Swiss Federal Institute of Technology (ETHZ)
Department of Information Technology and Electrical
Engineering (ITET)
IBM Research Zurich

Master of Science Thesis

Algorithms for Sensor Localization and Synchronization in Wireless Sensor Networks

by

Paraskevi Papoula

Supervisor: Dr. Olga Saukh, ETH Zurich, TIK
Tutor: Dr. Wolfgang Schott, IBM Zurich Research Laboratory
Professor: Prof. Dr. Lothar Thiele

Zurich 2011

Summary

Wireless Sensor Networks (WSN) are becoming more and more famous over the years as their cost decreases and their capabilities increase. They offer their services in various fields such as monitoring buildings, human health, environmental events and so on. In many cases it is not feasible to manually locate the sensors at specific positions, thus they are randomly deployed or thrown over an area. Therefore, self-localization mechanisms have been developed to determine the position of each node and the deviation of its clock from a reference clock with an accuracy which depends on the application. We address these challenges in the context of land seismic exploration where the accurate location estimation and time synchronization of the sensor nodes is essential for generating a good quality three dimensional seismic image.

In this master thesis, we introduce a joint localization and time synchronization algorithm for WSNs. A model is derived for jointly estimating the location and time parameters of sensor nodes by exchanging short communication messages between sensors and reference nodes. We compute the minimum error variance of the algorithm and use it to assess the derived estimation model. We developed a simulator which allows us to vary model parameters such as the noise, the signal to noise ratio, the bandwidth of the communication signals, and visualize the performance of the algorithm in these scenarios. Finally, we investigated the implementation feasibility and the computational cost of the proposed algorithm. According to the results, when using the Ultra Wideband technology for the communication between the sensors, the estimation accuracy is of the order of centimeters for location coordinates and of the order of sub-nanoseconds for the time parameters.

Acknowledgments

It is a pleasure to thank those who made this thesis possible. In the following lines some of them are gratefully acknowledged. However, these words cannot express the gratitude and respect I feel for all of those.

Firstly, I would like to thank the people who provided scientific support to make this work possible. I must thank Dr. Wolfgang Schott who initiated the thesis project and gave me the unique opportunity to join IBM Research. His encouragement supervision and support from the preliminary to the concluding level enabled me to develop an in-depth understanding on the subject of my thesis. My sincere thanks go to Prof. Dr. Lothar Thiele and Dr. Olga Saukh for their support and valuable input during my master thesis.

I am indebted to my colleagues in IBM Research for creating a motivating and pleasant working environment. In addition, I would like to thank my friends for their continuous support, advice and encouragement.

Last but not least, I would like to show my gratitude to my family for general education and the opportunity to start and pursue a career in science. I am particularly indebted to my parents for their never-ending encouragement and on-going support. Very special thanks go to my beloved brother Michael and sister Eleni for always being there to advice and support me with words of deep sense.

Contents

Chapter 1

Introduction

Oil and gas companies are interested in acquiring a three dimensional image of the sub-ground to decide upon extracting oil from different areas. Nowadays wired sensor networks are deployed to explore the sub-ground; sources emit signals at predetermined locations and sensors called *geophones* collect the reflected vibration signals from the geological sub-ground. The means to produce the transmitted signal vary with the sub-ground. Therefore, in marine survey the sources are vibrator, air gun, electric sparker and confined propane-oxygen explosions, while on land survey the sources are dynamites, vibrioses, weight drop, large caliber gun and vibrator. In the future, wireless sensor networks will be deployed to explore the sub-ground, decreasing the cabling cost, the attenuation introduced by the cables and the time needed to deploy the wired sensors along areas that cover many square kilometers.

In land seismic exploration, wireless geophone networks [1] consisting of thousands of sensors collect the reflected signals from the sub-ground, transmit them to a gateway along with their location and time so that an accurate three dimensional seismic image can be constructed. In such a network some of the sensors, called anchor nodes, know their location and are time synchronized to a reference clock (e.g. they may be equipped with GPS receivers, or be connected by a cable). The rest of the sensor nodes on the other hand are located in unknown positions and their clocks deviate from the reference clock. A wireless sensor network for this application faces two new challenges: very small time synchronization error and location estimation accuracy of less than one meter.

1.1 Goal of the Thesis

The goal of this thesis is to jointly localize sensor nodes in a sensor field and synchronize them to the reference clock by exchanging short messages between

3

each sensor node and at least three anchor nodes. Furthermore, this report includes a sensitivity study in terms of accuracy of localization algorithms, time-synchronization error, complexity of investigated algorithms and scalability issues in wireless sensor networks.

1.2   Thesis Outline

This thesis is organized as follows. The first chapter provides a brief review of the most important background information on the two problems that we address in this thesis, namely localization and time synchronization. Chapter 2 discusses the state of the art for localization and time synchronization methods in wireless sensor networks. In Chapter 3, we present a model for self-localizing sensors when they are all perfectly time synchronized; we validate it with a regression model and verify the results with simulations. Chapter 4 describes an algorithm to jointly localize and time synchronize sensor nodes by exchanging short messages with anchor nodes. A mathematical analysis of the algorithm is presented and the minimum error variance for any estimator is computed. In addition, the results of the simulations for different scenarios are compared. In Chapter 5, we present the computation complexity of the suggested algorithm in terms of mathematical operations and discuss the hardware limitations that arise.

1.3   Notation

During the thesis we use the following notations:

- The set of all $m \times n$ matrices is denoted by $A_{m \times n}$

- ^{T} stands for the transpose operation on a matrix

- E denotes the expectation operator

- Given a vector $\boldsymbol{x} = [x_1, x_2]^T$, its Euclidean norm is given by $||\boldsymbol{x}|| := \sqrt{x_1^2 + x_2^2}$

- $|\cdot|$ stands for the absolute value of a scalar or the determinant of a matrix

- $\odot$ denotes the element by element product known as Hadamard product and $\otimes$ denotes the Kronecker product

- $\mathbf{1}_{m \times n}$ denotes an $m \times n$ matrix of ones, $\boldsymbol{E}_{m \times 1}$ denotes an $m \times 1$ vector of alternating -1 and 1 respectively and $\boldsymbol{I}$ denotes an identity matrix

Chapter 2

Background Information - Basic Concepts

This chapter explains the most basic concepts in the fields of localization and time synchronization in wireless sensor networks.

2.1 Wireless Sensor Networks

Wireless Sensor Networks are becoming more and more famous over the years as their cost decreases and their capabilities increase. They offer their services in various fields such as monitoring buildings, human health, environmental events and so on. A WSN comprises nodes which are able to monitor physical or environmental events, communicate between them and self-organize. Each sensor node is typically equipped with a radio transceiver to communicate wirelessly, a power source such as battery or solar cells to power itself for some time, a processing unit such as a microcontroller, or a Digital Signal Processor to offer processing capabilities, a memory such as flash or RAM to store data and a sensor or actuator to sense an environmental or physical event. In a WSN, there is a trade-off between the resources and the size and cost of the sensors; more expensive sensors are equipped with higher computation speed, communication bandwidth and memory. Moreover the challenges in WSNs that are addressed by the algorithms refer to the sensor lifetime, the robustness and fault tolerance and the self-configuration.

2.2   Sensor Node Localization

In many cases it is not feasible to manually locate the sensors at specific positions, thus they are randomly deployed or thrown over an area. Therefore, self-localization mechanisms have been developed to determine the position of each node with an accuracy which depends on the application. For some of these applications, a very precise location estimation of each sensor node is required. In this section, we describe the existing mechanisms for the localization of sensor nodes along with their advantages and disadvantages.

The mechanisms designed to determine the location of a sensor node can be separated in two categories: range-based and range-free. In the former case, the node location is derived from point-to-point distance or angle estimates, while in the latter one proximity information is derived by beacons from anchor nodes. Further categorization exists depending on whether there is communication between nodes (active localization) or not (passive localization).

2.2.1   Range-Based Localization

There are four mechanisms to estimate the location of each node in a wireless sensor network. It is usual that a combination of two or more of them is used to achieve higher accuracy in locating the nodes. Once distance information to at least three anchors is available through any of these mechanisms, a simple trilateration or multilateration algorithm yields the location of the sensor nodes.

Received Signal Strength Information (RSSI)

Many algorithms take advantage of the power level at the receivers to infer their distance from the sender [2]. In a wireless sensor network, where nodes apply this mechanism to self-localize, anchors include their power level in the transmitted packet and receivers subtract it from the received power. This power difference indicates information about their geometrical distance according to an attenuation model and at the same time it removes any kind of dependency on the actual transmitted power. This approach is very attractive in terms of device complexity and cost, but the achieved accuracy is its major drawback; accuracy decreases when the distance increases. Additionally RSSI is sensitive to environmental changes such as walls, trees, weather, and also to multipath fading. The RSSI errors are multiplicative and furthermore the RSSI is a function of specific component values on a board which means that the original transmit power might not be known leading to an inaccurate distance estimate.

Time of Arrival (ToA)

The second range-based approach described here is the Time of Arrival (ToA). According to this mechanism, the receiver notes the time as displayed by its clock when a signal first arrives. Assuming that the receiver knows exactly when the packet was transmitted, it calculates the total time needed for the packet transmission and propagation delay. Based on this total time and knowing the propagation speed of the signal through the medium, the receiver can infer its distance to the sender. It is very important that the ToA of the Line of Sight (LOS) signal is identified, otherwise the computed distance will be larger than the real one.

The main sources of error [3] arise from the multipath propagation, the additive noise and the time synchronization. However, the following procedures reduce the effect of these errors:

- In case of multipath errors, there might exist early arriving multipath signals or attenuated LOS ones (hidden in noise) making it difficult to choose the correct LOS signal. In such cases, the ToA can be set to the time that the cross correlation between the received signals and the known transmitted one first crosses a threshold.

- In the case of additive noise, the receiver finds the ToA by calculating the cross correlation as in previous case. The multipath propagation errors are more severe than those caused by noise. Here it should be noted that the time delays in transmitter and receiver hardware and software are included in the measured ToA.

- As far as time synchronization is concerned, if the clock of each node is not synchronized with the reference clock, the receivers will falsely estimate the duration that the signal was actually being propagated (Time of Flight, ToF). A way to overcome this is to measure the round trip ToA, so that the same clock will be used in the timestamping. Another way is to consider the state of each sensor's clock as an unknown parameter and then give it as input along with the one way ToA to a localization algorithm to estimate both the sensor location and the biases of each sensor's clock. The round trip ToA can also be measured in asynchronous sensor networks, where multiple sensors reply to a signal at the same time using multiuser interference cancellation.

Acoustic and Radio Frequency signals are the most popular candidates for the ToA approach for reasons that are explained in the following section. As far as clock synchronization is concerned a precision of 10 μsec is adequate for acoustic signals (corresponding location accuracy $\approx 3.4 \ mm$) but not for radio frequency ones (corresponding location accuracy $\approx 3000 \ m$).

Time Difference of Arrival (TDoA)

This approach is similar to ToA in the sense that the receiver still notes the arrival time of signals, but in this case it receives two signals of different frequency. The difference between this approach and the ToA is that the receiver takes advantage of the difference in propagation times of the used frequencies to calculate the distance to the sender [3].

Usually a radio and an acoustic signal are used for this method. Upon reception of the Radio Frequency (RF) signal, the receiver starts a timer to measure the elapsed time until the acoustic signal is received. This elapsed time includes the propagation delay and the transmission time of the packet. Thus, there is no need for global time synchronization among the sensor nodes in this technique. When there is ambient noise in the environment, an alternative suggestion is to use radio-interferometric with RF signals to measure the phase difference between two senders and two receivers; the latter method is recommended when there is grassy environment because grass has a damping effect on acoustic signals. There are three basic reasons why acoustic signals are preferred for ToA and TDoA. At first the acoustic medium is isotropic with predictable attenuation. Furthermore, it yields reasonable accuracy and finally it requires inexpensive available hardware.

Angle of Arrival (AoA)

The last range-based mechanism refers to the Angle of Arrival of a signal to a receiver. This mechanism usually provides complementary information to ToA and RSS by indicating the direction of neighboring sensors. There are two ways to acquire AoA measurements and both require multiple antenna elements: the first one is to use a sensor array and array signal processing techniques and the second one is to use the RSS ratio between two directional antennas located on the sensor. The main sources of error are the same as the ones that affect ToA. The AoA measurements are modeled as Gaussian. When they are combined with time measurements they can overcome NLOS errors. One advantage of AoA over the other range-based mechanisms is that the errors in angle-measurements propagate slower [4].

