCU starts responding to that interrupt accordingly. However in reality there might be a sub-clock delay between the signalling of the interrupt by the radio and the MCU's response, due to the fact that the two components operate on different clocks. An approach that could mitigate these effects is to drive both the MCU and Radio from the same clock source.

To evaluate the delays associated with the timing between the nodes, the code on the *requester* node was altered to set a pin high immediately after the Send Packet command was strobed to the CC2500 Radio, and the code on the *responder* was altered to set a pin "high" immediately when a packet was received. Two small connections where then soldered to the transmitting and receiving antennas of the devices, and a Tektronix TDS2014 Four Channel Oscilloscope was then connected to the transmitting and receiving nodes. This is visualized in Figure 3.7(a). In Figure 3.7(b), Channel 1 and 2 of the Oscilloscope represent the MCU pin set "high" immediately after the transmit packet command was strobed; and the signal transmitted on the Antenna respectively. Channel 3 and 4 of the Oscilloscope represent the *responder's* antenna signal and the pin set high on successful reception of a packet. From the timing analysis it can be seen that the transmit to receive signal on the MCU takes approximately 540μs (at 250kbps) which corresponds to 8640 counts of an accurate 16MHz clock. From the actual experiments the values that are produced are very close to that number (8670 counts) for a different pair of nodes placed on similar distance apart. We thus assume that our timing values measured at the requester device correspond well to the total time measured by the oscilloscope.



(a) Oscilloscope Experiment Setup

(b) Oscilloscope Channels

FIGURE 3.7: Investigation of the timing uncertainties

3.7 Experimental Results

The two-way ToF method analysed previously was tested on field experiments in order to evaluate the ranging precision and overall performance of the method. The experimental setup consisted of a pair of ez430-rf2500 wireless nodes programmed with the "requester" and "responder" codes respectively. The ideal environment for this type of experiment is an obstacle free, plane area with good line-of-sight (LoS) for the two nodes. In addition, the interference from other wireless systems must be as low as possible. Since the CC2500 radio transceiver operates on the 2.4GHz band, it is expected that a number of other wireless networks, like WLAN, will cause significant interference in areas (ex. University campuses) where such networks are present. Experiments were carried out on three sites.

The first site is a level grass field at the University of Southampton campus where surrounding buildings could be a reason for multipath propagation and a number of WLAN university networks might cause interference. In this site the maximum communication range between the two nodes was limited because there was no more space available than 42m in LoS condition.

The second site is at the University of Southampton Wide Lane sports complex (Figure 3.8). Nodes were deployed on a grass field with no obstacles close to the experimental set-up. The maximum range between the two nodes that allowed the ranging system to run adequately was 70m in good LoS.

A number of experiments took place indoors. A narrow corridor at the Electronics Systems and Devices group of ECS school was used in order to test the ranging method indoors. The hallway was 40m long and had a maximum width of approximately 2m and minimum of 1.7. This environment is considered to be prone multipath effects, thus a decrease in accuracy should be expected.

3.7.1 Experimental Setup

The two EZ430-RF2500 devices were strapped on two wooden chairs in order to elevate them from the ground. The elevation was 90 cm off the ground. Both nodes were powered on from laptops (using the USB dongles) to ensure that they operated with full power supply. The laptops were also used in order to log the ranging data from the “requester ”node via its USB port. The transmission power was set at the maximum possible value of +1dBm. Two datarate settings were used for the node’s radio in these experiments at 250kbps and at 500kbps. In general the 250kbps is the nominal datarate and thus is the one used in the majority of the experiments.

Due to the ez430-rf2500 design the antenna orientation plays a significant role on the maximum communication range. We concluded that the best antenna orientation was with the two antennas facing each other and being slightly inclined at an angle from the vertical position, towards the ground. Of the two nodes the “responder” was positioned once at the beginning of the experiment, while the “requester” was positioned in different positions. Ranging data were collected from the “requester” in steps of 3m until the maximum possible communication range was reached. A tape measure was used as reference and in order to measure the “true” distance between the two nodes. Initially the reference two-way ToF was estimated by placing the two node at a minimum distance ($< 0.3m$) and averaging 100 two-way transactions. In these experiments 1000 two-way transactions were used to estimate the distance between the two nodes. Due to the inability of the EZ430-RF2500 hardware to store 1000 float values at the same time, the *requester* node performed the necessary calculations (compute std, exclude transactions, compute the average) whenever 100 transaction were achieved. The results were then forwarded to the its serial port and a new cycle of two-way ToF begun. The operation was terminated, when the *requester* had forwarded the results from 1000 two-way values (10 final estimates) from this location. Then the node was moved to the next location.

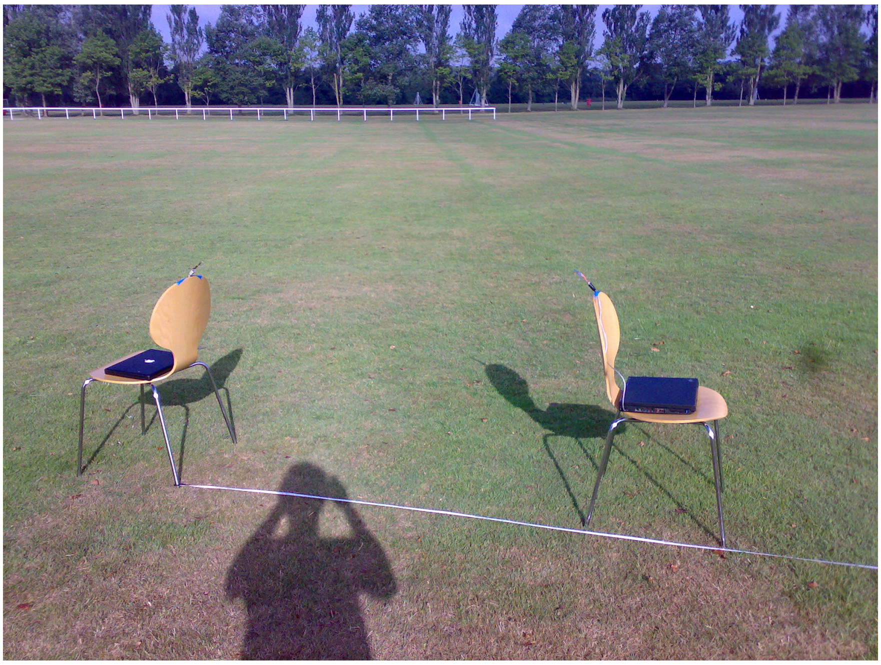


FIGURE 3.8: Experimental Setup for the ToF ranging experiments

The metric used to evaluate the system's accuracy is mainly the RMS error which is defined as follows. Assuming n different distances.

$$d_{rms} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(\mathbf{d}_{real} - \mathbf{d}_{esti})^2} \tag{3.11}$$

In Figure 3.9, Figure 3.10 and Figure 3.11 results from two different days of experiments at the University of Southampton campus site are presented. Figure 3.9 and Figure 3.10 illustrate results with the datarate set to 250kbps while Figure 3.11 corresponds to an experiment executed withe the node's datarate set to 500kbps.

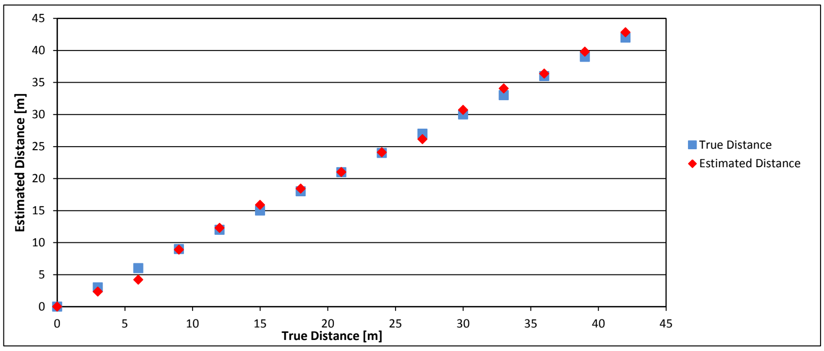


FIGURE 3.9: True Ranging vs Estimated Ranging- RMS error = 0.75m, Maximum Error 1.79m

Results from the same site with the data rate set to 500kbps are illustrated in Figure 3.11

Results from an experiment carried out at the University of Southampton Wide Lane sports

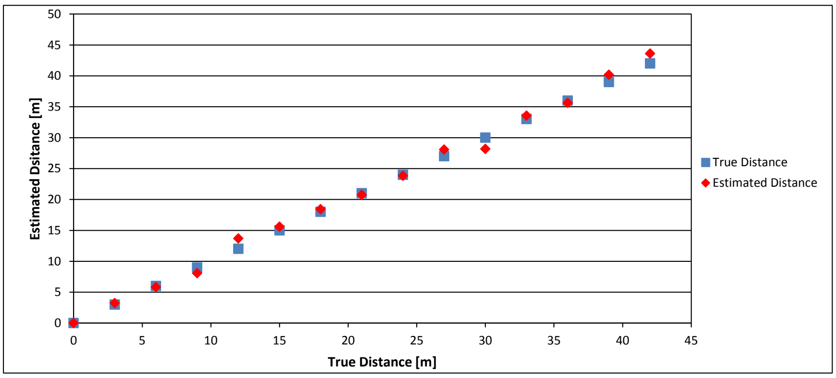


FIGURE 3.10: True Ranging vs Estimated Ranging- RMS error = 0.94m, Maximum Error 1.84m

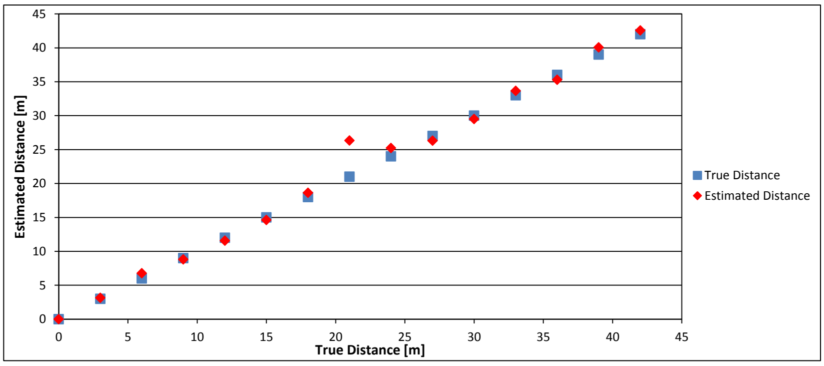


FIGURE 3.11: True Ranging vs Estimated Ranging- RMS error = 1.51m, Maximum Error 5.32m

complex is shown at Figure 3.12. At this site the maximum communication range that was achieved was 70m using the 250kbps datarate setting.

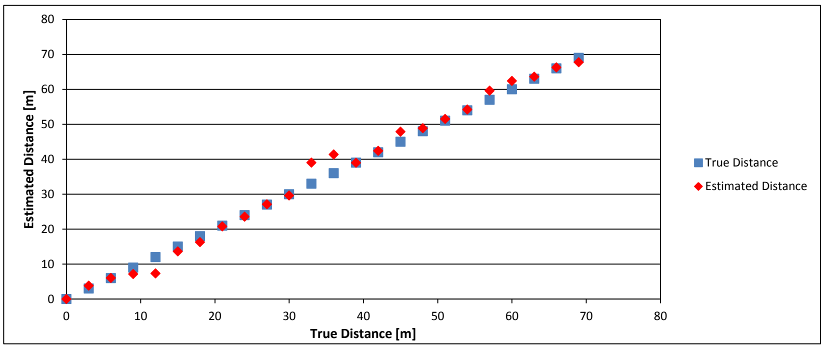


FIGURE 3.12: True Ranging vs Estimated Ranging - RMS error = 2.23m, Maximum Error 6m

Finally results from the indoor location are illustrated in Figure 3.13 and Figure 3.14. In Figure 3.14 the datarate used was 500kbps while in Figure 3.13 the datarate was set to 250kbps.

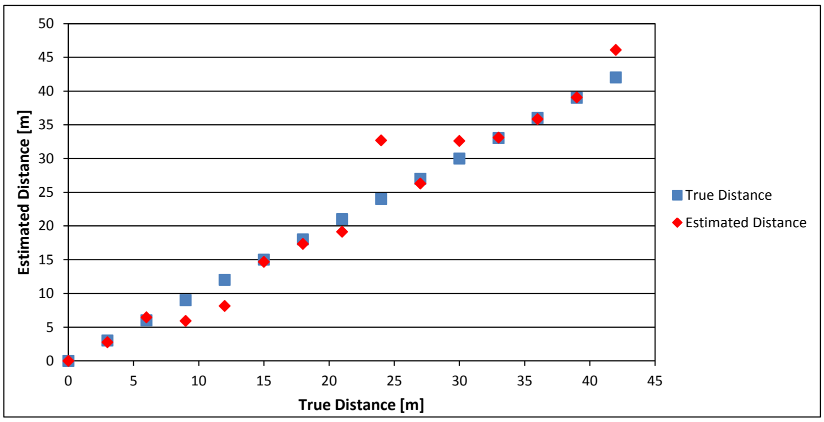


FIGURE 3.13: True Ranging vs Estimated Ranging in indoor environment- RMS error = 2.92m, Maximum Error 8.67m

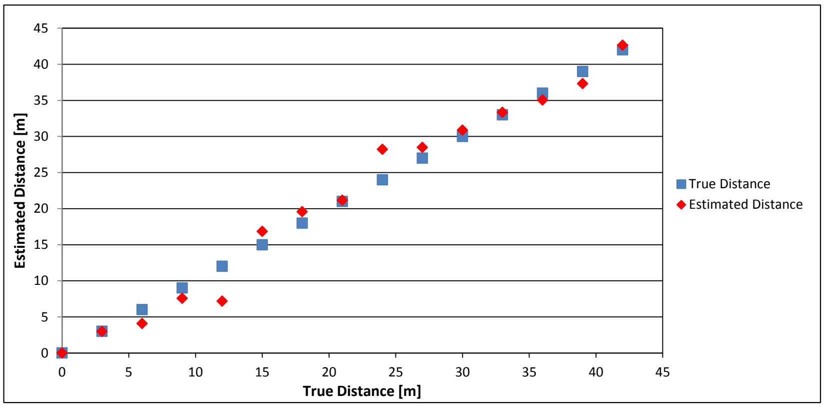


FIGURE 3.14: True Ranging vs Estimated Ranging in indoor environment- RMS error = 2.2m, Maximum Error 4.82m

3.7.2 Results Analysis

By investigating the results obtained in the three locations, we deduce that the overall performance of the ToF method is quite promising. By employing 1000 ranging transactions an RMS error between 1-3m is achieved for both datarates (250kbps,500kbps) considered. As expected results indoors are worst than outdoors for the same range of distances (42m). An increase of approximately 1.5m in the RMS error is observed between indoors and outdoors deployments. Results obtained on the campus site with the same datarate setting but on different days are

Ranging Results			
Location	Datarate	RMS error	Maximum Error
Location 1	250kbps	0.75m	1.79m
	250kbps	0.94m	1.84m
	500kbps	1.51m	5.32m
Location 2	250kbps	2.23m	6m
Location 3	250kbps	2.92m	8.67m
	500kbps	2.2m	4.82m

TABLE 3.1: Results from experiments of the proposed ToF method

consistent (see Figure 3.9 and Figure3.10). The difference of 24cm can be potentially attributed to the manual measurement of the true distance, the slightly different orientation of the node's antennas as well as the different environmental conditions between during the two days. Figure 3.11 illustrates results from the campus site obtained using the 500kbps datarate setting. One would expect the higher datarate to provide more accurate results. However, there is a very poor estimate at 21m in this experiment. This poor estimate is a result of a poor wireless communication link between the two nodes for this measurement. By excluding the estimate at 21m and using the rest 14 estimates the RMS error drops from 1.51m to 0.68m which is very close and slightly better than the one obtained with the 250kbps datarate.

Figure 3.12 illustrates the results obtained at the grass field outside of University of Southampton Wide Lane Sports Complex. The results obtained in this environment for the 250kbps datarate are in contrast to the expectations, worst in terms of accuracy from the ones obtained at the campus site. This is due to the fact that 3 positions out of the 23 were estimated with considerable error ($> 4m$). Different to the rest of the experiment the timing values for these three positions diverged from what the linear increase dictates. This behaviour is attributed to the possibility for the internal clock on the nodes to drift from the nominal value enough in order to result in poor ranging estimations. In the indoors deployment, the achieved accuracy is decreased, most probably due to multipath propagation. In both experiments (see Figure3.13 and Figure3.14) there are notable poor estimates at 12m and 24m. This is believed to be caused due to the structure of the corridor where the experiments took place. At 12m and 24m the corridor's wall had an opening (due to the presence of doors) that altered the quadrilateral geometry of the adjacent walls. As expected the higher data rate demonstrates improved accuracy.

3.7.3 Investigation of the clock drift

It is well known that clock oscillators are susceptible to drift from the nominal frequency. This effect can be intensified due to temperature and voltage supply fluctuations. Subsequently additional uncertainty is inserted in timing a two-way transaction. To measure the drift in clock frequency, we used a Hameg HM8123 frequency counter connected to a 10MHz SRS FS725 Rubidium Frequency Standard clock reference, and measured the clock frequency approximately every second over a period of 3 hours under room temperature and constant power supply. The HM8123 gating time was set to 100ms. We recorded the frequency from the HM8123 via a laptop's USB port. The results reveal that the DCO clock frequency is normally distributed with a standard deviation of 1.63KHz. The clock's accuracy is therefore in the region of 1% and the drift exhibits a standard deviation of 0.01% around the mean value. Due to this behaviour, an additional error of approximately 17cm per clock cycle will be inserted because of the clock's instability. This frequency distribution yields a distribution of the time values similar to the one illustrated in Figure3.17.

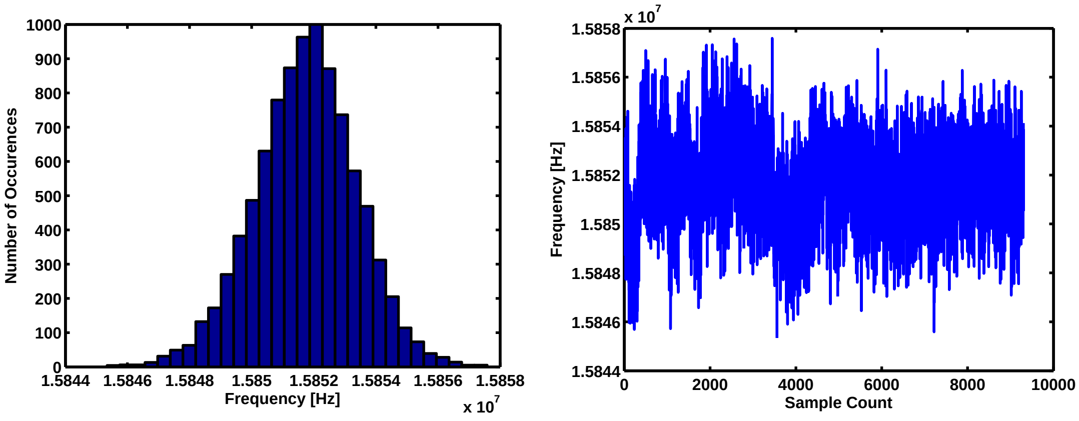


FIGURE 3.15: Investigation of the 16Mhz clock inaccuracy. Frequency Histogram (a); Frequency vs Time (b)

3.8 Comparison to theoretical expected performance

The timer that was used in the timing process is a 16MHz timer (maximum allowed value for the msp340 microcontroller). This value provides a resolution of $1/16MHz * c = 18.63m$. Given the system parameters, the theoretical lower bound of the variance in a two-way ToF system is given from the CRB in Eq3.6. Specifically, at 250kbps the CC2500 transceiver occupies 540KHz of bandwidth while at 500kbps 812KHz. Figure 3.16 illustrates the relationship of the SNR and the minimum achievable ranging error.

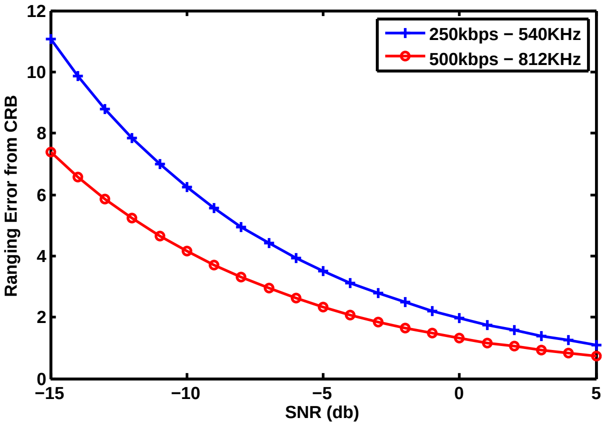


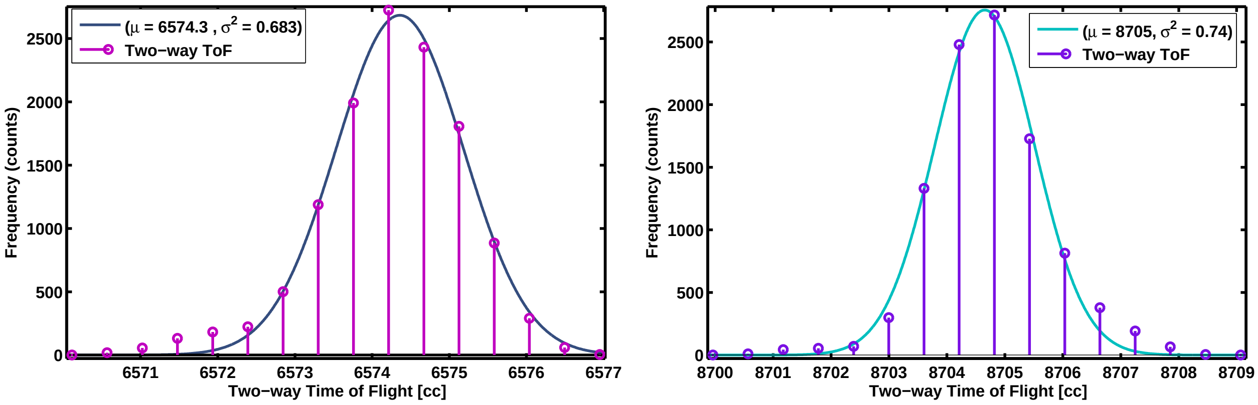
FIGURE 3.16: Minimum ranging error derived from CRB according to the EZ430-RF2500 hardware specifications

For example, considering a -5db SNR, the minimum variance of a timing value which is obtained after averaging 1000 measurements is $\sigma^2_{tof} = 137.3ns$ for the 250kbps and $60.7ns$ for the 500kbps respectively. These correspond to a minimum ranging error standard deviation of 3.5159m and 2.33m for each datarate respectively (calculated from $\sigma_{tof} \cdot c$).

As stated previously the node performed the necessary calculation whenever 100 two-way transactions were completed. Part of the process is the calculation of the standard deviation for these 100 transaction in order to exclude the timing values that fall outside the single deviant boundary. This procedure repeated itself 10 times in order to reach 1000 transactions. From all the experiments carried out the standard deviation of the timing values was initially in the range of $1.4cc - 1.8cc$ for the 500kbps setting and $2.4cc - 3cc$ when the 250kbps datarate was used. After averaging the values (excluding the ones outside the one deviant limit) the deviation was reduced to $0.3cc - 0.9cc$ for both the 500kbps and the 250kbps datarates. Assuming a value of $0.6cc$ and dividing this by two we get $\sigma_{ToF} = 0.3cc$. This value is expressed in clock cycles and a single clock cycle of the 16MHz timer is $(1/16MHz = 62.5ns)$. Thus the standard deviation of the proposed system can be approximated as $\sigma_{ToF} = 18.75ns$. This translates to a standard deviation of approximately 5.6m.

The deviation in the outdoors experiments was found to be smaller than the one indoors. Particularly for the 500kbps datarate setting the standard deviation that the two-way ranging values exhibit was approximately $0.4cc$ on average. This translates to a standard deviation of approximately $3.7m$ for the ranging estimates. This is of particular importance as this is the value that will be considered as the observation error distribution in the tracking experiments.

To also verify the distribution of the measurements that the proposed ToF ranging system yields, an experiment is designed where two nodes are placed at a short distance (2m) indoors and a vast number of ToF estimates is logged over a period of time. This experiment was executed twice with both the datarate values (250kbps and 500kbps). Approximately 10000 ToF estimations were logged. The values are plotted according to 15 equally spaced bins. From Figure 3.17 it is clear that the values can be considered as normally distributed around the mean value and exhibit a standard deviation which is very close to the one observed in the previous experiments.



(a) Histogram of 500kbps

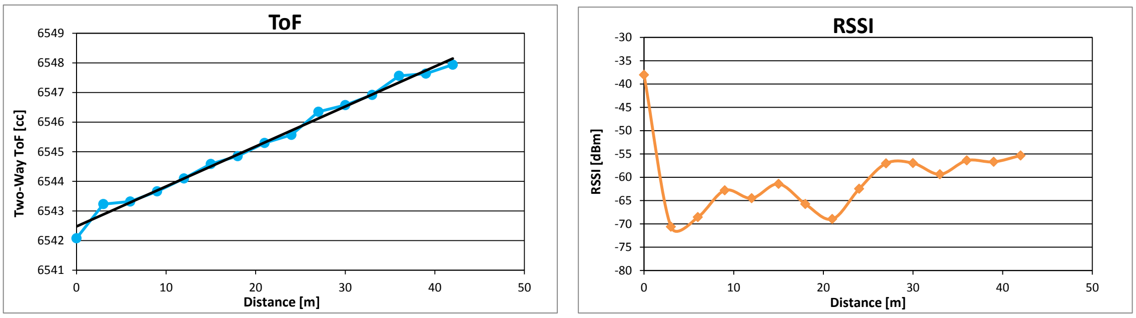
(b) Histogram of 250kbps

FIGURE 3.17: Timing Histogram of 10000 two-way values

3.9 ToF vs RSSI in EZ430-RF2500

This section provides a comparison between the two most well known techniques for estimating the range between two nodes in RF systems. The ez430-rf2500 offers the option of capturing the RSSI value of an incoming packet at the receiver. That option was used and in one of the experiments and the RSSI values of the ranging messages were captured from the RSSI register.

Calculation of the RSSI took place in a similar way like the ToF by averaging 100 RSSI values. Figure 3.18 illustrates the ToF values in clock cycles and the RSSI values in dBm. The RSSI values are converted to dBm with the following procedure for the CC2500 transceiver. After the value is converted from hexadecimal to decimal it is compared to ”128” and “if” ($RSSI_{value} > 128$) then the $RSSI_{dBm} = (RSSI_{dec} - 256)/2 - RSSI_{offset}$ else if ($RSSI_{dec} < 128$) then $RSSI_{dBm} = RSSI_{dec}/2 - RSSI_{offset}$. According to a number of previous works in the area, the RSSI value should follow an inverse power law in proportion to the distance between transmitter and receiver. It is clear from the graph that is difficult to establish an analytical relationship between the distance of the two nodes and the RSSI value. On the other hand the proposed ToF system, demonstrates linearity as expected.



(a) Distance vs ToF

(b) Distance vs RSSI

FIGURE 3.18: Distance vs ToF and Distance vs RSSI

3.10 Conclusions

This chapter presented a two-way RF-ToF method for ranging estimation in wireless embedded nodes. The multiple two-way transaction approach, achieves two major objectives. Firstly, it does not require the difficult task of synchronization among the participating nodes and secondly, amends the lack of fine resolution due to the low-frequency clocks that most WSNs are equipped with. The calibration method that is followed is effective since it caters for a number of delays difficult to be measured by using a single clock at the “reply” device. Sub-clock resolution is achieved by averaging multiple obtained time values. In addition, a simple yet effective procedure disposes any erroneous values that are present in the set of measurements.

The proposed ranging system is implemented on COTS hardware. It is therefore our belief that it can be implemented on different hardware platforms. Unlike other ToF ranging methods, the proposed system does not require any additional hardware. The entire procedure of obtaining and filtering the values as well as the calculation of the final ToF is completed on the nodes. Experimental results demonstrate an average accuracy of about 1m in outdoors deployments and about 2.25m indoors. In addition, through experimentation it was shown that the obtained two-way ToF ranging measurements are distributed in a Gaussian way around the mean (true) distance value with a standard deviation of approximately 5.6m. Particularly, for the settings (500kbps) and the environment (outdoors) where the tracking experiments were executed the observed standard deviation was approximately 3.7m.

Moving along the lines of implementing the complete tracking system in COTS hardware, the first step involved the design and development of a technique that can provide accurate enough ranging between wireless embedded nodes. The ranging method presented here is subsequently incorporated to the complete tracking system. The ultimate goal is to use the PF tracking algorithms, analysed in the following Chapters on the ranging data, obtained by the ToF-ranging method and produce the estimate of the target's kinematic variables.

Chapter 4

Theoretical Formulation of a Range-Only Tracking System

This chapter describes the theoretical work undertaken by the author in the area of algorithm design for implementation in a range-only tracking system for wireless embedded nodes. Section 4.1, provides the necessary background on representing a target tracking problem as a dynamical estimation problem using state-space representation. The fundamentals of Bayesian Estimation and Particle Filters are discussed in Sections 4.2 and 4.3 respectively. In Section 4.4 the mathematical formulation of the proposed range-only tracking system is provided. The two Particle-Filter based algorithms that were developed in order to be applied in the dynamic estimation problems, defined in Section 4.4, are discussed in Section 4.5. Finally, Section 4.6 concludes the chapter.

4.1 Fundamentals of Target Tracking Algorithms

4.1.1 Overview

In Section 2.1.1 the phases of a tracking system were discussed. This section focuses entirely on the tracking algorithm phase, which pertains to the processing stage of the acquired information in order to estimate the target's kinematics. The tracking algorithm is the mathematical operation that is employed by the tracking system to infer certain kinematic variables of the target (position, velocity, acceleration) based on observations that become available to the system.

The estimation of the target's kinematic variables, is based on the interaction between the target and one or multiple anchor points. The result of this interaction is some type of observations (measurements, data readings) that are useful in the sense that can be associated with the use of a physical or empirical law, to the targets kinematic variables. Hence, the associated data can then be processed appropriately with the use of an algorithmic procedure to produce the final estimation of the targets kinematics. For example the trilateration algorithm operates on three distance measurements to produce a unique set of coordinates. In tracking this algorithmic procedure is executed on each set of data made available from the target's interaction with the anchors. This can happen on regular or irregular (on request) time intervals, depending on the continuous or not nature of the tracking operation.

4.1.2 Problem Formulation in state-space representation

In mathematical terms, the problem of tracking a mobile object in real-time can be defined as:

"Sequentially estimating the state variables (state vector) of a dynamical system, using a sequence of noisy measurements made available to the system" [98]

The above definition considers the tracking problem as a dynamical system, modeled in a state-space manner. A state-space representation is a mathematical model of a physical system as a set of inputs, outputs (measurements) and state variables. The state variables are expressed as vectors and the evolution of the state-vectors in time is described by differential equations.

A state-space representation usually describes the system in the discrete time domain, where the time steps are defined as the time instances that observations become available. As a result the differential equations become equations of difference. The state vector is considered to contain all the vital information (in terms of variables) which describe the physical system. In the target tracking case, the state vector comprises some of the moving object's kinematic characteristics (e.g. position in terms of coordinates, velocity, acceleration). The measurements (observations), collected by the system are associated appropriately (mathematically) with the state vector, thus providing a mathematical model to infer the value of the state variables based on the batch of available observations.

The reason for opting to formulate the tracking problem as a recursive nonlinear state-space estimation problem is to reap the benefits of the two main approaches for tracking; tracking with respect to landmarks and dead-reckoning (see Section 2.1. These two approaches were analysed in Section 2.1. Dead-reckoning is used to formulate the mathematical model (state equation) based on which the state vector evolves in time. Instead of simply using this mathematical model to predict the state-vector (target's position, velocity) over time, observations with respect to landmarks obtained at regular time intervals, are associated to the state vector (observations equation) and used to refine the initial estimation produced exclusively from the state model. Adding to the mix the probabilistic nature of dynamical systems, since the final estimation is produced from a pdf with the use of a certain criterion (minimum mean square error), the resulting dynamical tracking system is more robust to noisy observations than both dead-reckoning and tracking with respect to landmarks (trilateration).

4.1.3 General Case

In the general case, the target tracking problem can be formulated as a discrete-time state-space representation model using the following set of nonlinear equations:

$$\mathbf{x}_k = f_{k-1}(\mathbf{x}_{k-1}, \mathbf{w}_{k-1}) \tag{4.1}$$

$$\mathbf{z}_k = h_k(\mathbf{x}_k, \mathbf{v}_k) \tag{4.2}$$

where :

- $\mathbf{k}$: is the discrete-time index
- $\mathbf{f}$ and $\mathbf{h}$: are nonlinear time-variant functions in the general case
  - $\mathbf{f}$: describes the evolution of the state vector in time
  - $\mathbf{h}$: describes the mathematical association of the measurements to the state
- $\mathbf{x}$: is the state vector
- $\mathbf{z}$: is the measurements vector
- $\mathbf{w}$ and $\mathbf{v}$: are additive noise distributions (that are known and can be sampled)

4.2 Bayesian Estimation Theory

Dynamic Bayesian Estimation is a general probabilistic approach for estimating an unknown probability density function recursively over time using incoming measurements and a mathematical process model.

In a state-space estimation problem, Bayesian Estimation can be used if we consider the state-space equations as the mathematical process model and the density function to be estimated, as the probability density function of the state vector at a certain time instance given the set of observations. An important assumption is that the state must evolve in time as a Markov process and the measurements should represent the observed states of a Hidden Markov Model (HMM). This means that the current state (at time k) depends only on the previous state (at time $k-1$) and also that the current set of measurements depends only upon the current state and not on any previous measurements. In principle, tracking systems satisfy the Markov assumption. The target's position at time $\mathbf{x}_k$ only depends on the motion pattern (state equation) that the target followed during the interval (k-1,k] and is independent of any previous, to k-1, states. Similarly the set of measurements $\mathbf{z}_k$ (considering the range) obtained by a number of observers at time k only depends on the target's position coordinates (state vector) at time k and again is independent of the target's previous states. Mathematically this is expressed in Eq.4.1 and Eq.4.2.

$$p = (\mathbf{x}_k|\mathbf{x}_{k-1}, \mathbf{x}_{k-2}, \ldots, \mathbf{x}_0) = p(\mathbf{x}_k|\mathbf{x}_{k-1}) \tag{4.3}$$

$$p = (\mathbf{z}_k|\mathbf{x}_k, \mathbf{x}_{k-1}, \mathbf{x}_{k-2}, \ldots, \mathbf{x}_0) = p(\mathbf{z}_k|\mathbf{x}_k) \tag{4.4}$$

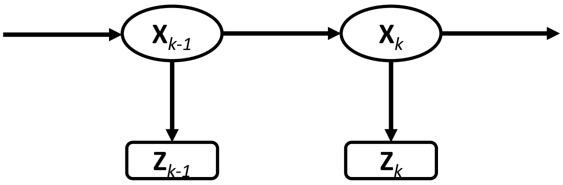


FIGURE 4.1: Hidden Markov Model. Arrows denote dependency

Bayesian Estimation Algorithms produce a recursive approximation of the probability density function (pdf) of the state variable $\mathbf{x}_k$ at a time instant k given the obtained set of measurements $\mathbf{Z}_k \triangleq \{z_i, i = 1, \cdots, k\}$ up to that time instant. This is termed as the posterior (after the observations have been included) pdf $p(\mathbf{x}_k|\mathbf{Z}_k)$ of the state vector. The approximation of the posterior pdf enables the estimation of the state-vector with the use of a specific criterion (e.g. minimum mean squared error).

$$\hat{\mathbf{x}}_{mmse} \triangleq \mathbb{E}\{\mathbf{x}_k|\mathbf{Z}_k\} = \int \mathbf{x}_k p(\mathbf{x}_k|\mathbf{Z}_k) d\mathbf{x}_k \tag{4.5}$$

The estimation algorithm considers the *initial* (before measurements become available) probability density function $p(x_0) \triangleq p(\mathrm{x}_0|\mathrm{z}_0)$ as known. Usually a Gaussian density function is selected to approximate the initial pdf. The desired posterior pdf can be obtained recursively using the system model and the following set of equations, in a two stage (predict-update) procedure [37].