2.2.2   Range-Free Localization

In range-free localization algorithms, there is no assumption about the availability of absolute point-to-point distance estimates between sensor nodes. Therefore, the location of sensor nodes is estimated by exploiting the radio connectivity information or the sensing capabilities of each sensor. These algorithms may require some reference nodes.

2.2.3 Active Localization

In order to characterize a localization method as active, we need to know whether the transmitted signals were intended for localizing the nodes. The stimuli in this category must be artificially generated for the localization process. A comprehensive survey on active localization methods can be found in [5]. Any of the four techniques described above can be characterized as active if the transmitted RF or acoustic signals are part of the localization process and their generation is controlled.

2.2.4 Passive Localization

In the case of passive localization methods, the stimuli needed for the localization process are not generated in a controlled way, but in a natural manner. As an example, Kwon et al in [6] describe a sensor network, where the sensors record global environmental events such as the sound of a thunder, vibrations of seismic events or shadows of clouds, and localize the sensors nodes based on the different ToA of the global events. These methods do not need any additional infrastructure and thus the maintenance cost is much lower than in the active localization methods. In addition, they are commonly used in applications such as indoor positioning, where global positioning systems are not available.

2.2.5 Global Positioning System (GPS)

Nowadays when the word *positioning* is mentioned the Global Positioning System comes automatically in mind. The GPS is a constellation of satellites overall that are in 12-hour orbits around the earth. For positioning four satellites are adequate but up to twelve synchronized satellites may be used to increase the accuracy of localization. There are two GPS signals: a civilian and a military one. The civilian signal can be received by Standard Positioning System (SPS) receivers and provides a "worst case" pseudorange accuracy of 7.8 m at 95% confidence level. The actual accuracy experienced by the users depends on atmospheric effects and receiver quality and can reach up to 3 m horizontal position accuracy. The military required special components to be received and it was intended for US and Allied military users. Due to encryption and jam-resistance it was more robust than the civilian GPS signal. After year 2000, it was made available to civilians but the military scrambled their radio signals to protect the location of military troops and operations that had taken place. Today higher accuracy can be achieved if GPS is used in combination with various augmentation systems. These enable real time positioning to within a few centimeters and post-processed positioning to within millimeters. The GPS accuracy can be affected by:

- the satellite position; in case of existing mountains between the satellites and the point of interest the error can be up to 30 m.

- noise in radio signal; in this case the error may be from 1 to 10 m.

- atmospheric conditions; they affect the speed of the radio signal leading to inaccurate time of flight and thus inaccurate distance calculation.

- multipath effects; when signals get reflected on objects before reaching the target leading to an increased time of flight. The receiver recognizes these multipath signals by a technique named narrow collector spacing.

If the Assisted GPS is used then even with less than four satellites positioning is possible but the accuracy is decreased up to 50 m. However, the Time To First Fix (TTFF) is decreased because of the assistance from the mobile base station. The Wide Area Augmentation System (WAAS) is an air navigation aid system that is developed to assist the GPS in terms of accuracy (1 m horizontally and 1.5 m vertically), integrity and availability. Furthermore, when someone considers using GPS for localization, he should bare in mind that the connection is not guaranteed, that the time for synchronizing with the satellites can reach up to 12 minutes.

2.3   Time Synchronization

Before we explain time synchronization, let us mention a few hardware details about sensors. Each sensor contains a crystal oscillator circuit which sustains oscillation by taking a voltage signal from the quartz resonator, amplifying it, and feeding it back to the resonator. The oscillator amplifies the signals coming out of the crystal, making the crystal's frequency stronger which eventually dominates the output of the oscillator. Natural resistance in the circuit and in the quartz crystal filters out all the unwanted frequencies.

One of the most important advantages of quartz crystal oscillators is that they exhibit very low phase noise, thus producing a pure tone. This makes them particularly useful in telecommunications where stable signals are needed, and in scientific equipment where very precise time references are essential.

The output frequency of a quartz oscillator is either the fundamental resonance or a multiple of the resonance.

Environmental changes of temperature, humidity, pressure, and vibration can change the resonant frequency of a quartz crystal. Due to aging and environmental factors such as temperature and vibration, constant adjustment is necessary to keep even the best quartz oscillators within a small deviation from their nominal frequency.

Although crystals can be fabricated for any desired resonant frequency within technological limits, engineers design crystal oscillator circuits around relatively few standard frequencies, such as 10 MHz, 20 MHz and 40 MHz. Using fre-

quency dividers, frequency multipliers and phase locked loop circuits, it is possible to synthesize any desired frequency from the reference frequency.

From the above description it is clear that although each sensor has identical hardware components, it is very likely that the resonant frequencies of the sensors will not be identical. Therefore a mechanism is needed to estimate the difference between the frequencies of the sensors and a reference clock rate. Since the clock rate of each sensor varies with time, this mechanism must be repeated frequently to have an accurate estimation of this clock drift. A typical model for the clock of a sensor as is explained in detail in the next chapter is described by the formula

$$t_B = \theta_s t + \theta_o \tag{2.1}$$

where

- t_B is the local time at a sensor node,

- t is the reference time,

- θ_s is the deviation of the sensor clock rate from the reference clock rate in parts per million (ppm) called *skew* and

- θ_o is the time difference from the reference clock at a particular time (usually it is measured at $t = 0$) called *offset* or *absolute jitter*.

The various time synchronization algorithms estimate the skew and the offset of each sensor and they use these estimations either to translate the timestamps or to correct their clocks. Two important factors that the design of each algorithm must take into account are the energy and time consumed to synchronize the nodes in a wireless sensor network where the energy resources are limited.

2.4   Ultra Wideband Communication

Ultra WideBand (UWB) technology employs narrow pulses of sub-nanosecond duration which in frequency domain corresponds to signals with relative bandwidth (BW) more than 20% (known as fractional bandwith) or to signals with absolute bandwidth more than $500MHz$. Traditional radio systems vary the power level, the frequency or phase to transmit data; UWB systems transmit data by generating signals with large BW at a specific time instant using pulse-position or pulse-time modulation. Although these signals occupy a large frequency spectrum, they do not interfere with other systems because their power is less than the ambient noise, namely 45 dBm/MHz as specified in the standard IEEE 802.15.4a [7]. UWB systems are ideal for high-precision location applications and range measurements because they demonstrate the following three characteristics:

1. high data rate connectivity

2. robustness to multipath fading

3. high transmission security and simple design

According to this IEEE standard, the UWB physical layer design and the process technology provides resistance to multipath fading while transmitting very low power. When UWB pulses are used for distance estimation, a ranging counter on the receiver determines the arrival time of the pulse. The time of flight of the UWB pulse can be calculated by a cross-correlation function of the received pulse and a transmitted reference pulse. Larger bandwidth is translated to more accurate distance estimates. If an actual physical ranging counter with an accuracy of sub-nanosecond was to be implemented, it would need to run at 64 GHz which is unlikely to be implemented on a low-cost battery powered device. Therefore, the ranging counter is an abstraction but its values can be generated using computational techniques such as linear interpolation of the cross-correlation function through a second-order approximation of the maximum likelihood estimate.

In [8], Gezici et al demonstrate the use of UWB technology in sensor networks for localization. The precise location estimation is facilitated by the high resolution of UWB technology in time. They show that UWB signaling in sensor networks achieves centimeter accuracy with low-power and low-cost implementation.

Savazzi et al in [1] present the WSN requirements for the physical and medium access control layer when UWB technology is used for oil and gas exploration. They argue that the throughput requirements per node are of the order of 150 $Kbps$ when three component accelerometers are deployed on each node. The localization error must be less than one meter to acquire a seismic image of good quality. In order to achieve this localization accuracy, the bandwidth of the transmitted UWB pulse must be at least 500 MHz. Furthermore as far as power consumption is concerned, they state that the WSN must be designed to work continuously for three to seven days.

Chapter 3

State of the Art

In this chapter, we present the related work on localization and time synchronization of sensors in sensor networks.

3.1 Related Work on Localization

Various localization methods for ad hoc sensor networks are presented in [9]. Guolin et al divide the localization methods in three categories: localization with beacons, with moving beacons and without beacons. In the first category, some nodes called beacon nodes are aware of their location while the remaining nodes use connectivity or ranging information to estimate their positions. For the case of moving beacons, the performance of the suggested algorithm is compared with the performance of DV-HOP and maximum likelihood estimation. In the latter category the algorithm is decentralized and the position is estimated by communication between neighboring nodes.

Zhuhong You et al in [10] suggest a range-free distributed method to self-localize the sensors in a wireless sensor network. A moving node equipped with six directional antennas gradually increases its sending signal strength, and includes in each transmitted message its current position, the sector of the antenna used, and the level of the transmitted power. The sensor nodes within range use this information to calculate their location. Simulations in a region of 100×100 m with 500 sensor nodes showed that the localization error depends on the trajectory of the moving beacon and also that it decreases when the number of directional antennas is increased.

Patwari et al in [11] present a survey on statistical models for AoA, TDoA, ToA and AoA locating methods along with their advantages and sources of errors. They also present a way to calculate the Cramer Rao Bound and apply it in numerical examples using ToA, RSS and AoA raging techniques. They also intro-

duce the Ultra Wideband pulses in localizing applications due to their increased accuracy. In addition they categorize the localization algorithms.

In [3], Kwon et al suggest dividing the localization problem in two parts. In the first part, a general layout of the network is derived and, in the second one, the nodes are assigned specific locations. The main idea is that the sender broadcasts a radio message followed by an acoustic signal. Receivers first detect the radio signal and wait for the acoustic tone. The time difference between those two receptions yields distance information between sender and receiver which is then provided as input for the multilateration scheme. However, there are a number of error sources that have to be taken into consideration, such as the time synchronization, the acoustic sensing and actuation delays, signal attenuation, environmental noise, the multipath effects and so on. To mitigate these sources of errors, they propose the following:

- In a noisy or grassy outdoor environment, the acoustic signal attenuates fast with distance having a limited range of 10 m. Hence, in order to increase the Signal-to-noise ratio (SNR) and therefore the range of the acoustic signal the transmitted tone is amplified by inexpensive hardware yielding a range of more than 35 m. Maroti et al in [12] suggest that in case of ambient noise, radio interferometry with RF signals should be used instead of RF and acoustic signals.

- Since the clocks of the sensor nodes are not perfectly synchronized, the exact time of flight of the RF and acoustic signals cannot be extracted and thus the distance information is inaccurate. Therefore, a synchronization algorithm must be applied to synchronize the nodes. When the same RF signal used for TDoA is also used for synchronization the error among the clocks is in the order of 50 μs which is translated to a positioning error of 15 cm.

- Environmental conditions can cause erroneous signal detection. Thus, two ways of eliminating this source of error are to transmit a sequence of tones instead of one single tone, and to decide that there is a signal detection if the sum of the samples is above a threshold value.

- The effects of echoes, low SNR, and environmental factors are mitigated by encoding a pattern in the acoustic signal and by adding small random delays between the elements of pattern.

- During the transmission and the reception processes there are non deterministic delays, such as message transmission, speaker activation, detector delay, which can be converted to deterministic by introducing a constant time interval between the transmission of the RF message and the corresponding acoustic tone.

- The last but not least source of errors comes from individual erroneous measurements due to faulty hardware or noise. To overcome these errors each pair of nodes performs multiple distance measurements and filters the results by taking either the median or the mode value. Therefore, inconsistency bidirectional checks discard incorrect measurements.

After acquiring the distance information, an extension of the Least-Square Scaling (LSS) algorithm is presented which uses radio-interferometry and nodes' acoustic measurements under soft constraints related to the topology of the network. The distances are estimated based on all available measurements and do not require measurements between all possible pairs in the network. Since for the localization process, each sensor needs to communicate with three anchor nodes, the number of anchor nodes is independent of the size of the network. On the other hand, multidimensional scaling is a centralized algorithm which needs the set of all possible distance nodes in the network for sensor localization [13].