- The Chapman-Kolmogorov Equation - Predict stage

$$p(x_k|Z_{k-1}) = \int p(x_k|x_{k-1})p(x_{k-1}|Z_{k-1})\mathrm{d}x_{k-1} \tag{4.6}$$

Then at time k a new measurement becomes available and an update to the predicted pdf is carried out via the Bayes' Rule

- Bayes' Rule - Update Stage

$$\begin{aligned} p(x_k|Z_k) &= p(x_k|z_k, Z_{k-1}) \\ &= \frac{p(z_k|x_k, Z_{k-1})p(x_k|Z_{k-1}}{p(z_k|Z_{k-1})} \\ &= \frac{p(z_k|x_k)p(x_k|Z_{k-1})}{p(z_k|Z_{k-1})} \end{aligned} \tag{4.7}$$

In case, one of the Eq. 4.1 and Eq. 4.2 or both is linear and time invariant then a Matrix replaces the $\mathbf{f}$ and/or $\mathbf{h}$ functions. In case both the state and update equations are linear the set of state-space equations becomes:

$$\mathbf{x_k} = \mathbf{A}\mathbf{x_{k-1}} + \mathbf{B}\mathbf{w_{k-1}} \tag{4.8}$$

$$\mathbf{z_k} = \mathbf{B}\mathbf{x_k} + \mathbf{D}\mathbf{v_k} \tag{4.9}$$

An example of a linear model for the state equation that has been proposed in order to describe the motion pattern of a moving object is the constant velocity model (details in 4.4.2.1). In case both the state and measurements equation exhibit linearity then the entire dynamical system is termed linear and the optimum Bayesian estimator for the state vector is calculated with the use of the well-known Kalman Filter [68].

However in situations where the state and/or measurements equation is nonlinear, the Kalman Filter is unable to solve the system effectively. In mathematical terms the problem lies on the inability to calculate the update integral of Eq. 4.7, in closed form, due to the nonlinear nature of the system. A source of nonlinearity, in tracking systems for example is the fact that the measurements equation both for range or bearings measurements is nonlinear. Hence regardless of the motion model employed in the state equation the resulting system will still be nonlinear.

Bayesian Estimation Theory has produced a number of alternatives methods capable of solving nonlinear dynamical systems, however without being able to provide the optimum Bayesian estimator but an approximation of it. Kalman Filters Derivatives (Extended Kalman Filter, Unscented Kalman Filter, Gaussian Sum Filters) provided that certain assumptions hold can be applied in nonlinear systems, producing however a suboptimal estimate [37, 98].

4.3 Particle Filters

The inherent inability of the above mentioned methods to deal effectively with nonlinear dynamical systems resulted in research efforts being conducted in the direction of providing solutions, capable to effectively solve nonlinear systems. The field of Dynamic Bayesian Estimation has been revolutionized during the past fifteen years due to the development of several prominent stochastic sampling algorithms which are collectively termed as Markov Chain Monte Carlo (MCMC) methods [99]. MCMC methods are simulation-based methods that have led to powerful numerical methods for computing posterior probability distributions [36]. As mentioned before, the state-vector estimates are derived from its posterior pdf. A class of MCMC methods that has attracted substantial interest over the past decade in the area of recursive Bayesian Estimation, is Particle Filters. Particle Filters (PF) need practically no assumptions regarding the dynamic system in order to be able to operate and produce an estimation of the posterior state pdf. Different to Kalman Filters that assume that the state's posterior pdf follows a Gaussian distribution, PF approximate the posterior pdf with a set of particles (samples) weighted appropriately. PF are also know by the term “Sequential Monte Carlo Methods (SMCM)” and are inspired by the principles of Importance Sampling and Monte Carlo Integration (analytical details regarding Particle Filter can be found on Appendix A) [37]. The flexibility due to the lack of assumptions, in this new approach for recursive Bayesian Estimation attracted significant research interest, in a wide area of applications where stochastic estimation of nonlinear dynamical systems is required [33,38]. An important aspect of PF estimation methods is the fact that, in general, they are computationally demanding methods. Hightower *et. al.* perform a case study investigating the accuracy of PF in tracking problems in connection to the required processing demands of these algorithms. The authors conclusion is that PF can be run practically on a number of devices [57,58].

The emergence of PF as an attractive alternative to existent methods led researches to consider PF for a number of tracking applications. In [12], the authors perform an overall review on the use of Particle Filters for non-linear non-Gaussian tracking. The authors analyse the different variations of the standard Particle Filter (bootstrap filter) and conclude by stating that for a nonlinear problem Particle Filters can prove to be able to provide the best suboptimal Bayesian estimator. Doucet *et al.* reach an identical conclusion and present a generic framework, where Particle Filters can be applied to a certain category of nonlinear systems. This category of systems is met in a number of application domains, one of them being target tracking [39]. Another framework specifically designed for positioning, navigation and tracking applications is presented in [53]. The authors consider seven different application scenarios, and propose and evaluate variations of the standard Particle Filter under these scenarios.

4.4 Modeling the proposed Target Tracking System

In order to model the proposed target tracking system, a state-space representation approach is employed. The state vector $\mathbf{x}$ is formed from the target’s planar coordinates and two axis velocity. Thus,

$$\mathbf{x} = [x \; y \; v_x \; v_y] \tag{4.10}$$

4.4.1 Measurements Model

To form the measurements equation, at first, we consider that a number of N_s anchor nodes are deployed to provide the system with ranging estimates. In this section the method that is used to obtain the range estimates is not considered but the focus is on the algorithm’s development.

Given a certain range method, at each sampling step k, the N_s anchor nodes transmit an estimate of the Euclidean distance between the target node and each anchor, hence the measurements equation is formed as follows:

$$\mathbf{z}_k = \begin{bmatrix} \sqrt{(y_k - y_1)^2 + (x_k - x_1)^2} \\ \sqrt{(y_k - y_2)^2 + (x_k - x_2)^2} \\ \sqrt{(y_k - y_3)^2 + (x_k - x_3)^2} \\ \vdots \\ \sqrt{(y_k - y_{N_s})^2 + (x_k - x_{N_s})^2} \end{bmatrix} + \mathbf{v}_k \tag{4.11}$$

- where time index k: is discrete: $k = 1, 2, \ldots, K$
- where $\mathbf{z}_k$ is the measurements vector ($N_s \times 1$) containing the ranging estimates from the N_s anchor nodes
- $\mathbf{v}_k$: is a $N_s \times 1$ independent and identically distributed (i.i.d) measurements noise vector sampled from a known noise distribution

4.4.2 State Evolution

The state equation describes the evolution of the state vector in time. In the tracking problem, the state equation represents the motion pattern followed by the target.

The state equation is formed as follows:

$$\mathbf{x}_k = \mathbf{F}\mathbf{x}_{k-1} + \mathbf{\Gamma}\mathbf{w}_{k-1} \tag{4.12}$$

For the purposes of the proposed tracking system we employ two different models for the state equation of our system. In simple terms, matrix $\mathbf{F}$ from Eq.4.12 is defined in two ways.

4.4.2.1 The Single Model Approach

In the single model approach $\mathbf{F}$ is formed according to the constant velocity (CV) model where the target is assumed to be moving with constant velocity around a certain value. In practice, this means that the target's velocity remains around the value that it has at time t=0, just before the commencement of the tracking operation. Considering constant velocity the target's coordinates and velocity will be given as:

$$x_k = x_{k-1} + v_x T_s \tag{4.13}$$

$$y_k = y_{k-1} + v_y T_s \tag{4.14}$$

$$v_x[k] = vx[k-1] \tag{4.15}$$

$$v_y[k] = v_y[k-1] \tag{4.16}$$

$$\tag{4.17}$$

When also considering the presence of noisy accelerations (α_x,(α_y)) (given as $\mathbf{w}_{k-1}$) along the two axes,

$$x_k = x_{k-1} + v_x T_s + \frac{1}{2}\alpha_x T_s^2 \tag{4.18}$$

$$y_k = y_{k-1} + v_y T_s + \frac{1}{2}\alpha_y T_s^2 \tag{4.19}$$

$$v_x[k] = vx[k-1] + \alpha_x T_s \tag{4.20}$$

$$v_y[k] = v_y[k-1] + \alpha_y T_s \tag{4.21}$$

$$\tag{4.22}$$

In this case the state equation is linear and matrix **F** is given as:

$$\mathbf{F} = \begin{bmatrix} 1 & 0 & T_s & 0 \\ 0 & 1 & 0 & T_s \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{4.23}$$

and

$$\Gamma = \begin{bmatrix} T_s^2/2 & 0 \\ 0 & T_s^2/2 \\ T_s & 0 \\ 0 & T_s \end{bmatrix} \tag{4.24}$$

where,

- T_s is the sampling period,
- $\mathbf{w}_{k-1}$ is a 2×1 i.i.d process noise vector with dimension of acceleration m/sec^2 sampled from a known distribution which represents any mismodeling effects or disturbances in the motion model. Note the $\mathbf{w}_{k-1}$ has accelerations units since it represents noisy accelerations along the x and y axis
- and $\mathbf{x}_k$ is the state vector, defined in Eq.4.10

4.4.2.2 The Multiple Model Approach

The multiple model approach was developed to provide support for targets that move in a more complex pattern, which typically includes turning manoeuvres, which the CV model can not handle adequately. The CV model represents trajectories where the target moves on near straight line. Therefore, turns or jerks to the right or left can not by sufficiently captures by the CV model. The addition of two extra state models that represent a clockwise (right) and an anti-clockwise (left) turn are used to represent these motion patterns faithfully. In the multiple model case, the state equation is modeled with the use of three switching dynamical models and is defined as follows,

$$\mathbf{x}_k = \mathbf{F}(\mathbf{r}_k)\mathbf{x}_{k-1} + \mathbf{\Gamma}\mathbf{w}_{k-1} \tag{4.25}$$

where $\mathbf{r}_k$ is a regime variable that defines which of the three models is in use for the sampling period $(t_{k-1}, t_k]$.

The coordinated turn model considers a turn with constant speed (constant turn rate) which means a constant yaw rate along a road of constant radius of curvature. Considering such a motion pattern and the generic form of the state equation (Eq.4.1) in a continuous timeline we obtain,

$$x'(t) = f(x(t)) + w(t) \tag{4.26}$$

For the constant turn model with turn rate ω that we consider, state function f is given as,

$$f = (v_x, v_y, -\omega v_y, \omega v_x)^T \tag{4.27}$$

Following the analysis of Blackman in [20], Eq.4.27 can be analytically integrated and discretised which results in the following state transition matrix for the dynamical system we consider.

$$\begin{aligned} x(t+T) = x(t) + \int_t^{t+T} & (f(x(\tau)) + w(\tau)d\tau \\ & x(t+T) = g(x(t)) \end{aligned}$$

$$g(x) = \begin{bmatrix} x + v_x \dfrac{sin(\omega T)}{\omega} - v_y \dfrac{1 - cos(\omega T)}{\omega} \\ y + v_x \dfrac{1 - (\omega T)}{\omega} - v_y \dfrac{sin(\omega T)}{\omega} \\ v_x cos(\omega T) - v_y sin(\omega T) \\ v_x sin(\omega T) - v_y cos(\omega T) \end{bmatrix} \tag{4.28}$$

By including the coordinated turn models in the state equation, the state transition matrix $\mathbf{F}$ is defined in the multiple model case as follows,

- $\mathbf{F}(1)$: The CV model defined in Eq.4.23

- The first coordinated turn model

$$\mathbf{F}(2) = \begin{bmatrix} 1 & 0 & \frac{sin(\omega T_s)}{\omega} & -\frac{1 - cos(wT_s)}{\omega} \\ 0 & 1 & \frac{1 - cos(\omega T_s)}{\omega} & \frac{sin(\omega T_s)}{\omega} \\ 0 & 0 & cos(\omega T_s) & -sin(\omega T_s) \\ 0 & 0 & sin(\omega T_s) & cos(\omega T_s) \end{bmatrix} \tag{4.29}$$

- Second Coordinated turn model

$$\mathbf{F}(3) = \begin{bmatrix} 1 & 0 & \frac{sin(\omega T_s)}{\omega} & -\frac{1 - cos(\omega T_s)}{\omega} \\ 0 & 1 & \frac{1 - cos(\omega T_s)}{\omega} & \frac{sin(\omega T_s)}{\omega} \\ 0 & 0 & cos(\omega T_s) & sin(\omega T_s) \\ 0 & 0 & -sin(\omega T_s) & cos(\omega T_s) \end{bmatrix} \tag{4.30}$$

  - where T_s: is the sampling interval
  - and w : is the constant coordinated turn rate, expressed in radians/sec

The two coordinated turn models are used to model turning manoeuvres in the anticlockwise and the clockwise direction respectively. These type of motion modeling has been used previously in scenarios involving bearings only tracking [11,71] as well as in aircraft navigation.

The reason for adopting multiple models is to best describe the state evolution in situations where a manoeuvring target is considered. Thus, sudden and abrupt changes of the state vector can be effectively tackled with the multiple-model approach.

4.5 Tracking Algorithms

To solve the dynamical systems analysed in the previous section an algorithm capable of dealing with nonlinear systems must be employed as the tracking algorithm for the proposed system. This section provides insight on the Particle Filters based tracking algorithms that were developed as part of this research project.

The reason for choosing PF as the background for developing the algorithms stems from the need of an algorithm capable of handling the nonlinear system models introduced in Section 4.4. These models are derived from the basic motivating idea of this research (see Section 1.2) which is the development of tracking system for wireless sensor networks which operates exclusively on range only measurements.

4.5.1 Basic Principles of Particle Filters

This section provides the theoretical framework that the tracking algorithms are based upon. We also provide the notation that will be used and also some basic principles adopted from [98]. In the following sections we explain the two algorithms that were developed in accordance with the two models that describe the evolution of the state.

To solve a system modeled with a state-space representation model means that the filter we employ must be able to produce an estimate of the state vector $\mathbf{x}_k$ at time k based on the sequence of available measurements (ranging data) $\mathbf{Z}_k$ up to that time instance. To be able to calculate a state estimate, the posterior probability density function (pdf) $p(\mathbf{x}_k|\mathbf{Z}_k)$ must be constructed. Obtaining the posterior pdf allows for an estimate of the state to be computed based on a certain criterion like the minimum mean square error (MMSE).

The fundamental concept of the Sequential MCMC algorithms is the representation of the required pdf with a set of particles and their corresponding weights. The set of Particles is drawn from a proposal distribution and then, weighted appropriately to represent the state's pdf. Let's denote the evolution of the state vector upto time 'k' as $\mathbf{x}_k = \{\mathbf{x}_j : j = 1, 2, \ldots, k\}$. Similar to this notation, the measurements made available to the system up to time 'k' are denoted as $\mathbf{Z}_k = \{\mathbf{z}_j : j = 1, 2, \ldots, k\}$. The required pdf $p(\mathbf{x}_k \,|\, \mathbf{Z}_k)$ is approximated by a set of N particles denoted as $\mathbf{x}_k^i$ and their corresponding weights, w_k^i.

An approximation of the state pdf at time k is given from the following:

$$p(\mathbf{X}_k \,|\, \mathbf{Z}_k) = \sum_{i=1}^{N} w_k^i \delta(\mathbf{x}_k - \mathbf{x}_k^i) \tag{4.31}$$

where $\delta(\dot{})$: is Dirac's delta function.

As mentioned previously, particles $\mathbf{x}_k^i$ are sampled from a proposal distribution $q(\mathbf{x}_k|\mathbf{Z}_k)$. The importance weight for each particle is computed according to the following relationship:

$$w_k^i \propto \frac{p(\mathbf{x}_k^i \,|\, \mathbf{Z}_k)}{q(\mathbf{x}_k^i \,|\, \mathbf{Z}_k)} \tag{4.32}$$

An important aspect of Particle Filters is the fact that these type of algorithms usually suffer from the so called "degeneracy effect". In simple terms the degeneracy effect causes, after a number of iterations, all but one particles to have negligible weights. The result is, an important amount of computation to be devoted in updating particles with minimal contribution to the approximation of the pdf. To avoid the degeneracy effect in PF algorithms, a measure called effective sample size N_{eff}, is introduced and defined as follows:

$$N_{eff} = \frac{1}{\sum_{i=1}^{N}(w_k^i)^2} \tag{4.33}$$

A resampling step is carried out whenever N_{eff} is found to be smaller than a predefined threshold N_{thr}. Resampling eliminates samples with low importance weights and multiplies samples with high importance weights. Details regarding the resampling step is given in Appendix A.

Inspired by the documented ability of PF to deal effectively with nonlinear systems and based on the principles analyzed in the previous section, this research project introduces two PF-based tracking algorithms. The details of the algorithms are presented in the following sections.

4.5.2 Range Only Tracking Particle Filter Algorithm - ROT-PF

The algorithm described in this section is intended for a tracking scenario which follows the state-space model analyzed in Section 4.4.2.1. To begin with, we considered that both the state and measurements noise follow known distributions that can be sampled. The transitional prior $p(\mathbf{x}_k|\mathbf{x}_{k-1})$ is chosen as the importance density function to sample particles from. Initial particles (at time $t = 0$) are drawn from a distribution $p(x_0)$ which represents the system's prior knowledge regarding the target's initial state condition. This "knowledge" can be considered as the first detection of the target.

To produce a sample from the transitional prior, a noise sample $\mathbf{w}_{k-1}^i$ is generated initially and used in Eq.4.12 to produce a sample $\mathbf{x}_k^i$ distributed accordingly to the transitional prior. Upon receiving a new measurement the weight for each particle is computed. Because the transitional prior is chosen as the importance density function, Eq.4.32, which calculates the weight for each particle, simplifies to $\tilde{w}_k^i \propto p(\mathbf{z}_k|\mathbf{x}_k^i)$ which is the likelihood of the measurement vector (real observation) $\mathbf{z}_k = [z_1, z_2 \cdots z_j]$, given the predicted observation $\mathbf{z}_k^i$, calculated from Eq.4.11, using the sampled particle $\mathbf{x}_k^i$.

Taking also under consideration that the measurements $\mathbf{z}_k$ follow a Gaussian distribution $N(\mu_v, \sigma_v^2)$ with μ_v given by Eq. 4.11 and σ_v^2 is the observations noise covariance, the weight $\tilde{w}_k^i$ for particle $\mathbf{x}_k^i$ is calculated from the following:

$$\tilde{w}_k^i = p(\mathbf{z}_k|\mathbf{x}_k^i) = \prod_{j=1}^{N_s} \frac{1}{\sqrt{2\pi\sigma_v^2}} exp\left(-\frac{(\mathbf{z}_k - \mathbf{z}_k^i)(\mathbf{z}_k - \mathbf{z}_k^i)}{2\sigma_v^2}\right) \tag{4.34}$$

The final step in the ROT-PF algorithm involves resampling, whenever N_{eff} is found to be smaller than N_{thr}.

A single iteration of the ROT-PF algorithm is given in Algorithm 1.

Algorithm 1 ROT-PF Algorithm

```
Initialize
- Draw Initial Particles
for $i = 1$ to $N$ do
    $\mathbf{x}_0^i \sim p(x_0)$, ($\sim$: denotes sampling from)
end for
Sequential Importance Sampling Step
- Sample Particles and Calculate Weights
for $i = 1$ to $N$ do
    $\mathbf{x}_k^i \sim p(x_k|x_{k-1})$
    $\tilde{w}_k^i = p(z_k|x_k^i)$ (using Eq.4.34)
end for
- Calculate total weight
$t = \sum_{i=1}^{N} \tilde{w}_k^i$
- Normalize weights
for $i = 1$ to $N$ do
    $w_k^i = t^{-1}\tilde{w}_k^i$
end for
Resampling Step
if $N_{eff} < N_{thr}$ then
    - Resample with replacement to obtain N new particles distributed according to $p(x_k|Z_{0:k})$
end if
```

4.5.3 Range only Tracking Multiple Model Particle Filter Algorithm ROT-MMPF

To solve the switching dynamical model presented in Section 4.4.2.2, a multiple model PF algorithm is employed. The state vector in the multiple-model case is the augmented state vector which contains both the state $\mathbf{x}_k$ and the regime variable $\mathbf{r}_k$. The augmented state vector is denoted as, $\mathbf{y}_k = [\mathbf{x}_k \; r_k]$.

Initial particles are drawn from two distributions $p(\mathbf{r}_0)$ and $p(\mathbf{x}_0)$ which represent the system's initial knowledge regarding the system's state. Particles for the state $\mathbf{x}_k$ are sampled from the transitional prior similar to the ROT-PF algorithm, while particles for the regime variable are sampled according to the transitional probability matrix $\Pi = [\pi_{mn}]$. The rule that is followed for that is; if $r_{k-1}^i = m$, then r_k^i should be set to n with probability π_{mn}.

The transitional probabilities are illustrated in the following figure where 1,2,3 represent the three dynamical models (CV and two coordinated turn models) and the arrows represent the transitional probabilities between the dynamical models.

the transitional probability matrix is formed based on that as:

$$\mathbf{\Pi} = \begin{bmatrix} \pi_{11} & \pi_{12} & \pi_{13} \\ \pi_{21} & \pi_{22} & \pi_{23} \\ \pi_{31} & \pi_{31} & \pi_{33} \\ & & \end{bmatrix} \tag{4.35}$$

As with the ROT-PF algorithm, whenever a new measurement vector becomes available the

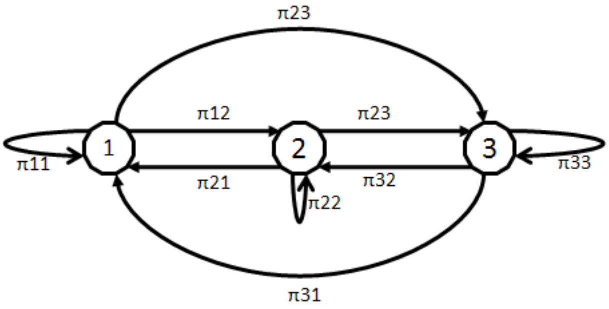


FIGURE 4.2: Transitional Probabilities

weights for each particle are computed by using the likelihood function $p(\mathbf{z}_k \,|\, \mathbf{y}_k^i)$, which in this case depends on the augmented state vector. Similar, to the single-modal algorithm the predicted observation z_k^i is calculated based on the sampled particles of the state vector $\mathbf{x}_k^i$, using Eq.4.11. Finally Eq.4.36 is calculated to produce a value for every weight $\mathbf{w}_k^i$ in a similar way with the ROT-PF case only this time the projected value of the observation based on the sampled particles also depends on the state model in use for time step k. The final step of the ROT-MMPF algorithm includes the resampling step whenever this is necessary.

$$\tilde{w}_k^i = p(\mathbf{z}_k|\mathbf{x}_k^i, r_k^i) = \prod_{j=1}^{N_s} \frac{1}{\sqrt{2\pi\sigma_v^2}} exp\left(-\frac{(\mathbf{z}_k - \mathbf{z}_k^i|r_k^i)(\mathbf{z}_k - \mathbf{z}_k^i|r_k^i)}{2\sigma_v^2}\right) \tag{4.36}$$

An iteration of the ROT-MMPF algorithm is given in Algorithm 2.

Algorithm 2 ROT-MMPF Algorithm

```
Initialize
- Draw Initial Particles
for $i = 1$ to $N$ do
    $\mathbf{r}_0^i \sim p(r_0)$ ($\sim$ denotes sampling from)
    $\mathbf{x}_0^i \sim p(x_0)$
end for
Sequential Importance Sampling Step
- Sample Particles and Calculate Weights
for $i = 1$ to $N$ do
    $\mathbf{r}_k^i \sim \pi ij$
    $\mathbf{x}_k^i \sim p(x_k|x_{k-1})$
    $\tilde{w}_k^i = p(z_k|x_k^i, r_k^i)$ (using Eq.4.34)
end for
- Calculate total weight
$t = \sum_{i=1}^{N} \tilde{w}_k^i$
- Normalize weights
for $i = 1$ to $N$ do
    $w_k^i = t^{-1}\tilde{w}_k^i$
end for
Resampling Step
if $N_{eff} < N_{thr}$ then
    - Resample with replacement to obtain N new particles distributed according $p(y_k|Z_{0:k})$
end if
```

4.6 Discussion and Summary

In this chapter, we provided the formulation of the tracking problem as a dynamical state-space estimation problem. In the sequel the basics of Bayesian Estimation and the fundamental princniples of Particle Filters were analysed. The two algorithms (ROT-PF, ROT-MMPF) presented in this Chapter were developed based on the dynamical tracking system presented in Section 4.4.

The approach followed in this research differentiates in the following points from other current approaches in the literature.

- This research proposes a tracking system entirely based on range (distance) observations. A number of published approaches in the relevant literature utilized additional modalities such as velocity and bearings information. To the author's view, ranging between WSN nodes is a modality that can be acquired in a cost-efficient manner using a number of methods without the use of additional per-node hardware.
- The proposed system, models the target's dynamics using two different approaches. The constant velocity model which has been employed in the past, has been used to verify the ability of the system and the tracking algorithm to track an object that moves on relatively simple motion pattern. This model was then extended to a multiple-model approach which has not been considered in range-only tracking systems published in the literature. To the author's belief, the multiple-model approach enhances the support for manoeuvring targets as this is the case in the majority of real-world applications.
- The PF algorithms that were designed within the scope of this research project aim at providing a framework for solving the non-linear tracking system. Different to other approaches, where each different node represented a distinct particle and updated the state belief based on its local measurements, the proposed system operates on the accumulated data acquired from a number of anchor nodes, thus maximizing the amount of information used for updating the state belief at every time step.

These innovative points are targeted to be combined with the RF-ToF ranging method(see Chapter 3, on a real-time range-only tracking system for embedded nodes. In the following chapter, the proposed tracking algorithms are extensively evaluated in a simulation environment which permits the performance assessment of the proposed tracking system.

Chapter 5

System Evaluation under Simulations

This chapter presents results from simulating the range-only tracking system presented in Chapter 4. MATLAB was chosen as the environment to conduct the simulations because of flexibility it offers in the implementation of the PF tracking algorithms. The experimentation in the simulation environment enabled us to assess the algorithms performance in terms of accuracy and also observe how accuracy is affected from a number of factors like the sampling interval, the number of particles that are used in the implementation of the algorithm, the initial knowledge of the target's state and the number of deployed anchor nodes. The robustness of the system is examined with respect to the additive noise that corrupts the accuracy of the ranging measurements. To provide a thorough investigation of the proposed tracking system, various scenarios were simulated under different parameters.

Section 5.1 provides an overview of the simulation setup. Sections 5.2 and 5.3 present simulations results of the ROT-PF and ROT-MMPF algorithms respectively. In Section 5.5, the tracking system is simulated by taking into account the achieved accuracy of the ToF ranging technique, presented in Chapter 3. In Section 5.6 the two approaches (single- and multiple-model) are compared under a manoeuvring scenario. In the sequel, the theoretical Cramer-Rao lower bound of the proposed tracking system are derived and used as a benchmark to evaluate the error performance of the proposed system (Section5.8). A summary of this chapter highlighting the main outcomes of the simulation investigation is provided in Section 5.9.

5.1 Simulation Environment

Both the ROT-PF and the ROT-MMPF algorithms were simulated under different conditions to evaluate their performance in terms of tracking accuracy and robustness. In the simulations environment considered, four anchor nodes are deployed in known locations to provide the system with range observations, while a single ground mobile target is the object of interest. A predefined amount of time steps is set for every simulation run. The target begins from a given initial position and then moves randomly based on the model used for its dynamics (CV model or multiple models). Uncertainty is added from the state noise distribution. At every time step the range from each one of the four anchors is calculated with the addition of the observations noise. The tracking algorithm (either ROT-PF or ROT-MMPF) is initialised from an initial distribution and at each time step the preset number of particles is generated. Following, the four range estimates are used to calculate the weights of the particles. Finally an estimation of the state vector is produced for every sampling step. At the end of the simulation the "real" state vectors $\mathbf{x}$ and the estimation $\hat{\mathbf{x}}$ are compared with the use of the Root Mean Square Error (RMSE). A flow chart of the simulation is illustrated in Figure 5.1.

In practice, to acquire a range estimate from each of the anchor nodes requires a certain amount of time. Moreover, processing the ranging measurements and inferring an estimation regarding the target's position requires a certain amount of time to be devoted to that process. However for the purposes of simulation, the ranging estimates are considered to be instantly acquired by the central node. Additionally, in praxis the position of the anchor nodes is extremely significant for the system operation. To perform the ToF ranging the anchors nodes must be constantly within the communication range of the target. This restriction was disregarded in simulations, thus the anchor's locations was not affecting the system's performance.

To quantify the accuracy achieved by the system, the Root Mean Square Error (RMSE) is employed. The RMSE is a well-known metric used in the majority of tracking systems to evaluate their performance. The RMSE for the target's planar position is defined as follows:

$$\mathbf{RMSE} = \sqrt{\frac{1}{T}\sum_{t=1}^{T}(x_t - x_{t,est})^2 + (y_t - y_{t,est})^2} \tag{5.1}$$

- where T: is the total number of time steps.
- x_t, y_t are the real coordinates of the target
- $x_{t,est}$, $y_{t,est}$ are the estimated coordinates

Sometime for specific investigation purposes, the exact same scenario is simulated multiple times in order to access the robustness of the system's performance with Monte Carlo simulations. In this occasion, to quantify the effect of different parameters, the average RMSE $\mathbf{RMSE}_{avg}$ will be also used. The average $\mathbf{RMSE}_{avg}$ over L executions of the same scenario is defined as:

$$\mathbf{RMSE_{avg}} = \frac{1}{L}\sqrt{\frac{1}{T}\sum_{t=1}^{T}(x_t - x_{t,est})^2 + (y_t - y_{t,est})^2} \tag{5.2}$$

To define different experimental conditions, a range of values was used for the most important simulations parameters. The state noise and observations noise distributions represent the amount of noise that corrupt the state model and the observations noise respectively. The observations noise is related to the accuracy of the ranging technique while the state noise is related to the motion pattern of the target. In the simulations, we employ zero-mean Gaussian distributions with covariances w_k and v_k to approximate the state and observations noise respectively. Increased covariance values translates to higher noise conditions. Therefore three noise levels are considered in the simulations. A low noise scenario with $w_k < 0.1$ and $v_k < 0.1$, an intermediate noise scenario with $0.1 < w_k < 2$ and $0.1 < v_k < 2$ and a high scenario where $w_k > 2$ and $v_k > 2$. Another important parameter is the target's initial state $\mathbf{x}_0$ and in particular its initial velocity values. Due to the fact that the CV model is used the target is expected to move with speed around its initial value. We therefore, consider slow target's the ones with initial velocity $v < 1m/s$, targets moving at intermediate speeds $1m/s < v < 3m/s$ and fast targets with $v > 3m/s$. Ultimately, the system's initial knowledge regarding the target's initial position is a key parameter. The tracking system employs a pre-defined distribution to sample initial particles from. In the simulations that are presented here, a Gaussian (N_0) distribution was chosen for that matter. Improved system performance is obtained when the mean μ_0 of this Gaussian distribution, in particular the values that correspond to the target's initial coordinates are close to the target's real initial coordinates as defined in $\mathbf{x}_0$. In a number of simulations μ_0 is chosen to be much different than the actual initial state of the target, to approximate a scenario where the system was poor knowledge regarding the target's initial state.

In the multiple model case, the transitional probability regime variable denotes the likeliness of a target to remain to the regime between successive sampling intervals. Subsequently the transitional probability m is used to define the proneness of target to perform turning manoeuvres. The lower m is the highest the probability for the target to perform manoeuvres. Therefore $m = 0.85$ represents a low manoeuvrability target, while $m = 0.65$ describes a target inclined to manoeuvre more frequently. Of the rest simulation parameters the sampling interval and number of particles of great importance for the system and for that they are analysed thoroughly. For the multiple model case we chose a higher amount of total simulation time (75 sec) as opposed to the single model (50 sec) to allow time to the target to transit between regimes and perform manoeuvres.

5.2 Evaluating the ROT-PF algorithm

This section presents simulation results of the ROT-PF algorithm under various scenarios. The details of the ROT-PF tracking algorithm were given in Section 4.5.2.

5.2.1 First Scenario, low-noise conditions and slow speed target

A wireless network consisting of four anchor nodes is considered to be deployed. The coordinates of the anchor nodes are, $s_1 = [20m\ 20m], s_2 = [20m\ 60m], s_3 = [80m\ 20m], s_4 =$

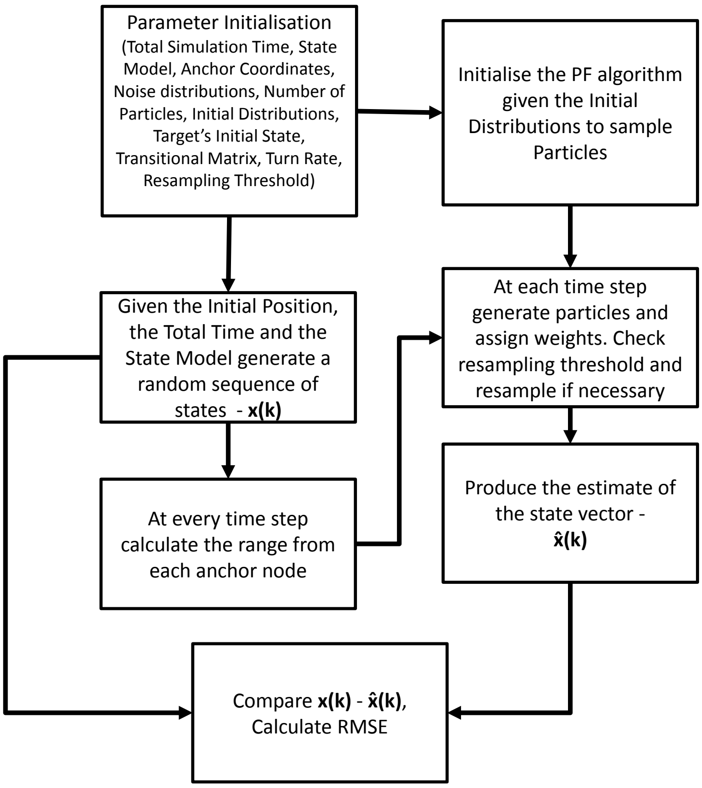

Parameter Initialisation (Total Simulation Time, State Model, Anchor Coordinates, Noise distributions, Number of Particles, Initial Distributions, Target's Initial State, Transitional Matrix, Turn Rate, Resampling Threshold)
Initialise the PF algorithm given the Initial Distributions to sample Particles
Given the Initial Position, the Total Time and the State Model generate a random sequence of states - x(k)
At each time step generate particles and assign weights. Check resampling threshold and resample if necessary
At every time step calculate the range from each anchor node
Produce the estimate of the state vector - x̂(k)
Compare x(k) - x̂(k), Calculate RMSE

FIGURE 5.1: Flow Chart of the Simulation Environment

$[80m\ 60m]$. The state vector of the target evolves in time as defined in Eq. 4.23, while the measurements are associated to the target's state according to Eq. 4.11. The sampling period is set to $T_s = 1sec$ and the system evolves for $T = 75sec$. In the implementation of the ROT-PF tracking algorithm $N = 500$ particles were used. Since a low-noise environment is considered the measurements and state noise sources are considered to follow zero mean Gaussian distributions as follows.

$$\mathbf{w}_k \sim \mathcal{N}(0, 0.03)$$
$$\mathbf{v}_k \sim \mathcal{N}(0, 0.04)$$

The target's initial state is $\mathbf{x}_0 = [10m\ 15m\ 0.4m/s\ 0.3m/s]$. Initial particles are sampled from a Gaussian distribution with mean the actual initial state of the target ($\mu = \mathbf{x}_0$) and covariance matrix $\mathbf{S}_0 = \mathbf{I}_4$. The distribution that initial samples are taken from represents the system's "knowledge" regarding the target's initial state. In this case the system has very good "initial knowledge" ($\mu = \mathbf{x}_0$).

where : $\mathbf{I}_n$ is the $n \times n$ identity matrix.

Trajectory estimation results from a single run are illustrated in Figure 5.2. The estimation of the two-axis velocity of the object is illustrated in Figures 5.3, 5.4.

The RMSE for this run was calculated: $\mathbf{RMSE} = 0.2236m$ over a 70m x 50m area. Due to

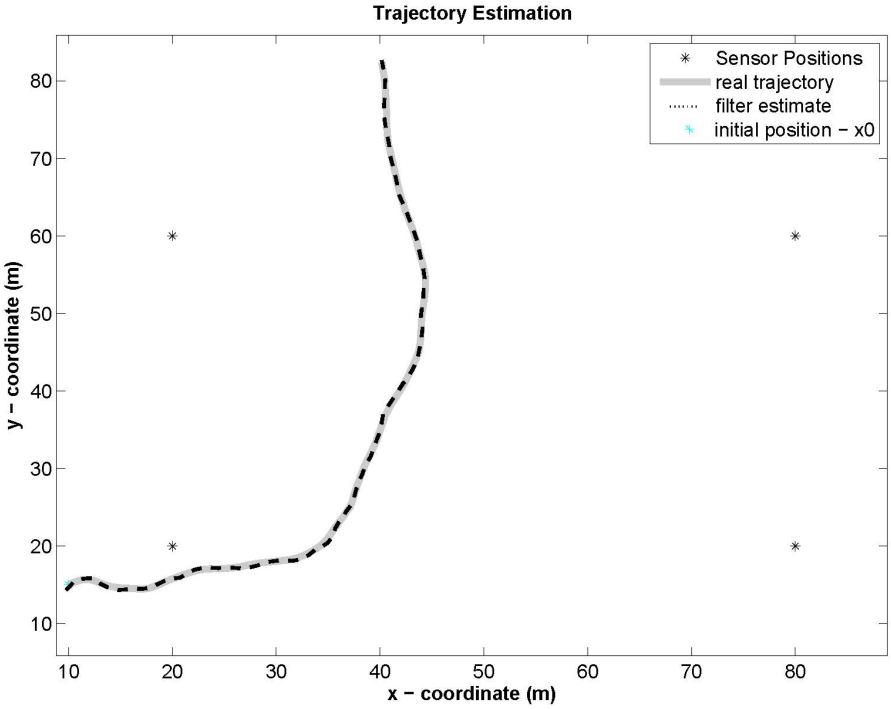


FIGURE 5.2: Trajectory Estimation for the exemplar run of Scenario 1

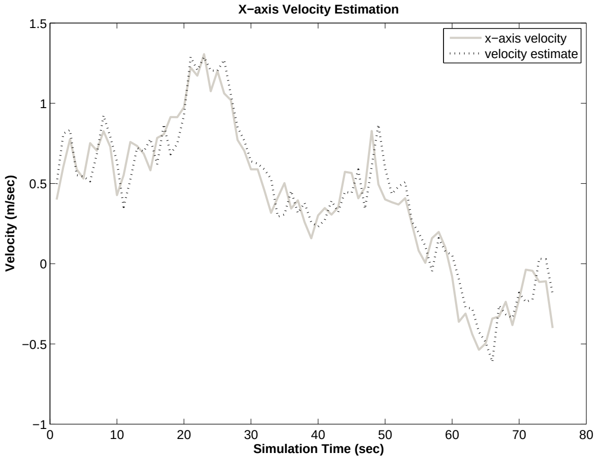


FIGURE 5.3: X-axis velocity for the exemplar run of Scenario 1

the fact that the additive noise as well as the target's speed, are very low, the system achieves great accuracy in tracking the target. This previous set-up was simulated for 50 times and the average RMSE was calculated and found to be $\mathbf{RMSE_{avg}} = 0.3649m$. The proposed system is very accurate under such "ideal" conditions. The RMSE obtained from 50 executions of this scenario is depicted in Figure5.5.

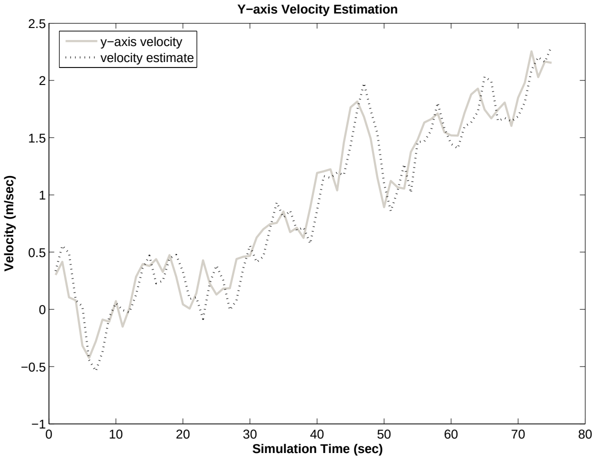


FIGURE 5.4: Y-axis velocity for the exemplar run of Scenario 1

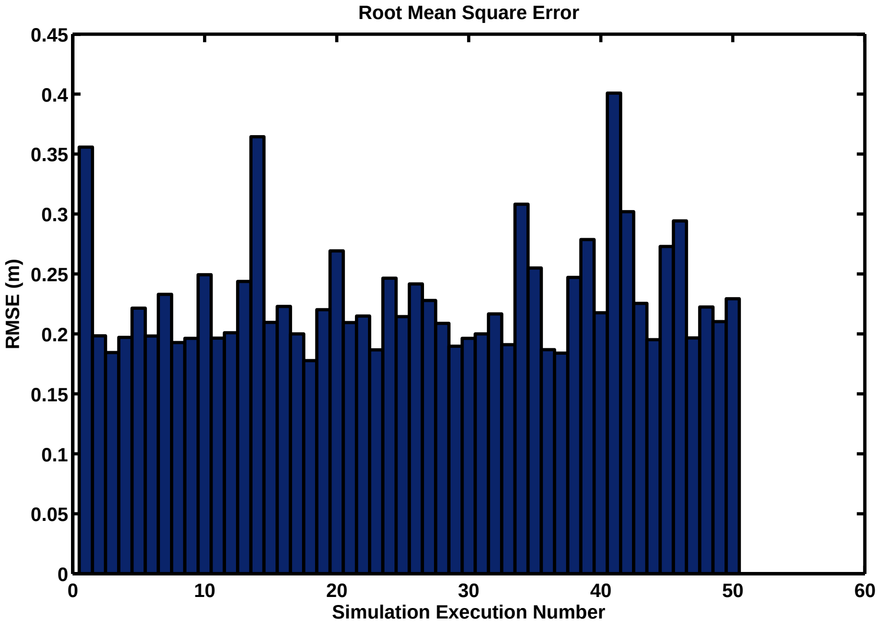


FIGURE 5.5: RMSE for 50 of Scenario 1

5.2.2 Second Scenario - faster target, higher noise level

For this simulation the anchor nodes were considered to be placed in the same positions, and the same sampling period ($T_s = 1$) and total time were applied ($T = 75sec$). Also the same number of particles was employed in the implementation of the ROT-PF algorithm. The measurements and state noise covariances were increased from the previous scenario to simulate an environment with increased uncertainty.

$$\mathbf{w}_k \sim \mathcal{N}(0, 0.5)$$
$$\mathbf{v}_k \sim \mathcal{N}(0, 1.5)$$

The target's initial state is $\mathbf{x}_0 = [5m\ 2m\ 3m/s\ 2.5m/s]$. The target is now initialised with faster speed the the previous scenario. Finally initial particles were sampled from the same distribution as in the First Scenario. The trajectory and velocity estimation results are provided in Figure 5.6 and Figure 5.7 and Figure 5.8

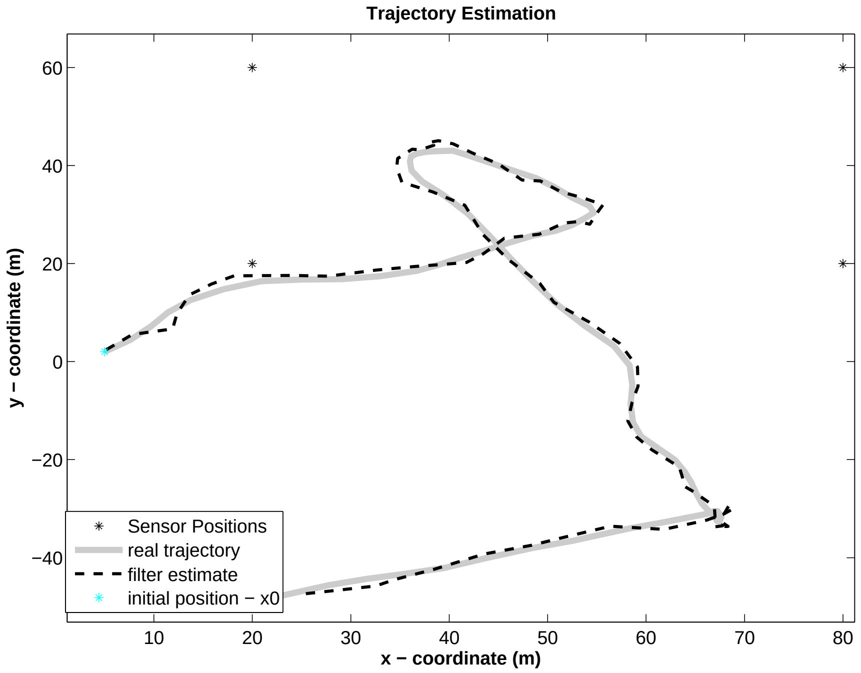


FIGURE 5.6: Trajectory Estimation for the exemplar run of Scenario 2

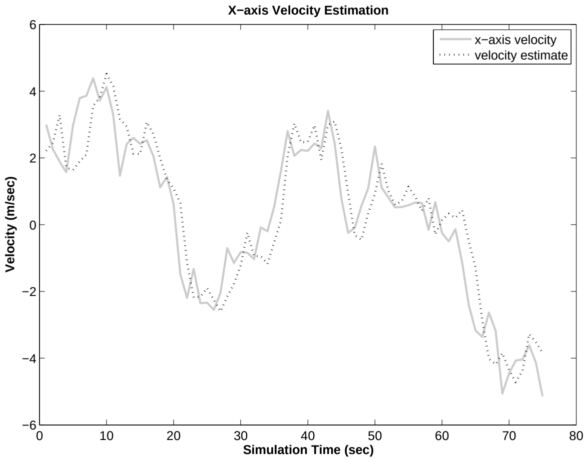


FIGURE 5.7: X-axis velocity for the exemplar run of Scenario 2

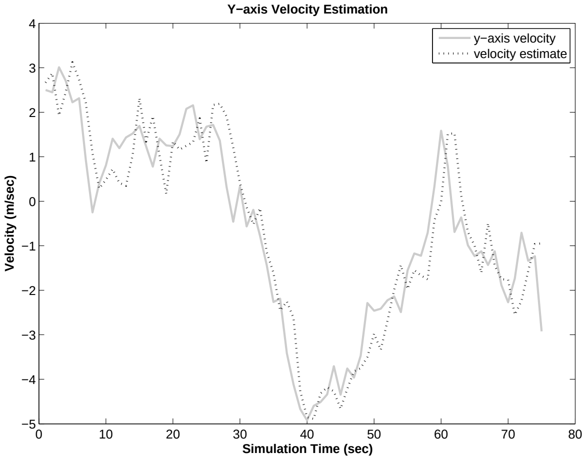


FIGURE 5.8: Y-axis velocity for the exemplar run of Scenario 2

The RMSE for this exemplar run is calculated $\mathbf{RMSE} = 2.29m$ over a 60 x 100m area. Similar to the previous investigation this scenario was simulated for 50 times and the RMSE was calculated for each run. Results are illustrated below.

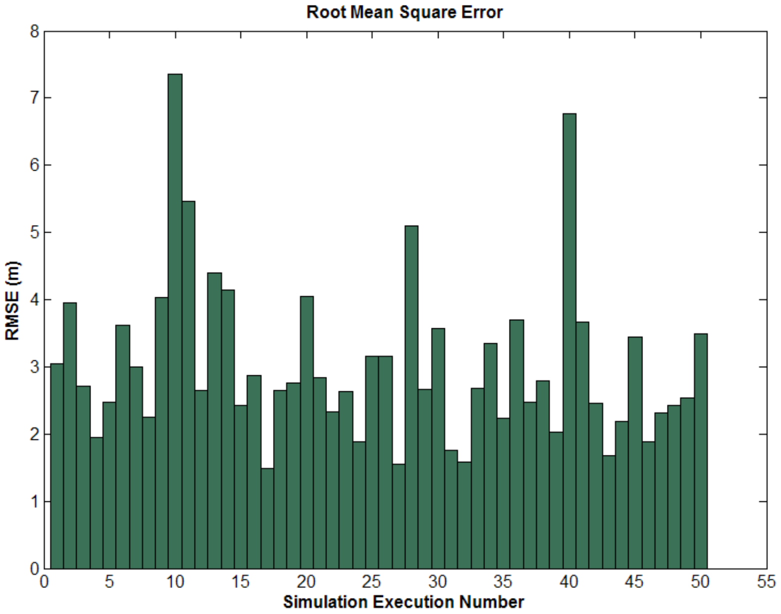


FIGURE 5.9: RMSE for 50 executions of Scenario 2

Very good accuracy has been achieved since the RMSE never becomes higher than 8m and is usually below 5m.

5.2.3 Third Scenario, noisy environment, high initial uncertainty

A factor that affects the accuracy of the system is the system's knowledge regarding the target's initial state. In situations where the system does not have "good knowledge" about the initial state of the target, the system's accuracy decreases because a substantial error is added in the beginning of tracking or sometimes tracking fails completely due to that initial error. To investigate the effects in the performance of the system under "poor knowledge" regarding the target's initial position the following scenario is simulated.

The anchor nodes are positioned in the same locations as in the previous scenario. The sampling period was maintained in $T_s = 1sec$ and the total simulation time was left $T = 75sec$. Finally the same number of particles was employed in the implementation of the ROT-PF algorithm.

The measurements and state noise sources for this scenario are considered in such a way to represent high-noise conditions:

$$\begin{aligned}\mathbf{w}_k &\sim \mathcal{N}(0, 2)\\ \mathbf{v}_k &\sim \mathcal{N}(0, 4)\end{aligned}$$

which means that both our motion model and measurements were corrupted from a high amount of noise.

The target's initial state is $\mathbf{x}_0 = [20m\ 15m\ 4m/s\ 5m/s]$. A faster target than previously was considered. Initial particles were drawn from a Gaussian distribution with zero mean $\mu_0 = 0$ and $S_0 = \mathbf{I}$. Is is obvious that in this scenario the system has poor knowledge regarding the target's initial state. The main reason for simulating such a scenario is to demonstrate the robustness of the proposed tracking system under condition of a fast target, high noise and with "poor knowledge" about the target's initial state.

An exemplar execution of this high-noise scenario, with significant initial uncertainty is illustrated in Figure5.10. Due to the poor initial regarding the target's initial condition the system requires some iterations to converge to the target's trajectory. The RMSE for this run is 21m over a 200m x 300m area.

The above scenario was simulated for 100 times and the RMSE was calculated in every execution.

Only in one execution the system lost track of the target. In 96 of the executions the RMSE remained below 50m while and in 76 of the execution the RMSE was lower than 20m in a 100m x 200m area. These results indicate that the system efficiently copes with the noisy and "poor initial knowledge" conditions and achieves acceptable accuracy in its tracking operation. It must be noted that the an initial error of approximately 25m is added to the RMSE in every execution due to the poor "initial knowledge".

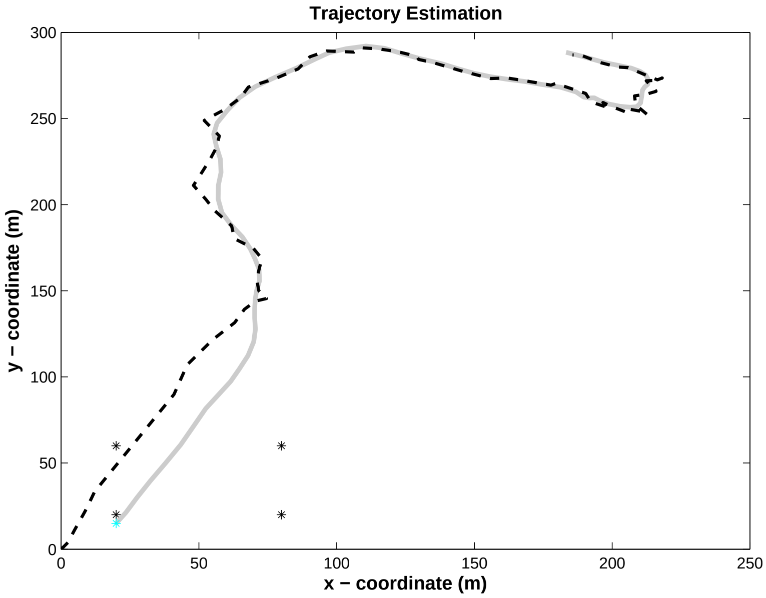


FIGURE 5.10: Trajectory Estimation for the exemplar run of Scenario 3

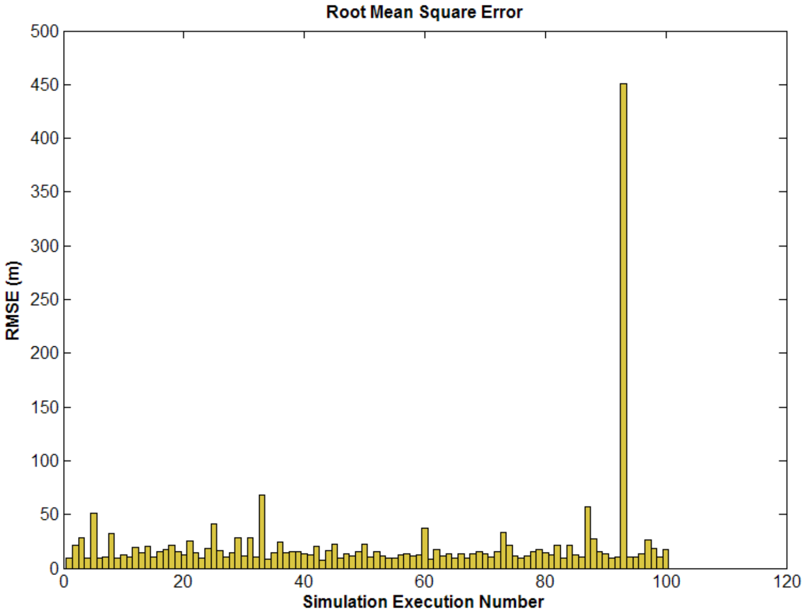


FIGURE 5.11: RMSE for 100 executions of Scenario 3

5.2.4 Effect of the Sampling Period and Number of Particles in the System's Performance

5.2.4.1 Investigating the Sampling Interval

This section investigates the effect on the system's accuracy of the number of particles employed in the ROT-PF tracking algorithm as well as the sampling period. The sampling period (or interval) is defined as the elapsed time between successive estimations of the target's state vector. Sources of latency, that result in increasing sampling interval is the time that it takes to fuse the ranging estimates form the anchor nodes to the tracking node, as well as the time required by the anchors node to produce a ranging estimate. Moreover, the time required for the algorithm to run into completion must also be considered. This depends heavily on the number of particles employed by the PF algorithm. All these sources of latency result in a total amount of time T_{agg} between successive state estimation of the target. When this amount of time increases, the result is a lesser amount of data to be provided to the system for tracking. In this case, the temporal resolution of the data decreases resulting in poor performance.

To investigate how the increasing sampling period affects the system's performance, a scenario was simulated multiple times for different sampling periods. The average RMSE was calculated for every sampling period. To isolate the effect of the sampling period in the system's performance a scenario were the system had "good knowledge" regarding the target's initial state was considered. The scenario parameters are, as follows.

Four anchor nodes deployed, in the following coordinates $s_1 = [40m\, 60m]$, $s_2 = [40m\, 140m]$, $s_3 = [100m\; 60m]$, $s_4 = [100m\; 140m]$. The number of particles in the implementation was kept low $N = 500$ and the total simulation time is $T = 75sec$.

The measurements and state noise sources were considered to follow :

$$\begin{aligned} \mathbf{w}_k &\sim \mathcal{N}(0, 1.5) \\ \mathbf{v}_k &\sim \mathcal{N}(0, 3) \end{aligned}$$

which represent significantly noisy conditions.

The target's initial state is $\mathbf{x}_0 = [10m\;\, 8m\;\, 5m/s\;\, 5m/s]$. Finally initial particles were drawn from a Gaussian distribution with mean $\mu_0 = \mathbf{x}_0 + N(0,1)$ and covariance $S_0 = I$. In the beginning the sampling period was set $T_s = 1$ and was increased to 2,3,. . . ,7sec. Each different sampling period was simulated for 100 times.

Figure 5.12 illustrates the degradation of the system's performance under increasing sampling period. The average RMSE increases almost linearly with increasing sampling interval. The discontinuity for $T_s = 7sec$ where better performance than expected is observed should not be misleading. Given that the total time is T=75sec, the system has 10 estimation points for $T_s = 7sec$ and 12 for $T_s = 6$. Subsequently the RMSE calculation involves more erroneous points for $T = 6sec$ than for $T = 7sec$. This leads, to a similar performance for $T_s = 6sec$ and $T_s = 7sec$ and in some runs the systems may even achieve a smaller error for $T_s = 7sec$. However when increasing the total time to 300sec, allowing for a substantial number of estimation points for each sampling interval value, and repeating the same experiment we obtain the results illustrated in Figure5.13, where the average RMSE values follow a more faithful linear pattern with increasing sampling period.

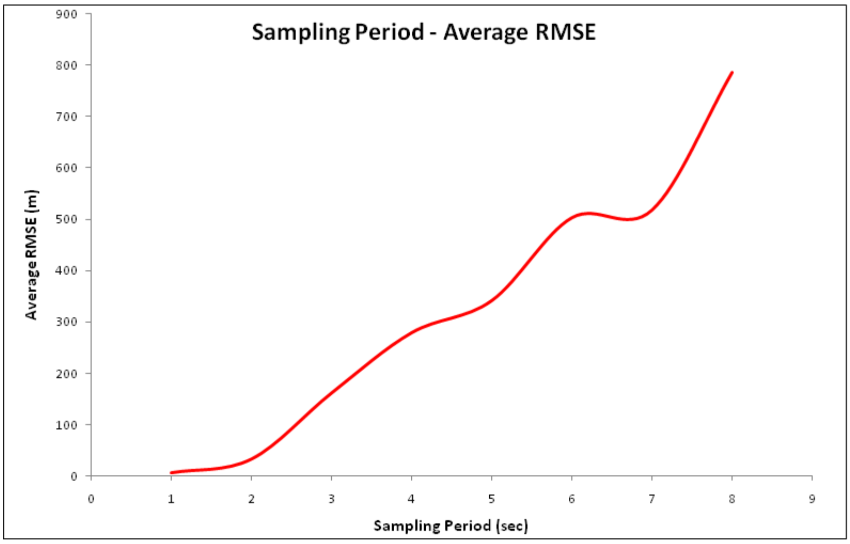


FIGURE 5.12: Sampling Period - Average RMSE for ROT-PF algorithm (total time 75 sec)

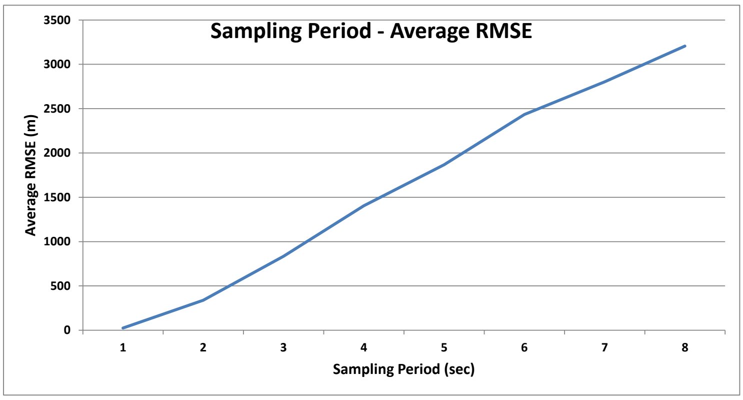


FIGURE 5.13: Sampling Period - Average RMSE for ROT-PF algorithm (total time 300 sec)

5.2.4.2 Investigating the Number of Particles

ROT-PF algorithm performance depends heavily on the number of particles that the algorithm samples from the proposal distribution (in this case the transitional prior) in order to estimate the current state of the system. If a bigger number of particles is used the algorithms performance improves. Nevertheless increasing the number of particles will increase the execution time and complexity of the algorithm since more calculations must be carried out. Particularly in situations where the algorithm runs on real-time data and must produce real-time estimates the execution time is a very important factor. Thus, although increasing the number of particles will result in better performance this research considers 3500 as the maximum number of particles to be used in a single implementation of the tracking algorithm.

To quantify the improvement that the increased particle size achieves, we simulate the previous scenario with sampling period $T_s = 3$ for different particle sizes (500, 1000, 1500, 2000, 2500, 3000, 3500) for 100 times and calculate the average RMSE. Results are depicted in Figure 5.14. Clearly increasing the number of particles results in an improvement in the performance of the system. The improve is greater when increasing the particle size from 500 to 2000, while the increase is significantly smaller in for particle sizes from 2000 to 4000. Therefore it is concluded, that given the effect that increased number of particles has on the execution time of PF algorithm, for a real time system aiming at a high accuracy a particle size between 1000-2000 would be the best choice, balancing accuracy and execution time.

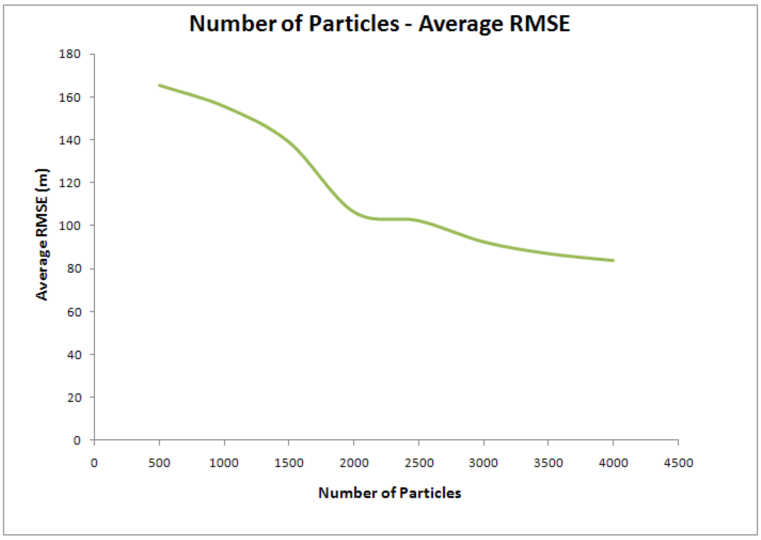


FIGURE 5.14: Number of Particles - Average RMSE for ROT-PF algorithm

5.3 Evaluating the ROT-MMPF algorithm

This section presents simulation results of the ROT-MMPF algorithm under various scenarios. The details of the ROT-MMPF tracking algorithm can be found in section 4.5.3. This is the multiple-model approach which provides better support for targets that are likely to manoeuvre and in general exhibit sudden changes in their position and velocity vectors. The manoeuvrability of a target is quantified with the use of the transitional probability matrix. High manoeuvring target's will have a higher probability to switch between the three dynamical models defined in section 4.4.2.2. The scenario set-up is the same as in the investigation of the ROT-PF algorithm. Finally the same factors that affect the accuracy in the ROT-PF case are also analysed for the ROT-MMPF.

5.3.1 Scenario 1: Low manoeuvrability, low noise scenario

A wireless network consisting of four anchor nodes is considered to be deployed. The coordinates of the anchor nodes are, $s_1 = [10m\ 0m], s_2 = [50m\ 0m], s_3 = [10m\ 25m], s_4 = [50m\ 25m]$. The measurements equation is the same as before, defined in Eq. 4.11. The evolution of the state vector in time follows the dynamic model described in section 4.4.2.2. The sampling period is initially set to $T_s = 1sec$ and the system evolves for $T = 50sec$. In the implementation of the ROT-PF tracking algorithm $N = 500$ particles were used. The measurements and state noise sources are considered to follow zero mean Gaussian distributions. Specifically:

$$\begin{aligned} \mathbf{w}_k &\sim \mathcal{N}(0, 0.05) \\ \mathbf{v}_k &\sim \mathcal{N}(0, 0.5) \end{aligned}$$

The regime variable $\mathbf{r}$ is defined as a first order homogeneous Markov chain with transition probability $m = 0.85$. Thus the transition probability matrix is,

$$\mathbf{P}(r_t \,|\, r_{t-1}) = \begin{bmatrix} 0.85 & 0.075 & 0.075 \\ 0.075 & 0.85 & 0.075 \\ 0.075 & 0.075 & 0.85 \end{bmatrix} \tag{5.3}$$

The target's initial state is $\mathbf{x}_0 = [5m\ 5m\ 1m/s\ 1m/s]$. The turning rate $w = pi/3$ rad/sec. Finally initial particles for the regime variable are sampled with equal initial probability $\mathbf{P}_0 = [1/3\ 1/3\ 1/3]$ while for the state initial particles are drawn from a Gaussian distribution with zero mean and unity covariance.

Results in estimating the target's trajectory and two-axes velocity are illustrated in Figure 5.15. Figures 5.16 and 5.17

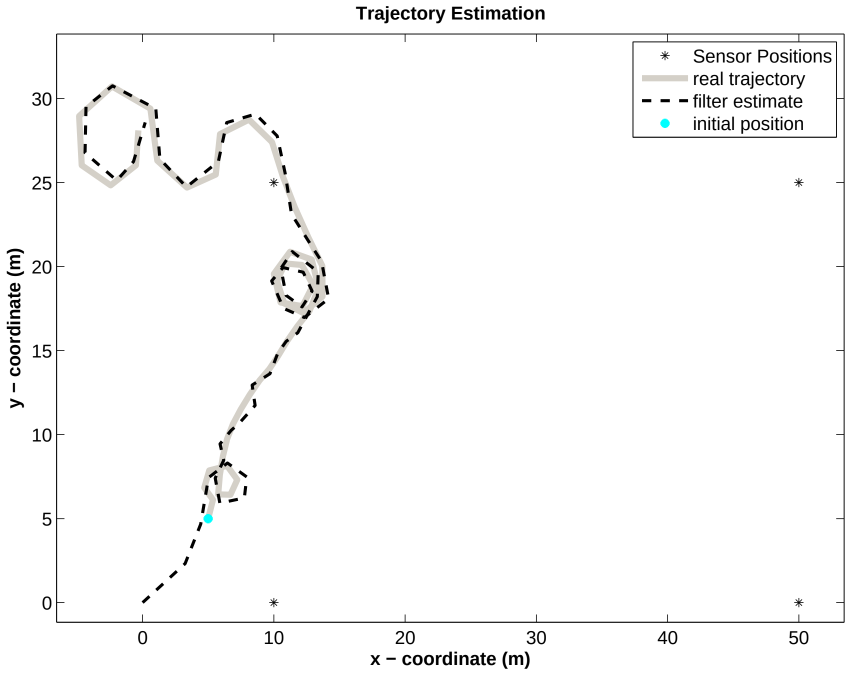


FIGURE 5.15: Trajectory Estimation for the exemplar run of Scenario 1

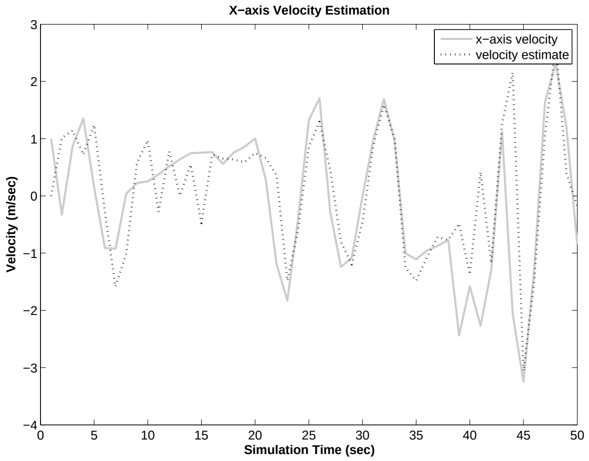


FIGURE 5.16: X-axis velocity for the exemplar run of Scenario 1

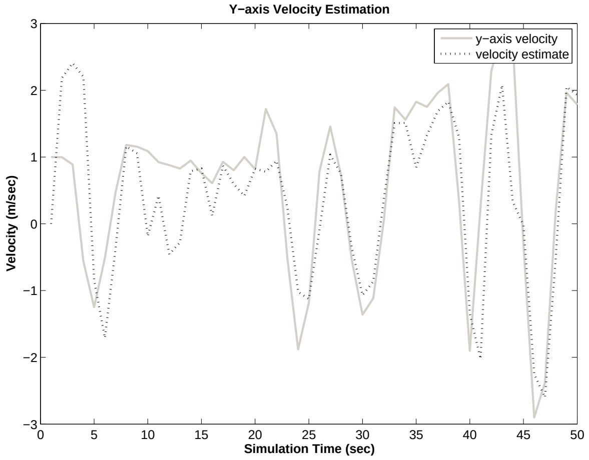


FIGURE 5.17: Y-axis velocity for the exemplar run of Scenario 1

The RMSE was calculated, $\mathbf{RMSE} = 0.7024m$ over a 50m x 30m area. Finally RMSE Results from 50 executions of this scenario are depicted in Figure5.18.