Girod et al in [14] present a case study where wideband audible acoustic signal was used for sensor localization; the transmitter advertises that it will emit an acoustic signal with a given code at a particular time referenced to its own time scale and therefore the receiver knows when to start listening. After detecting the received signal, the time of flight is derived which is then translated into distance. This approach was chosen for two reasons; firstly, the wavelength of audible signals spans from a meter to a few centimeters increasing the resilience of the signal to scattering. Secondly, different emitters can select orthogonal codes that are detected even during collisions. The outliers are rejected based on an adaptive threshold equal to three times the median of the set of points not yet rejected. The Reference Broadcast Synchronization (RBS) algorithm achieved an error of 10 μsec in synchronization which is translated to a localization error of $11, 5 \ cm$.

In [15], a range-independent, decentralized approach is described for self localization when the anchor nodes have antenna sectors. Sensor nodes define a positioning area based on beacons from anchors. The novel idea introduced here is a grid score table which indicates the overlapping possible locations for each node; the points of this grid table have coordinates and are marked when the beacon from a sector of an anchor is within communication range. In that case, the values of the marked grid points are incremented by one. Thus the nodes do not need to perform any kind of computations or AoA measurements; they assume that they are at the centroid of the area defined by the grid points with the maximum values. In terms of performance of the SerLoc algorithm, it achieves a localization error of 0,5 times the sensor communication range when four anchors with three-sectored antennas are used. However, the error decreases when the density of anchors increase.

Patwari et al in [2] combine two range-based mechanisms for relative sensor

localization, namely RSSI and ToA. Cramer Rao Lower Bounds and maximum likelihood estimators are derived for ToA and RSSI assuming Gaussian and log-normal models respectively. For the RSSI measurements a free space path loss is assumed while range information is derived from ToA through inquiry-response message exchange among pairs of anchors and sensor nodes. The achieved accuracy is 1m when applying the algorithm in a network of 44 devices in an indoor area 14×13 m; the four devices in the corners are chosen as anchor nodes and the rest as devices to be localized.

In [16] Kozick and Sadler suggest a mobile Access Point (AP) which emits radio and acoustic sound to localize the sensors while taking advantage of the Doppler shift. CRLB is computed for the accuracy of the localization algorithm and the impact of atmospheric turbulence is taken into account. Since there is no need for communication between the AP and the sensor nodes, there is no need for time synchronization among nodes; the radio messages of the AP contain timing, AP location and motion and information about the acoustic signal. The sensor nodes estimate at least three quantities to localize themselves: Doppler shift, time delay and angle of arrival. Simulations show that in sunny conditions the accuracy is several meters while in cloudy weather the accuracy can reach up to 3 cm.

3.2   Related Work on Time Synchronization

In [1] and [17], a description of the wireless sensor network for oil and gas exploration is provided. The new features and requirements of such networks are explained for the physical and the MAC layer and a mixture of new technologies are proposed for a large-scale, real-time, synchronous, and spatially dense wireless geophone network.

For applications such as land seismic exploration, time synchronization between the sensor nodes is crucial. There are two categories of time synchronization algorithms; unidirectional and bidirectional depending on whether there is a handshake between receivers or not. In the first category, we see algorithms like Reference Broadcast Synchronization (RBS), Flooding Time Synchronization Protocol (FTSP) and in the latter one we meet Hierarchical Reference Time Synchronization (HRTS) and Individual Time Request (ITR) algorithms.

In [18], a self-correcting time synchronization algorithm is proposed according to which no packets are exchanged among the sensor nodes. A sequence of successive reference broadcast packets are sent to drive the non synchronized clocks on the receiver nodes to gradually approach and eventually be locked to the reference clock of the beacon node. The accuracy of this algorithm is arbitrarily defined as it depends on the synchronization period. The authors performed experiments on a small network of 10 Mica2 motes.

Solis, Borkar and Kumar in [19] suggest a distributed time synchronization algorithm which achieves an accuracy of below $2\mu s$ when implemented on MICA2 motes. They used the linear form $T_i = a_i t + O_i$ to model the clock drift of each node i where a_i is the skew, O_i is the offset with respect to a reference node and t is the real time. In order to estimate skew and offset parameters, the nodes exchange timestamped packets and compute the recursive Least Square (LS) error estimators. The topology is not necessarily known a priory and furthermore multihop time synchronization is possible under the global constraint that in a closed loop the sum of all nodes' offset is zero and the product of all the nodes' skew is one.

In a later work [20], Kumar and Giridhar adopt the same method for estimating clock differences between nodes. They view the network as a graph, and use the same global constraint for the offset and the skew in case of closed loops. They compare the LS error variance of the drift parameters when the topology is a tree, a clique and a planar network. By relating the sensor network to an electrical circuit, and the variance of the drift LS error to the electrical resistance, they provide an intuitive explanation of how the drift parameters are affected when topology changes. Thus, the error variance in a tree is proportional to the diameter of the graph, just as the resistance is additive in a similar structured circuit. In a clique, the LS error decreases inversely with the size of the network while in a planar network the maximum resistance distance is $O(1)$. In addition, [20] points out that in order to obtain a bounded estimation error, the packet exchange must be repeated frequently because the drift parameters vary with time and also because more bilateral estimates yield better estimates.

Graham and Kumar in [21] introduce the Control Time Protocol (CTP) to provide synchronization in control applications with random delay and jitter. The clock model is assumed linear as before, and in order to estimate the skew and the offset, timestamped packets are exchanged between pairs of nodes. They assume that the propagation delay between a pair of nodes is symmetric and they prove that the offset is undetermined without this assumption. They analyze the impact of timestamping on system reliability and they state that the precision of offset estimation within upper and lower bounds depends on delay symmetry and on knowledge of minimum delays. Thus being able to identify erroneous measurements (outliers) can improve the precision. In terms of estimators, Graham and Kumar explain why the windowed linear least squares approach has more advantages than the recursive or linear least squares methods. In order to further increase robustness against outliers, the exchange rate of timestamped packets may increase when an outlier is detected.

The work in [22] shows that in a wireless sensor network it is feasible to simultaneously localize and synchronize all the nodes. Zheng and Wu consider two cases: in the first one the anchor nodes are all perfectly time synchronized

with the reference time, and in the second one the anchors are not synchronized among them. In both scenarios, the sensor nodes are self-localized and time-synchronized with the anchors using the two way message exchange mechanism as described in Section 4.2.

The joint approach descibed in [22] time-synchronizes each node separately with all the anchor nodes. Comparing this algorithm with other time synchronizing algorithms [23], we identify some basic differences:

- While FTSP [24] synchronizes all recipients with the root with a single radio message, the joint approach synchronizes one node at a time. However, FTSP does not compensate for the propagation delay. For both protocols multihop synchronization is possible in a mesh network topology.

- RBS [25] uses receiver-to-receiver synchronization on a tree topology and it neglects the propagation delay between root and recipients. The time difference of the reception of the same message by two or more recipients is used to calculate the time offset (ignoring the time skew) of their clocks.

- TPSN [26] is a sender-to-receiver time synchronization protocol that needs a tree topology. The two way communication is initiated by sending a synchronization pulse by the node at level 1 to the root of the tree. The time synchronization pulse is propagated along the tree in such a way that all level i nodes are synchronized to the level $i-1$ nodes. The clock drift and offset are calculated by the same four timestamps that are described in the joint approach.

- Glossy [27] synchronizes nodes without additional messages during flooding. The synchronization accuracy reaches $0.5 \mu sec$, which is the basic requirement to achieve constructive interference. The difference from the joint approach is that it uses one way communication between nodes; a root initiates the flooding and the recipients retransmit the same message upon reception. The neighboring nodes retransmit the same message at the same time. The key of success here is that all the non-deterministic parts such as radio processing delay and software delay of the time interval from transmission to reception have been investigated and their uncertainty has been taken into account. A rough comparison between Glossy and the joint approach shows that the number of messages M sent in Glossy per node is less than $A \times M \times K$ for the joint approach, where K is the non synchronized neighbors, A is the number of neighbors and M the number of rounds. Furthermore, the convergence time of Glossy, $T_{slot} \times Diameter$, is shorter than the convergence of joint approach where one node is synchronized at a time. However, the joint approach solves the problem of localization and time synchronization at the same time, with a time synchronization accuracy of the order of sub-nanosecond.

Chapter 4

Joint Localization and Synchronization

In this chapter, we present an algorithm for localizing and time synchronizing sensor nodes in a wireless sensor network. At first, we state the assumptions considered. Then, we present a model for jointly estimating the location in a two dimensional plane and the timing parameters of clock signal of a node, and show simulation results for various scenarios. Furthermore, we derive the minimum error variance for the location and timing parameters for any estimation scheme for the proposed algorithm and use it to assess the simulation results and the theoretical analysis.

4.1 Assumptions

We propose an algorithm to localize and time synchronize sensor nodes in a network, where some nodes called *anchors* share the same global time. Moreover, during the localization procedure, the sensor nodes determine their two dimensional coordinates, thus we assume that there are at least three anchors within their communication range. In our model, we assume that all the transmitted messages are received.

4.2 Joint Localization and Time Synchronization Algorithm Description

The proposed joint localization and time synchronization algorithm uses two way message communication for ranging and estimation of the timing parameters skew θ_s and offset θ_o. A sensor node with unknown location coordinates x

and y is self-localized by exchanging M rounds of messages with each anchor l. These messages include as a preamble a known UWB pulse. The sensor node running this algorithm determines the unknown parameters x, y, θ_s, θ_o by using the propagation delay and a time translation model. During a bidirectional communication between a sensor node B and an anchor node A, we observe the following events:

- at local time T, sensor node B transmits a known pulse

- anchor node A detects and receives the pulse at local time R

- node A transmits a known pulse at local time $\overline{T}$, and includes the timestamps $\overline{T}$ and R in the message

- node B detects the known pulse at local time $\overline{R}$ and receives the message.

In the end of this bidirectional communication, node B knows all four timestamps in order to deduce its distance from anchor node A and its time skew and offset from the reference clock. This message exchange is repeated for M rounds with each anchor in the vicinity of node B and is described by the following system of equations

$$T_{lm} = \theta_s(R_{lm} - t_1 - n_{lm}) + \theta_o \tag{4.1}$$

and

$$\overline{R}_{lm} = \theta_s(\overline{T}_{lm} + t_1 + n_{lm}) + \theta_o, \tag{4.2}$$

where we assume that there are $3 \leq l \leq L$ anchors in the neighborhood of the sensor node B, and that there are $m = M$ rounds of Two Way Ranging (TWR) communications between the node B and each anchor node. With t_1, we denote the propagation delay between the two nodes and $n_{lm}, \overline{n}_{lm}$ are the ToA detection errors, which are assumed to be Gaussian distributed. Since the measurements are noisy, the system of Equations 4.1 and 4.2 will not have a unique solution, thus we seek the solution with the minimum error.

4.3   Joint Localization and Time Synchronization Model Analysis

The TWR communication between sensor node B and anchor node A, which was described in previous section, is depicted in Figure 4.3. The horizontal line represents the time as measured on the anchor node, which is equal to the reference time, and the inclined line represents the time as measured on node B. The message sent by the anchor on each round contains the timestamps of
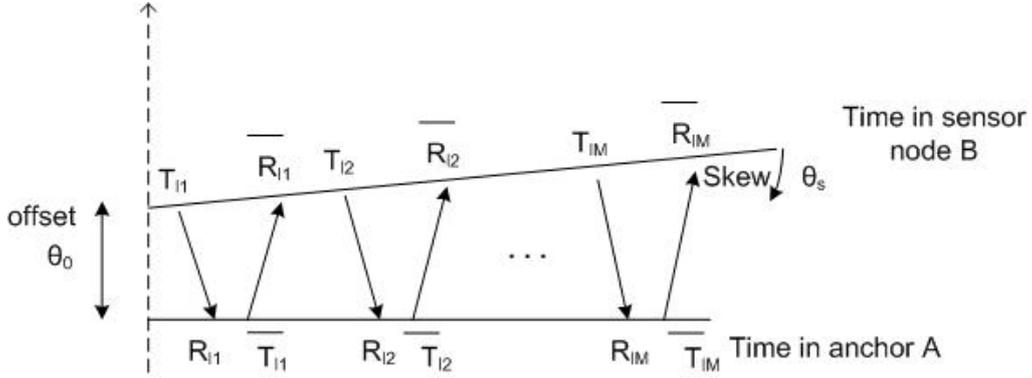
Figure 4.1: Two way message exchange between anchor A and sensor node B

reception R and transmission $\overline{T}$ by the anchor node. It is important to note here that the timestamps are measured by two different time clocks, because these two nodes are not time synchronized. The translation between these clocks is described by the formula

$$t_B = \theta_s t + \theta_o, \tag{4.3}$$

where θ_s is the frequency skew of node B's clock and θ_o is the relative offset. The reception and transmission of the two messages per round is expressed by Equations 4.1 and 4.2.