The parameters considered in this scenario are close to the ideal case. This scenario was simulated under very low additive noise and the target considered was moving in low speed and with low possibility for manoeuvres.

5.3.2 Scenario 2: intermediate manoeuvrability, faster target

For this simulation the anchor nodes were considered to be placed in the same positions, and the same sampling period ($T_s = 1$) and total time were applied ($T = 75sec$). Also the same number of particles was employed in the implementation of the ROT-PF algorithm.

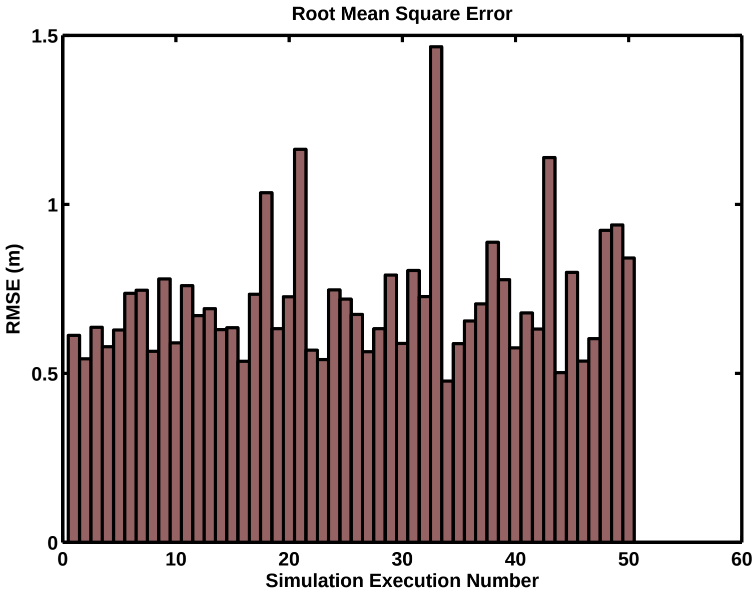


FIGURE 5.18: RMSE for 50 executions of Scenario 1

The measurements and state noise sources were considered to follow :

$$\mathbf{w}_k \sim \mathcal{N}(0,1)$$
$$\mathbf{v}_k \sim \mathcal{N}(0,3.5)$$

The target's initial state is $\mathbf{x}_0 = [10m\ 10m\ 2m/s\ 2m/s]$. In this scenario to increase the level of uncertainty in the system regarding the target's initial position, initial particles are sampled from a Gaussian distribution with mean $\mu_0 = \mathbf{x}_0 + N(0,3)$ and unity covariance. The turning rate $w = pi/4$ rad/sec. Finally initial particles for the regime variable are sampled with equal initial probability $\mathbf{P}_0 = [1/3\ 1/3\ 1/3]$. The transitional probability is set to $m = 0.65$.

Results regarding the estimation of the target's trajectory from a single execution of the above scenario are depicted in Figure 5.19.

To evaluate the robustness of the multiple-model tracking algorithm, this scenario was simulated for a total of 50 times. The RMSE was calculated for every execution and results are illustrated in Figure 5.20.

The RMSE remains below 20m in 45 of the execution while in 43 of the execution the RMSE is lower than 10m over an 200m x 120m area. These results demonstrate the ability of the system to cope with manoeuvring targets despite the presence of noise and the initial uncertainty.

5.3.3 Scenario 3: High speed high manoeuvrable target

The scenario investigated in this section pertains to a target moving at high speed that is capable of performing abrupt manoeuvres. The parameters are as follows. The anchor nodes are

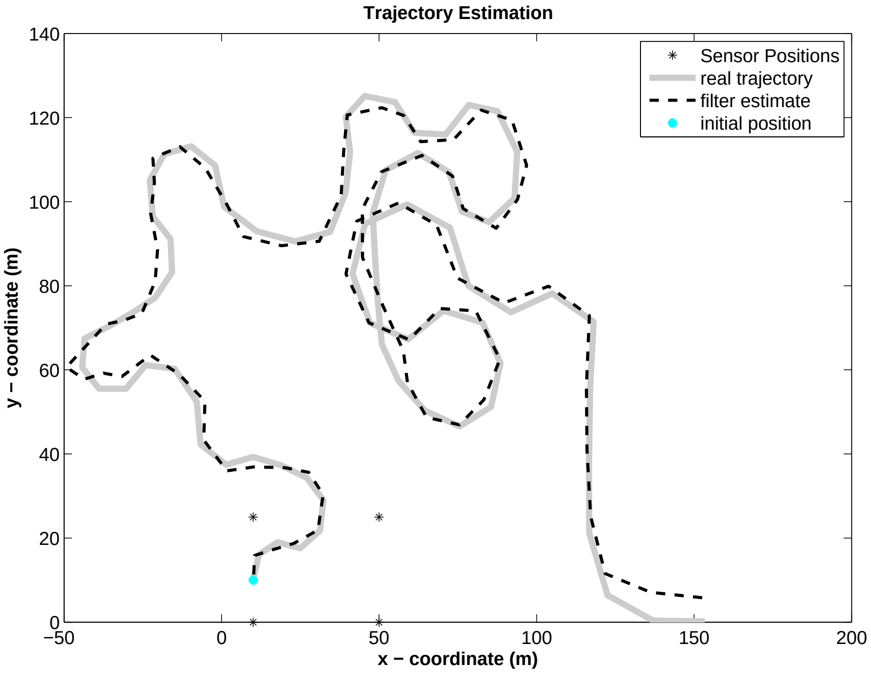


FIGURE 5.19: Trajectory Estimation for the exemplar run of Scenario 2

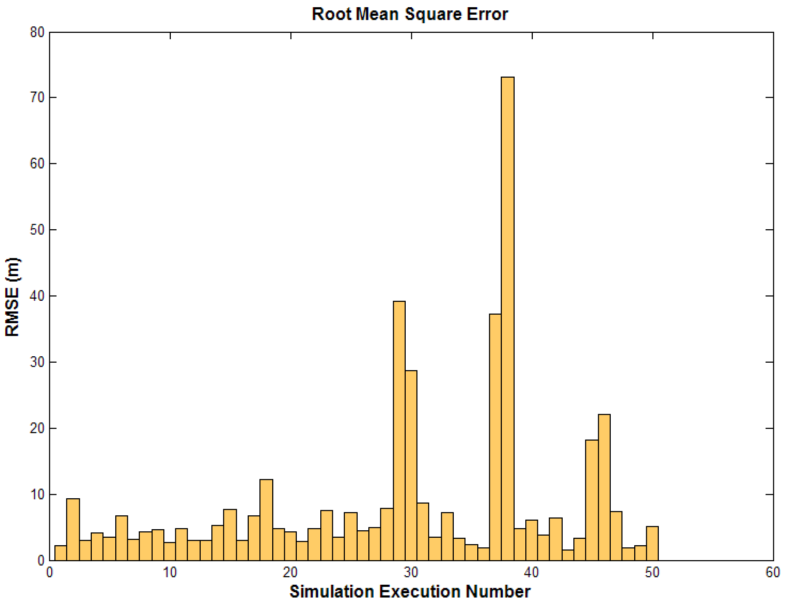


FIGURE 5.20: RMSE for 50 executions of Scenario 2

positioned in the same coordinates, the sampling period the total time of the simulation and the number of particles employed in the implementation of the algorithm, are also left unchanged. The distributions of the additive corruptive noise for the state model and the measurements is given below. The turning rate remained at $w = pi/4$ rad/sec.

$$\mathbf{w}_k \sim \mathcal{N}(0, 2)$$
$$\mathbf{v}_k \sim \mathcal{N}(0, 3.5)$$

The target's initial state is $\mathbf{x}_0 = [10m\ 10m\ 5m/s\ 5m/s]$. In this scenario to increase the level of uncertainty in the system regarding the target's initial position, initial particles are sampled from a Gaussian distribution with zero mean and unity covariance. Finally initial particles for the regime variable are sampled with equal initial probability $\mathbf{P}_0 = [1/3\ 1/3\ 1/3]$ and the transitional probability is set to $m = 0.3$.

The trajectory estimation from executing an exemplar run of this scenario is illustrated in Figure 5.21.

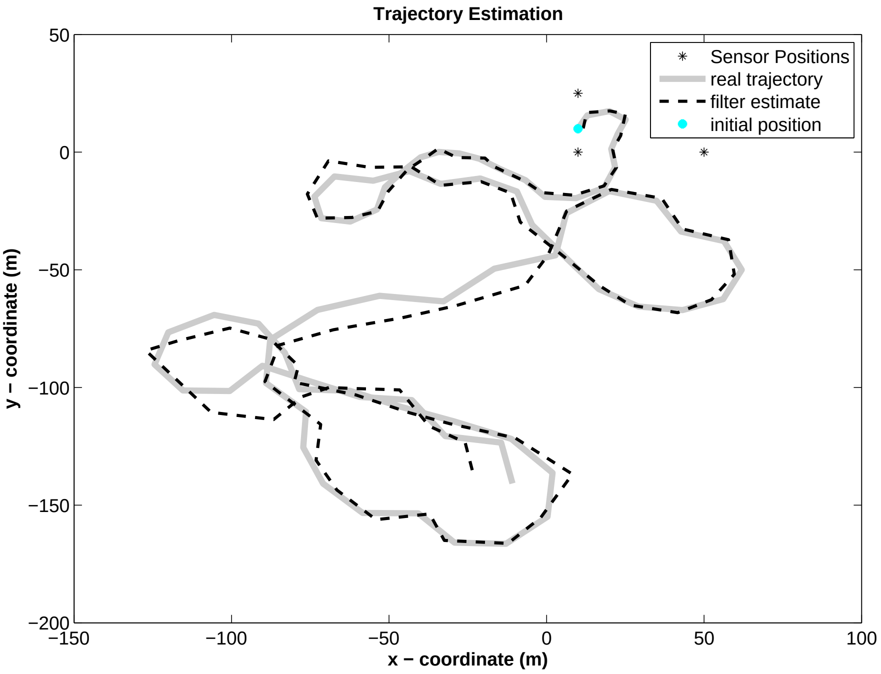


FIGURE 5.21: Trajectory Estimation for the exemplar run of Scenario 3

Again the evaluation of the algorithm's performance is carried out by simulating this scenario for a total of 50 times. The RMSE was calculated for every execution and results are illustrated in Figure 5.22. The average RMSE is $18.41m$. In 39 out of 50 execution the RMSE remains lower than 20m and the average RMSE is $RMSE_{avg} = 18.41m$ over a 200m x 200m area. This investigation reveals that even under noisy conditions with high initial uncertainty and a fast moving target the ROT-MMPF algorithm provides acceptable accuracy and robustness.

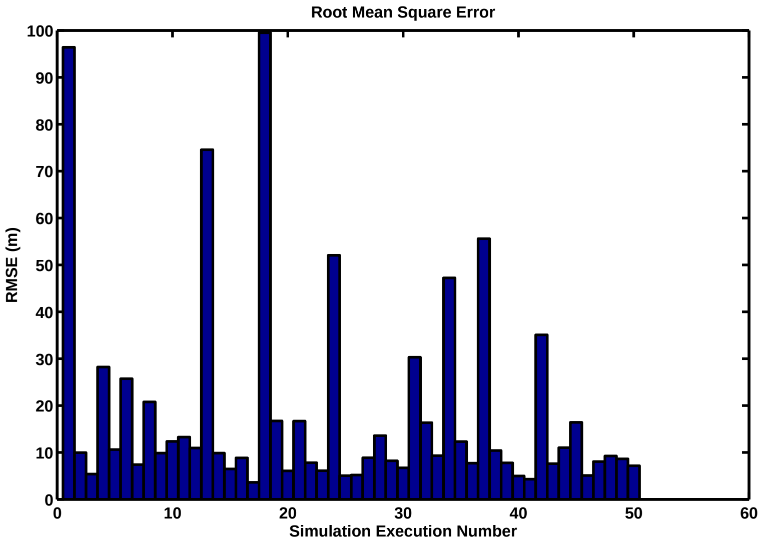


FIGURE 5.22: RMSE for 50 executions of Scenario 3

5.3.3.1 Investigating the Sampling Period

A similar investigation to the one described in Section 5.2.4.1 is conducted for the ROT-MMPF algorithm in order to evaluate the effect of increasing sampling interval in the achieved accuracy. The scenario parameters used are the following;

The anchor nodes are positioned in the following locations, $s_1 = [10m\ 0m], s_2 = [50m\ 0m], s_3 = [10m\ 25m], s_4 = [50m\ 25m]$.The number of particles employed in the implementation of the algorithm, are $N = 500$. The total simulation time is set to T=100ate is set at $w = pi/3$ rad/sec.

$$\mathbf{w}_k \sim \mathcal{N}(0, 1.5)$$
$$\mathbf{v}_k \sim \mathcal{N}(0, 3)$$

The target's initial state is $\mathbf{x}_0 = [10m\ 8m\ 5m/s\ 5m/s]$.

Finally initial particles for the regime variable are sampled with equal initial probability $\mathbf{P}_0 = [1/3\ 1/3\ 1/3]$. The transitional probability is set to $m = 0.85$.

Figure 5.23 illustrated the results obtained from this investigation which demonstrate a similar (linear) pattern as the one observed in the PF-ROT algorithm with increasing sampling interval. Again it is evident from this investigation that increasing the sampling interval results in the accuracy of the system decaying.

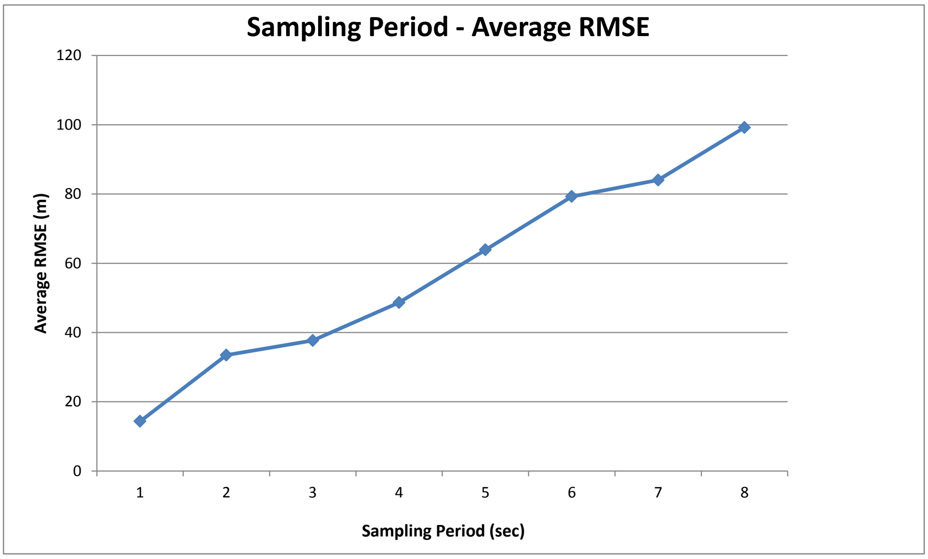


FIGURE 5.23: Sampling Period - Average RMSE for ROT-MMPF algorithm

5.3.3.2 Investigating the Number of Particles

The same investigation with the ROT-PF algorithm is carried for the ROT-MMPF algorithm in order to assess the effect of increasing particle size. The set-up for this investigation is the following. The anchor nodes are positioned in the same coordinates as in the investigation of the sampling period. The sampling period itself was set at $T_s = 3sec$. The total simulation time is $T = 50sec$. The turning rate is set at $w = pi/3$ rad/sec. The noise distributions are defined in the following.

$$\mathbf{w}_k \sim \mathcal{N}(0, 2)$$
$$\mathbf{v}_k \sim \mathcal{N}(0, 5)$$

The target's initial state is $\mathbf{x}_0 = [10m \;\; 8m \;\; 5m/s \;\; 5m/s]$. Initial particles are drawn from a Gaussian distribution with mean $\mu_0 = \mathbf{x}_0 + N(0, 1)$ and unity covariance. Finally initial particles for the regime variable are sampled with equal initial probability $\mathbf{P}_0 = [1/3 \; 1/3 \; 1/3]$ and the transitional probability is set to $m = 0.75$.

Results obtained from this investigation are presented in Figure 5.24. A similar, to the ROT-PF algorithm, behaviour is observed for the ROT-MMPF algorithm. Increasing the number of particles from 500 to 2000 results in a substantial improvement in performance. From that point on the system's accuracy is improved incrementally and based on the discontinuity observed for 3000 we argue that no significant improvement is achieved for a particle size greater than 3000.

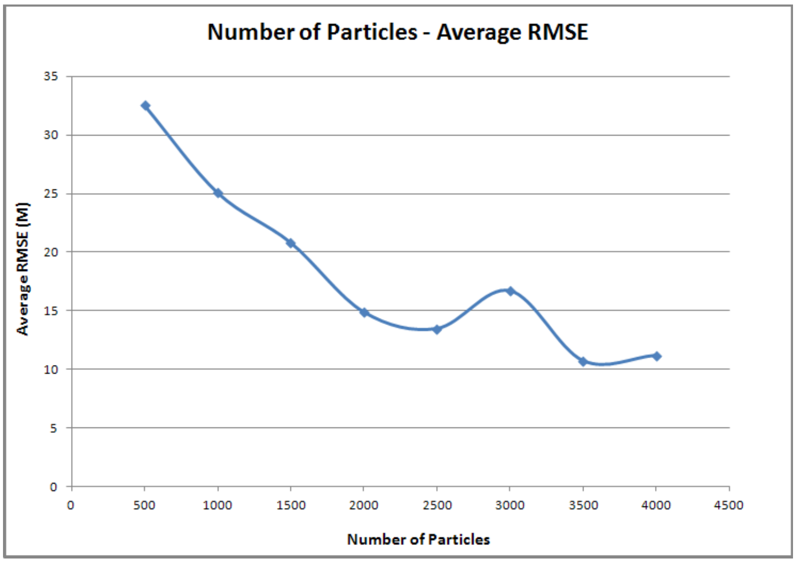


FIGURE 5.24: Sampling Period - Average RMSE for ROT-MMPF algorithm

5.4 Comparison against non-modeling tracking algorithms

5.4.1 Preliminaries

In this section an investigation is presented where the proposed PF-ROT and MMPF-ROT are compared against non-dynamic tracking algorithms. The simulation set-up is similar to the one used for the evaluation of the PF-ROT and MMPF-ROT algorithms. The only difference is that the ranging observations instead of being inserted to the PF algorithms they are used as input to two non-dynamic algorithms, iterative trilateration and non-linear least squares.

The trilateration algorithm is a simple geometric algorithm which in essence computes the planar coordinates of the intersection of three circles. Since, for 2D tracking two coordinates (x,y) need to be estimated ranging information and the location of three anchors are sufficient for trilateration. Considering that three anchors in positions $((x_1, y_1), (x_2, y_2), (x_3, y_3))$ provide range estimates (r_1, r_2, r_3) at each sampling step and the target's unknown coordinates are (x_t, y_t):

$$r_1^2 = (x_t - x_1)^2 + (y_t - y_1)^2 \tag{5.4}$$

$$r_2^2 = (x_t - x_2)^2 + (y_t - y_2)^2 \tag{5.5}$$

$$r_3^2 = (x_t - x_3)^2 + (y_t - y_3)^2 \tag{5.6}$$

$$\tag{5.7}$$

By subtracting the third relationship from the first and the second we obtain the following set of equations with two unknowns (x_t, y_t):

$$r_1^2 - r_3^2 = (x_t - x_1)^2 + (y_t - y_1)^2 - (x_t - x_3)^2 - (y_t - y_3)^2 \tag{5.8}$$

$$r_2^2 - r_3^2 = (x_t - x_2)^2 + (y_t - y_2)^2 - (x_t - x_3)^2 - (y_t - y_3)^2 \tag{5.9}$$

By expanding the relationships we obtain the solution for (x_t, y_t)

$$\begin{bmatrix} (x_3 - x_1) & (y_3 - y_1) \\ (x_3 - x_2) & (y_3 - y_2) \end{bmatrix} \begin{bmatrix} x_t \\ y_t \end{bmatrix} = \begin{bmatrix} \frac{(r_1^2 - r_3^2) + (x_3^2 - x_1^2) + (y_3^2 - y_1^2)}{2}) \\ \frac{(r_2^2 - r_3^2) + (x_3^2 - x_2^2) + (y_3^2 - y_2^2)}{2} \end{bmatrix}$$

or

$$AX = B \Leftrightarrow X = A^{-1}B \tag{5.10}$$

By considering the system of four anchors and rearranging Eq.5.7, the error of each one of the ranging measurements is given from:

$$e_1 = (x_t - x_1)^2 + (y_t - y_1)^2 - r_1^2 \tag{5.11}$$

$$e_2 = (x_t - x_2)^2 + (y_t - y_2)^2 - r_2^2 \tag{5.12}$$

$$e_3 = (x_t - x_3)^2 + (y_t - y_3)^2 - r_3^2 \tag{5.13}$$

$$e_4 = (x_t - x_4)^2 + (y_t - y_4)^2 - r_4^2 \tag{5.14}$$

$$\tag{5.15}$$

By beginning from a starting point $\mathbf{x}_{b,t}$, the non-linear least squares (LSQ) algorithm tries to solve the system of equations in Eq.5.15 at every time step (when new ranging measurements become available), by finding the vector of unknowns ($\mathbf{x}_est = (x_t, y_t)$) that best fits the system in the sense that minimizes the sum of square of the errors. That is,

$$E = \sum_{i=1}^{4} e_i \tag{5.16}$$

In this investigation the starting point for each time step was taken as the estimation of the non-linear LSQ algorithm at the previous step ($\mathbf{x}_{b,t} = \mathbf{x}_{est,t-1}$) while the for the first iteration, a point that represents the system's initial knowledge of the target's first location (similar to the PF algorithms) is used.

5.4.2 Comparative Results

Simulations are conducted to compare the performance of the three algorithms PF-ROT/MMPF-ROT,iterative trilateration and non-linear least squares. We chose the noise levels as the varying parameter in this investigation and kept the rest of the parameters constant. The scenario setup was set as follows; four anchors deployed in $s_1 = [20m\ 20m], s_2 = [20m\ 60m], s_3 = [80m\ 20m], s_4 = [80m\ 60m]$, $T_s = 1sec$, total time $T = 50sec$, $\mathbf{x}_0 = [20m15m1m/s1m/s]$. The distribution to sample initial particles from for the PF algorithms as well as sample the beginning point of the non-linear LSQ algorithm is a Gaussian distribution with mean $\mu_0 = \mathbf{x}_0 + N(0,1)$ (which is the target's real position at t=0 corrupted from a Gaussian with zero mean and unity standard deviation) and covariance $S_0 = I$. Both the PF-algorithms employ 500 particles. Finally for the multiple model case the transition probability was set to $m = 0.75$.

In this investigation varying noise levels were applied. For the state noise took, 3 levels were chosen; low-noise (w $_{\text{k}}$ = 0.5), medium (w $_{\text{k}}$ = 1) and high (w $_{\text{k}}$ = 2). For each of the state noise levels the three algorithms were evaluated for varying levels of observations noise (n $_{\text{k}}$ = 0.05,0.5,1,2,3,4,5,6). Since trilateration and non-linear least squares are non-modeling approaches they are primarily affected by the observations noise.

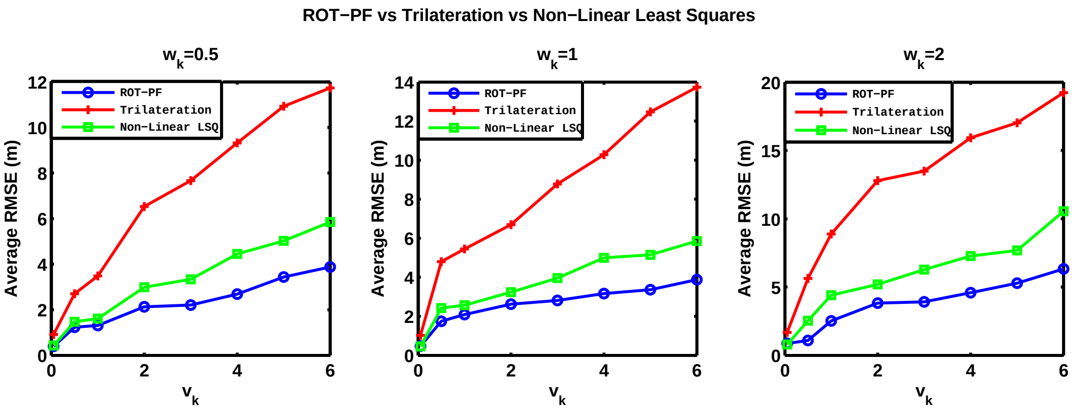


FIGURE 5.25: Comparison between the ROT-PF, Trilateration, Non-linear least squares algorithms for different noise levels (w$_{\text{k}}$,v$_{\text{k}}$)

The obtained performance, in terms of average RMSE, results comparing the single-model and the multiple-model case are illustrated in Figure5.25 and Figure5.26 respectively. The three algorithms were simulated, for permutation of noise levels, for a total of 50 times and in each execution the RMSE for each algorithm was calculated. The final results are produces by averaging the RMSE from the all the executions. From Figure5.25 and Figure5.26 it is evident that the PF-based algorithms outperform both non-linear LSQ and trilateration, particularly in high-noise conditions ($v_k > 2$). It is also clear that in absence of observations noise ($v_k = 0.05$) the three algorithms have almost identical performance. Nevertheless, such ideal conditions are not expected in real-world applications, thus the robustness ($RMSE < 5m$ over a 60m x 40m area) that PF-based methods achieve justifies their choice as tracking algorithms over traditional methods.

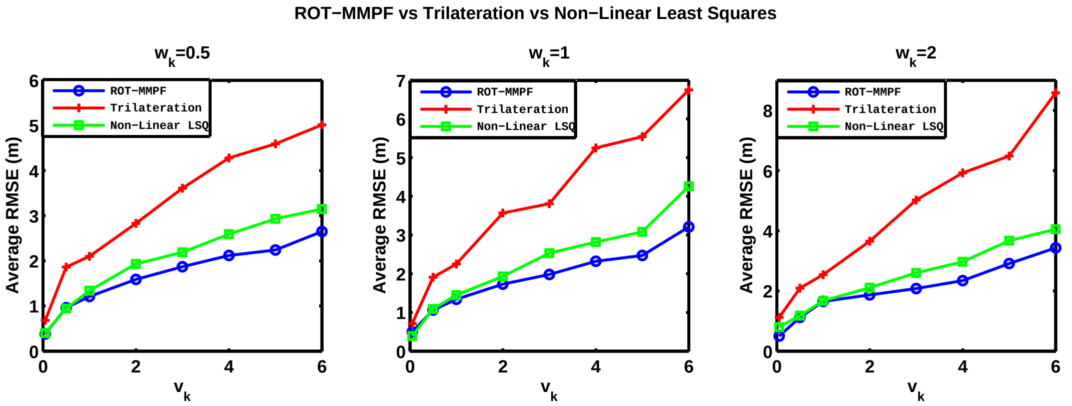


FIGURE 5.26: Comparison between the ROT-MMPF, Trilateration, Non-linear least squares algorithms for different noise levels (w_k,v_k)

5.5 Evaluation of the tracking system given the obtained accuracy of the ToF ranging method

The ToF method analysed in Chapter 3 is intended to provide the tracking system with the required point to point ranging between the anchor nodes and the target node. Having in mind the performance results of the ranging technique after being implemented in the EZ430-RF2500 hardware and based on the developed simulation environment we run simulations given the achieved accuracy of the ranging method.

From Section 3.8 the expected standard deviation of the two-way ToF ranging method is $\sigma_{tof} = 3.7m$ for an outdoor deployment and at the 500kbps datarate. By importing the experimental deduced accuracy for our ranging technique to the simulation environment developed and executing the simulations given the $\sigma_v = 3.7m$ standard deviation the following results are obtained for 100 executions of a similar scenario to the one in Section 5.3.2. The good performance remains for the accuracy limits posed by the two-way ToF technique. Figure 5.27 illustrates a histogram of the RMSE for 100 executions of the scenario under investigation. In 77 of the simulations runs an RMSE smaller than $10m$ is achieved and the average RMSE is $11.72m$.

Although in the full-scale experiments (presented in Chapter 6) the target moved at a much slower velocity and with less manoeuvres this investigation aims at justifying the ability of the MMPF algorithms to perform accurately taking into account the achieved accuracy of the two-way ToF method.

5.6 Single Model vs Multiple model

Here a comparison of the two approaches we employ to form the state equation, takes place. A scenario where the target follows a predefined trajectory which involves a quick two-turn manoeuvre is simulated with the single and multiple-model approach. As we can observe from

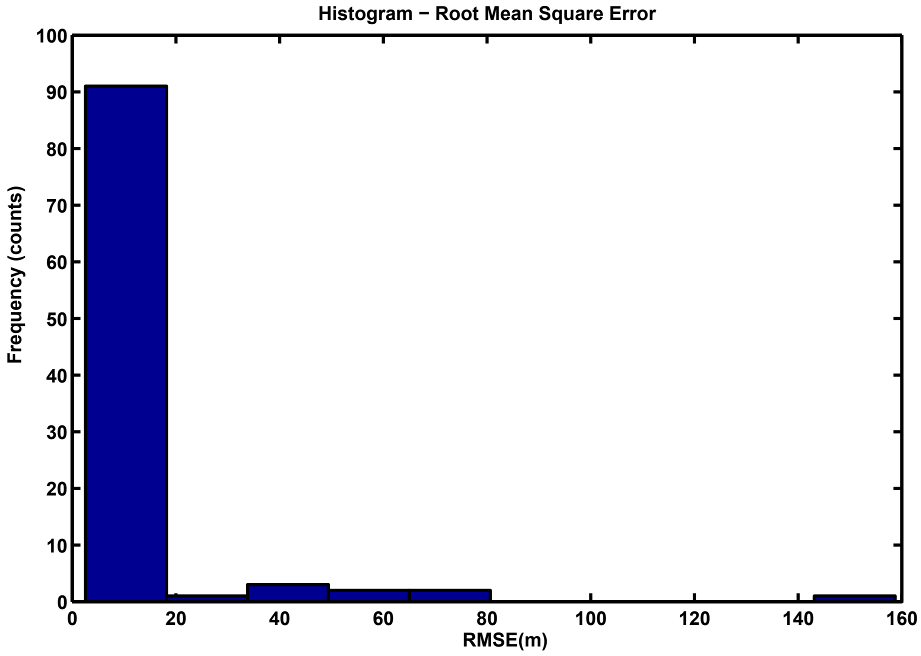


FIGURE 5.27: Histogram of the RMSE considering the ToF ranging accuracy

the Figure 5.28 the ROT-MMPF algorithm can successfully track the target's manoeuvres. On the other hand the ROT-PF algorithm keeps track of the target during the first turn but looses focus in the second turn and requires some time until the algorithm's output converges back to the target's real trajectory. The ROT-PF algorithm converges back to the the target's real trajectory after the target's resumes its straight line pattern which is adequately represented with the CV model.

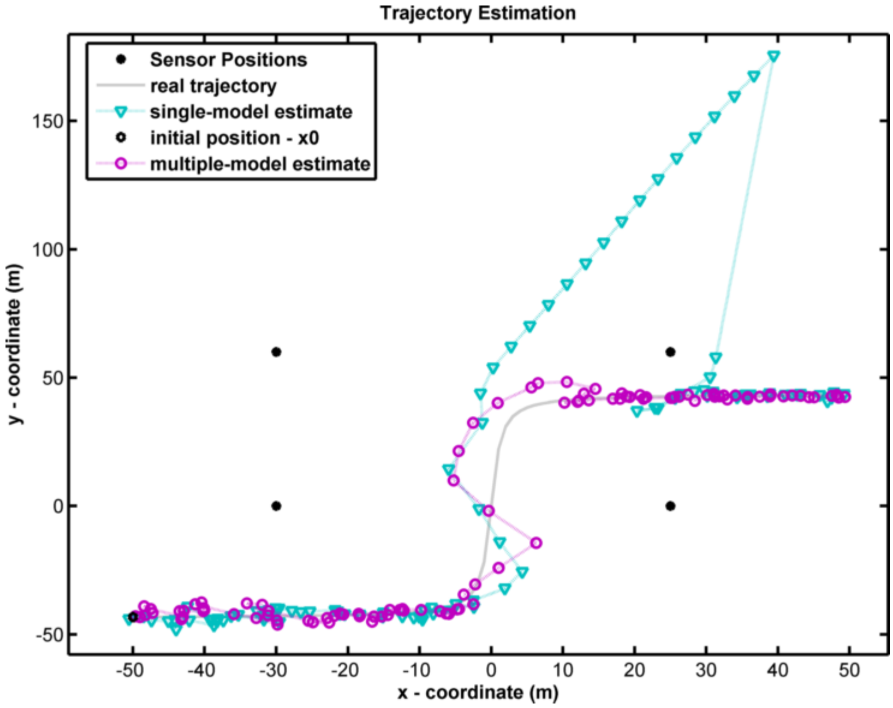


FIGURE 5.28: Comparison of the two models under a quick manoeuvring scenario

5.7 Number of Deployed Sensors

A number of approaches in the area of PF tracking algorithms in multisensor environments stress that the algorithm's performance can be improved if there is an increase in the number of observers that provide the system with data [2]. This argument was investigated for the proposed system and the outcome of this investigation was that no significant improvement was achieved just because an increased number of anchor nodes was employed. The performance in the presence of more than four observers was similar to the one demonstrated, for the same conditions, with the initial deployment of four anchor nodes.

To the author's belief the number of anchor nodes must be kept at a minimum level which however guarantees good performance. Adding wireless nodes will result in increasing the network communication overhead as well the algorithm's execution time (since data from more nodes need to be fused). There are situations however where an increase in the number of anchor nodes is mandatory due to other reasons. For example when the area that needs to be monitored exceeds the communication range of the four anchor nodes. In that case additional nodes must be deployed, however the number of anchor that contributes for the estimation of each track can remain at the predefined minimum.

5.8 Cramer-Rao Lower Bounds

The Cramer-Rao lower bound (CRLB) is a theoretically constructed lower bound of the second-order (mean-squared) error performance of an unbiased estimator. It is used extensively in non-linear dynamic estimation problems where the recursive optimal Bayesian solution can not be derived in closed form. In these situations the CRLB is utilised as a benchmark for evaluating the performance of sub-optimal algorithms [23,120]. The mathematical derivation of the CRLB has itself attracted considerable research interest . Nowadays in the majority of the situations the CRLB is calculated recursively with the use of the Fisher information matrix (the inverse of this matrix is taken as the CRLB). The bound calculated using the Fisher information matrix is called "posterior" since it is applicable in systems modeled with nonzero process noise [37,111,122].

In this section we derive the posterior CRLB for the two approaches (single and multiple model) that we used to formulate the state dynamics of the proposed range-only tracking system. Moreover simulations are presented to assess the performance of the proposed system with the theoretically derived lower bound.