The propagation delay t_1 for a signal that travels with speed c is described by formula

$$t_1 = \frac{||\boldsymbol{x} - \boldsymbol{a}_l||}{c}, \tag{4.4}$$

where $\boldsymbol{x} = [x,\ y]^T$ are the location coordinates of the sensor node to be localized and $\boldsymbol{a}_l = [a_{lx},\ a_{ly}]^T$ are the known location coordinates of the anchor node l.

The goal of this analysis is to build a model for estimating the location and timing parameters of each sensor node using the joint localization and time synchronization algorithm. From Estimation Theory we know that the Maximum Likelihood (ML) estimator is the optimal estimator. Since there is no closed form solution for the ML estimator, we compute the Least Squared Error (LSE) estimator and we compare it with the minimum error variance of this algorithm bounded by the Cramer Rao Lower Bound (CRLB).

The LSE estimation of the unknown location coordinates x and y, and the timing parameters θ_s and θ_o of each sensor is derived in a closed form equation following two steps: first there is a linearization step providing a rough solution for the unknown vector $\boldsymbol{\phi} = [x,\ y,\ \theta_s,\ \theta_o]^T$, which is used in the second step to

improve the accuracy of the estimators. Dividing Equations 4.1 and 4.2 by θ_s and introducing two variables $\theta_1 = 1/\theta_s$ and $\theta_2 = \theta_o\theta_s$, we re-parametrize the system

$$- R_{lm} + t_1 = -T_{lm}\theta_1 + \theta_2 - n_{lm} \tag{4.5}$$

$$\overline{T}_{lm} + t_1 = \overline{R}_{lm}\theta_1 - \theta_2 - n_{lm}. \tag{4.6}$$

Squaring and rearranging the above two equations we get

$$2\left[\frac{1}{c^2}x^T a_l - T_{lm}R_{lm}\theta_1 + R_{lm}\theta_2 + \frac{1}{2}T_{lm}^2\theta_1^2 - T_{lm}\theta_1\theta_2 + \frac{1}{2}\left(\theta_2^2 - \frac{1}{c^2}||x||^2\right)\right]$$
$$= \frac{1}{c^2}||a_l||^2 - R_{lm}^2 + e_{lm} \tag{4.7}$$

$$2\left[\frac{1}{c^2}x^T a_l - \overline{T}_{lm}\overline{R}_{lm}\theta_1 + \overline{T}_{lm}\theta_2 + \frac{1}{2}\overline{R}_{lm}^2\theta_1^2 - \overline{R}_{lm}\theta_1\theta_2 + \frac{1}{2}\left(\theta_2^2 - \frac{1}{c^2}||x||^2\right)\right]$$
$$= \frac{1}{c^2}||a_l||^2 - \overline{T}_{lm}^2 + \overline{e}_{lm}, \tag{4.8}$$

where all the error terms are included in the following equations

$$e_{lm} = 2(-T_{lm}\theta_1 + \theta_2 + R_{lm})n_{lm} - n_{lm}^2 \tag{4.9}$$
$$\overline{e}_{lm} = 2(\overline{R}_{lm}\theta_1 - \theta_2 - \overline{T}_{lm})\overline{n}_{lm} - \overline{n}_{lm}^2. \tag{4.10}$$

Since the TWR communication between a sensor node and each anchor is repeated M times, we use vectors and matrices to represent the $2ML$ equations of the algorithm. In the above non-linear system of equations, the unknown vector is $\boldsymbol{\omega} = [\boldsymbol{x}, \theta_1, \theta_2]^T$. Furthermore, let us introduce the non linear variables $\xi_5 = \frac{1}{2}\theta_1^2$, $\xi_6 = (1/2)(\theta_2^2 - ||x||^2/c^2)$, and $\xi_7 = \theta_1\theta_2$ to represent linearly the M communication rounds between the sensor node B and an anchor A_l as follows

$$\boldsymbol{A}_l\boldsymbol{\xi} = \boldsymbol{b}_l + \boldsymbol{e}_l \tag{4.11}$$

where

$$\boldsymbol{A}_l = 2\begin{bmatrix} \frac{\boldsymbol{a}_l}{c^2} & -T_{l1}R_{l1} & R_{l1} & T_{l1}^2 & 1 & -T_{l1} \\ \frac{\boldsymbol{a}_l}{c^2} & -\overline{T}_{l1}\overline{R}_{l1} & \overline{T}_{l1} & \overline{R}_{l1}^2 & 1 & -\overline{R}_{l1} \\ \vdots & & & & & \\ \frac{\boldsymbol{a}_l}{c^2} & -T_{lM}R_{lM} & R_{lM} & T_{lM}^2 & 1 & -T_{lM} \\ \frac{\boldsymbol{a}_l}{c^2} & -\overline{T}_{lM}\overline{R}_{lM} & \overline{T}_{lM} & \overline{R}_{lM}^2 & 1 & -\overline{R}_{lM} \end{bmatrix}_{2M\times 7}$$

$$\boldsymbol{\xi} = \begin{bmatrix} \boldsymbol{x}^T \\ \theta_1 \\ \theta_2 \\ \xi_5 \\ \xi_6 \\ \xi_7 \end{bmatrix}_{7 \times 1}$$

$$\boldsymbol{b}_l = \begin{bmatrix} \frac{\|a_l\|^2}{c^2} - R_{l1}^2 \\ \frac{\|a_l\|^2}{c^2} - \overline{T}_{l1}^2 \\ \vdots \\ \frac{\|a_l\|^2}{c^2} - R_{lM}^2 \\ \frac{\|a_l\|^2}{c^2} - \overline{T}_{lM}^2 \end{bmatrix}_{2M \times 1}$$

$$\boldsymbol{e}_l = [e_{l1}, \overline{e}_{l1}, ..., e_{lM}, \overline{e}_{lM}]_{2M \times 1}^T.$$

Since the sensor node B exchanges messages with all the anchors in its vicinity, we collect all the equations for $l = 1, 2, ..., L$ anchors together and we have

$$\boldsymbol{A\xi} = \boldsymbol{b} + \boldsymbol{e}, \tag{4.12}$$

where $\boldsymbol{A} = [\boldsymbol{A}_1^T, ..., \boldsymbol{A}_L^T]^T$, $\boldsymbol{b} = [\boldsymbol{b_1}^T, ..., \boldsymbol{b}_L^T]^T$, and $\boldsymbol{e} = [\boldsymbol{e}_1^T, ..., \boldsymbol{e}_L^T]^T$.

The well known Least Squares (LS) solution of Equation 4.12 is then

$$\boldsymbol{\xi}_{LS} = (\boldsymbol{A}^T\boldsymbol{A})^{-1}\boldsymbol{A}^T\boldsymbol{b}. \tag{4.13}$$

This solution can only be a rough estimation for the following two reasons:

- all the components of the error term $\boldsymbol{e}$ are assumed to have the same variances

- there are no constraints applied between the elements of the unknown vector, leading to an inconsistent result.

Consequently in the refinement step these two statements are taken into consideration. The rough estimation of Equation 4.13 is used to approximate the covariance matrix $\boldsymbol{C}_e$ of the noise vector $\boldsymbol{e}$ and thus to improve the accuracy of the solution. This way we assign a different weight on the error of each measurement. The solution, which is then given by the Weighted Least Squares (WLS) approach, is

$$\boldsymbol{\xi}_{WLS} = (\boldsymbol{A}^T\boldsymbol{C}_e^{-1}\boldsymbol{A})^{-1}\boldsymbol{A}^T\boldsymbol{C}_e^{-1}\boldsymbol{b}, \tag{4.14}$$

where the noise vector $\boldsymbol{e}$ is simplified if we ignore the squared terms n_{lm}^2 and $\overline{n}_{lm}^2$ resulting to the form

$$\boldsymbol{e} = 2\boldsymbol{T}_e \odot \boldsymbol{n} \tag{4.15}$$

and its covariance matrix

$$\boldsymbol{C}_e \approx 4(\boldsymbol{T}_e \boldsymbol{T}_e^T) \odot \boldsymbol{C}_{nn} \tag{4.16}$$

with

$$\boldsymbol{T}_e = \theta_1 \boldsymbol{T}_b - \theta_2 \boldsymbol{E}_{2M \times 1} - \boldsymbol{T}_a \tag{4.17}$$
$$\boldsymbol{T}_{al} = [-R_{l1}, \overline{T}_{l1}, ..., -R_{lM}, \overline{T}_{lM}]^T \tag{4.18}$$
$$\boldsymbol{T}_a = [\boldsymbol{T}_{a1}^T, ..., \boldsymbol{T}_{aL}^T]^T \tag{4.19}$$
$$\boldsymbol{T}_{bl} = [-T_{l1}, \overline{R}_{l1}, ..., -T_{lM}, \overline{R}_{lM}]^T \tag{4.20}$$
$$\boldsymbol{T}_b = [\boldsymbol{T}_{b1}^T, ..., \boldsymbol{T}_{bL}^T]^T. \tag{4.21}$$

As a last step, we exploit the relation between the elements of $\boldsymbol{\xi}$. Matrix $\boldsymbol{G}_1$ is used to acquire the unknown variables x, y, θ_1, θ_2 from $\boldsymbol{\xi}$

$$\boldsymbol{G}_1 \boldsymbol{\omega} = \boldsymbol{\xi}_{WLS} + \boldsymbol{n}_{WLS}, \tag{4.22}$$

where $\boldsymbol{n}_{WLS}$ is the estimation error in $\boldsymbol{\xi}_{WLS}$, $\boldsymbol{\omega} = [\boldsymbol{x}^T, \boldsymbol{\theta}^T]^T$, $\boldsymbol{G}_1 = [\boldsymbol{I}_4, \widetilde{\boldsymbol{G}}_1]^T$ and

$$\widetilde{\boldsymbol{G}}_1 = \frac{1}{2} \begin{bmatrix} 0 & 0 & \theta_1 & 0 \\ -\frac{x}{c^2} & -\frac{y}{c^2} & 0 & \theta_2 \\ 0 & 0 & \theta_2 & \theta_1 \end{bmatrix}.$$

To compute $\boldsymbol{\omega}$, we use the covariance matrix $\boldsymbol{C}_{WLS}$ of $\boldsymbol{\xi}_{WLS}$ which is equal to $\boldsymbol{C}_{WLS} = (\boldsymbol{A}^T \boldsymbol{C}_e^{-1} \boldsymbol{A})^{-1}$. Thus, when applying the joint localization and time synchronization algorithm, the estimation of the location and the time parameters is

$$\boldsymbol{\omega}_{CWLS} = (\boldsymbol{G}_1^T \boldsymbol{C}_{WLS}^{-1} \boldsymbol{G}_1)^{-1} \boldsymbol{G}_1 \boldsymbol{C}_{WLS}^{-1} \boldsymbol{\xi}_{WLS} \tag{4.23}$$
$$= (\boldsymbol{G}_1^T \boldsymbol{A}^T \boldsymbol{C}_e^{-1} \boldsymbol{A} \boldsymbol{G}_1)^{-1} \boldsymbol{G}_1^T \boldsymbol{A}^T \boldsymbol{C}_e^{-1} \boldsymbol{b}. \tag{4.24}$$

The vector $\boldsymbol{\omega}_{CWLS}$ contains the estimations of x, y, θ_1 and θ_2. Therefore, we need to derive θ_s and θ_o from them.

4.4 Cramer-Rao Lower Bound

The Cramer Rao Lower Bound (CRLB) sets a lower bound on the error variance of any unbiased estimator. This is useful in several ways. Namely, the CRLB provides a benchmark against which we can compare the performance of any unbiased estimator. Hence, CRLB is useful in feasibility studies like this thesis.

We consider correlated residuals $\boldsymbol{n}$ that are Gaussian distributed with zero mean and known covariance $\boldsymbol{C}_{nn}$. Our goal is to find the CRLB of the covariance matrix of any unbiased estimator.

From Equations 4.1, 4.2 and 4.17, we rewrite the system for ease of computation in the form

$$\boldsymbol{T}(\boldsymbol{x}) = \boldsymbol{R\theta} - \boldsymbol{n}. \tag{4.25}$$

Then we compute the CRLB in two steps: first, when vector $\boldsymbol{\theta} = [1,0]^T$, and then, when the vector $\boldsymbol{x}$ is known. Therefore in the following two sections, we compute the CRLB in these two cases, and afterwards we combine them to calculate the CRLB for the joint localization and time synchronization algorithm.

4.4.1 Cramer Rao Lower Bound of Location Estimate with Perfect Timing

In this section, we assume that $\theta_s = 1$ and $\theta_o = 0$, yielding $\theta_1 = 1$ and $\theta_2 = 0$, thus, we modify the system of Equations 4.1 and 4.2 in the following way

$$\boldsymbol{n} = \boldsymbol{R\theta} - \boldsymbol{T}(\boldsymbol{x}). \tag{4.26}$$

When the Probability Density Function (PDF) is viewed as a function of the unknown parameter with $\boldsymbol{x}$ fixed it is denoted as likelihood function. Since we have assumed Gaussian distributed residuals, the joint likelihood function conditioned on the unknown vector $\boldsymbol{x}$ is given by p_1

$$p_1(\boldsymbol{R\theta}|\boldsymbol{x}) = \frac{1}{(2\pi \cdot |\boldsymbol{C}_{nn}|)^{ML}} \exp\left[-\frac{1}{2}(\boldsymbol{R\theta} - \boldsymbol{T}(\boldsymbol{x}))^T \boldsymbol{C}_{nn}^{-1}(\boldsymbol{R\theta} - \boldsymbol{T}(\boldsymbol{x}))\right]. \tag{4.27}$$

The more sharp the likelihood function is, the more accurately we can estimate the unknown parameters. The sharpness or *curvature* of this function is measured by the negative of the second derivative of the logarithm of the likelihood function. If the second derivative depends on the unknown parameters the curvature is better represented by the average of the second derivative. The connection between the curvature and the CRLB is easily obtained by stating the fact that the curvature is the Fisher information ($\boldsymbol{I}(\boldsymbol{x})$) function (or matrix in our case since we have vectors), and the variance of the estimators should satisfy the Equation 4.28

$$\sigma^2(\boldsymbol{x}) = \boldsymbol{C}_{nn} \geq [\boldsymbol{I}^{-1}(\boldsymbol{x})]. \tag{4.28}$$

Thus we need to compute the logarithm of the likelihood function as shown below

$$\begin{aligned}
\log p_1(\boldsymbol{R\theta}|\boldsymbol{x}) &= -\frac{1}{2}(\boldsymbol{R\theta} - \boldsymbol{T}(\boldsymbol{x}))^T \boldsymbol{C}_{nn}^{-1}(\boldsymbol{R\theta} - \boldsymbol{T}(\boldsymbol{x})) - ML\log(2\pi|\boldsymbol{C}_{nn}|) \\
&= -\frac{1}{2}\boldsymbol{n}^T \boldsymbol{C}_{nn}^{-1}\boldsymbol{n} - ML\log(2\pi|\boldsymbol{C}_{nn}|).
\end{aligned} \tag{4.29}$$

The next step in calculating the curvature is to take the first and second order derivatives of Equation 4.29 with respect to $\boldsymbol{x}$. Note that due to the way that $\boldsymbol{n}$ depends on x and y the derivatives with respect x and y are similar. Thus we present only the derivatives with respect to x

$$\begin{aligned}
\frac{\partial \log p_1}{\partial x} &= -\frac{1}{2}\left[\boldsymbol{n}^T \boldsymbol{C}_{nn}^{-1}\frac{\partial \boldsymbol{n}}{\partial x} + \frac{\partial \boldsymbol{n}}{\partial x}\boldsymbol{C}_{nn}^{-1}\boldsymbol{n}^T\right] \\
&= -\boldsymbol{n}^T \boldsymbol{C}_{nn}^{-1}\frac{\partial \boldsymbol{n}}{\partial x}
\end{aligned} \tag{4.30}$$

$$\frac{\partial^2 \log p_1}{\partial x \partial y} = -\boldsymbol{n}^T \boldsymbol{C}_{nn}^{-1}\frac{\partial^2 \boldsymbol{n}}{\partial x \partial y} - \frac{\partial \boldsymbol{n}^T}{\partial y}\boldsymbol{C}_{nn}^{-1}\frac{\partial \boldsymbol{n}}{\partial x} \tag{4.31}$$

$$\frac{\partial^2 \log p_1}{\partial x^2} = -\left[\boldsymbol{n}^T \boldsymbol{C}_{nn}^{-1}\frac{\partial^2 \boldsymbol{n}}{\partial x^2} + \frac{\partial \boldsymbol{n}^T}{\partial x}\boldsymbol{C}_{nn}^{-1}\frac{\partial \boldsymbol{n}}{\partial x}\right]. \tag{4.32}$$

The Fisher Information matrix $\boldsymbol{I}(\boldsymbol{x})$ is constructed by taking the expectations of the above derivatives with respect to the likelihood function p_1

$$\boldsymbol{I}(\boldsymbol{x}) = \begin{bmatrix} -[E(\frac{\partial^2 (\log p_1)}{\partial x^2})] & -[E(\frac{\partial^2 (\log p_1)}{\partial x \partial y})] \\ -[E(\frac{\partial^2 (\log p_1)}{\partial y \partial x})] & -[E(\frac{\partial^2 (\log p_1)}{\partial y^2})] \end{bmatrix}.$$

Thus filling in the table $\boldsymbol{I}(\boldsymbol{x})$, we get

$$\boldsymbol{I}(\boldsymbol{x}) = \begin{bmatrix} [\frac{\partial \boldsymbol{n}^T}{\partial x}\boldsymbol{C}_{nn}^{-1}\frac{\partial \boldsymbol{n}}{\partial x}] & [\frac{\partial \boldsymbol{n}^T}{\partial y}\boldsymbol{C}_{nn}^{-1}\frac{\partial \boldsymbol{n}}{\partial x}] \\ \left[\frac{\partial \boldsymbol{n}^T}{\partial x}\boldsymbol{C}_{nn}^{-1}\frac{\partial \boldsymbol{n}}{\partial y}\right] & [\frac{\partial \boldsymbol{n}^T}{\partial y}\boldsymbol{C}_{nn}^{-1}\frac{\partial \boldsymbol{n}}{\partial y}] \end{bmatrix}.$$

In order to calculate the CRLB, we recall the equations describing the terms involved in the Fisher Information table

$$\boldsymbol{n} = \boldsymbol{R\theta} - \boldsymbol{T}(\boldsymbol{x}),$$
$$\boldsymbol{T}(\boldsymbol{x}) = \boldsymbol{T}_a + t_1, \tag{4.33}$$
$$t_1 = \frac{||\boldsymbol{x} - \boldsymbol{a}_l||}{c}.$$

Thus the first derivative of $\boldsymbol{n}$ with respect to the unknown vector $\boldsymbol{x}$ is calculated below

$$\begin{aligned}
\frac{\partial \boldsymbol{n}}{\partial x} &= \frac{\partial (\boldsymbol{R\theta} - \boldsymbol{T}(\boldsymbol{x}))}{\partial \boldsymbol{x}} = \frac{\partial \boldsymbol{R\theta}}{\partial \boldsymbol{x}} - \frac{\partial \boldsymbol{T}(\boldsymbol{x})}{\partial \boldsymbol{x}} \\
&= -\frac{\partial (\boldsymbol{T}_a + t_1)}{\partial x} = -\frac{\partial \boldsymbol{T}_a}{\partial x} - \frac{\partial t_1}{\partial x} \\
&= -\frac{\partial ||\boldsymbol{x} - \boldsymbol{a}_l||}{c\partial x} = -\frac{(\boldsymbol{x} - \boldsymbol{a}_l)^T}{c||\boldsymbol{x} - \boldsymbol{a}_l||} \\
&= -\frac{(\boldsymbol{x} - \boldsymbol{a}_l)^T}{c^2 t_1},
\end{aligned}$$

where $\boldsymbol{x}$ is the coordinate vector of the sensor to be localized and $\boldsymbol{a}_l$ is the coordinate vector of the anchor l. Since the derivative must be calculated for all the L anchors in the vicinity of this sensor node, we present the results in the following matrix form:

$$\boldsymbol{CRLB}(\boldsymbol{x}) = \left[\frac{\partial \boldsymbol{n}^T}{\partial \boldsymbol{x}} \boldsymbol{C}_{nn}^{-1} \frac{\partial \boldsymbol{n}}{\partial \boldsymbol{x}} \right]^{-1}, \tag{4.34}$$

where

$$\begin{aligned}
\frac{\partial \boldsymbol{n}}{\partial \boldsymbol{x}} &= \boldsymbol{B}_1 \\
\boldsymbol{B}_1 &= -\frac{1}{c^2} \left[\frac{(\boldsymbol{x} - \boldsymbol{a}_1)^T}{t_1}, \dots, \frac{(\boldsymbol{x} - \boldsymbol{a}_L)^T}{t_1} \right]^T \otimes \boldsymbol{1}_{2M \times 1}.
\end{aligned} \tag{4.35}$$

Therefore, $\boldsymbol{CRLB}(\boldsymbol{x}) = \left(\boldsymbol{B}_1^T \boldsymbol{C}_{nn}^{-1} \boldsymbol{B}_1 \right)^{-1}$ is a 2×2 matrix with the diagonal elements being the variances of the location estimates x, y.

4.4.2 Cramer Rao Lower Bound of Timing Estimates with Known Locations

In this section, we consider that all the sensor nodes know their location but they are not synchronized. We derive the CRLB for the time parameters following the same steps as in the case of perfect time synchronization.

The residuals n_{lm} and $\bar{n}_l m$ are Gaussian distributed with zero mean and covariance matrix $\boldsymbol{C}_{nn}$, and are given by the following equations

$$\boldsymbol{n} = \boldsymbol{R\theta} - \boldsymbol{b},$$
$$\boldsymbol{\theta} = [\theta_s, \; \theta_o]^T, \tag{4.36}$$
$$\boldsymbol{R} = \boldsymbol{T}_b - \boldsymbol{E}_{2LM \times 1}.$$

Following exactly the same analysis as before, we have the same joint likelihood function p_1 conditioned on the unknown vector $\boldsymbol{\theta}$

$$p_1(\boldsymbol{b}|\boldsymbol{\theta}) = \frac{1}{(2\pi \cdot |\boldsymbol{C}_{nn}|)^{ML}} \cdot \exp\left[-\frac{1}{2}(\boldsymbol{R\theta} - \boldsymbol{b})^T \boldsymbol{C}_{nn}^{-1}(\boldsymbol{R\theta} - \boldsymbol{b}))\right], \tag{4.37}$$

which we use to calculate the curvature as before. Thus, we compute the logarithm of p_1, then the first and second derivative of the logarithm and finally we take expectations as shown below

- Logarithm of joint conditional PDF

$$\begin{aligned}
\log p_1(\boldsymbol{b}|\boldsymbol{\theta}) &= -\frac{1}{2}(\boldsymbol{R\theta} - \boldsymbol{b}))^T \boldsymbol{C}_{nn}^{-1}(\boldsymbol{R\theta} - \boldsymbol{b}) - ML\log(2\pi|\boldsymbol{C}_{nn}|) \\
&= -\frac{1}{2}\boldsymbol{n}^T \boldsymbol{C}_{nn}^{-1}\boldsymbol{n} - ML\log(2\pi|\boldsymbol{C}_{nn}|)
\end{aligned} \tag{4.38}$$

- First derivative of $\log p_1$

$$\begin{aligned}
\frac{\partial \log p_1}{\partial \theta_i} &= -\frac{1}{2}\left[\boldsymbol{n}^T \boldsymbol{C}_{nn}^{-1}\frac{\partial \boldsymbol{n}}{\partial \theta_i} + \frac{\partial \boldsymbol{n}}{\partial \theta_i}\boldsymbol{C}_{nn}^{-1}\boldsymbol{n}^T\right] \\
&= -\boldsymbol{n}^T \boldsymbol{C}_{nn}^{-1}\frac{\partial \boldsymbol{n}}{\partial \theta_i}
\end{aligned} \tag{4.39}$$