For a discrete nonlinear estimation problem which is described from the model in Eq. 4.1 and Eq. 4.2, the covariance matrix $\mathbf{P}_k$ of an unbiased estimator $\hat{\mathbf{x}}_k$ of the state vector at time k has a lower bound (CRLB) which is expressed as [98] :

$$\mathbf{P}_k \triangleq \mathbb{E}\{(\hat{\mathbf{x}}_k - \mathbf{x}_k)(\hat{\mathbf{x}}_k - \mathbf{x}_k)^T\} \geq \mathbf{J}_k^{-1} \tag{5.17}$$

where $\mathbf{J}_k$ is the Fisher's information matrix defined as.

$$\mathbf{J}_k = \mathbb{E}\{[\nabla_{\mathbf{x}_k} \log p(\mathbf{x}_k|\mathbf{Z}_k)][\nabla_{\mathbf{x}_k} \log p(\mathbf{x}_k|\mathbf{Z}_k)]^T\} \tag{5.18}$$

Tichavsky *et al.* provided an elegant way for the recursive computation of the information matrix. From [122] the information matrix (at a given time instance) can be expressed as:

$$\mathbf{J}_k = \mathbf{D}^{22}_{k-1} - \mathbf{D}^{21}_{k-1}(\mathbf{J}_{k-1} + \mathbf{D}^{11}_{k-1})^{-1}\mathbf{D}^{12}_{k-1} \tag{5.19}$$

where:

$$\mathbf{D}^{11}_{k-1} = -\mathbb{E}\{\nabla_{\mathbf{x}_{k-1}}[\nabla_{\mathbf{x}_{k-1}} \log p(\mathbf{x}_k|\mathbf{x}_{k-1})]^T\} \tag{5.20}$$

$$\mathbf{D}^{12}_{k-1} = -\mathbb{E}\{\nabla_{\mathbf{x}_k}[\nabla_{\mathbf{x}_{k-1}} \log p(\mathbf{x}_k|\mathbf{x}_{k-1})]^T\} \tag{5.21}$$

$$\mathbf{D}^{21}_{k-1} = -\mathbb{E}\{\nabla_{\mathbf{x}_{k-1}}[\nabla_{\mathbf{x}_k} \log p(\mathbf{x}_k|\mathbf{x}_{k-1})]^T\} = [\mathbf{D}^{12}_{k-1}]^T \tag{5.22}$$

$$\mathbf{D}^{22}_{k-1} = -\mathbb{E}\{\nabla_{\mathbf{x}_k}[\nabla_{\mathbf{x}_k} \log p(\mathbf{x}_k|\mathbf{x}_{k-1})]^T\} - \mathbb{E}\{\nabla_{\mathbf{x}_k}[\nabla_{\mathbf{x}_k} \log p(\mathbf{z}_k|\mathbf{x}_k)]^T\} \tag{5.23}$$

To begin the recursion, the information matrix at time $k = 0$ is taken from the initial density of the state vector $p(\mathbf{x}_0)$ as follows.

$$\mathbf{J}_0 = \mathbb{E}\{[\nabla_{\mathbf{x}_0} \log p(\mathbf{x}_0)][\nabla_{\mathbf{x}_0} \log p(\mathbf{x}_0)]^T\} \tag{5.24}$$

If the initial density is chosen to be Gaussian ($p(\mathbf{x}_0) = N(\mathbf{x}_0; \mu_o, \mathbf{P}_0)$) then the Eq. 5.24 becomes $\mathbf{J}_0 = \mathbf{P}_0^{-1}$

The generic recursive way to obtain the CRLB is summarized in Eq. 5.19 - Eq. 5.23. Based on that recursion we derive the bound for the two cases of range-only tracking that we proposed in Chapter 4.

5.8.1 CRLB for the ROT-PF algorithm

In this case the state equation is modeled with the use of the constant velocity model and thus is linear (Section 4.4.2.1). The observations equation is nonlinear since it is formulated as the Euclidean distance (range) between the anchors and the target. Considering the process noise to be zero ($\mathbf{w}_k = 0$), which means a purely deterministic trajectory, the recursion of Eq. 5.19 becomes:

$$\mathbf{J}_k = [\mathbf{F}^{-1}]^T \mathbf{J}_{k-1} \mathbf{F}^{-1} + \sum_{i=1}^{N_s} H^T_{k,i} \mathbf{R}^{-1}_{k,i} \mathbf{H}_{k,i} \tag{5.25}$$

where: N_s is the number of anchors, $H_{k,i}$ is the Jacobian of the measurements equation ($\nabla_{\mathbf{x}_k} h(\mathbf{x}_k)$) with respect to the state vector, evaluated at the true value of $\mathbf{x}_k$ and $\mathbf{R}_{k,i}^{-1}$ is the inverse of the observations noise covariance matrix.

The Jacobian matrix of the measurements equation is a 1×4 matrix given in Eq. 5.26.

$$\begin{aligned} H_{k,i}[1,1] &= \frac{(x_k - x_i)}{\sqrt{(x_k - x_i)^2 + (y_k - y_i)^2}} \\ H_{k,i}[1,2] &= \frac{(y_k - y_i)}{\sqrt{(x_k - x_i)^2 + (y_k - y_i)^2}} \\ H_{k,i}[1,3] &= 0 \\ H_{k,i}[1,4] &= 0 \end{aligned} \tag{5.26}$$

To calculate the CRLB of the ROT-PF algorithm and evaluate the performance of the proposed system in accordance to the CRLB we simulate the following scenario. Four anchor nodes are considered deploy at coordinates $s_1 = [10\ 0], s_2 = [50\ 0], s_3 = [10\ 25], s_4 = [50\ 25]$. In line with the theoretical assumptions the process noise is considered zero while the observations noise is considered Gaussian. The target's initial state vector is $x_0 = [10m\ 10m\ 0.1m/s\ 0.1m/s]$ and initial particles are sampled from a Guassian distribution with $\mu_0 = x_0 + N(0,1)$ and covariance matrix $\mathbf{S}_0 = \mathbf{J}_0^{-1} = diag[1\ 1\ 1\ 1]$. This scenario was simulated for 400 time steps for a total of 500 Monte Carlo runs and the CRLB for position was calculated as:

$$\text{CRLB}_{pos} = \sqrt{\mathbf{J}_k^{-1}[1,1] + \mathbf{J}_k^{-1}[2,2]} \tag{5.27}$$

where $\mathbf{J}_k^{-1}[1,1]$ and $\mathbf{J}_k^{-1}[2,2]$ are the diagonal elements of the information matrix corresponding to the CRLB for x- and y- coordinates respectively.

Results are illustrated in Figure 5.29, from where it is clear that the achieved RMS error follows a similar trend as the CRLB.

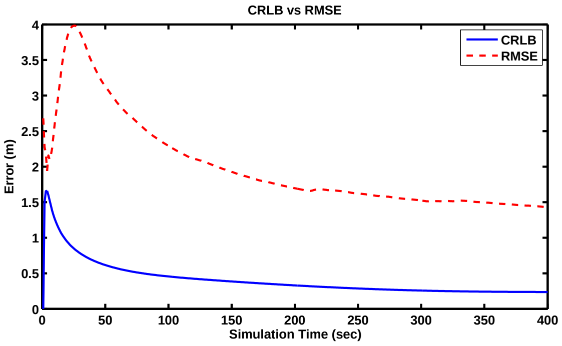


FIGURE 5.29: Comparison of CRLB and RMSE for 500 runs

5.8.2 CRLB for the ROT-MMPF algorithm

In the multiple-model case the derivation of CRLB is being done using the same approach as for the single-model case presented previously. However, in this case the CRLB can not be derived analytically due to the fact that this would require the differentiation of terms like log$p(r_k|r_{k-1})$ which is not possible for a discrete regime variable [98]. Consequently, taking into account a zero process noise system, the CRLB is computed in a similar to the single-model case, recursive way for a given sequence of regime variables [19].

Considering a specific sequence of regime variables $R^l_k \triangleq \{r^l_1, r^l_2, \ldots, r^l_k\}$, with $l = 1, 2, \ldots, s^k$ being the possible regime values up to time k, the covariance of an estimator of the state vector is given by Eq. 5.17 conditioned on the particular regime sequence.

$$\mathbf{P}_k \triangleq \mathbb{E}\{(\hat{\mathbf{x}}_k - \mathbf{x}_k)(\hat{\mathbf{x}}_k - \mathbf{x}_k)^T | R^l_k\} \geq \mathbf{J}^{-1}_k \tag{5.28}$$

Fisher's information matrix $\mathbf{J}^l_k$ is computed for the specific regime variable sequence R^l_k from Eq. 5.33. The conditional (on the regime sequence R^l_k) CRLB is given as the inverse of the information matrix:

$$CRLB^l(\mathbf{x}_k) \triangleq [\mathbf{J}^l_k]^{-1} \tag{5.29}$$

Considering that at time k the regime variable can be any of the possible s^k different permutations, the unconditional CRLB is calculated as the expectation of the conditional bounds [19],

$$\begin{aligned} CRLB(\mathbf{x}_k) &= \mathbb{E}\{CRLB^l(\mathbf{x}_k)\} \\ &= \sum_{l=1}^{s^k} pr(R^l_k)[\mathbf{J}^l_k]^{-1} \end{aligned} \tag{5.30}$$

where $pr(R^l_k)$ is the forward probability of a particular sequence of states of the first order Markov chain defined by the transition probability matrix $\Pi = [\pi(i,j)]$ (see Eq. 4.35) and the initial probability matrix $p(\mathbf{r}_0 : p_i \triangleq prr1 = i, i = 1, 2 \ldots, s)$. That probability is calculated as:

$$pr(R^l_k) = \left[\prod_{j=1}^{s} p_j^{\delta(r^l_1, j)}\right] \prod_{i=1}^{s} \left[\prod_{j=1}^{s} \pi_{ij}^{n_{ij}(R^i_k)}\right] \tag{5.31}$$

where $\delta(r_i, j) = 1$ if $r_i = j$ and "0" elsewhere and

$$n_{ij}(R_k^l) = \sum_{\eta=2}^{k} \delta(r_{\eta-1}, i)\delta(r_\eta, j) \tag{5.32}$$

The computational complexity of the CRLB defined in Eq. 5.30 increases exponentially with time and requires the enumeration of the growing regime sequences [46]. As a result this bound can only be calculated for small numbers of k [98]. For this the majority of books on target tracking [18, 21] consider an a-priori known regime sequence R_k^ϵ. This sets the probability of that particular sequence to "1" and the probability of any other regime sequence to "0" in Eq. 5.30. The CRLB in this case is simplified to the following equation (similar to Eq. 5.33.

$$\mathbf{J}_k^\epsilon = [[\mathbf{F}_{k-1}^\epsilon]^{-1}]^T \mathbf{J}_{k-1}^\epsilon [\mathbf{F}_{k-1}^\epsilon]^{-1} + \sum_{i=1}^{N_s} [H_{k,i}^\epsilon]^T \mathbf{R}_{k,i}^{-1} \mathbf{H}_{k,i}^\epsilon \tag{5.33}$$

The evaluation of the CRLB for the MMPF took place both with the enumeration method as well as for a purely deterministic trajectory (absent process noise, apriori known regime sequence). The simulation setup was similar to the one for the investigation of the CRLB for ROT-PF algorithm. The target's initial state vector is $x_0 = [10m \ 10m \ 1m/s \ 1m/s]$ and initial particles are sampled from a Guassian distribution with $\mu_0 = x_0 + N(0,1)$ and covariance matrix $\mathbf{S}_0 = \mathbf{J}_0^{-1} = diag[1\,1\,1\,1]$. The manoeuvring turning rate was set to $w_r = \pi/3$. Initial particles for the regime variable were sampled with equal probability $P_0 = [1/3 \ 1/3 \ 1/3]$ and the transitional probability is set to $m = 0.95$.

In the enumeration method the total time steps were set to $K = 12$. Results of 500 Monte Carlo runs are illustrated in Figure 5.30.

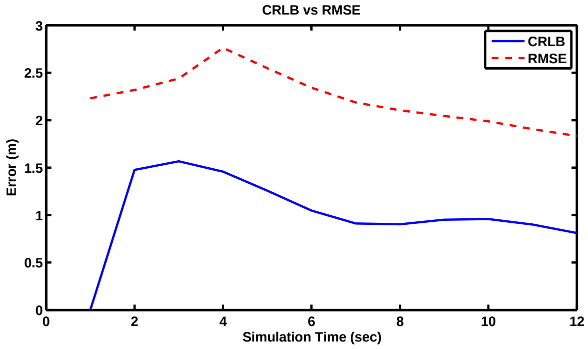


FIGURE 5.30: CRLB for the MMPF-ROT algorithm

A realisation of a regime sequence for $K = 100$ time steps was produced and it was used to calculate the CLRB in a deterministic way for that particular regime sequence. Figure 5.31 illustrates the trajectory that corresponds to the regime sequence. Following, this scenario was simulated for a total of 500 Monte Carlo runs and the results are shown in Figure5.32.

From Figure 5.30 and Figure 5.32 it is clear that the RMSE of the multiple-model system follows a similar trend as the theoretical CRLB.

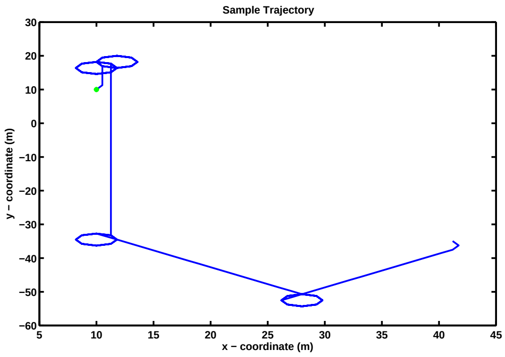


FIGURE 5.31: Sample Trajectory for CRLB computation

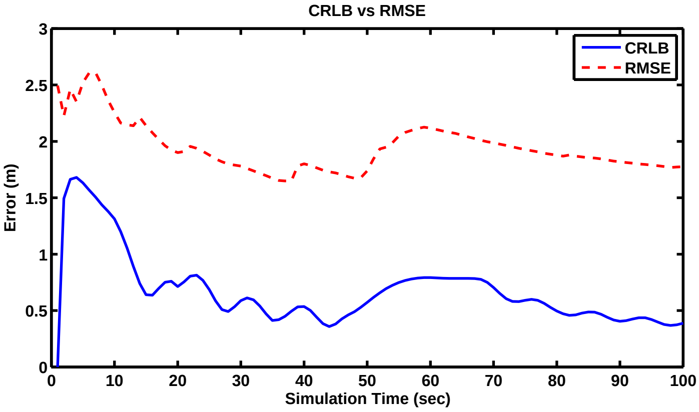


FIGURE 5.32: Comparison of CRLB and RMSE for 500 runs

5.9 Summary and Conclusions

This chapter provided an extensive investigation of the proposed tracking system. Although this investigation was carried out on a simulation environment, a number of factors that would affect a real-world implementation (like the update interval) are also included. The following conclusions are deduced based on the results obtained from the simulation evaluation of the tracking system.

→ The system's performance depends upon the following factors. The accuracy of the system's "knowledge" regarding the target's initial position, the additive disruptive noise distributions (the measurements noise represents the accuracy of the ranging technique employed), the time interval between successive estimations of the target's state (which includes the sampling period, the algorithms execution time and the successful aggregation of all data) and finally the number of particles employed in the implementation of the tracking algorithms

→ The additive noise depends heavily on the accuracy of the ranging technique (measurements noise) employed to provide the measurements and also on how well the evolution of the target's dynamics are modeled with the motion model (state noise) used in the algorithm. The system's knowledge regarding the target's initial position is also an application-specific factor. In applications where the target itself initializes tracking the initial knowledge can be very good. One the other hand in situations where the system must both detect and track the target, initial knowledge depends on the detection operation

→ The parameters that may lead to the increase of the time interval between tracks where analyzed in section 5.2.4.1. Subsequently, the number of particles used in the tracking algorithm implementation must be tailored to the system processing abilities since increasing the number of particles may also lead to an increase in the execution time of the algorithm

→ Specifically, these factors presented previously affect the system's accuracy in the following way:

  - If all of these parameters are in average levels, the achieved accuracy is quite promising. Particularly the particle size of the algorithm can be further reduced to 250-400 particles, resulting in lower execution time and complexity provided that the rest of the parameters are in "ideal" levels
  - Accuracy deteriorates in the event where two or more of these factors are not in good levels. However, unless these factors reach derogatory levels there is good possibility for the system to perform more than adequately (RMSE $<$ 20m) over a 200m x 200m area even under non-ideal situations
  - The reduction in accuracy can be amended by employing a higher number of particles. The ROT-PF algorithm demonstrated a 35% improvement in its performance by employing four times more particles. In addition, the ROT-MMPF algorithm improves its performance by 56% when a similar (4 times) increase in its particle size takes place

→ Tracking of manoeuvring targets is supported with the use of the multiple models that form the state equation. Simulations reveal the soundness of this choice

→ It has been demonstrated that the attained RMSE for both the ROT-PF and ROT-MMPF algorithms confirms the trend of the theoretical Cramer-Rao lower bound

Chapter 6

Implementation and Evaluation of the Tracking System

Chapter 3 presented the two-way ToF ranging method for wireless embedded nodes while the design of a range-only tracking system was discussed in Chapter 4 and simulated in Chapter 5. In this Chapter these developments are combined on a ToF tracking system for embedded nodes. Section 6.1 provides details regarding the implementation of the system on the EZ430-RF2500 nodes. In Section 6.2 we present an energy consumption investigation of the ToF ranging method. The details of the experimentation setup are given in Section 6.3. The results from executing 25 experiments are then discussed in Section 6.4. Also in this section we compare the experimental results with theoretical results obtained under simulations of similar scenarios to the ones tested in our outdoor experiments. Ultimately, Section 6.5 concludes this Chapter.

6.1 Design and implementation of the tracking system on T.I. EZ430-RF2500 platform

The implementation procedure comprises of integrating the various operations of the tracking system on hardware. The software that was developed and used to program the embedded nodes is heavily based on the software which was used in the two-way ToF ranging method and was presented in Section 3.6.3.

According to the predefined scenario in Section 1.2, tracking of a mobile target is achieved through the PF algorithms presented in Chapter 5 by using a number of anchor nodes deployed in known positions to provide the necessary ranging information through the ToF range method. One of the nodes was chosen as the mobile target. The development of the required software for that scenario was thus divided into three categories which are given in the following and attempts to conduce to the production of a tracking system that demonstrates, acceptable latency which allows for real-time operation, efficient data fusion and accurate performance.

6.1.1 Anchor nodes software

The anchor nodes are a number of embedded nodes deployed in known locations and their mission is to interact with the target-node in order for the ranging data to be extracted from the two-way ToF technique. Due to this and in order for the implementation to be as efficient as possible, the "responder" software code as it was analysed in Section 3.6.3 was used for the anchor nodes as more suitable for their role.

6.1.2 Target Node Software

The choice for using the *responder* software for the anchor nodes was dictated after it was deduced that the *requester* software is more efficient to be implemented on the target-node. Considering the functionality of the requester node, a decision was made where a single target-*requester* and four anchors-*responders* would result to a more efficient system than having four anchors-*requesters* and one target-*responder*. The four requesters - one responder set up would require some extra networking to avoid anchors transmitting ranging messages simultaneously. This can be achieved by ordering the four anchors and including a message exchange between successive anchors where one anchor informs the other that has completed ranging so the next anchor can take over. On the other hand the one requester - four responders approach has no such demand because the requester controls the entire operation. In the resulting system the target-node initiated the communication between itself and the anchors. Initially, the target-node engages in a ranging process with the first anchor node; as soon as the nominal number of transactions is achieved and the two-way ToF estimate is calculated, the value is fused to the central node and the target-node carries on and engages in a ranging operation with the next anchor node. A data cycle is completed when the target-node has acquired one ranging estimate from every anchor node. The four estimates are fused to the central node at the moment of their production sequentially, therefore the target node is not required to store any ranging estimations from previous anchors.

As it was highlighted in Chapter 3, the *responder* device enters the two-way ToF process only after being initiated by the *requester*. As analysed before the target-node must exchange ranging transactions on a one-to-one basis with each one of the anchors within a single sampling period. To guarantee this and prevent message collisions between the target-node and the anchors, the anchors' CC2500 radio is programmed on different communication channels. This approach allows the target-node to complete the ranging process with a specific anchor without the risk of another anchor node intercepting this process which would result in faulty time readings. The target-node is aware of the communication channel that each anchor operates on and loops through these during every sampling interval.

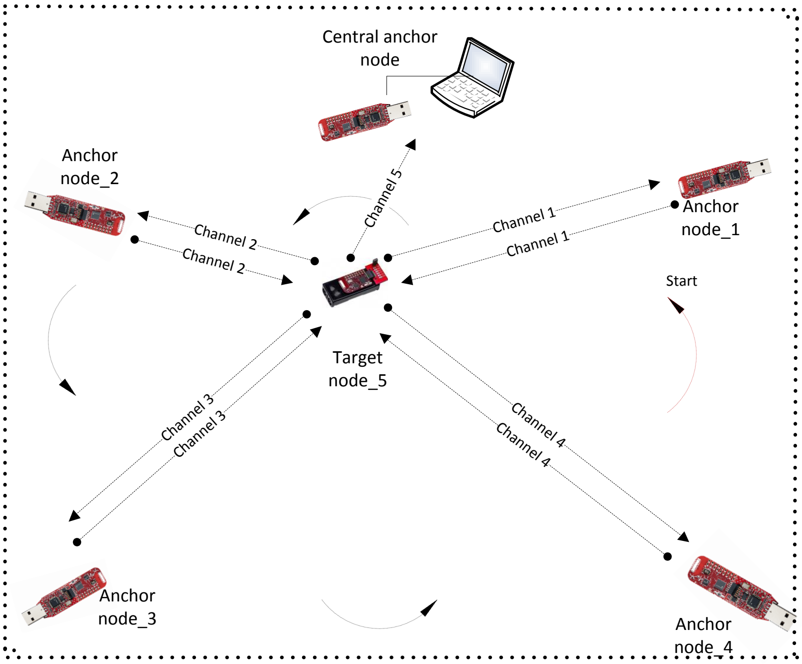


FIGURE 6.1: The communication cycle between the target and the anchors

6.1.3 Development of the software for the central node

The central node is responsible for the collection of ranging estimates and for the execution of the tracking algorithm. Although efforts took place in order to implement the PF algorithms on EZ430-RF2500 this was not possible because of the processing demands of the algorithms. The PF algorithms require substantial memory storage for their implementation, mainly because of the need to generate and assign weights to a relatively large number of particles. This was proven impossible in the memory-constrained EZ430-RF2500 platform. Additionally the operation of calculating the likelihood (see Eq.4.34, Eq.4.36) of each particle based on the incoming set of observations, involves complex mathematical operations to be executed in every iteration of the filter. By considering the processing capabilities of the available MCU for WSNs, even if the memory constraint was non-existent, the real-time operation of the system would be jeopardised since it would require a considerable amount of time for the embedded processor to compete the required operations. Attempts to implement a PF algorithm on dedicated hardware, present in the literature [25, 104] focus on minimising the complexity of the distinct operations of the PF algorithms (particle generation, weights calculation, resampling). Nevertheless, the proposed architectures are implemented on powerful mains-powered platforms such as FPGAs. In the concluding Chapter ideas are discussed for a potential architecture that would enable the implementation of the PF algorithms on a wireless node with increased processing and memory capabilities.

Several important challenges were considered in the design of the central-node software. Firstly the issue of time synchronization between the data acquired and the estimates produced. As soon as the central node acquires the required data from the anchor nodes, it initializes the execution of the tracking algorithm and produces an estimation. Following, a new set of observations will be available at the central node and the algorithm is executed based on the new set of data to produce the estimate of the next time instant. It is imperative to ensure, that both the operations of continuous data accumulation and execution of the algorithm will run in the central node effectively. The data fusion scheme must also ensure that data from different nodes do not collide.

Under these conditions a choice was made to employ a laptop computer as a central node to execute the PF tracking algorithm. An EZ430-RF2500 node acts as the bridge between the target-node and the laptop. The target-node communication with the designated EZ430-RF2500 takes place in a pre-specified wireless channel different from the ones that are used between the target-node and the anchor nodes. The designated node forwards each ranging estimate to its UART port and then the software on the laptop takes over for further processing.

The tracking algorithm (either MMPF-ROT or PF-ROT) is implemented as a MATLAB routine. Another MATLAB script initializes the procedure. At the initial stages of this script a number of parameters must be set. These involve the noise levels, the distribution details from which the initial particles for PF algorithm are sampled, the target's initial location before the commencement of tracking, the anchor's positions and the connection of the software script to the serial port where the central node is connected are all set before the initialisation of the tracking operation.

The front-end script monitors the serial port where the EZ430-RF2500 is connected and is programmed to signal an interrupt whenever the required amount of bytes (i.e. four range estimates from each anchor node) is present at the laptop's serial port. A MATLAB interrupt routine is scheduled to run whenever the nominal number of bytes is reached and stores the range estimates into the MATLAB environment. The PF algorithm routine is called within that interrupt routine and is executed given the acquired ranging data. After the tracking algorithm runs to completion another set of ranging data is available and the procedure enters the next cycle. The results from each successive execution are stored on MATLAB workspace.

Different to our simulation experiments, discussed in Chapter 5, the sampling interval in the real-world experiments was not set to a constant value. The sampling interval is in essence the time elapsed between two successive executions of the tracking algorithm or in terms of our model, the time elapsed between the current and the previous state vectors. It is an important parameter, which also affects the state model that is employed to describe the dynamics of the target's motion. In the real-world experiments the sampling interval is affected primarily by the amount of time required to collect a ranging estimate from all the anchors.

In the event that the connection between the target and the anchors is not very good a greater number of transmissions may be required in order to reach the nominal number of 100 two-way transactions, which is required to obtain a ToF range estimate between the target and the respective anchor. Subsequently this will result in an increase of the sampling interval since more time is going to be required to obtain the range estimates. To avoid using an erroneous sampling interval in our model, an adaptive scheme is employed. The sampling interval is calculated in MATLAB as the required amount of time to obtain the four range estimates. The value of a real-world clock is captured whenever an interrupt (that signals the availability of four range estimates in the serial port) is raised. By subtracting the previous value of that clock the sampling interval can be calculated. Using this method, we guarantee that the state-update model that is employed takes into account the varying amount of time that has lapsed between two successive executions of the tracking algorithm.

A block diagram of the operations that are carried out in the system's main MATLAB script illustrated in Figure 6.2.

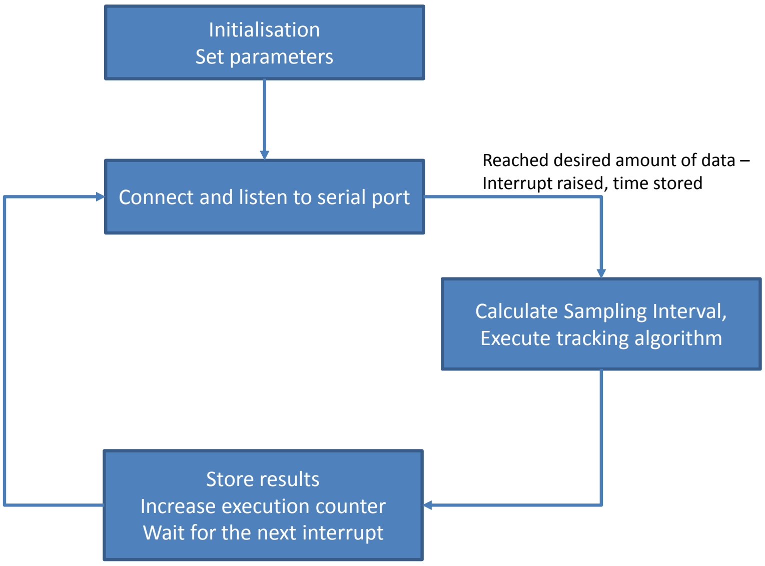

Initialisation
Set parameters
Connect and listen to serial port
Reached desired amount of data –
Interrupt raised, time stored
Calculate Sampling Interval,
Execute tracking algorithm
Store results
Increase execution counter
Wait for the next interrupt

FIGURE 6.2: Block diagram of the developed software in MATLAB

6.2 Current Consumption and Timing Investigation of the ToF Ranging Method

In this section an investigation of the current consumption of the two-way ToF ranging technique is provided. The reason for focusing on the ToF technique is that it constitutes the majority of the anchors and target node operation. The following figures illustrate the current consumption of the "requester" node during the execution of the two-way ToF ranging. A simple circuit was used for this experiment where a 10 Ohm resistor facilitated the calculation of the current drawn by the node. The node under investigation was powered by a voltage generator with the nominal 3.6V value. A second node was powered at the same level though a computer USB port. The spikes in Figure 6.3 correspond to the transmission and reception of the ranging packets while a smaller spike afterwards corresponds to the calculations and the extraction of the average timing.

In Figure 6.4 the maximum instantaneous value of the current is approximately 20.7mA which verifies the CC2500 radio specification [62]. Figure 6.5 provides insight on the required amount of time in order for the nodes to complete a single ranging operation. The minimum time, given that all 100 two-way are completed successfully without the need for retransmissions, that is required for the two node nodes to complete is 150ms. According to this measurement an expected sampling period of 1s for the tracking system is reasonable given a good communication link (small number of dropped messages) between the wireless nodes.

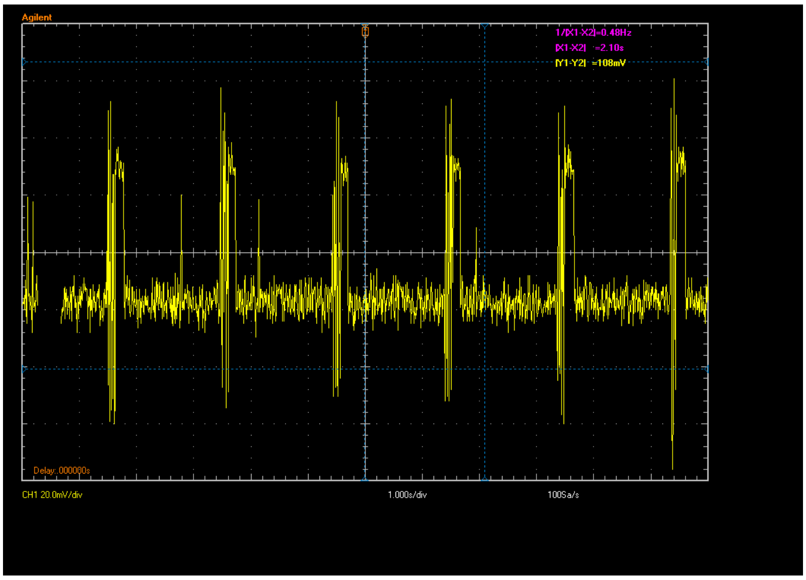


FIGURE 6.3: Current consumption investigation of the ToF method

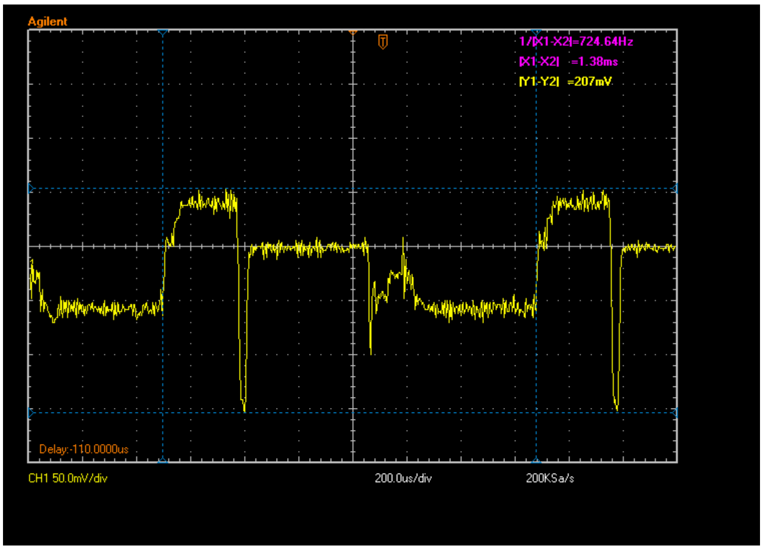


FIGURE 6.4: Higher resolution version of Figure 6.3 to measure the maximum current drawn

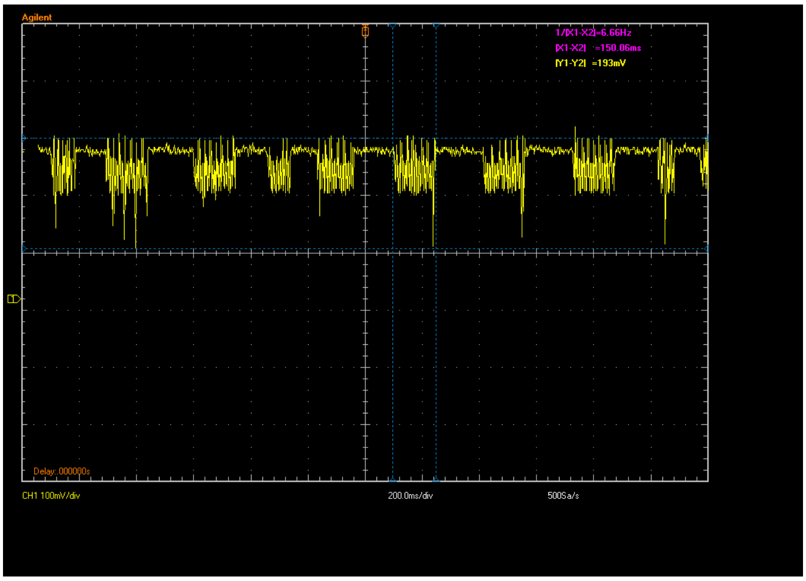


FIGURE 6.5: Measuring the minimum required amount of time to complete a single ranging operation

6.3 Deployment setup

6.3.1 Embedded Nodes

As it was discussed previously the tracking system was implemented on T.I. EZ430-RF2500 hardware. However in order to achieve the best wireless connection between the nodes during the demanding process of cycling through the anchor nodes, instead of using the designated T.I. battery pack, which has proven to be problematic for the node communication when the in-node distance increased, we used custom battery packs in order to power the EZ430-RF2500 nodes (figure 6.6). These battery packs provided a more constant voltage supply.

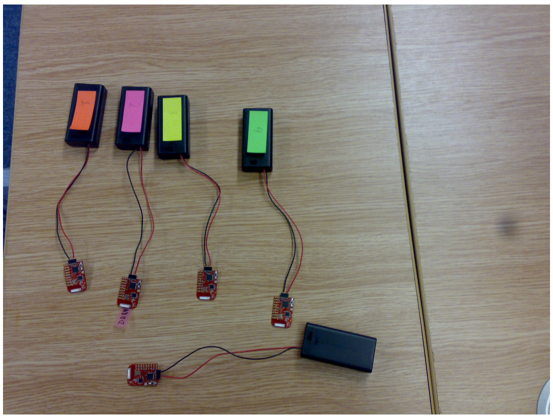


FIGURE 6.6: The five nodes (four anchors and one target node) prior to deployment

6.3.2 Deployment Area

The experimental setup included the deployment of four anchor nodes over a designated area. The four anchors were positioned in the four vertices of a square region. A custom grid made of builders string was also deployed in the square area to facilitate the execution of the experiments. The grid is illustrated in Figure 6.7.