- Second derivative of $\log p_1$

$$\frac{\partial^2 \log p_1}{\partial \theta_i^2} = -\left[\boldsymbol{n}^T \boldsymbol{C}_{nn}^{-1}\frac{\partial^2 \boldsymbol{n}}{\partial \theta_i^2} + \frac{\partial \boldsymbol{n}^T}{\partial \theta_i}\boldsymbol{C}_{nn}^{-1}\frac{\partial \boldsymbol{n}}{\partial \theta_i}\right] \tag{4.40}$$

$$\frac{\partial^2 \log p_1}{\partial \theta_i \theta_j} = -\left[\boldsymbol{n}^T \boldsymbol{C}_{nn}^{-1}\frac{\partial^2 \boldsymbol{n}}{\partial \theta_i \theta_j} + \frac{\partial \boldsymbol{n}^T}{\partial \theta_j}\boldsymbol{C}_{nn}^{-1}\frac{\partial \boldsymbol{n}}{\partial \theta_i}\right] \tag{4.41}$$

- Take expectations to construct the Fisher Information matrix

$$\boldsymbol{I}(\boldsymbol{x}) = \begin{bmatrix} -[E(\frac{\partial^2 (\log p_1)}{\partial \theta_s^2})] & -[E(\frac{\partial^2 (\log p_1)}{\partial \theta_s \partial \theta_o})] \\[2em] -[E(\frac{\partial^2 (\log p_1)}{\partial \theta_o \partial \theta_s})] & -[E(\frac{\partial^2 (\log p_1)}{\partial \theta_o^2})] \end{bmatrix}$$

Thus filling in the matrix $\boldsymbol{I(x)}$, we get

$$
\boldsymbol{I(x)} =
\begin{bmatrix}
[\frac{\partial \boldsymbol{n}^T}{\partial \theta_s} \boldsymbol{C}_{nn}^{-1} \frac{\partial \boldsymbol{n}}{\partial \theta_s}] & [\frac{\partial \boldsymbol{n}^T}{\partial \theta_s} \boldsymbol{C}_{nn}^{-1} \frac{\partial \boldsymbol{n}}{\partial \theta_o}] \\[2ex]
[\frac{\partial \boldsymbol{n}^T}{\partial \theta_o} \boldsymbol{C}_{nn}^{-1} \frac{\partial \boldsymbol{n}}{\partial \theta_s}] & [\frac{\partial \boldsymbol{n}^T}{\partial \theta_o} \boldsymbol{C}_{nn}^{-1} \frac{\partial \boldsymbol{n}}{\partial \theta_o}]
\end{bmatrix}
$$

Using the Equations 4.36, we calculate the first derivative of $\boldsymbol{n}$

$$
\begin{aligned}
\frac{\partial \boldsymbol{n}}{\partial \theta_s} &= \frac{(\boldsymbol{R\theta} - \boldsymbol{b})}{\partial \theta_s} \\[1ex]
&= \frac{\partial(\frac{\boldsymbol{T}_b}{\theta_s} - \boldsymbol{E}_{2LM \times 1} \frac{\theta_o}{\theta_s})}{\partial \theta_s} \\[1ex]
&= -\frac{\boldsymbol{T}_b}{\theta_s^2} + \boldsymbol{E}_{2LM \times 1} \frac{\theta_o}{\theta_s^2}
\end{aligned}
\tag{4.42}
$$

$$
\begin{aligned}
\frac{\partial \boldsymbol{n}}{\partial \theta_o} &= \frac{(\boldsymbol{R\theta} - \boldsymbol{b})}{\partial \theta_o} \\[1ex]
&= \frac{\partial(\frac{\boldsymbol{T}_b}{\theta_s} - \boldsymbol{E}_{2LM \times 1} \frac{\theta_o}{\theta_s})}{\partial \theta_o} \\[1ex]
&= -\frac{\boldsymbol{E}_{2ML \times 1}}{\theta_s}.
\end{aligned}
\tag{4.43}
$$

If we group all the derivatives for all the anchor nodes, we have

$$
CRLB(\boldsymbol{x}) = \left[\frac{\partial \boldsymbol{n}}{\partial \boldsymbol{\theta}}^T \boldsymbol{C}_{nn}^{-1} \frac{\partial \boldsymbol{n}}{\partial \boldsymbol{\theta}} \right]^{-1},
\tag{4.44}
$$

where

$$
\begin{aligned}
\frac{\partial \boldsymbol{n}}{\partial \boldsymbol{\theta}} &= \boldsymbol{B}_2, \\[1ex]
\boldsymbol{B}_2 &= \left[\frac{1}{\theta_s^2}(-\boldsymbol{T}_b + \theta_o \boldsymbol{E}_{2ML \times 1}), \ \frac{-\boldsymbol{E}_{2ML \times 1}}{\theta_s} \right]^T.
\end{aligned}
\tag{4.45}
$$

Therefore, $\boldsymbol{CRLB(x)} = \left(\boldsymbol{B}_2^T \boldsymbol{C}_{nn}^{-1} \boldsymbol{B}_2 \right)^{-1}$.

4.4.3 Cramer Rao Lower Bound of Location and Timing Estimates

In this section, we assume that the nodes are not time synchronized and also they are not aware of their position. We want to derive the CRLB of the variance of the residuals $\boldsymbol{n}$, where $\boldsymbol{n} = \boldsymbol{R\theta} - \boldsymbol{T(x)}$. The unknown vector to be estimated is $\boldsymbol{\phi} = [x, \ y, \ \theta_s, \ \theta_o]^T$. Following exactly the same analysis as the two previous sections, we derive the $\boldsymbol{CRLB(\phi)} = \left(\boldsymbol{B}^T \boldsymbol{C}_{nn}^{-1} \boldsymbol{B} \right)^{-1}$ where $\boldsymbol{B} = [\boldsymbol{B}_1 \ \boldsymbol{B}_2]$.
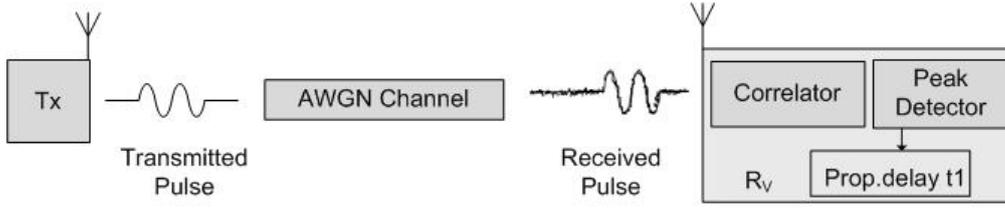
Figure 4.2: Simulation Model

4.5   Simulation Parameters

So far we have presented the joint localization and time synchronization algorithm
and the model analysis. This section describes the simulation model that we
developed in order to verify that the algorithm behaves as expected in various
cases. We used Matlab 2010 to build a simulator for the localization algorithm
described earlier. We are interested in the mean squared error of each sensor's
location and timing parameters when the variance σ^2 of the ToA error variables
in Equations 4.1 and 4.2 varies as follows:

$$10 \log(1/\sigma^2) \in [-15, 5] \ dB \tag{4.46}$$

Figure 4.2 depicts the modules comprising the simulation model:

- Tx is the transmitter, which sends a known pulse. The first derivative of
  a Gaussian pulse given by the following formula was chosen because it has
  no side lobes and a sharp roll-off compared to other pulses

$$v = 4\pi \frac{t}{\tau^2} \exp\left[-2\pi \left(\frac{t}{\tau}\right)^2\right], \tag{4.47}$$

  where τ is the pulse width parameter.

- This pulse is attenuated according to the log-distance propagation model

$$P(d) = P(1) - 10\gamma \log(d), \tag{4.48}$$

  where

    - $P(1)$ is the power in dBm at 1 meter from the sensor node, used as
      reference point,
    - d is the distance in meters between that the pulse needs to traverse,
    - γ is the path loss exponent equal to 2, and
    - $P(d)$ is the power in dBm at the distance d.
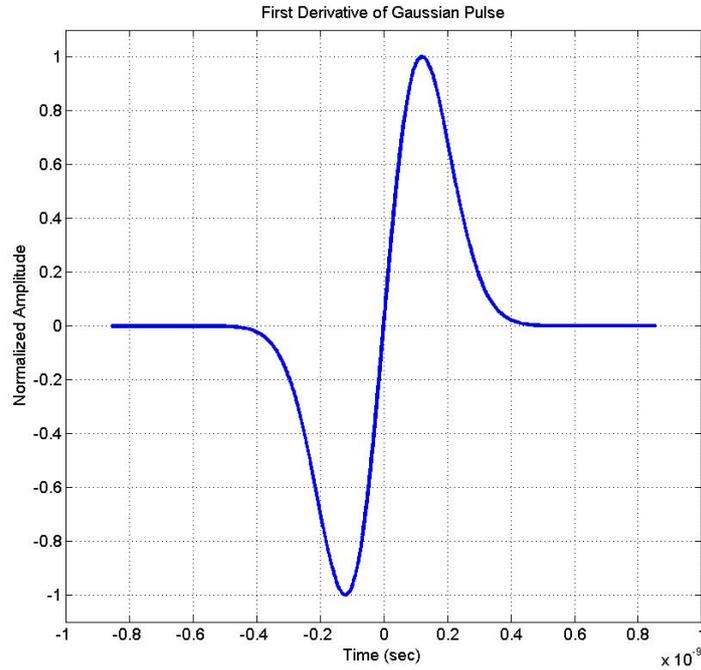
Figure 4.3: First derivative of Gaussian Pulse for $\tau = 0.43 \ ns$

Figure 4.3 shows the UWB transmitted pulse.

- The receiver Rv detects the known pulse using a correlator and a peak detector. Since the channel noise is white and random, the correlation between the received signal and the known pulse in Equation 4.47 will have a peak higher than the noise and will be detected by the peak detector. However, it is not so trivial to detect the right peak because the cross correlation will also be a noisy signal with many local minima and maxima. Therefore, we use a peak detection function which searches for the point with the largest value, around which there are points lower by a peak threshold δ on both sides. For our model, any threshold value above 0.4 times the maximum value leads to a correct peak detection. After detecting the peak, the time *lag* yields the propagation delay between the transmitter and the receiver.

The parameters of each sensor node that applies the algorithm are randomly chosen:

- an offset value θ_o randomly drawn from the set $[1, \ 10] \ ns$

- a skew value θ_s randomly drawn from the set $[-30, \ 30]$ parts per million (*ppm*)
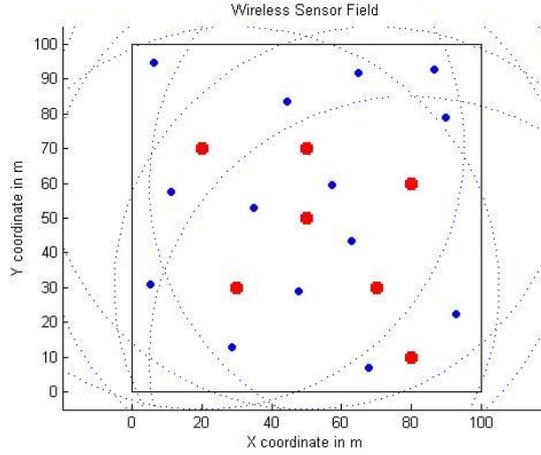
Figure 4.4: Sensor field with anchor nodes (marked with red) and sensor node (marked with blue). The dashed lines represent the communication range of each node

- location coordinates randomly chosen from $[1, \ 100] \ m$.

As far as the topology is concerned, we assume an area of $100 \times 100 \ m$ with 7 anchors and 1000 different possible locations for sensors. Since each node self localizes itself after communication with the neighboring anchor nodes, we run the localization algorithm 1000 times and average the results over all locations. In Figure 4.5, the wireless sensor field is depicted, where the red squares are the anchor nodes and the blue circle is a sensor node. The blue dashed circles show the communication range of each anchor node when RF signals are used for the localization algorithm.

The output of the simulator consists of two types of diagrams: the cumulative plot and the Mean Squared Error (MSE) of the WLS solution given by the equations

$$MSE_x = E(x^2 - \hat{x}^2)$$
$$MSE_{\theta_s} = E(\theta_s^2 - \hat{\theta}_s^2)$$
$$MSE_{\theta_o} = E(\theta_o^2 - \hat{\theta}_o^2).$$

Both types are used for timing and location parameters versus a changing variable depending on the scenario. We are interested in the effect of the following model parameters on the variance of the four LS errors:

- The error variance σ^2 measured in nanoseconds, denoted as n_{lm} and $\overline{n}_{lm}$ in Equations 4.1 and 4.2. We refer to the variance of n_{lm} and $\overline{n}_{lm}$ as jitter in

the next sections.