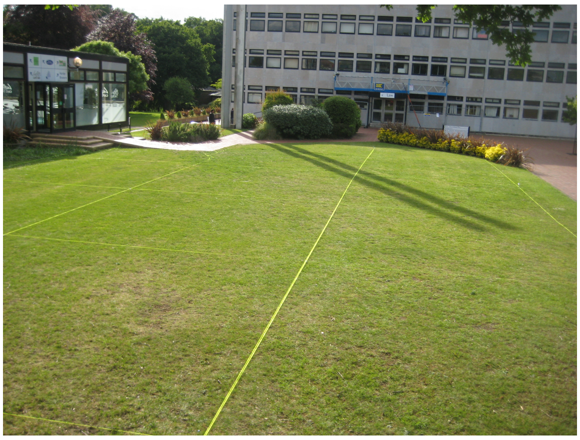


FIGURE 6.7: The deployed grid

The square region where the experiments were executed was chosen to be 15m x 15m. These dimensions allowed enough space to run the tracking experiments without increasing the distance between the nodes to levels that would jeopardize the connectivity of the nodes and would result in considerable latencies.

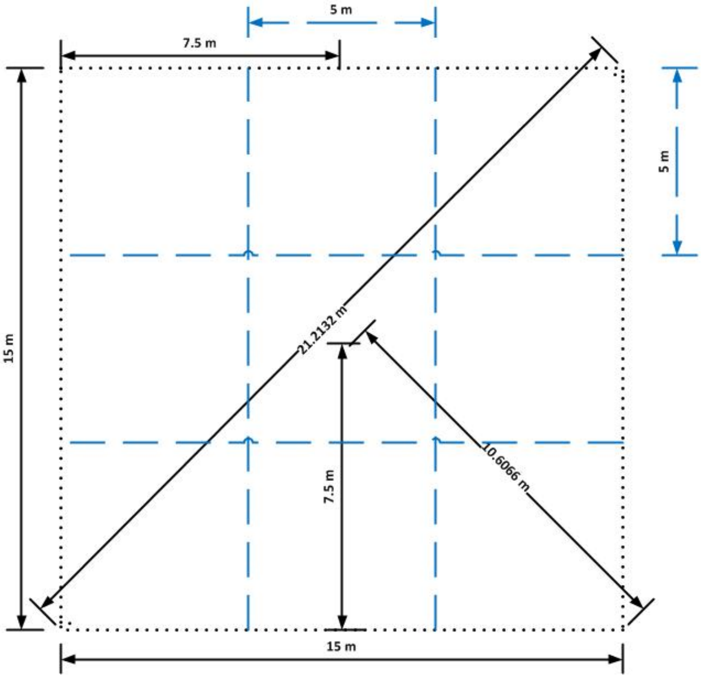
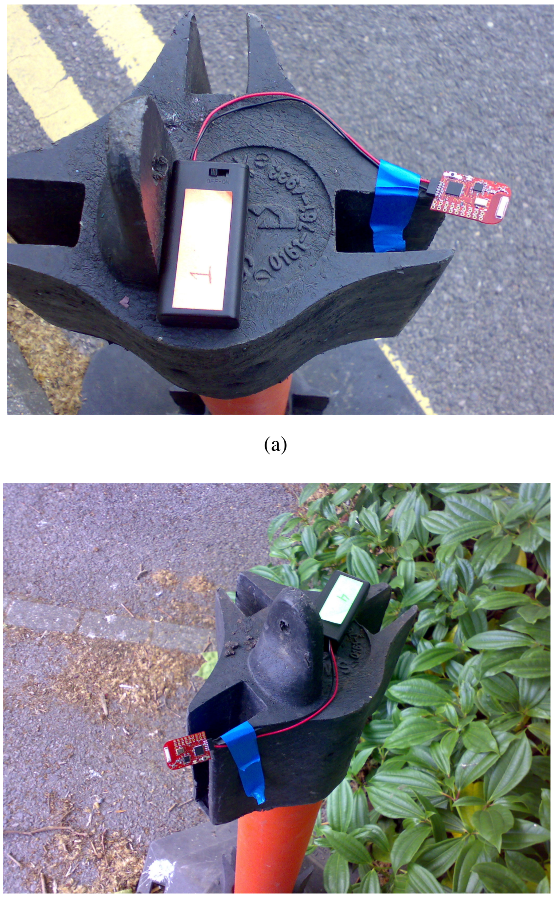


FIGURE 6.8: The dimensions of the area were nodes were deployed

Finally the nodes were strapped on plastic traffic poles (Figure 6.9) and elevated from the ground

(1m) in order to avoid potential deflection of the RF signals.



(a)

(b)

FIGURE 6.9: EZ430-RF2500 nodes attached to a traffic poles

6.4 Experiments execution and results

6.4.1 Preliminaries

In the experiments carried out the four anchors were placed in known positions in the corners of the 15m x 15m square which defined our experimentation area. The central processing node, where the ranging estimates were fused and the system's estimates were produced was placed outside the 15m x 15m square area. In our experiment we restrained in tracking a single mobile node. The target mobile node was carried in the hands of a person which was walking in the designated square area. The target node was carried at a similar height as the anchor nodes. In that sense the height difference of the target node and the anchors during the experiments was neglected and the experiments were constrained to 2-d planar tracking.

In all experiments the nominal number of two-way ranging transactions with each anchor node was set to 100. That is ten times lower that the 1000 that were used in the ranging experiments.

The ranging estimates are expected to follow a Gaussian distribution around the mean ToF value with a $\sigma_v = 3.7m$ standard deviation as it was revealed from the experiments presented in Section 3.8 . The reason for choosing 100 two-way transactions to estimate the range between the target and each anchor is related to the required real-time operation. First of all, the EZ430-RF2500 does not have enough memory to store 1000 timing values, which means that in order to utilise more than 100 transactions the ranging routine should be executed multiple times. This was done without a problem in the ranging experiments where real-time operation was not a critical component. However in the tracking system where the ranging estimates must reach the central node in a timely manner with minimum delay such an approach would add significant latency that would hinder the ability for real-time operation. Due to the these issues, we chose to proceed with setting the nominal number of ranging transactions to 100. The EZ430-RF2500 datarate was set to 500kbps.

The rest of the parameters for the tracking algorithms were defined as follows. The state noise was defined as zero-mean white Gaussian noise with covariance $\sigma_w^2 = 0.5$. Similarly the observations noise is also defined as zero-mean white Gaussian with $\sigma_v = 3.7$ as it is standardized in Section 3.8 from the ranging experiments. The system also was aware of the target's actual initial position. The adaptive method to monitor the sampling interval was described previously (6.1.3). In the implementation of the PF tracking algorithms the particle size was set to $N = 1500$ to provide improved robustness in case the sampling period increased. The distribution to sample the initial particles from was a Gaussian distribution with zero mean and unity covariance. Finally the transition probability for the regime variable was set at $m = 0.8$ for the MMPF-ROT algorithm. The constant turning rate was set at $\pi/4rad/sec$.

To calculate the proposed system's tracking accuracy the output obtained from the systems must be compared with respect to the ground truth of the target. Nevertheless, measuring the target's ground truth position during the experiment both spatially and temporally is not an easy task. To simplify this task, the target's trajectory was predefined before the experiments. With the help of the grid the target's trajectory was divided in individual segments at which the target moved at a straight line with approximate constant speed. During the execution of the experiment we recorded the times when the ranging data from all nodes were collected (reached the central node). These time instances defined the update interval for the state-space model. Finally the total time of each individual straight line segment was recorded.

This facilitated the computation of the target's true velocity in each segment of it's trajectory. In addition by knowing the estimation times, we were able to interpolate and calculate the target's ground truth position and velocity at the time of the execution of the algorithm. In conclusion, following this approach, the estimation produced by the system could be compared against an accurate enough approximation of the true target variables.

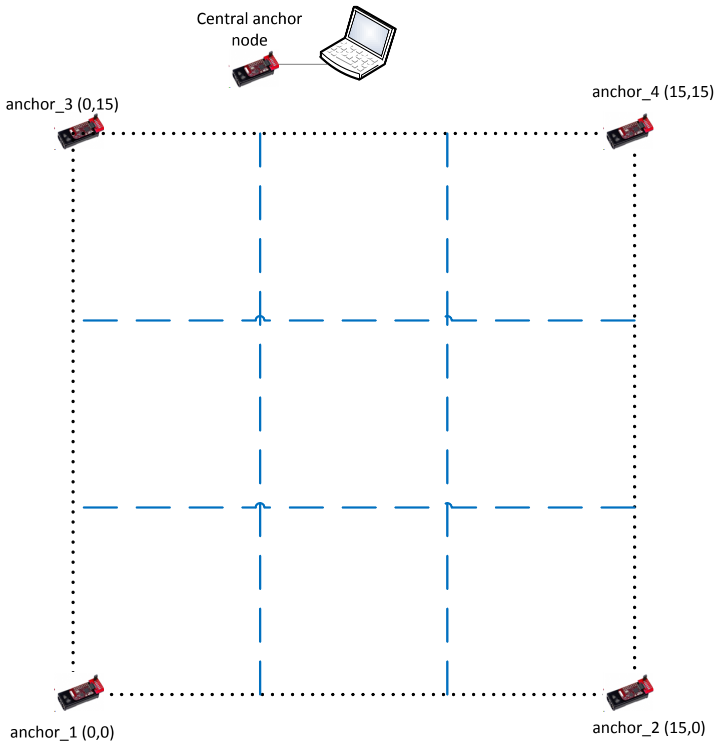


FIGURE 6.10: The deployment of the anchor nodes. In the brackets the x,y coordinates of each anchor are provided

6.4.2 Experimental Results

In this section we present analytical results, of position and velocity estimation from a number of experiments that were carried out with the proposed tracking system. The experiments are categorized in three groups based on the target's trajectory.

1. Straight Line Trajectories
2. Trajectories involving one or two manoeuvres
3. Trajectories involving more than two manoeuvres

The motivation behind dividing the experiments into these three categories is to include both straight line trajectories as well as trajectories involving manoeuvres. The later are to be used for the evaluation of the ROT-MMPF algorithm. As analysed before the purpose of the ROT-MMPF algorithm is to provide enhanced support in situations where the target performs a turning manoeuvre. By including trajectories involving both straight line segments (modeled with the CV model) and turns we access the performance of the ROT-MMPF algorithm.

In the experimental runs that are presented in the following, we plot the position estimates produced by the tracking algorithm against the ground truth. The ground truth is produced based on the time and distance covered in each intermediate segment. Additionally, we plot the velocity vector, by combining the velocit estimation for each axis (v_x and v_y), in every estimation point.

6.4.2.1 Straight line trajectories

In experiments 1-4 (Fig. 6.11-Fig. 6.14) we present results from straight line trajectories that the target node followed. It is clear from the results that the system is able to track the target with very good accuracy. The achieved accuracy, in the four experiments (Experiments-1 to Experiments-4) with straight line trajectories presented here was between 1.8m - 2.8m. Due to the simple motion pattern the target followed in these experiments, the system demonstrated the highest performance.

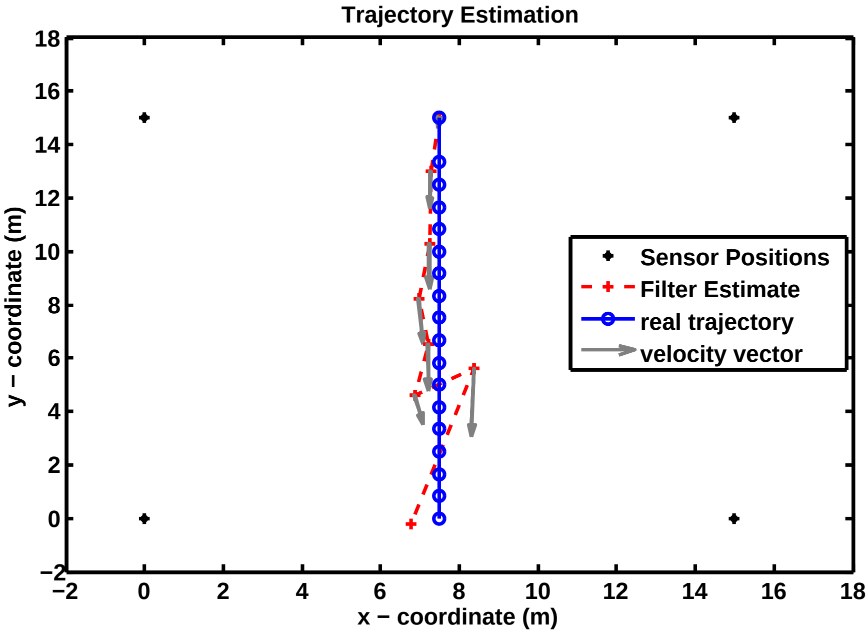


FIGURE 6.11: Experiment-1: A straight line North-to-South trajectory. Crosses are the target's position estimated by the filter and arrows indicate the estimated velocity vector at every sampling update time instance. The circles indicate the ground truth. RMSE = 1.9m

6.4.2.2 Trajectories involving manoeuvres

In experiments 5-8 (Fig. 6.15-Fig. 6.18) we present results from experiments where the target's trajectory included manoeuvres. These particular cases were tackled with the multiple-model state representation and the use of the MMPF-ROT tracking algorithm. For the four experiments (Experiment-5 to Experiment-8) presented here, the observed accuracy was in the range of 2m-3.2m.

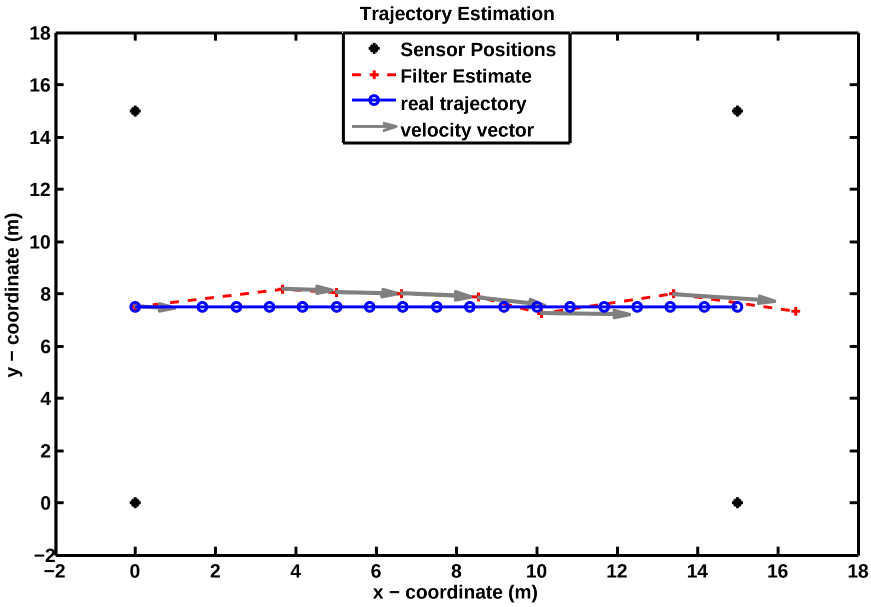


FIGURE 6.12: Experiment-2: A straight line West-to-East trajectory. RMSE = 1.8m

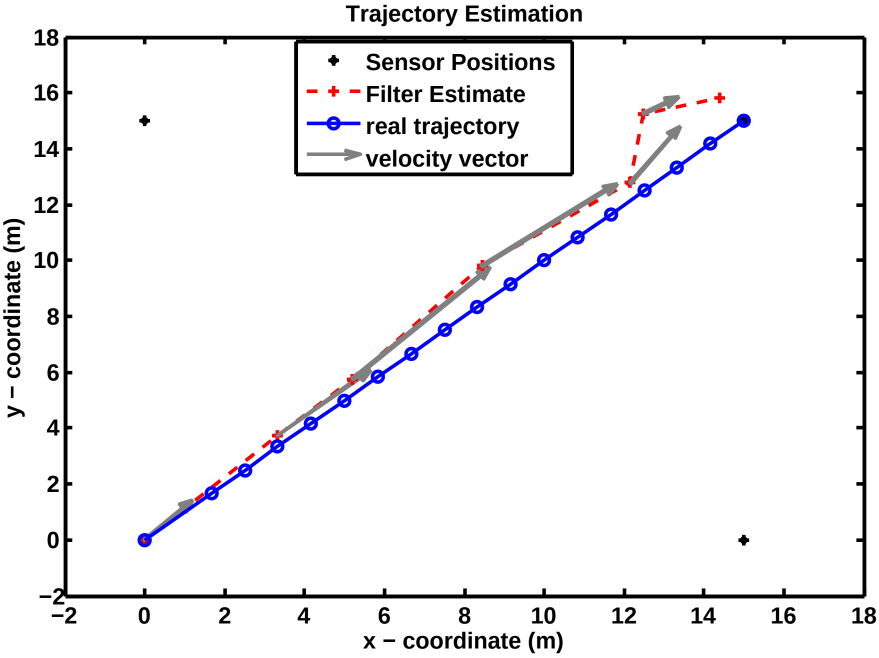


FIGURE 6.13: Experiment-3: A straight line diagonal trajectory from anchor_1 to anchor_4. RMSE = 2.3m

6.4.2.3 Cyclic Trajectories

In the final batch of experimental results (Fig. 6.19-Fig. 6.21) we provide results from executions where the target node followed a cyclic trajectory. In one execution the target followed a perimetrical trajectory, while the other two executions involve the target passing through the midpoints of the four sides of the square area in succession. The RMSE for these experiments (Experiment-9 to Experiment-11) was in the range of 2.9m-3.6m. Because of the multiple manoeuvres the target performed, the highest RMSE was observed in these trajectories.

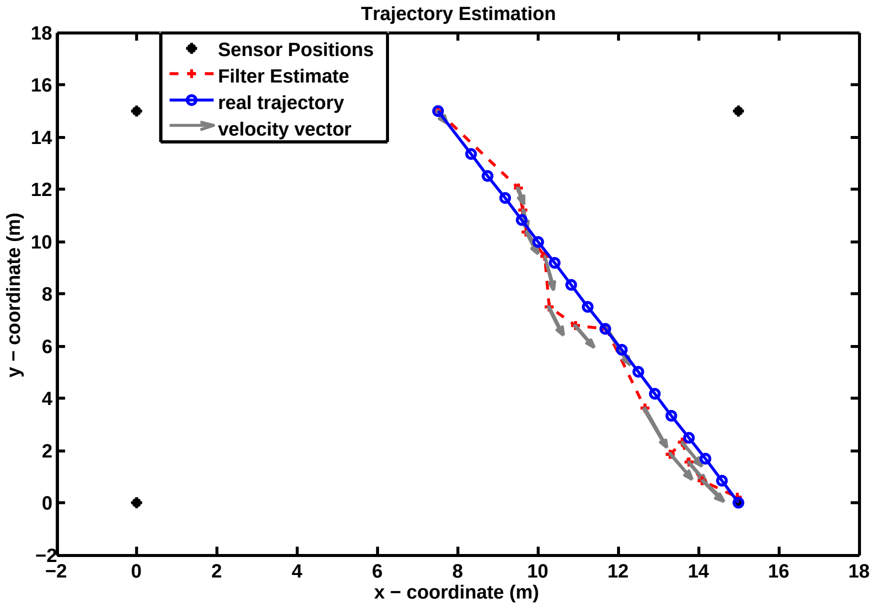


FIGURE 6.14: Experiment-4: A straight line diagonal trajectory from the midpoint of the top side of the square to anchor_2. RMSE = 2.8m

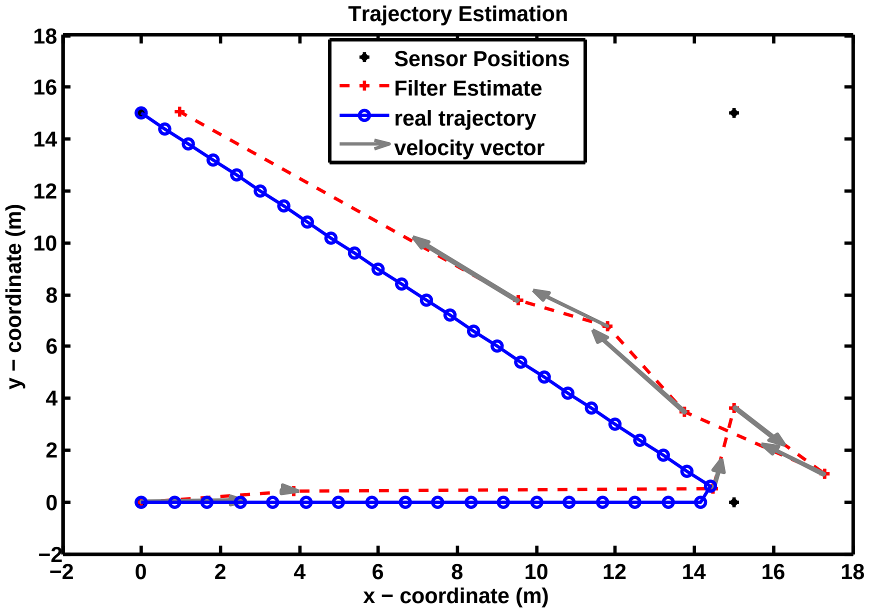


FIGURE 6.15: Experiment-5: A manoeuvring experiment where the target moves from ancor_1 to anchor_2 and then moves diagonally towards anchor_3. RMSE=2.7m

In experiment-10 illustrated in Fig. 6.19 the estimation of the target's dynamics ends abruptly. This is due to the fact that the script in MATLAB is programmed to receive only numeric values. In the event that because of erroneous wireless transmission between the embedded nodes a transmitted byte is interpreted as a character and not a numeric value the MATLAB script signals an error and the program's execution stops. As a result the final part of the target's trajectory was not estimated. In the following experiment (experiment-11, Fig 6.20) we repeated the same trajectory.

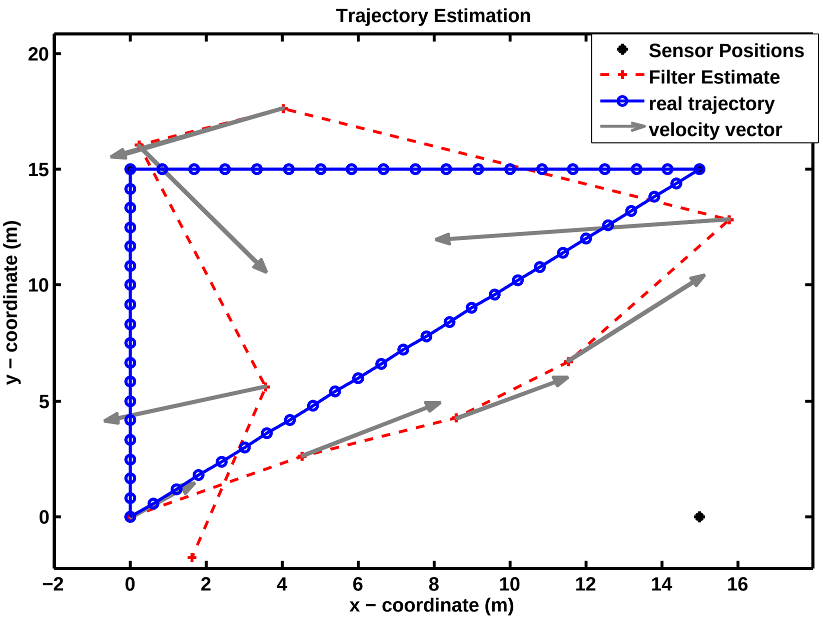


FIGURE 6.16: Experiment-6: A manoeuvring experiment where the target moves from ancor_1 diagonally to anchor_4 then moves back to anchor_3 and returns to anchor_1. RMSE=3.2m

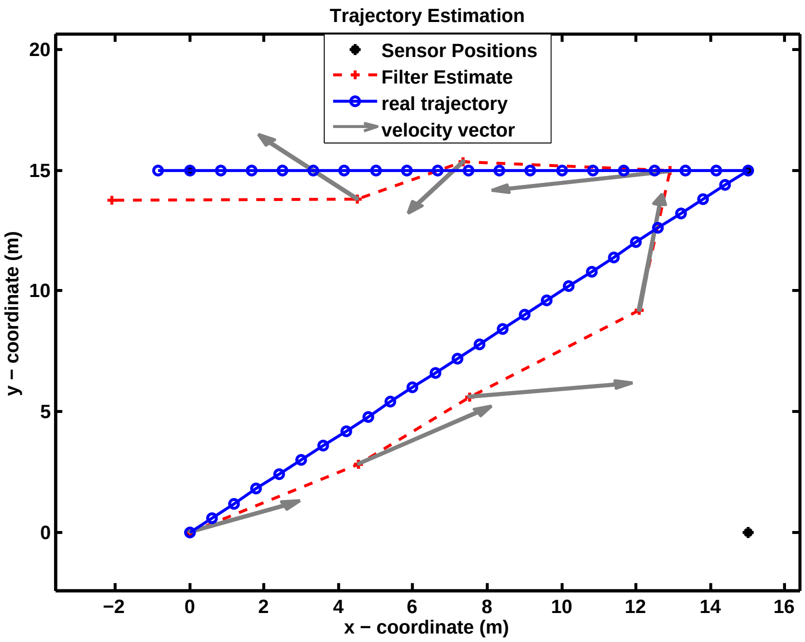


FIGURE 6.17: Experiment-7: A manoeuvring experiment where the target moves from ancor_1 diagonally to anchor_4 then moves back to anchor_3. RMSE=2.7m

6.4.3 Performance Analysis

6.4.3.1 Aggregated Experimental Results

This section provides a quantitative analysis of the achieved accuracy for the proposed tracking system. Two types of error are investigated the position error and the velocity error with respect to the ground truth. Based on the time the target spent in the intermediate (straight line)

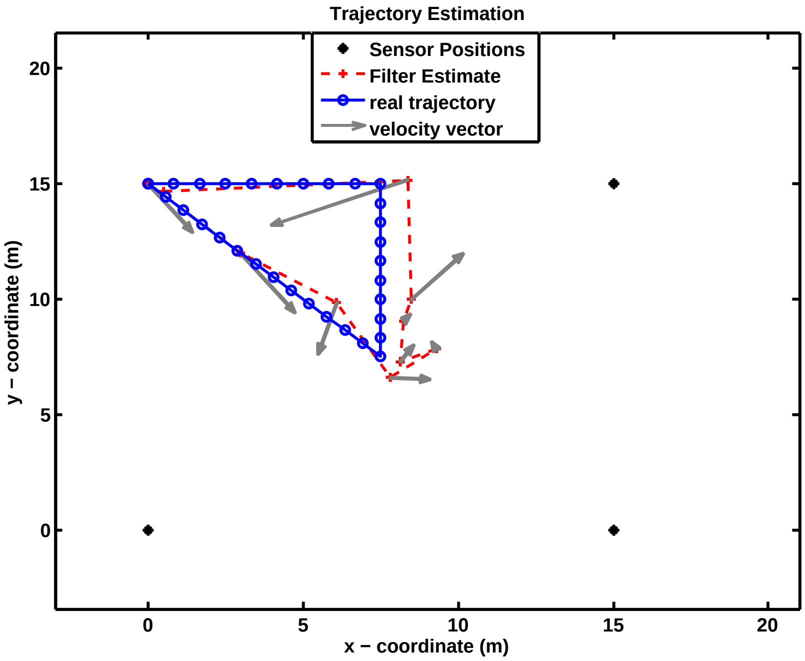


FIGURE 6.18: Experiment-8: A manoeuvring experiment where the target moves from ancor_3 diagonally to the center of the square area then to the midpoint of the top side and then back to anchor_3. RMSE=2.1m

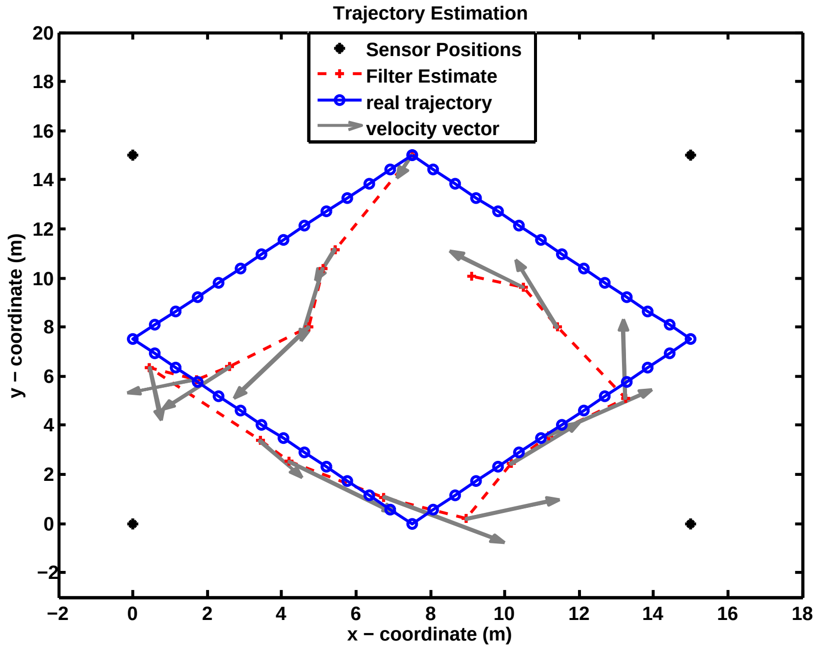


FIGURE 6.19: Experiment-9: A manoeuvring experiment where the target starts from the midpoint of the top side and passes from the midpoint of each successive side of the square area before it returns to the beginning point. RMSE=3.2m

segments between the waypoints, we extrapolate the ground truth values and compare it against the algorithm's estimations. A total of 25 tracking experiments were used in this analysis. The collective performance results are illustrated in the following table and in Figure 6.23. From the 25 experiments under investigation here, 12 of them had an RMSE for position smaller than 2.5m, while 10 experiments demonstrated an RMSE accuracy for x-axis velocity smaller than

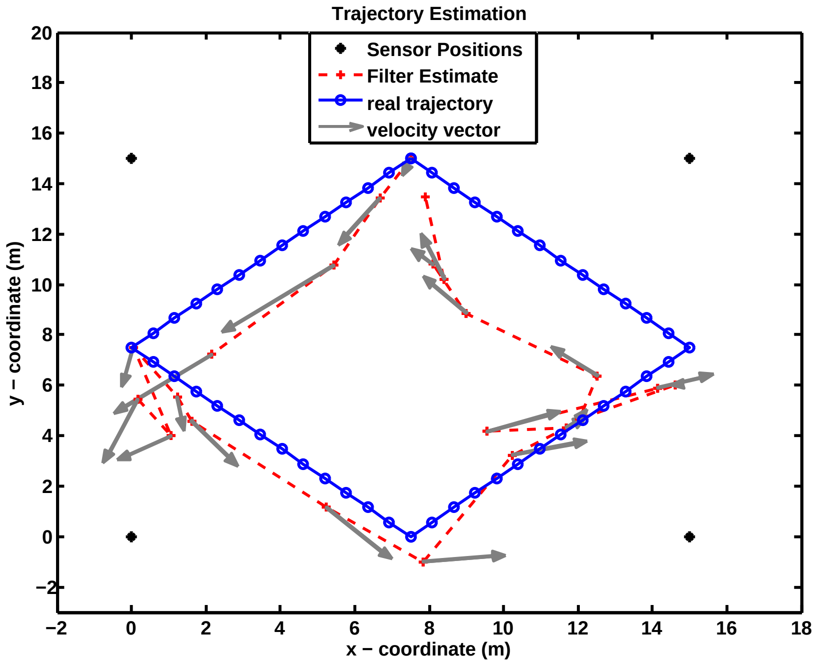


FIGURE 6.20: Experiment-10: A second execution of the previous experiment. RMSE=2.8m

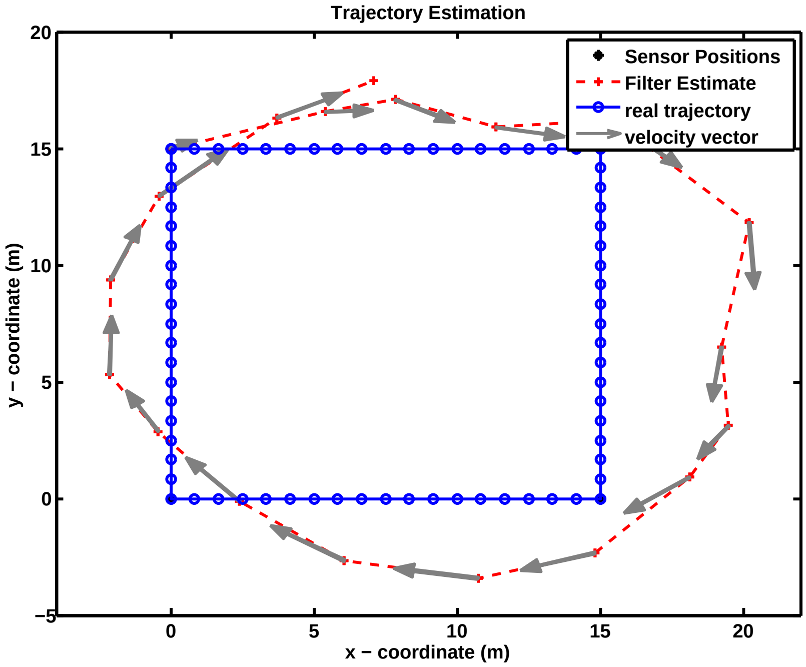


FIGURE 6.21: Experiment-11: In this experiment the target moves peripherally around the square area starting from anchor_3. RMSE=3.6m

1m/s and 12 an RMSE accuracy for y-axis velocity smaller than 1m/s.

The obtained accuracy of 2.6m over a 15m x 15m (225m^2) area is deemed to be very satisfactory. This performance corresponds to an 11% of the area size and considering the four application scenarios discussed in Section1.2, it is the author's belief that such accuracy is adequate for the purposes of the aforementioned scenarios.

Number of Experiment	Position RMSE (m)	x-velocity RMSE (m/s)	y-velocity RMSE (m/s)
1	1.9485	0.1883	0.8953
2	1.8029	0.8264	1.4227
3	2.7523	1.7175	0.389
4	3.7179	1.4809	1
5	1.8107	0.6485	0.3636
6	1.8863	2.0214	0.828
7	1.7267	0.8929	0.9135
8	2.6931	2.6847	0.7864
9	2.0331	1.5975	0.9288
10	3.8687	3.4838	2.1649
11	3.2702	1.9796	1.6948
12	3.2118	1.7699	1.5459
13	3.6512	2.5605	2.2207
14	2.3595	1.0189	1.124
15	2.7006	2.0866	1.8795
16	2.1974	1.7801	0.9031
17	2.2185	2.1407	0.5504
18	2.8265	0.9955	0.7951
19	1.7516	1.0389	1.2148
20	2.9787	0.7059	1.9519
21	2.4612	1.9639	1.2003
22	2.835	0.8075	0.8758
23	2.2902	1.148	1.0661
24	4.0774	2.7443	2.5435
25	2.4339	1.016	1.2048
Average	2.62	1.57	1.22
Best Case	1.4612	0.1883	0.7864
Worst Case	4.0774	3.4838	2.5435

TABLE 6.1: Accuracy results from 25 experimental executions

6.4.3.2 Comparison to Simulation Results

Here we present a comparison between the results obtained from the full-scale experiments with simulations results obtained after simulating multiple times a tracking scenario similar to the one we experimented with in the full-scale experiments. Four anchors were considered placed in coordinates ((0m,0m), (15m,0m), (0m,15m), (15m,15m)) exactly as in the outdoors deployment. The system parameters (target's initial state, distribution to sample initial particles), the PF algorithm parameters (particle size) and the noise levels, where the same as they were defined in the full-scale experiments (see Section6.4.1).