- The number of message exchange rounds M between a sensor node and an anchor.

- The bandwidth of the UWB known pulse transmitted given by

$$BW = \frac{2}{2\pi\tau\log(\sqrt{e})}, \tag{4.49}$$

where the pulse width τ was introduced in Equation 4.47 and $e = 2.718$ is the mathematical constant.

- The signal to noise ratio.

We refer to each of the above cases as scenario and present the results of the simulations, along with the analysis and the CRLB in the next section. For the analysis results, we use the covariance matrix from Equation 4.23 given by

$$\boldsymbol{C}_{CWLS} = (\boldsymbol{G}_1^T \boldsymbol{A}^T \boldsymbol{C}_e^{-1} \boldsymbol{A} \boldsymbol{G}_1)^{-1}. \tag{4.50}$$

The four diagonal elements of $\boldsymbol{C}_{CWLS}$ are the variances of the two coordinates (x, y) and the two timing parameters θ_1 and θ_2 respectively. Since we are interested in the variance of the parameters θ_s, and θ_o, we use the first order Taylor series expansion to obtain them as shown below

$$\sigma_{\theta_s}^2 = \frac{d\left(\frac{1}{\theta_1}\right)}{d\theta_1}\sigma_{\theta_1}^2 = \frac{1}{\theta_1^4}\sigma_{\theta_1}^2 \tag{4.51}$$

and

$$\sigma_{\theta_o}^2 = \left(\frac{\partial\left(\theta_2/\theta_1\right)}{\partial\boldsymbol{\theta}^T}\right)^2 \boldsymbol{C}_{CWLS}\frac{\partial\left(\theta_2/\theta_1\right)}{\partial\boldsymbol{\theta}} = \left[-\frac{\theta_2}{\theta_1^2}, \frac{1}{\theta_1}\right]^T \boldsymbol{C}_{CWLS}\left[-\frac{\theta_2}{\theta_1^2}, \frac{1}{\theta_1}\right]. \tag{4.52}$$

4.6 Simulation Results

This section contains the simulation results for the different scenarios described above.

4.6.1 Algorithm Performance when Jitter Varies

The varying parameter in this scenario is $1/jitter$. All the other parameters are fixed during the simulation to the values:

- two way message exchange rounds $M = 4$

- ultra wideband bandwidth of known pulse $BW = 1.5\ GHz$
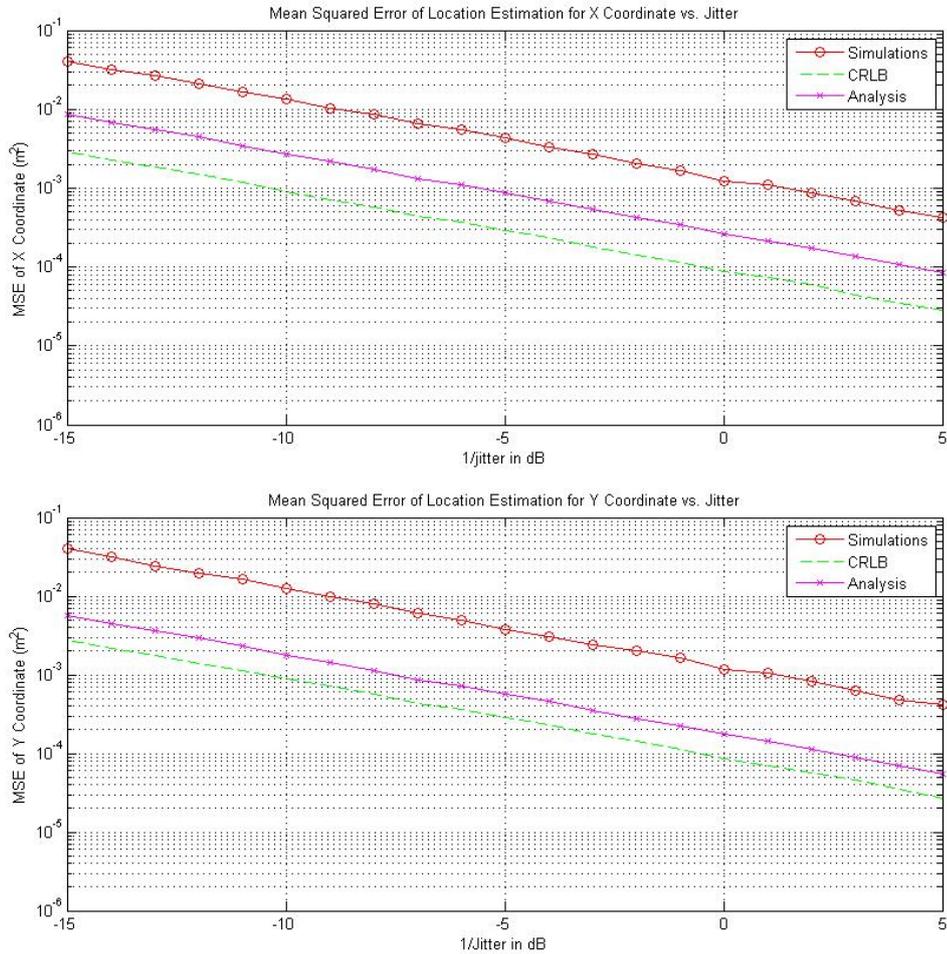
- $SNR = 40\ dB$.

Figure 4.5: MSE of location coordinates estimates vs jitter

Figure 4.5, depicts the MSE of the location estimates for the two coordinates x and y of the sensor node while the variance of error decreases. We observe three main points in this figure:

- As $1/jitter$ increases the MSE of location estimates decreases and vice versa

- The standard deviation of the error of the three curves is:

  - $\sigma_{simulation\ error} \in [0.0158,\ 0.1413]\ m$
  - $\sigma_{CWLS\ error} \in [0.0089,\ 0.089]\ m$

    – $\sigma_{CRLB\ error} \in [0.004,\ 0.0398]\ m$.

This practically means that even in very bad conditions in terms of noise, the position of the sensor node can be estimated with an accuracy of 14 cm, while in very good conditions the accuracy increases to $1.6cm$ as simulations show. This algorithm achieves an accuracy of 4 mm according to the CRLB
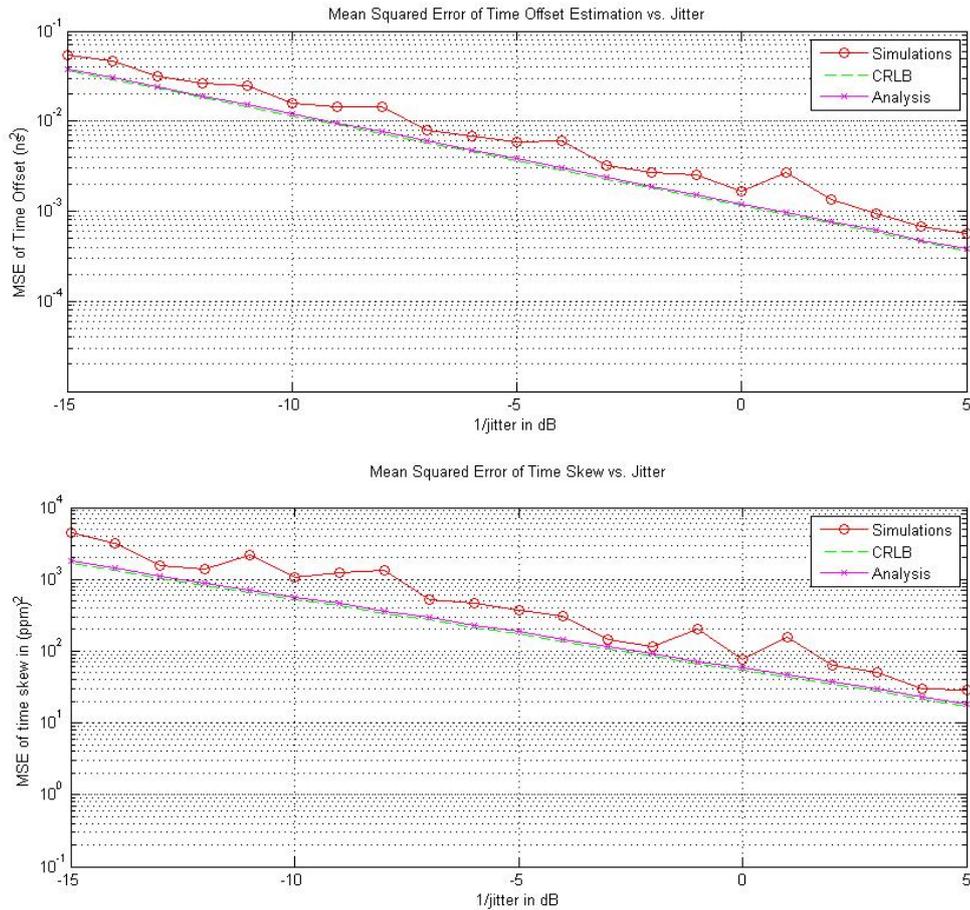
Figure 4.6: MSE of time parameters estimates vs jitter

- For the relevant position of the three curves, we observe that the green line corresponds to the CRLB and is the minimum error variance of this algorithm, therefore it is the lowest curve. The pink line represents the results of the analysis and the red line represents the simulation results. The distance between the red and the pink curves shows that the algorithm performs better in theory than in simulation. The difference however between the green and the pink curve stems from the fact that we want to estimate

the location coordinates x and y but in the model we are given information data about t_1 which is a non-linear transformation of them. Therefore, the estimator is not efficient [28].

However, as we see in the timing parameters results of Figure 4.6, the difference between the green and the pink curves is smaller, due to the fact that the information data about the time parameters is a linear transformation of them, given by Equation 4.3. While the offset value of the sensor node's clock is randomly chosen from $[0, \; 10] \; ns$, in Figure 4.6 we see that the standard deviation of the time offset parameter is of the order of sub-nanoseconds:

- $\sigma_{simulation \; error} \in [0.0178, \; 0.1778] \; m$

- $\sigma_{CWLS \; error} \approx \sigma_{CRLB \; error} \in [0.0141, \; 0.1585] \; m.$

The time skew estimation results depicted in the same figure, show that the standard deviation of the estimation reaches $3.548 \; ppm$, when as already mentioned the skew between the sensor nodes is randomly chosen from $[0, \; 30] \; ppm$.
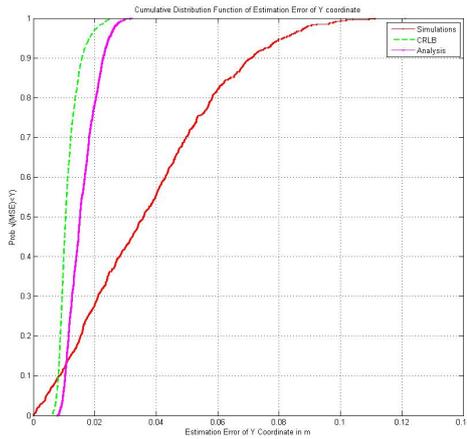
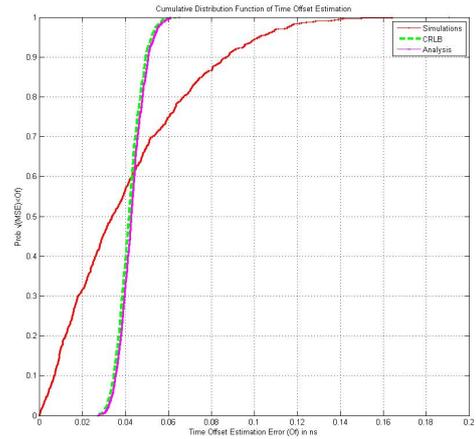Figure 4.7: Cumulative distribution of y coordinate estimation error when $1/jitter = -2dB$

Figure 4.8: Cumulative distribution of time offset estimation error when $1/jitter = -2dB$

The cumulative distribution function (CDF) of the estimation parameters informs us about the probability that the estimation error will attain a value less than or equal to each value that this error can take when $1/jitter$ is fixed. In Figure 4.7, we see the CDF of y coordinate estimate when $1/jitter$ is equal to $-2 \; dB$. The three curves of this figure are not symmetric, which can be verified by the fact that the median is less than the mean value, indicating that there is a positive skew;

- $median_{simulations} = 0.03585$ m, $\mu_{simulations} = 0.0447$ m

- $median_{CWLS} = 0.01517$ m , $\mu_{CWLS} = 0.0167$ m

- $median_{CRLB} = 0.01055$ m, $\mu_{CRLB} = 0.0119$ m.