To approximate the behaviour the system demonstrated in the full-scale experiments, we used a varied sampling interval. Based on observations from the real-world experiments, the minimum observed value of the sampling interval during the experiments was around 0.8sec and the maximum one around 2.2sec. To approach this in our simulations we randomized the sampling

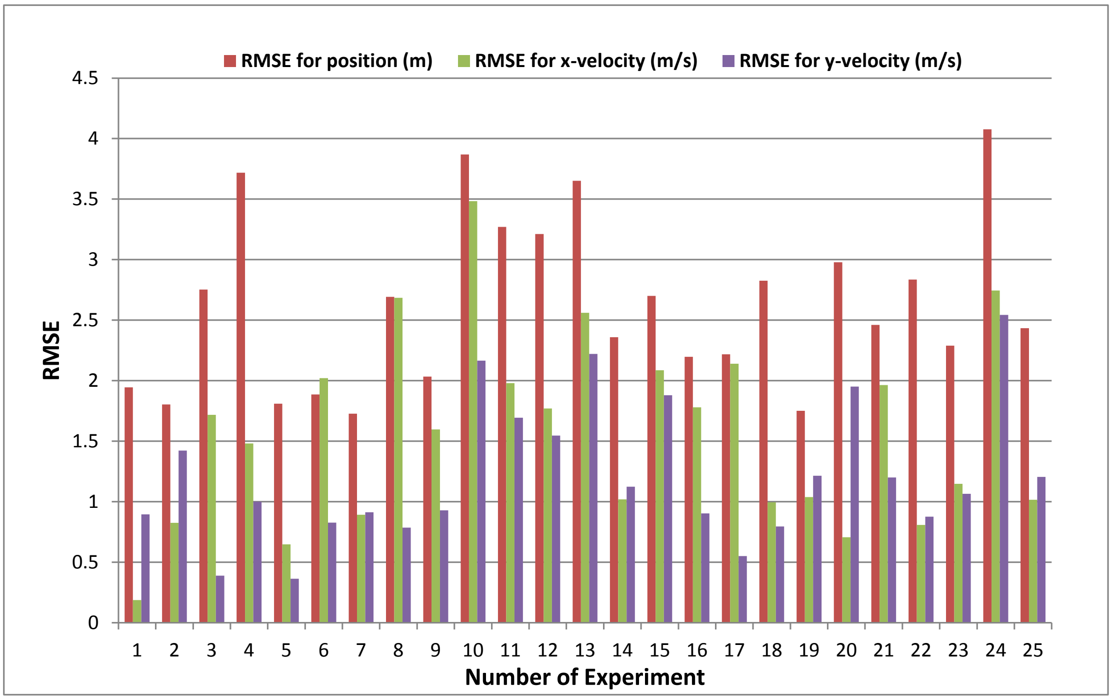


FIGURE 6.22: RMSE of position and velocity estimates

interval variable between 0.8 and 2.2sec. The simulations were run for 30 time steps and with each step having varying sampling interval. This approach resulted in simulation executions that run for a total time similar to the one that the real-world experiments lasted for.

We simulated this scenario for 100 runs and included both random and deterministic trajectories. The deterministic trajectories were the same as the ones presented previously in Section6.4.2and included straight line trajectories as well trajectories with predefined manoeuvres.

The average RMSE obtained from the simulation analysis is 2.5m, a result which is very close to the one (2.6m) observed in our real-world experiments. Additionally the accuracy in the velocity estimation obtained from simulations is 1.89m/s which again is similar to the one obtained in the full-scale experiments (1.9m/s).

Conclusively, this simulation investigation attests and verifies the accuracy levels and the system's performance demonstrated in the real-world experiments. The combined RMSE for position results, from 100 simulations of this simulation set-up are illustrated in Figure 6.23.

6.5 Summary

This chapter is the ultimate technical chapter of this thesis. It describes the incorporation, of the outcomes described in previous chapters namely the PF based tracking algorithms (Chapter 5) and the ToF ranging method (Chapter 4), into a novel real-time ToF tracking system for low-power embedded nodes. The software that was developed for the implementation in hardware

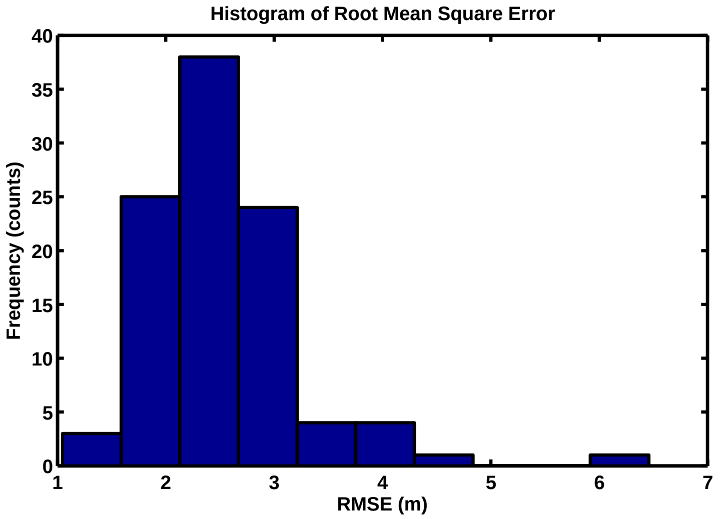


FIGURE 6.23: Histogram of RMSE evaluated from 100 simulation under similar conditions with our real-world experiments and under varying sampling interval

was heavily based on the software for the ranging technique adapted in such way to facilitate the tracking experiments. The system designated three different categories of nodes, the target node, the anchors and the central node, all of which programmed with a different piece of software that provided the desired functionality. Although it was attempted to implement the PF tracking algorithm on the central node, due to processing demands and memory limitations the PF algorithm could not be implemented in its current form on the EZ430-RF2500 nodes. Ultimately, for convenience, the PF algorithm was implemented on the MATLAB environment and run on a designated laptop which was connected to the base station node in order to receive the required ranging data.

The system was deployed on an outdoors location with good line-of-sight between the participating nodes. A mobile wireless node was the target to be tracked. The experiments included straight line trajectories as well as trajectories involving manoeuvres. From the cumulative analysis an average RMS error of 2.6m for position and 1.5m for x-axis and 1.2 for y-axis velocity respectively was achieved in a 15m x 15m area. The obtained results demonstrate the applicability of the system on tracking manoeuvring targets. These results also justify the expected good performance of the algorithm as it was revealed through simulations.

Chapter 7

Concluding Remarks and Future Directions

This dissertation described the design, implementation and evaluation of a real-time, range-only target tracking system for wireless embedded nodes. The motivating idea of this project considers a small number of wireless embedded nodes (anchors) to be deployed in known coordinates tasked with acquiring ranging information and a central node which receives the accumulated data and executes the tracking algorithm in real-time to estimate the target's position and velocity. The challenges that the proposed research attempted to address are accuracy, real-time operation and support for manoeuvring targets. In the implementation of the system, we targeted at fulfilling these demands. The proposed system is based on a two-way ToF ranging method and a PF tracking algorithm which is employed to infer the target's dynamics.

This chapter provides a summary of the work that was carried out in this thesis. Section 7.1 highlights the major research contributions that this research achieves. In the following, a number of lessons learned during the implementation process of the proposed system are discussed in Section 6.2. Finally, ideas for possible future research directions are listed and analysed in Section6.3.

7.1 Research Contributions

To acquire the necessary ranging we implemented a two-way ToF ranging method which achieves subclock resolution by averaging multiple ToF estimates to extract the ToF value between a pair of embedded nodes. The advantage this method yields over previous ranging approaches is the fact that does not require any additional hardware to be installed on the wireless nodes. AoA methods that were consider for tracking in WSNs, require arrays of antennas and TDoA techniques demand an ultrasound transceiver to be attached on the nodes. In addition, contrast to the radio-interferometric ranging method which requires good synchronisation between the two nodes to transmit signals simultaneously, the proposed ToF method achieves a mean accuracy of about 1.5m outdoors, without such requirements.

The proposed range-only tracking system was formulated as a dynamical state-space problem which aims at providing enhanced support in tracking manoeuvring targets by adopting a multiple-model scheme to represent the dynamics of manoeuvring targets. Different to the majority of the approaches present in the relevant literature, which only consider a single linear model (also investigated in this thesis) to describe the target's motion pattern, this research work proposes the utilization of three switching dynamic models in the formulation of the state-update equation. By doing so, we believe that the sudden changes in the kinematic variables of manoeuvring targets are described with higher precision.

To effectively solve the defined dynamical system, this research develops two PF tracking algorithms. Depending on the choice of the state model (single or multiple). The PF algorithms operate on the accumulated ranging data provided by a number of anchor nodes deployed in known coordinates. From the simulation investigation as well as the full-scale experiments we deduce that four anchors are able to provide ranging data with enough spatial-resolution in order for the PF algorithms to accurately estimate the kinematic variables of the target.

The proposed tracking system is evaluated under both simulations as well at full-scale. The simulation experimentation enabled identification of the system parameters that will affect the system's performance. From that investigation we conclude that the two most significant parameters are the sampling interval and the number of particles. To guarantee accurate performance the sampling interval must be kept as low as possible. By increasing the number of particles of the PF algorithm accuracy is also increased. However an increase in the number of particles results in increase in the execution time of the algorithm and as result the real-time operation of the system could be hindered. From all the simulations carried out, the conclusion was that for a sampling interval of around $1s$ a number of $N \sim 500$ particles should suffice as to guarantee accurate performance, given that four observers (anchors) provided the system with range data.

The system was implemented on COTS platforms (T.I. EZ430-RF2500) and tested under different types of trajectories. A total of 25 individual experiments are presented in this thesis. From the 25 experiments considered, the average RMSE was 2.6m for position and 1.9m/s for velocity over an 15m x 15m area. These results are further compared to simulation results conducted under similar conditions which justify the performance of the system. Such performance is comparable to the one reported by the systems reviewed in Section 2.5. In addition, the evaluation of the proposed system considered a number of different trajectories and the results are extracted as the average of 25 experiments. Most of the reported systems report results from a limited number of experimental executions.

7.2 Lessons learned during implementation and experimentation

Here we discuss our experiences from the procedure of implementing the tracking system on hardware. We highlight several aspects of the tracking system operation that play a significant role in the system's overall performance and cannot be identified through simulations. The hurdles that were faced in the implementation and experimentation process of the system are also analysed.

First of all, the designed tracking system requires significant communication between the participating nodes (anchors, target, central) as this is a requirement of the ToF ranging method. If the network performance is diminished and the packet loss is higher than normal, additional transmissions are required to reach the nominal number in order for the ranging to be completed. Extra transmissions, will increase the time required for the ranging estimations and subsequently increase the latency of the tracking system to produce an estimation. This condition can severely affect the real-time operation as well as the overall performance of the tracking system.

For example, during the experimentation of the ranging technique we deployed the nodes on an area in the University of Southampton campus where wireless WiFi networks operating on a similar band (2.4Ghz) added enough noise to make the execution of the ranging experiments impossible for considerable distances between the two nodes. Also, physical obstacles that interject the line-of-sight between the nodes also result in diminishing network performance. Finally environmental conditions like humidity can also affect wireless communication and cause an increase of the packet loss. The aforementioned issues can be tackled to a point with the use of more powerful hardware equipped with whip antennas capable of more robust wireless communication in longer distances. Power supply is another important factor which affects the network performance. In particular for the EZ430-RF2500 hardware, which is equipped with a simple chip antenna, the power supply must be at maximum levels in order for the nodes to achieve communication over a distance that would allow experimentation. That was the reason why we chose laptops to power-up the nodes during our ranging experiments. For the tracking experiments the AA battery packs proved to provide adequate voltage levels with no fluctuations and sufficient current draw for the distances used in our experimentation.

Ultimately the PF tracking algorithms proved to be too computationally complex to be implemented on the EZ430-RF2500 platform. The main limitation arises from the memory capacity (32Kb flash / 1Kb RAM) of the platform which is not adequate for the implementation of the PF algorithms. In reality most of the well known WSNs platforms do not have sufficient amount of memory for the implementation of the PF algorithms.

7.3 Future Directions

7.3.1 Amendments to the Ranging Method

The proposed two-way ToF ranging method demonstrated prominent results with acceptable accuracy for the purposes of the tracking system. Nevertheless, a number of amendments can be considered for future investigation in order to obtain improved accuracy in the estimation of the locating node. Moreover the following propositions target at reducing the required number of two-way transactions in order to achieve similar performance.

To begin with, as presented in Chapter 3 the achieved accuracy of the two-way ToF ranging method can be improved by synchronising the operation of the radio and the MCU. In essence this means driving both pieces of hardware from the same clock source. That would mitigate the delay within the interrupt routine that is caused when the routine is raised from the radio (for example on receiving a message) until the micro-controller takes handle of the interrupt. The delay is caused because the two devices that take part on the interrupt process, operate under different clocks.

Another potential research direction in connection to the ToF method is related to the basics of the two-way ToF ranging. A possible amendment could include a more accurate modeling of the phase offset between the two nodes over time which would allow calculation of the specific number of required transactions for a given accuracy. Moreover incorporating in the ToF method models that correlate the clock's drift on a single node to temperature and voltage supply fluctuations can result in more accurate estimation of the range between the two nodes.

7.3.2 Distributed target tracking system

A proposal for further developing the designed target tracking system is to implement the system in a distributed manner. The concept is to employ a hierarchical WSN with the use of cluster leaders.

Each cluster will be responsible for tracking a target in the vicinity of the cluster. The leader of its cluster will assume a role similar to the central node in the proposed system. Consequently the cluster head will be a node of the same technology, with enhances processing capabilities, able to execute the tracking algorithm. Each cluster head accumulates data from a small number (4) of anchor nodes-members of its cluster. The benefits that yield from such an implementation can be summarized as follows.

The energy demands of the tracking system will be distributed across the clusters, resulting in increased system lifetime. This is achieved since only one cluster is responsible of tracking the target at a given time period. The rest of the clusters will remain in an idle state until an adjacent cluster initiates them. Extending the previous proposal, one can also consider the scenario where each cluster monitors its remaining energy levels, considering both the remaining energy on the cluster head as well as on the anchors. In case this energy falls below a threshold an adjacent cluster with adequate remaining energy may be initiated to continue tracking based on its anchors. Moreover communication connectivity between WSNs nodes is known to degrade with increasing distance. To address this issue the distributed approach is designed in such a way that when a cluster node realizes that the distance between the target node and anchor nodes exceeds a predefined threshold, passes the operation to an adjacent cluster node which is closer to the target. This approach ensures that tracking is conducted with the observations that are likely to provide higher accuracy. The immediate following cluster node uses the final estimation of the previous cluster node as the initial knowledge for the tracking operation.

7.3.3 Development of a wireless nodes capable to handle computationally intensive tasks

In conjunction with the previous proposal regarding a distributed tracking system, a potential approach can be the development of a wireless node capable of handling computationally intensive tasks like the PF algorithm. The EZ430-RF2500 platform although provided the necessary flexibility for the implementation of the tracking system, has some inherent limitations in terms of processing ability.

The suggestion that is proposed is to investigate commercially available DSP (e.g. T.I. TMS320 -C6713DSK) or FPGAs and attempt a connection to an EZ430-rf2500 platform through a serial interface. This approach is motivated by the relevant work in the area of PF hardware implementation. The sole purpose of the DSP will be the intensive number crunching required by the PF with acceptable latency, while EZ430-RF2500 will act in a similar way as the central node in our experiments forwarding the data obtained from the anchors. An illustration of the distributed tracking system is provided in Figure 7.1.

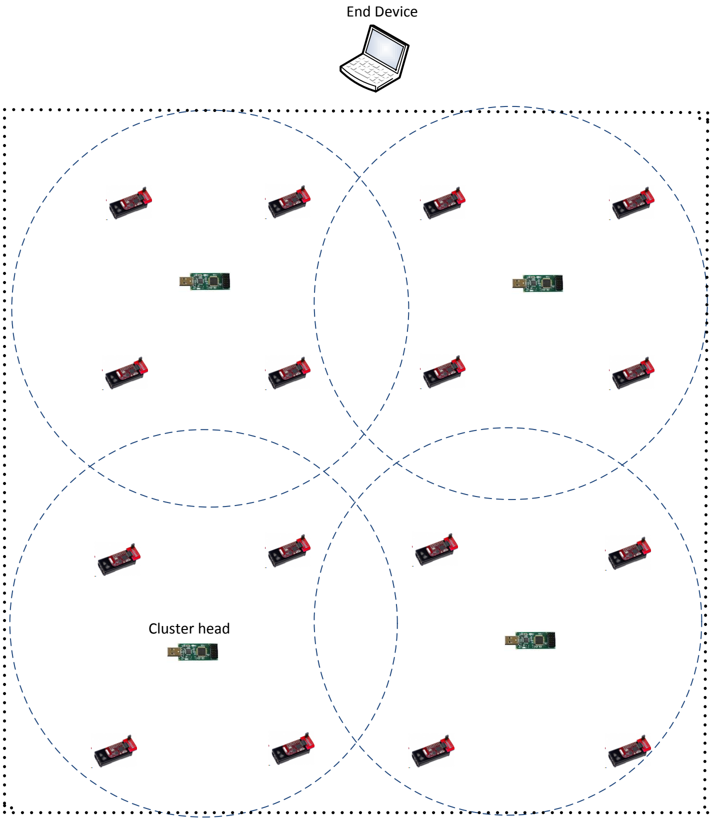


FIGURE 7.1: An illustration of the proposed distributed target tracking system

Appendix A

Bayesian Approach to target tracking - Particle Filters

This Appendix is based on the definition of the tracking problem found in [98]. The mathematical notation used in this Appendix is also taken from [98].

Defining the problem using state-space representation

The state-space representation of a system facilitates Bayesian Estimation Theory to be employed. In the state-space representation the state vector contains variables which describe the system (kinematic characteristics). On the other hand the measurement vector contains variables regarding the observations (measurements) that are available. In state-space representation these two vectors evolve in time.

Based on the state-space representation, a generic tracking problem is defined as follows:

Let $x_k \in \mathbb{R}^{n_x}$, where n_x is the dimension of the state vector and $\mathbb{R}$ the set of real numbers.
Also let $k \in \mathbb{N}$ to denote the time index. The time index in the system can be considered as continuous or discrete.
Based on that approach the state vector is evolving in time through the following equation.

$$x_k = f_{k-1}(x_{k-1}, v_{k-1}) \tag{A.1}$$

where f_{k-1} is a known function (nonlinear in the generic case) representing the evolution of the state vector in time and v_{k-1} is referred to as a process noise sequence. The set of measurements that become available at every time step are associated to the state vector with the following equation.

$$z_k = h_k(x_k, w_k) \tag{A.2}$$

where h_k is a known, nonlinear in the general case, function and w_k is a measurement noise sequence.

The noise sequences v_{k-1} and w_k are assumed to be white with known probability density functions. Also the initial target state $p(x_0)$ is assumed to be known.

In a real world implementation, the sequence of measurements y_k are observed from the system (sensors measurements, ToF measurements, etc). The non linear functions f_k and h_k are supposed to be known and the noise sequences are assumed Gaussian. Thus the only unknown factor is the state vector x_k at every time step k.

In the above defined system, the state vector x_k usually contains the kinematics characteristics of the target. This includes position coordinates, two axis velocity, acceleration and maybe direction of motion. However it must be stated, that by increasing the number of the state vector factors the complexity of the problem increases.

The objective is to obtain filtered estimates of the state x_k based on the set of available measurements $Z_k \triangleq \{z_i, i = 1, \cdots, k\}$ until time k. From a probabilistic point of view the problem is defined as to recursively gain some degree of belief in the state x_k at time k, given the data Z_k up to time k. Hence, the posterior probability density function $p(x_k/Z_k)$ is required to be constructed. The initial density is $p(x_0) \triangleq p(x_0/z_0)$ where z_0 denotes no measurements. Then based on Bayesian Theory the probability density function at time k can be computed in a two stage procedure using the following equations.

If the required pdf $p(x_{k-1}|Z_{k-1})$ is known at time $k-1$, equation A.1 can be used to predict the pdf at the next time step k by using the Chapman-Kolmogorov equation.

$$p(x_k|Z_{k-1}) = \int p(x_k|x_{k-1})p(x_{k-1}|Z_{k-1})\mathrm{d}x_{k-1} \tag{A.3}$$

In the above equation it is considered that $p(x_k|xk-1, Z_{k-1}) = p(x_k|x_{k-1})$. This statement, in statistics means that the time sequence of state x_k is a first order Markov Process. Mathematically a Markov Process is a stochastic time-series, in which *only the current state affects the next state*. More simplistically in the context of this problem the Markov Process definition is translated as: *in order to define state x_k only state x_{k-1} must be known*. This is defined by the state evolution equation A.1, where it is obvious that the current state x_k depends only upon the previous state x_{k-1} and the noise v_{k-1} statistics.

At time k a set of measurements Z_k becomes available to the system. The update stage is carried out where the prior (before the measurement is considered) pdf is updated based on that measurement to give the posterior pdf of the state vector. This is done using the known Bays's rule:

$$\begin{aligned} p(x_k|Z_k) &= p(x_k|z_k, Z_{k-1}) \\ &= \frac{p(z_k|x_k, Z_{k-1})p(x_k|Z_{k-1})}{p(z_k|Z_{k-1})} \\ &= \frac{p(z_k|x_k)p(x_k|Z_{k-1})}{p(z_k|Z_{k-1})} \end{aligned} \tag{A.4}$$

where the normalising constant is defined as:

$$p(x_k|Z_{k-1}) = \int p(z_k|x_k)p(x_k|Z_{k-1})\mathrm{d}x_k \tag{A.5}$$

Equations A.3, A.4 form the basis to compute the optimal Bayesian solution and define the posterior pdf $p(x_k|Z_k)$. Knowing the pdf enables to compute an estimation for the state vector x_k based on a certain criterion.

For example the minimum mean square error (MMSE) estimate is the conditional mean of x_k: $\hat{x}^{\text{MMSE}}_{k|k} = \int x_k \cdot p(x_k|Z_k)\mathrm{d}x_k$

The optimal Bayesian solution can be computed in some cases using known and widely used strategies like the Kalman Filter or grid based methods. The situations where a finite dimensional algorithm can be used to lead to the optimal Bayesian estimation can be formulated in the following cases.

- In a linear gaussian case, equations A.3 and A.4 become the Kalman Filter equations
- If the state is discrete - valued with a finite number of states, the grid-based methods provide the optimal algorithm
- For certain subclasses of nonlinear systems, an optimal algorithm is possible to be formulated

Particle Filters

This section provides the basic concepts of a Particle Filter algorithm. Particle Filters are suboptimal filters. The theoretical background of Particle Filters stems from sequential Monte Carlo Estimation of a probability density function based on mass points (or particles).

Monte Carlo Integration

Monte Carlo Integration is the basis for sequential Monte Carlo methods. Suppose the following integral needs to be computed.

$I = \int \mathrm{g(x)}d\mathrm{x}$

where $\mathrm{x} \in \mathbb{R}$

Monte Carlo methods for numerical integration, factorise g(x)=f(x) $\cdot\ \pi(\mathrm{x})$ so that $\pi(\mathrm{x})$ can be considered as a probability density function, $\pi(\mathrm{x}) \geq 0$ and $\int \pi(\mathrm{x}) = 1$. Monte Carlo methods consist of drawing N≫1 samples $\mathrm{x}^i; i = 1, \ldots, N$ distributed according to $\pi(\mathrm{x})$ and use them to calculate an estimate of the integral.

$$I = \int \mathrm{f(x)}\pi(\mathrm{x}) \tag{A.6}$$

The estimate is given as the sample mean:

$$I_N = \frac{1}{N}\sum_{i=1}^{N} f(x^i) \tag{A.7}$$

If the samples x^i are independent then I_N is an unbiased estimate and according to the law of large numbers will almost surely converge to I.

Importance Sampling

At this point we introduce an important procedure for Particle Filters called “Importance Sampling”. This procedure is explained in the following. In the ideal situation samples can be generated directly from π(x) and estimate I from the above equation. When π(x) is not available but a density function q(x) “similar” to π(x) is available, then sampling can take place using q(x) and a correct weighting of the sample set still makes Monte Carlo estimation possible. Function q(x) is referred to as importance or *proposal density*.

The world “similar” between π(x) and q(x) is summarised in the following:

$\pi(\mathrm{x}) > 0 \Rightarrow q(\mathrm{x}) > 0$ for all $\mathrm{x} \in \mathbb{R}^{\mathrm{n}_x}$

The above condition is necessary to hold and if valid, any integral; of the form A can be rewritten as:

$$I = \int \mathrm{f(x)}\pi(x)d\mathrm{x} = \int \mathrm{f(x)}\frac{\pi(\mathrm{x})}{q(\mathrm{x})}q(\mathrm{x})d\mathrm{x} \tag{A.8}$$

A Monte Carlo estimate can be computed using $N \gg 1$ samples distributed according to q(x) and forming the following weighted sum.

$$I_N = \frac{1}{N}\sum_{i=1}^{N} \mathrm{f}(\mathrm{x}^i)\tilde{w}(\mathrm{x}^i) \tag{A.9}$$

where $w(\mathrm{x}^i) = \dfrac{\pi(\mathrm{x}^i)}{q(\mathrm{x}^i)}$ are the importance weights. If the normalising factor of π(x) is unknown then normalisation is performed over the weights as follows.

$$w(\mathrm{x}^i) = \frac{\tilde{w}(\mathrm{x}^i)}{\sum_{j=1}^{N} \tilde{w}(\mathrm{x}^j)} \tag{A.10}$$

then the integral I_N can be computed as before.

$$I_N = \frac{1}{N} \sum_{i=1}^{N} \mathrm{f}(\mathrm{x}^i) w(\mathrm{x}^i) \tag{A.11}$$

using the normalised weights.

Sequential Importance Sampling

The importance sampling method is a general Monte Carlo integration upon which Particle Filter Methods are based for state-space estimation. The key idea is:

> “To represent the posterior probability density function as a set of samples(particles) with associated weights and to compute estimates based on these particles and weights”

Let $\mathrm{X}_k = \{\mathrm{x}_j, j = 0, \ldots, k\}$, represent the sequence of all target states up to time k. Let the joint posterior density be denoted as $p(\mathrm{X}_k|\mathrm{Z}_k)$ which is marginal at $p(\mathrm{x}_k|\mathrm{Z}_k)$. Also let $\{\mathrm{X}_k^i, i = 1, \ldots, N\}$ denote a set of support points and $\{w_k^i, i = 1, \ldots, N\}$ is a set of associated weights, that together characterise the posterior density $p(\mathrm{X}_k|\mathrm{Z}_k)$. Weights are normalised so that, $\sum w_k^i = 1$. Then the posterior probability density function at time k can be given from the following:

$$p(\mathrm{X}_k|\mathrm{Z}_k) \approx \sum_{i=1}^{N} w_k^i \delta(\mathrm{X}_k - \mathrm{X}_k^i) \tag{A.12}$$

The samples X_k^i are chosen from the “importance” density $q(\mathrm{X}_k|\mathrm{Z}_k)$. Then, according to A.10 the weights are given:

$$w_k^i \propto \frac{p(\mathrm{X}_k|\mathrm{Z}_k)}{q(\mathrm{X}_k^i|\mathrm{Z}_k)} \tag{A.13}$$

suppose at time $k-1$ we have samples that can approximate $p(\mathrm{X}_{k-1}|\mathrm{Z}_{k-1})$. At time k a new measurement becomes available and we wish to approximate $p(\mathrm{X}_k|\mathrm{Z}_k)$; The associated weights are computer from A.13 where after computing the update equation becomes:

$$w_k^i \propto w_{k-1}^i \frac{p(\mathrm{z}_k|\mathrm{x}_k^i)p(\mathrm{x}_k^i|\mathrm{x}_{k-1}^i)}{q(\mathrm{x}_k^i|\mathrm{x}_{k-1}^i, \mathrm{z}_k)} \tag{A.14}$$

and the posterior probability density function of the state vector x_k is given by:

$$p(\mathrm{x}_k|\mathrm{Z}_k) \approx \sum_{i=1}^{N} w_k^i \delta(\mathrm{x}_k - \mathrm{x}_k^i) \tag{A.15}$$

It can be shown that as $N \to \infty$ the approximation A.15 approaches the true $p(\mathrm{x}_k|\mathrm{Z}_k)$

This rather simple approach forms the basic Particle Filter.

In algorithmic terms the above procedure is summarised in the following table:

```
$[\{x_k^i, w_k^i\}_{i=1}^N]$ = SIS $[\{x_{k-1}^i, w_{k-1}^i\}_{i=1}^N, z_k]$
FOR $i$=1:N

- Draw $\mathrm{x}_k^i \sim q(\mathrm{x}_k|\mathrm{x}_{k-1}^i, z_k)$
 - Evaluate the importance weights up to a normalising constant according to A.14
$w_k^i \propto w_{k-1}^i \frac{p(\mathrm{z}_k|\mathrm{x}_k^i)p(\mathrm{x}_k^i|\mathrm{x}_{k-1}^i)}{q(\mathrm{x}_k^i|\mathrm{x}_{k-1}^i, \mathrm{z}_k)}$
 END FOR
 Calculate total weight  $t$ = SUM $[z\{\tilde{w}_k^i\}_{i=1}^N]$
 FOR $i$=1:N Normalise:  $w_k^i = t^{-1}\tilde{w}_k^i$

 END FOR
```

Resampling

Resampling eliminates samples with low importance weights and multiplies samples with high importance weights. Resampling is a mapping of random measure $(\mathbf{x}_k^i, w_k^i)$ into random measure $(\mathbf{x}_k^i*, 1/N)$ with uniform weights. The new set of random particles is generated by sampling with replacement from an approximate discrete representation of $(p(\mathbf{x}_k|\mathbf{Z}_k)$ given as:

$$p(\mathbf{x}_k|\mathbf{Z}_k) \approx \sum_{i=1}^{N} w_k^i \delta(\mathbf{x}_k - \mathbf{x}_k^i) \tag{A.16}$$

so that P($\mathbf{x}_k^i * = \mathbf{x}_k^j$) $= w_k^j$. The resulting sample is an i.i.d from the above density function and thus the new weights are uniform.

The resampling algorithm is given in the following

```
$[\{x_k^i, w_k^i, i^j\}_{i=1}^N]$ = RESAMPLE $[\{x_k^i, w_k^i\}_{i=1}^N, z_k]$
initialize the CSW (cumulative sum of weights): $c1 = w_k^1$

FOR i=2:N
    - Construct CSW: $c_i = c_{i-1} + w_k^i$
ENDFOR
- Start at the bottom of the CSW: i=1
- Draw a starting point $u_1 \sim U[0, N^{-1}]$

FOR j=1:N
    Move along the CSW: $u_j = U1 + N^{-1}(j-1)$
    WHILE $u_j > c_i$
        $i = i + 1$
    END WHILE
    -Assign sample $\mathbf{x}_k^{j*} = \mathbf{x}_k^i$
    -Assign sample $w_k^{j*} = N^{-1}$
    -Assign parent $i^j = i$
ENDFOR
```

Appendix B

Developed Software

The following flow diagrams describe the pieces of software developed for the *range requester* node, the *range responder* node the *target* node and the *central* node. The anchors nodes used the *responder* software on different communication channels.

Range Responder

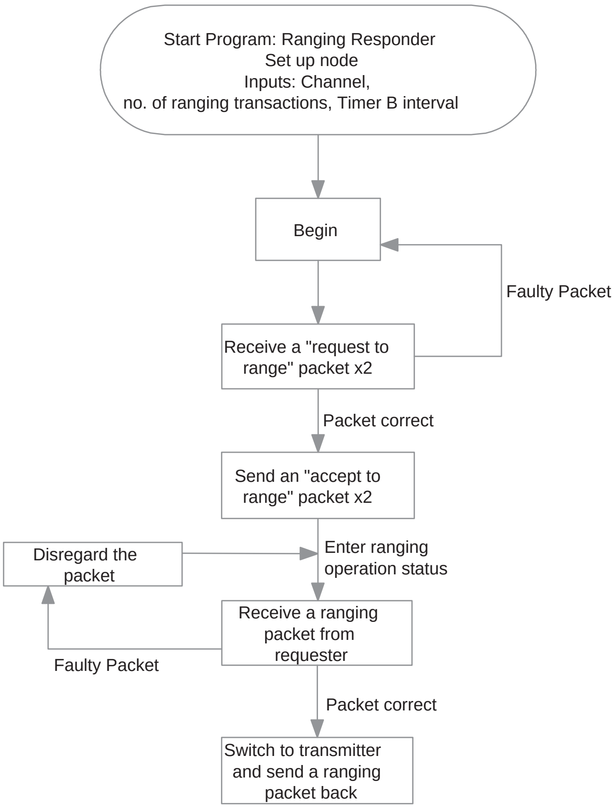

Start Program: Ranging Responder
Set up node
Inputs: Channel,
no. of ranging transactions, Timer B interval
Begin
Faulty Packet
Receive a "request to range" packet x2
Packet correct
Send an "accept to range" packet x2
Disregard the packet
Enter ranging operation status
Receive a ranging packet from requester
Faulty Packet
Packet correct
Switch to transmitter and send a ranging packet back

FIGURE B.1: Flow Diagram of the *responder* node software

Range Requester

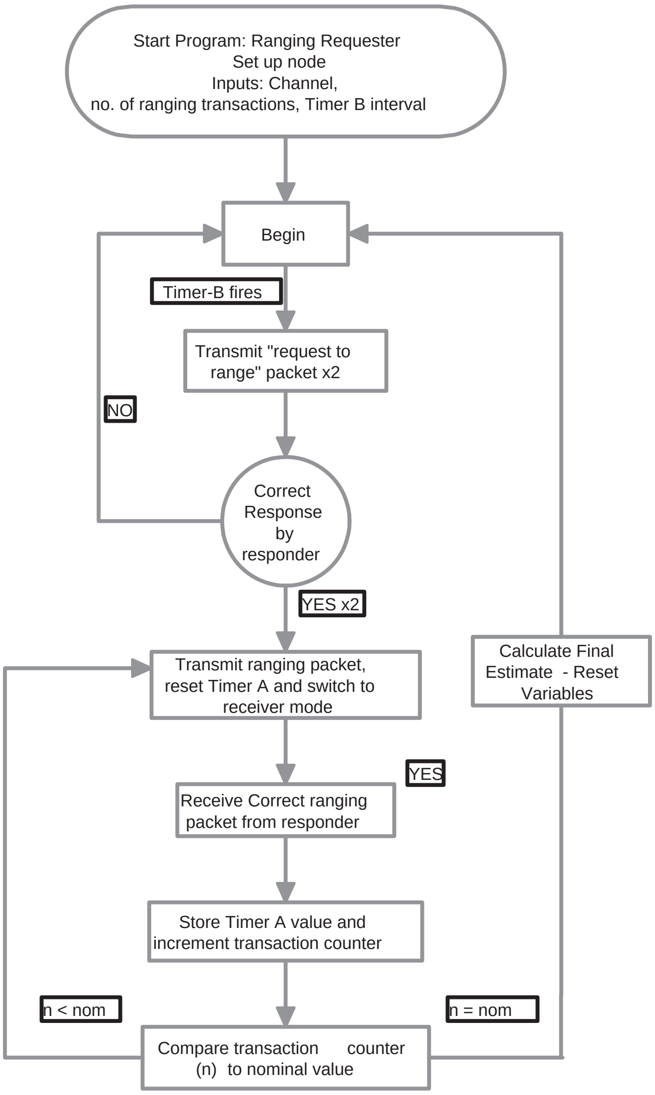

Start Program: Ranging Requester
Set up node
Inputs: Channel,
no. of ranging transactions, Timer B interval
Begin
Timer-B fires
Transmit "request to range" packet x2
NO
Correct Response by responder
YES x2
Transmit ranging packet, reset Timer A and switch to receiver mode
YES
Receive Correct ranging packet from responder
Store Timer A value and increment transaction counter
n < nom
n = nom
Compare transaction counter (n) to nominal value
Calculate Final Estimate - Reset Variables

FIGURE B.2: Flow Diagram of the *requester* node software

Target Node

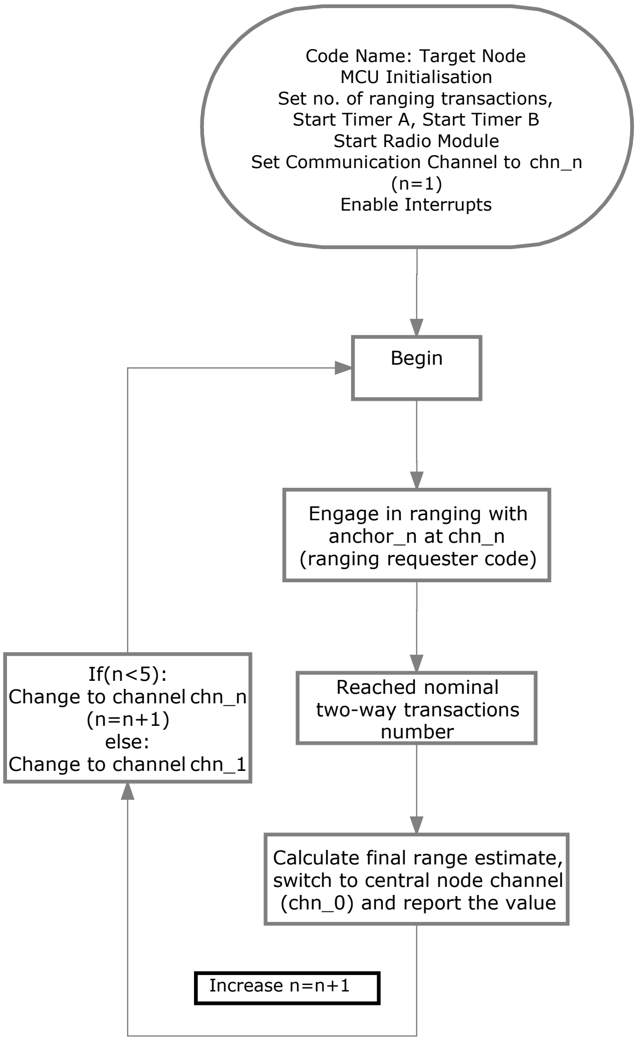


FIGURE B.3: Flow Diagram of the *target* node software

Central Node

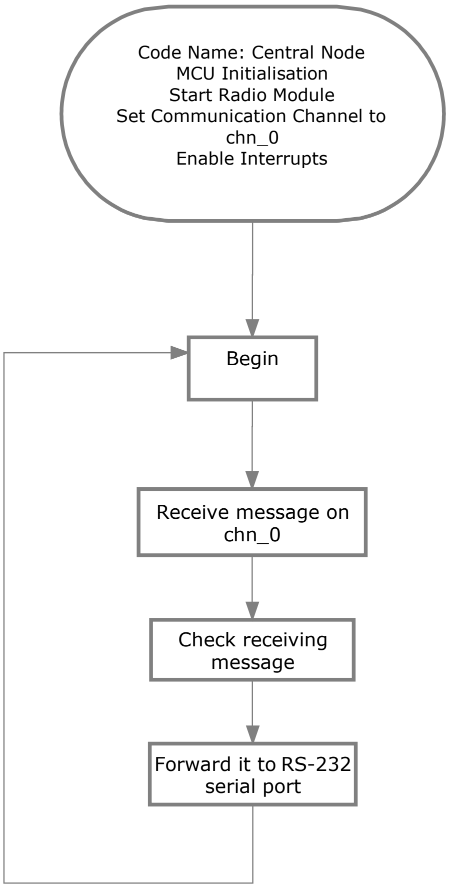


FIGURE B.4: Flow Diagram of the *cental* node software

Bibliography

[1] N. Ahmed, Y. Dong, T. Bokareva, S. Kanhere, S. Jha, T. Bessell, M. Rutten, B. Ristic, and N. Gordon. Detection and tracking using wireless sensor networks. In *Proceedings of the 5th International Conference on Embedded Networked Sensor Systems (SenSys '07)*, pages 425–426, 2007.

[2] N. Ahmed, Y. Dong, S.S. Kanhere, S. Jha, M. Rutten, T. Bessell, and N. Gordon. Performance evaluation of a wireless sensor network based tracking system. In *5th IEEE International Conference on Mobile Ad Hoc and Sensor Systems, (MASS 2008)*, pages 163–172, 2008.

[3] N. Ahmed, M. Rutten, T. Bessell, S.S. Kanhere, N. Gordon, and S. Jha. Detection and tracking using particle-filter-based wireless sensor networks. *IEEE Transactions on Mobile Computing*, 9(9):1332 –1345, Sep 2010.

[4] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci. Wireless sensor networks: a survey. *Computer Networks*, 38(4):393–422, 2002.

[5] The Zigbee Alliance. Zigbee specification document. `http://www.zigbee.org/ZigBeeSpecificationDownloadRequest/tabid/311/Default.aspx`, 2007.

[6] Y. Ammar, A. Buhrig, M. Marzencki, B. Charlot, S. Basrour, K. Matou, and M. Renaudin. Wireless sensor network node with asynchronous architecture and vibration harvesting micro power generator. In *Proceedings of the 2005 joint conference on Smart objects and ambient intelligence (sOc-EUSAI '05)*, pages 287–292, 2005.

[7] I. Amundson, X. Koutsoukos, and J. Sallai. Mobile sensor localization and navigation using rf doppler shifts. In *Proceedings of the first ACM international workshop on Mobile entity localization and tracking in GPS-less environments (MELT '08)*, pages 97–102, 2008.

[8] I. Amundson, J. Sallai, X. D. Koutsoukos, and A. Lédeczi. Radio interferometric angle of arrival estimation. In *Prceedings of the European Conference on Wireless Sensor Networks (EWSN 2010)*, pages 1–16, 2010.

[9] T. Arampatzis, J. Lygeros, and S. Manesis. A survey of applications of wireless sensors and wireless sensor networks. In *Proceedings of the 13th Mediterrean Conference on Control and Automation (MEDA 2005)*, pages 719–724, 2005.

[10] A. Arora, P. Dutta, S. Bapat, V. Kulathumani, H. Zhang, V. Naik, V. Mittal, H. Cao, M. Demirbas, M. Gouda, Y. Choi, T. Herman, S. Kulkarni, U. Arumugam, M. Nesterenko, A. Vora, and M. Miyashita. A line in the sand: a wireless sensor network for target detection, classification, and tracking. *Computer Networks*, 46(5):605–634, 2004.

[11] M. Sanjeev Arulampalam, B. Ristic, N. Gordon, and T. Mansell. Bearings-only tracking of manoeuvring targets using particle filters. *EURASIP J. Appl. Signal Process.*, 2004:2351–2365, January 2004.

[12] M.S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp. A tutorial on particle filters for online nonlinear/non-gaussian bayesian tracking. *IEEE Transactions on Signal Processing*, 50(2):174–188, Feb 2002.

[13] J. N. Ash and L. C. Potter. Sensor network localization via received signal strength measurements with directional antennas. In *Proceedings of the 2004 Allerton Conference on Communication, Control, and Computing*, pages 1861–1870, 2004.

[14] IEEE Standards Association. IEEE standard for information technology-telecommunications and information exchange between systems - local and metropolitan area networks - specific requirements - part 15.4: Wireless lan medium access (mac) and physical layer (phy) for low-rate wireless personal area networks(lr-wpans) (ieee std 802.15.4-2003), October 2003.

[15] IEEE Standards Association. IEEE standard for information technology-telecommunications and information exchange between systems - local and metropolitan area networks - specific requirements - part 15.4: Wireless lan medium access (mac) and physical layer (phy) for low-rate wireless personal area networks(lr-wpans) (ieee std 802.15.4-2007), September 2006.

[16] P. Bahl and V. N. Padmanabhan. Radar: an in-building rf-based user location and tracking system. In *Proceedings of the IEEE Nineteenth Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM 2000)*, pages 775–784 vol.2, 2000.

[17] Y. Bar-Shalom. *Multitarget-multisensor tracking: applications and advances.* Artech House Publishers, Dedham, MA, USA, 1992.

[18] Y. Bar-Shalom, T. Kirubarajan, and X.R. Li. *Estimation with Applications to Tracking and Navigation.* John Wiley & Sons, Inc., New York, NY, USA, 2002.

[19] A. Bessell, B. Ristic, A. Farina, X. Wang, and M.S. Arulampalam. Error performance bounds for tracking a manoeuvring target. In *Proceedings of the Sixth International Conference of Information Fusion*, pages 903 – 910, 2003.

[20] S. Blackman and R. Popoli. *Design and Analysis of Modern Tracking Systems.* Artech House Publishers, Dedham, MA, USA, 1999.

[21] W.D. Blair, G.A. Watson, T. Kirubarajan, and Y. Bar-Shalom. Benchmark for radar allocation and tracking in ecm. *IEEE Transactions on Aerospace and Electronic Systems*, 34(4):1097 –1114, oct 1998.

[22] D. Blatt and A.O. Hero. Energy-based sensor network source localization via projection onto convex sets. *IEEE Transactions on Signal Processing,*, 54(9):3614–3619, Sept. 2006.

[23] B. Bobrovsky and M. Zakai. A lower bound on the estimation error for markov processes. *IEEE Transactions on Automatic Control*, 20(6):785 – 788, dec 1975.

[24] T. Bokareva, W. Hu, S. Kanhere, B. Ristic, N. Gordon, T. Bessell, M. Rutten, and S. Jha. Wireless sensor networks for battlefield surveillance. In *Proceedings of The Land Warfare Conference 2006 (LWC 2006).*, 2006.

[25] M. Bolic, P.M. Djuric, and Sangjin Hong. Resampling algorithms and architectures for distributed particle filters. *IEEE Transactions on Signal Processing*, 53(7):2442 – 2450, july 2005.

[26] Qing Cao, Yan Ting, John Stankovic, and Tarek Abdelzaher. Analysis of target detection performance for wireless sensor networks. In *Proc of The 1st IEEE International Conference on Distributed Computing in Sensor Systems (DCOSS '05*, pages 276–292, 2005.

[27] V. Cevher, R. Velmurugan, and J.H. McClellan. A range-only multiple target particle filter tracker. In *Proceedings of 2006 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP 2006)*, volume 4, pages IV–IV, 2006.

[28] Y.T. Chan and S.W. Rudnicki. Bearings-only and doppler-bearing tracking using instrumental variables. *IEEE Transactions on Aerospace and Electronic Systems*, 28(4):1076–1083, Oct 1992.

[29] P. Chen, S. Oh, M. Manzo, B. Sinopoli, C. Sharp, K. Whitehouse, G. Tolle, J. Jeong, P. Dutta, J. Hui, S. Shaffert, S. Kim, J. Taneja, B. Zhu, T. Roosta, M. Howard, D. Culler, and S. Sastry. Experiments in instrumenting wireless sensor networks for real-time surveillance. In *Proc. of the International Conference on Robotics and Automation (ICRA 2006)*, 2006.

[30] W. C. Chung and D. Ha. An accurate ultra wideband (uwb) ranging for precision asset location. In *IEEE Conference Ultra Wideband Systems and Technologies (UWBST 2003)*, 2003.

[31] M. Coates. Distributed particle filters for sensor networks. In *Third International Symposium on Information Processing in Sensor Networks, (IPSN 2004)*, pages 99–107, 2004.

[32] M. Coates and G. Ing. Sensor network particle filters: motes as particles. In *IEEE 13th Workshop on Statistical Signal Processing (SSP 2005)*, pages 1152–1157, 2005.

[33] N. de Freitas. Rao-blackwellised particle filtering for fault diagnosis. In *Proc. of the IEEE Aerospace Conference (AeroSpace 2002)*, volume 4, pages 4–1767–4–1772 vol.4, 2002.

[34] Dept. of the Army, Washington D.C. *U.S. Army, Map Reading and Land Navigation, FM 21-26*, March 1956.

[35] D.M. Dooling and N. Sitar. Wireless sensors for wildlife monitoring. In *Proceedings of SPIE Symposium on Smart Structures and Materials - NDE*, 2005.

[36] A. Doucet and C. Andrieu. Iterative algorithms for optimal state estimation of jump markov linear systems. In *IEEE International Conference Proceedings of the Acoustics, Speech, and Signal Processing (ICASSP '99)*, pages 2487–2490, Washington, DC, USA, 1999. IEEE Computer Society.

[37] A. Doucet, N. De Freitas, and N. Gordon. *Sequential Monte Carlo methods in practice*. Springer - Verlag, 2001.

[38] A. Doucet, N. de Freitas, K.P. Murphy, and S.J. Russell. Rao-blackwellised particle filtering for dynamic bayesian networks. In *Proceedings of the 16th Conference on Uncertainty in Artificial Intelligence (UAI '00)*, pages 176–183, 2000.

[39] A. Doucet, N.J. Gordon, and V. Krishnamurthy. Particle filters for state estimation of jump markov linear systems. *IEEE Transactions on Signal Processing*, 49(3):613–624, Mar 2001.

[40] M. F. Duarte and Y. H. Hu. Vehicle classification in distributed sensor networks. *J. Parallel Distrib. Comput.*, 64(7):826–838, 2004.

[41] P. Dutta, J. Hui, J. Jeong, S. Kim, C. Sharp, J. Taneja, G. Tolle, K. Whitehouse, and D. Culler. Trio: enabling sustainable and scalable outdoor wireless sensor network deployments. In *The Fifth International Conference on Information Processing in Sensor Networks, 2006. IPSN 2006.*, pages 407–415, 2006.

[42] E. Elnahrawy, Xiaoyan Li, and R. P. Martin. The limits of localization using signal strength: a comparative study. In *First Annual IEEE Communications Society Conference on Sensor and Ad Hoc Communications and Networks, 2004. IEEE SECON*, pages 406–414, 2004.

[43] D. Estrin, D. Culler, K. Pister, and G. Sukhatme. Connecting the physical world with pervasive networks. *IEEE Pervasive Computing*, 1(1):59–69, 2002.

[44] D. Estrin, L. Girod, G. Pottie, and M. Srivastava. Instrumenting the world with wireless sensor networks. In *In International Conference on Acoustics, Speech, and Signal Processing (ICASSP 2001*, pages 2033–2036, 2001.

[45] D. Estrin, R. Govindan, and J. Heidemann. Embedding the internet: introduction. *Communications of the ACM*, 43(5):38–41, 2000.

[46] A. Farina, B. Ristic, and L. Timmoneri. Cramer-rao bound for nonlinear filtering with pd lt;1 and its application to target tracking. *IEEE Transactions on Signal Processing*, 50(8):1916 –1924, aug 2002.

[47] R.J. Fontana and S.J. Gunderson. Ultra-wideband precision asset location system. In *IEEE Conference on Ultra Wideband Systems and Technologies (UWBST 2002)*, 2002.

[48] T. Gao, C. Pesto, L. Selavo, Y. Chen, J. Ko, J. Lim, A. Terzis, A. Watt, J. Jeng, B. Chen, K. Lorincz, and M. Welsh. Wireless medical sensor networks in emergency response: Implementation and pilot results. In *IEEE Conference on Technologies for Homeland Security (IEEE HSET '08*, pages 187–192, 2008.

[49] D. Gay, P. Levis, R. von Behren, M. Welsh, E. Brewer, and D. Culler. The nesc language: A holistic approach to networked embedded systems. In *Proceedings of the ACM SIGPLAN 2003 conference on Programming language design and implementation (PLDI '03)*, pages 1–11, 2003.

[50] I Getting. The global positioning system. *IEEE Spectrum*, 30(12):36–47, Dec 1993.

[51] L. Gu, D. Jia, P. Vicaire, T. Yan, L. Luo, A. Tirumala, Q. Cao, T. He, J. A. Stankovic, T. Abdelzaher, and B. H. Krogh. Lightweight detection and classification for wireless sensor networks in realistic environments. In *Proceedings of the 3rd international conference on Embedded networked sensor systems (SenSys '05)*, pages 205–217, 2005.

[52] F. Gustafsson and F. Gunnarsson. Mobile positioning using wireless networks: possibilities and fundamental limitations based on available wireless network measurements. *IEEE Signal Processing Magazine*, 22(4):41–53, July 2005.

[53] F. Gustafsson, F. Gunnarsson, N. Bergman, U. Forssell, J. Jansson, R. Karlsson, and P.-J. Nordlund. Particle filters for positioning, navigation, and tracking. *IEEE Transactions on Signal Processing*, 50(2):425–437, Feb 2002.

[54] T. He, S. Krishnamurthy, L. Luo, T. Yan, L. Gu, R. Stoleru, G. Zhou, Q. Cao, P. Vicaire, J.A. Stankovic, T.F. Abdelzaher, J. Hui, and B. Krogh. Vigilnet: An integrated sensor network system for energy-efficient surveillance. *ACM Trans. Sen. Netw.*, 2(1):1–38, 2006.

[55] T. He, S. Krishnamurthy, J. A. Stankovic, T. Abdelzaher, L. Luo, R. Stoleru, T. Yan, Lin Gu, Jonathan Hui, and B. Krogh. Energy-efficient surveillance system using wireless

sensor networks. In *Proceedings of the 2nd international conference on Mobile systems, applications, and services (MobiSys '04)*, pages 270–283, 2004.

[56] J. Hightower and G. Borriello. Location systems for ubiquitous computing. *IEEE Computer*, 34(8):57 –66, Aug 2001.

[57] J. Hightower and G. Borriello. Location systems for ubiquitous computing. *Computer*, 34(8):57–66, Aug 2001.

[58] J. Hightower and G. Borriello. Particle filters for location estimation in ubiquitous computing: A case study. In *Sixth International Conference on Ubiquitous Computing (Ubicomp 2004)*, 2004.

[59] J. Hightower, G. Borriello, and R. Want. Spoton: an indoor 3d localizatechnology based on rf signal strength. Technical report, #Technical Report 2000-02-02, UWashington, Computer Science and Engineering, Feb 2000.

[60] Texas Instruments. Ez430-rf2500 development tool:user's guide, 2008.

[61] Texas Instruments. Simpliciti overview, 2008.

[62] Texas Instruments. Cc2500: Low-power 2.4ghz transceiver datasheet, 2009.

[63] Y. Ji, S. Biaz, S. Pandey, and P. Agrawal. Ariadne: a dynamic indoor signal map construction and localization system. In *Proceedings of the 4th international conference on Mobile systems, applications and services (MobiSys '06*, pages 151–164, 2006.

[64] X. Jiang, J. Polastre, and D. Culler. Perpetual environmentally powered sensor networks. In *Fourth International Symposium on Information Processing in Sensor Networks (IPSN 2005)*, pages 463–468, 2005.

[65] K. Kaemarungsi. Distribution of wlan received signal strength indication for indoor location determination. In *Proceedings of the 1st International Symposium on Wireless Pervasive Computing*, pages 1–6, 2006.

[66] J. M. Kahn, R. H. Katz, Y. H. Katz, and Kristofer S. J. Pister. Emerging challenges: Mobile networking for "smart dust". *Journal of Communications and Networks*, 2:188–196, 2000.

[67] J. M. Kahn, R. H. Katz, and K. S. J. Pister. Next century challenges: mobile networking for "smart dust". In *Proceedings of the 5th annual ACM/IEEE international conference on Mobile computing and networking (MobiCom '99)*, pages 271–278, 1999.

[68] R. E. Kalman. A new approach to linear filtering and prediction problems. *Transactions of the ASME Journal of Basic Engineering*, 3(82 (Series D)):35–45, 1960.

[69] C. Kam and W.S. Hodgkiss. Distributed target tracking in a wireless sensor network. In *Fortieth Asilomar Conference on Signals, Systems and Computers,(ACSSC '06)*, pages 1999–2003, 2006.

[70] T.C. Karalar and J. Rabaey. An rf tof based ranging implementation for sensor networks. In *Proceeding of the IEEE International Conference on Communications(ICC '06)*, volume 7, pages 3347–3352, 2006.

[71] K.Radhakrishnan, A. Unnikrishnan, and K.G Balakrishnan. Bearing only tracking of maneuvering targets using a single coordinated turn model. *International Journal of Computer Applications*, 1(1):25–33, February 2010.

[72] B. Kusý, G. Balogh, J. Sallai, Á. Lédeczi, and M. Maróti. intrack: High precision tracking of mobile sensor nodes. In *Proceedings of the 2007 European Conference in Wireless Sensor Networks, (EWSN 2007)*, volume 4373, pages 51–66, 2007.

[73] B. Kusy, A. Ledeczi, and X. Koutsoukos. Tracking mobile nodes using rf doppler shifts. In *Proceedings of the 5th International Conference on Embedded Networked Sensor Systems (SenSys '07)*, pages 29–42, 2007.

[74] B. Kusy, A. Ledeczi, M. Maroti, and L. Meertens. Node density independent localization. In *Proceedings of the 5th International Conference on Information Processing in Sensor Networks, (IPSN '06)*, pages 441–448, 2006.

[75] B. Kusy, J. Sallai, G. Balogh, A. Ledeczi, V. Protopopescu, J. Tolliver, F. DeNap, and M. Parang. Radio interferometric tracking of mobile wireless nodes. In *Proceedings of the 5th international conference on Mobile systems, applications and services (MobiSys '07)*, pages 139–151, 2007.

[76] S. Lanzisera. *RF Ranging for Location Awareness*. PhD thesis, University of California, Berkeley, May 2009.

[77] S. Lanzisera, D.T. Lin, and K.S.J. Pister. Rf time of flight ranging for wireless sensor network localization. In *International Workshop on Intelligent Solutions in Embedded Systems*, 2006.

[78] J.Y. Lee and R.A. Scholtz. Ranging in a dense multipath environment using an uwb radio link. *IEEE J Sel. Areas Commun.*, 20(9):1677 – 1683, Dec. 2002.

[79] K. Lin, J. Yu, J. Hsu, S. Zahedi, D. Lee, J Friedman, A Kansal, V. Raghunathan, and M. Srivastava. Heliomote: enabling long-lived sensor networks through solar energy harvesting. In *Proceedings of the 3rd international conference on Embedded networked sensor systems (SenSys '05)*, pages 309–309, 2005.

[80] M. Maróti, B. Kusy, G. Simon, and Á. Lédeczi. The flooding time synchronization protocol. In *2nd ACM Conference on Embedded Networked Sensor Systems (Sensys 2004)*, 2004.

[81] M. Maróti, P. Völgyesi, S. Dóra, B. Kusý, A. Nádas, Á. Lédeczi, G. Balogh, and K. Molnár. Radio interferometric geolocation. In *Proceedings of the 3rd international conference on Embedded networked sensor systems (Sensys 2005)*, pages 1–12, 2005.

[82] S. J. Maybank. Bearings-only tracking in the plane. *SIAM J. Appl. Math.*, 58(3):975–998, 1998.

[83] D.D. McCrady, L. Doyle, H. Forstrom, T. Dempsey, and M. Martorana. Mobile ranging using low-accuracy clocks. *IEEE Trans. Microw. Theory Techn.*, 48(6):951 –958, Jun 2000.

[84] D. Mušicki. Bearings only single-sensor target tracking using gaussian mixtures. *Automatica*, June 2009.

[85] S. Oh, P. Chen, M. Manzo, and S. Sastry. Instrumenting wireless sensor networks for real-time surveillance. In *Proc. of the International Conference on Robotics and Automation (ICRA 2006)*, 2006.

[86] S. Oh, S. Russell, and S. Sastry. Markov chain monte carlo data association for general multiple-target tracking problems. In *43rd IEEE Conference on Decision and Control (CDC 2004)*, volume 1, pages 735–742 Vol.1, 2004.

[87] S. Oh, S. Sastry, and L. Schenato. A hierarchical multiple-target tracking algorithm for sensor networks. In *Proceedings of the 2005 IEEE International Conference on Robotics and Automation (ICRA 2005)*, pages 2197–2202, 2005.

[88] P. Padhy, K. Martinez, A. Riddoch, H. L . Royan Ong, and J. K. Hart. Glacial environment monitoring using sensor networks. In *Real-World Wireless Sensor Networks (REALWSN 05)*, 2005.

[89] D. Pandya, R. Jain, and E. Lupu. Indoor location estimation using multiple wireless technologies. In *14th IEEE Proceedings on Personal, Indoor and Mobile Radio Communications (PIMRC 2003)*, volume 3, pages 2208–2212 vol.3, 2003.

[90] N. Patwari, J.N. Ash, S. Kyperountas, III Hero, A.O., R.L. Moses, and N.S. Correal. Locating the nodes: cooperative localization in wireless sensor networks. *IEEE Signal Process. Mag.*, 22(4):54 – 69, Jul 2005.

[91] N. Patwari, Hero, M. Perkins, N. S. Correal, and R. J. O'Dea. Relative location estimation in wireless sensor networks. *IEEE Transactions on Signal Processing*, 51(8):2137–2148, 2003.

[92] B. Priyantha, N, A. Chakraborty, and H. Balakrishnan. The cricket location-support system. In *Proceedings of the 6th annual international conference on Mobile computing and networking (MobiCom '00)*, pages 32–43, New York, NY, USA, 2000.

[93] N. B. Priyantha, H. Balakrishnan, E. Demaine, and S. Teller. Mobile-Assisted Localization in Wireless Sensor Networks. In *Proceedings of the 24th IEEE International Conference on Computer Communications (INFOCOM 2005)*, 2005.

[94] N. B. Priyantha, A. K.L. Miu, H. Balakrishnan, and S. Teller. The cricket compass for context-aware mobile applications. In *Proceedings of the 7th annual international conference on Mobile computing and networking*, MobiCom '01, pages 1–14, 2001.

[95] V. Rajaravivarma, Yi Yang, and Teng Yang. An overview of wireless sensor network and applications. In *Proceedings of the 35th Southeastern Symposium on System Theory*, pages 432–436, 2003.

[96] B. Ristic and M. S. Arulampalam. Tracking a manoeuvring target using angle-only measurements: algorithms and performance. *Signal Process.*, 83(6):1223–1238, 2003.

[97] B. Ristic and S. Arulampalam. Integrated detection and tracking of multiple objects with a network of acoustic sensors. In *10th International Conference on Information Fusion*, pages 1–7, 2007.

[98] B. Ristic, S. Arulampalam, and N. Gordon. *Beyond the Kalman Filter - Particle Filters For Tracking Applications*. Artech House Publishers, Dedham, MA, USA, 2004.

[99] C. P. Robert and G. Casella. *Monte Carlo Statistical Methods (Springer Texts in Statistics)*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2005.

[100] T. Roos, P. Myllymäki, H. Tirri, P. Misikangas, and J. Sievänen. A probabilistic approach to wlan user location estimation. *International Journal of Wireless Information Networks*, 9(3):155–164, July 2002.

[101] T. Roosta, S. M. Mishra, and A. Ghazizadeh. Robust estimation and detection in ad hoc and sensor networks. In *IEEE International Conference on Mobile Adhoc and Sensor Systems (MASS 2006)*, pages 236–245, 2006.

[102] P.A. Rosenqvist. Passive doppler-bearing tracking using a pseudo-linear estimator. *IEEE Journal of Oceanic Engineering*, 20(2):114–118, Apr 1995.

[103] M.I. S. *Radar Handbook*. McGraw-Hill Book Co., New York, NY, 1990.

[104] S. Saha, N.K. Bambha, and S.S. Bhattacharyya. Parameterized design framework for hardware implementation of particle filters. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP 2008)*, pages 1449 –1452, 2008.

[105] J. Sallai, G. Balogh, M. Maroti, A. Ledeczi, and B. Kusy. Acoustic ranging in resource-constrained sensor networks. In *International Conference on Wireless Networks (ICWN 2004)*, Las Vegas, Nevada, USA, 2004.

[106] A. Savvides, C. C. Han, and M. B. Strivastava. Dynamic fine-grained localization in ad-hoc networks of sensors. In *Proceedings of the 7th Annual ACM International Conference on Mobile Computing and Networking (MobiCom '01)*, pages 166–179, 2001.

[107] S.Y. Seidel and T.S. Rappaport. 914 mhz path loss prediction models for indoor wireless communications in multifloored buildings. *IEEE Transactions on Antennas and Propagation*, 40(2):207 –217, Feb 1992.

[108] X. Sheng and Y-H. Hu. Energy based acoustic source localization. In *Proceedings of the Internation Workshop in Infrormation Processing in Sensor Networks (IPSN 2003)*, pages 285–300, 2003.

[109] X. Sheng and Y.-H. Hu. Maximum likelihood multiple-source localization using acoustic energy measurements with wireless sensor networks. *IEEE Transactions on Signal Processing*, 53(1):44–53, Jan. 2005.

[110] Q. Shi, N. Correal, S. Kyperountas, and F. Niu. Performance comparision between toa ranging technologies and rssi ranging technologies for multi-hop wireless networks. In *IEEE 62nd Vehicular Technology Conference, 2005. VTC-2005-Fall*, volume 1, pages 434–438, 2005.

[111] M. Simandl, J. Krlovec, and P. Tichavsk. Filtering, predictive, and smoothing cramr-rao bounds for discrete-time nonlinear dynamic systems. *Automatica*, 37(11):1703 – 1716, 2001.

[112] F. Sivrikaya and B. Yener. Time synchronization in sensor networks: a survey. *IEEE Network*, 18(4):45 – 50, Jul 2004.

[113] A. Smith, H. Balakrishnan, M. Goraczko, and N. B. Priyantha. Tracking Moving Devices with the Cricket Location System. In *2nd International Conference on Mobile Systems, Applications and Services (Mobisys 2004)*, 2004.

[114] Taek Lyul Song. Observability of target tracking with range-only measurements. *IEEE Journal of Oceanic Engineering*, 24(3):383–387, Jul 1999.

[115] R. Szewczyk, A. Mainwaring, J. Polastre, J. Anderson, and D. Culler. An analysis of a large scale habitat monitoring application. In *Proceedings of the 2nd international conference on Embedded networked sensor systems (SenSys 2004)*, pages 214–226, 2004.

[116] R. Szewczyk, J. Polastre, A. Mainwaring, and D. Culler. Lessons from a sensor network expedition. In *Proc. of the First European Workshop on Wireless Sensor Networks (EWSN 2004)*, pages 307–322, 2004.

[117] L.G. Taff. Target localization from bearings-only observations. *IEEE Transactions on Aerospace and Electronic Systems*, 33(1):2–10, Jan. 1997.

[118] X.-J. Tao, C.-R. Zou, and Z.-Y. He. Passive target tracking using maximum likelihood estimation. *IEEE Transactions on Aerospace and Electronic Systems*, 32(4):1348–1354, Oct 1996.

[119] C. Taylor, A. Rahimi, J. Bachrach, H. Shrobe, and A. Grue. Simultaneous localization, calibration, and tracking in an ad hoc sensor network. In *Proceedings of the 5th international conference on Information processing in sensor networks (IPSN '06)*, pages 27–33, 2006.

[120] J.H. Taylor. The cramer-rao estimation error lower bound computation for deterministic nonlinear systems. In *IEEE Conference on Decision and Control including the 17th Symposium on Adaptive Processes (CDC)*, volume 17, pages 1178 –1181, 1978.

[121] B. Thorbjornsen, N. White, A. Brown, and J. Reeve. Radio frequency (rf) time-of-flight ranging for wireless sensor networks. *Measurement Science and Technology*, 21(3):1–12, March 2010.

[122] P. Tichavsky, C.H. Muravchik, and A. Nehorai. Posterior cramer-rao bounds for discrete-time nonlinear filtering. *IEEE Transactions on Signal Processing*, 46(5):1386 –1396, may 1998.

[123] S. P. M. Tran and T. A. Yang. Evaluations of target tracking in wireless sensor networks. In *Proceedings of the 37th SIGCSE technical symposium on Computer science education (SIGCSE '06)*, pages 97–101, 2006.

[124] US NAVY. *Introduction to Naval Weapons Engineering Course Syllabus*. Federation of American Scientists, Military Analysis Network.

[125] H. L. Van Trees. *Detection, Estimation, and Modulation Theory, Part I*. Wiley-Interscience, 1 edition, September 2001.

[126] J. E. Vargas and Z. Wu. Real-time multiple target tracking using networked wireless sensors. In *Second Conference on Autonomous Intelligent Networks and Systems*, 2003.

[127] K. Whitehouse and D. Culler. Calibration as parameter estimation in sensor networks. In *1st ACM Workshop on Wireless Sensor Networks and Applications*, 2002.

[128] K. Whitehouse, C. Karlof, and D. Culler. A practical evaluation of radio signal strength for ranging-based localization. *SIGMOBILE Mob. Comput. Commun. Rev.*, 11:41–52, January 2007.

[129] M. Youssef, A. Agrawala, and A. U. Shankar. Wlan location determination via clustering and probability distributions. In *In Proceedings of IEEE International Conference on Pervasive Computing and Communications (Percom 2003)*, 2003.

[130] Q. Zhang, G. Sobelman, and T. He. Gradient-driven target acquisition in mobile wireless sensor networks. In *Proceedings of the 2nd Second International conference on Mobile ad-hoc and sensor networks (MSN 2006)*, pages 365–376, 2006.

[131] F. Zhao, J. Liu, J. Liu, L. Guibas, and J. Reich. Collaborative signal and information processing: an information-directed approach. *Proceedings of the IEEE*, 91(8):1199–1209, Aug. 2003.

[132] F. Zhao, J. Shin, and J. Reich. Information-driven dynamic sensor collaboration. *IEEE Signal Processing Magazine*, 19(2):61–72, Mar 2002.