The CDF yields information about the PDF of the estimation parameter. Thus, Figure 4.7 shows that the variance of the CRLB and the analysis is less than the variance of the simulation, as expected.

In Figure 4.8, similar observations apply for the time offset parameter.

4.6.2 Algorithm Performance when SNR Varies

The varying parameter in this scenario is the SNR. All the other parameters were fixed during the simulation runs to the values:

- two way message exchange rounds $M = 4$

- ultra wideband bandwidth of known pulse $BW = 1.5\ GHz$

- $1/jitter = -5\ dB$.

As we see in Figures 4.9 - 4.10, the SNR does not effect the accuracy of the LS estimator. Even at 10 dB where the signal is equal noise, the UWB signal is detected, verifying the robustness of UWB technology in a noisy environment. The estimation error of the location coordinates is of the order of cm, of the time offset is of the order of sub-nanoseconds and of time skew is less than 20 ppm.

4.6.3 Algorithm Performance when the Number of Message Exchange Rounds (M) Varies

In this scenario, we vary the number of two way communication rounds between each sensor node and each anchor in its vicinity. For the simulation we used the following parameters which ensure that the enhancement of the algorithm is only due to the increase of rounds:

- $SNR = 100\ dB$

- ultra wideband bandwidth of known pulse $BW = 1.5\ GHz$

- $1/jitter = 10\ dB$

Figures 4.11 to 4.12 show that there is an improvement in the accuracy of the algorithm as the number of rounds increases. Furthermore, we see that the CRLB of location estimates asymptotically approaches the value of $10^{-5.5}$, while the the MSE of simulation and analysis results approach the value of 10^{-5}. Increasing the communication rounds between a sensor node and an anchor decreases the
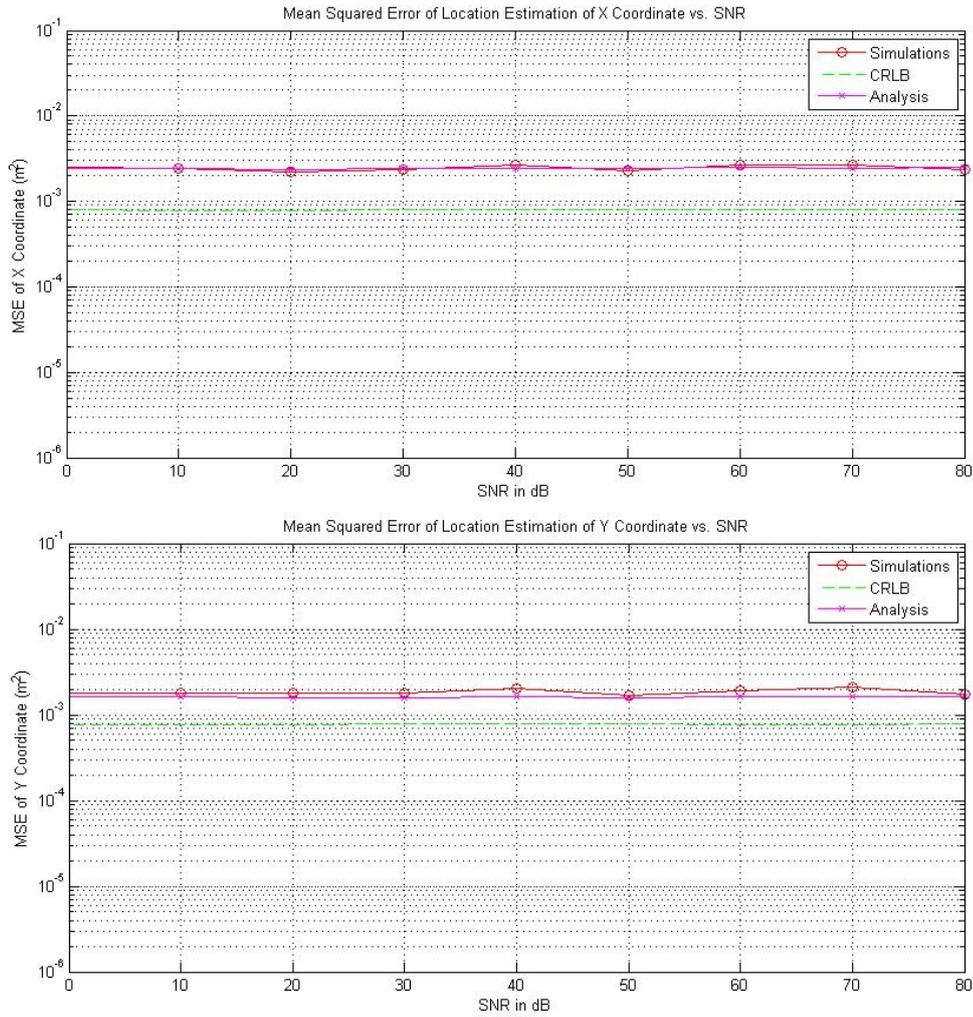
Figure 4.9: MSE of location coordinates estimates vs SNR

estimation error as has been proved by Kumar et al in [20] and is shown in the above figures.

4.6.4   Algorithm Performance when the UWB Pulse Bandwidth Varies

In this section, we are interested in the performance of the joint localization and time synchronization algorithm performance when the bandwidth of the UWB pulse varies. We vary the pulse width in time, which is translated in bandwidth change according to Equation 4.49.

The rest of the parameters are fixed and equal to

- two way message exchange rounds $M = 4$

- $SNR = 40\ dB$
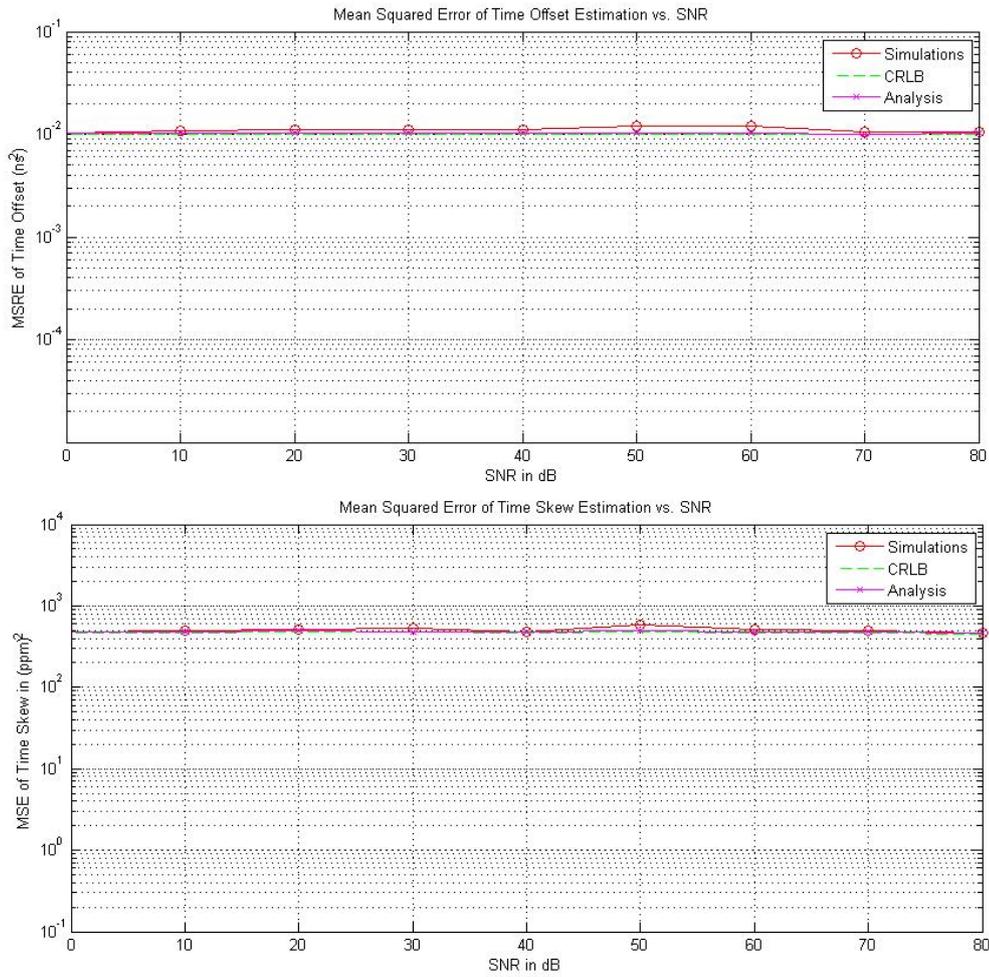
Figure 4.10: MSE of time parameters estimates vs. SNR

- $1/jitter = 1\ dB$.

Figures 4.13 - 4.14 show that by increasing the bandwidth of the transmitted UWB pulse, the MSE decreases three orders of magnitude and thus the performance of the algorithm is enhanced.

When using a very narrow pulse in time, the peak of the cross correlation function can be determined with high precision and therefore the propagation delay estimation is more accurate. Consequently, the estimation of location coordinates and time parameters have a smaller MSE. This behavior is verified by our simulation results.
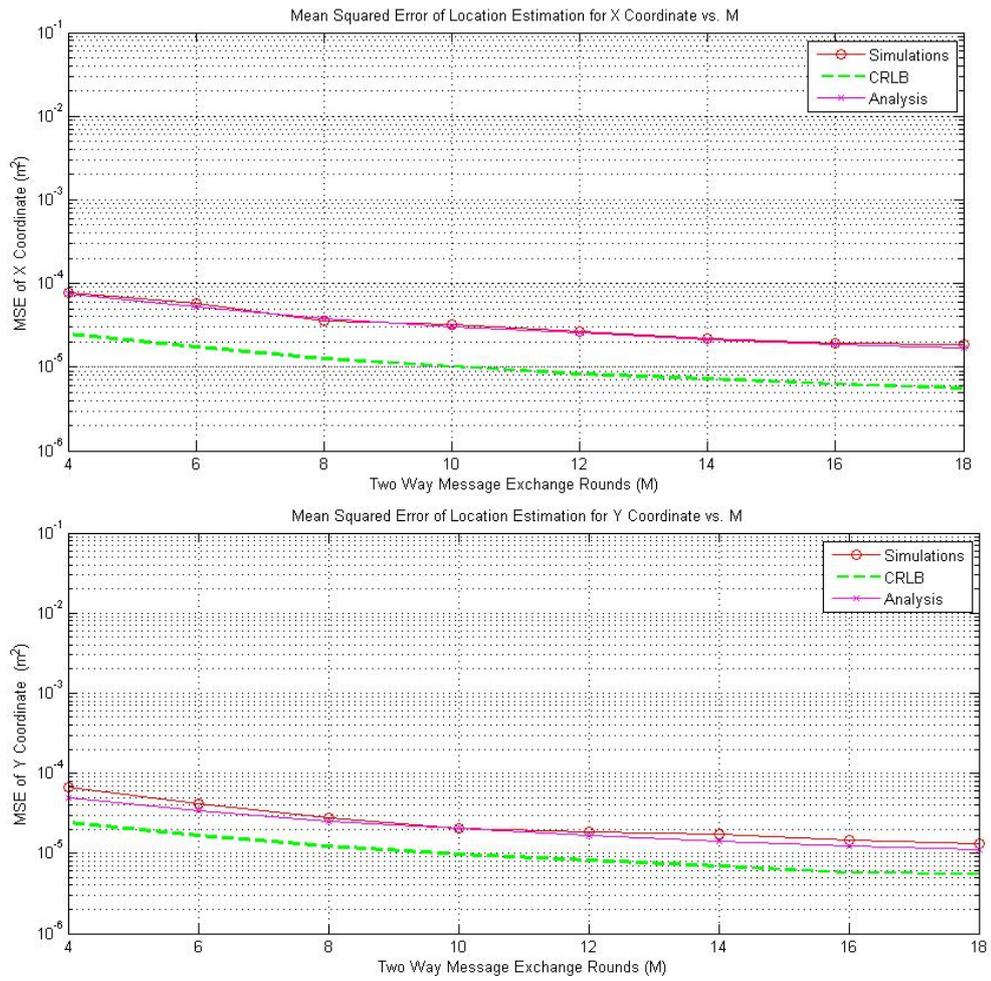
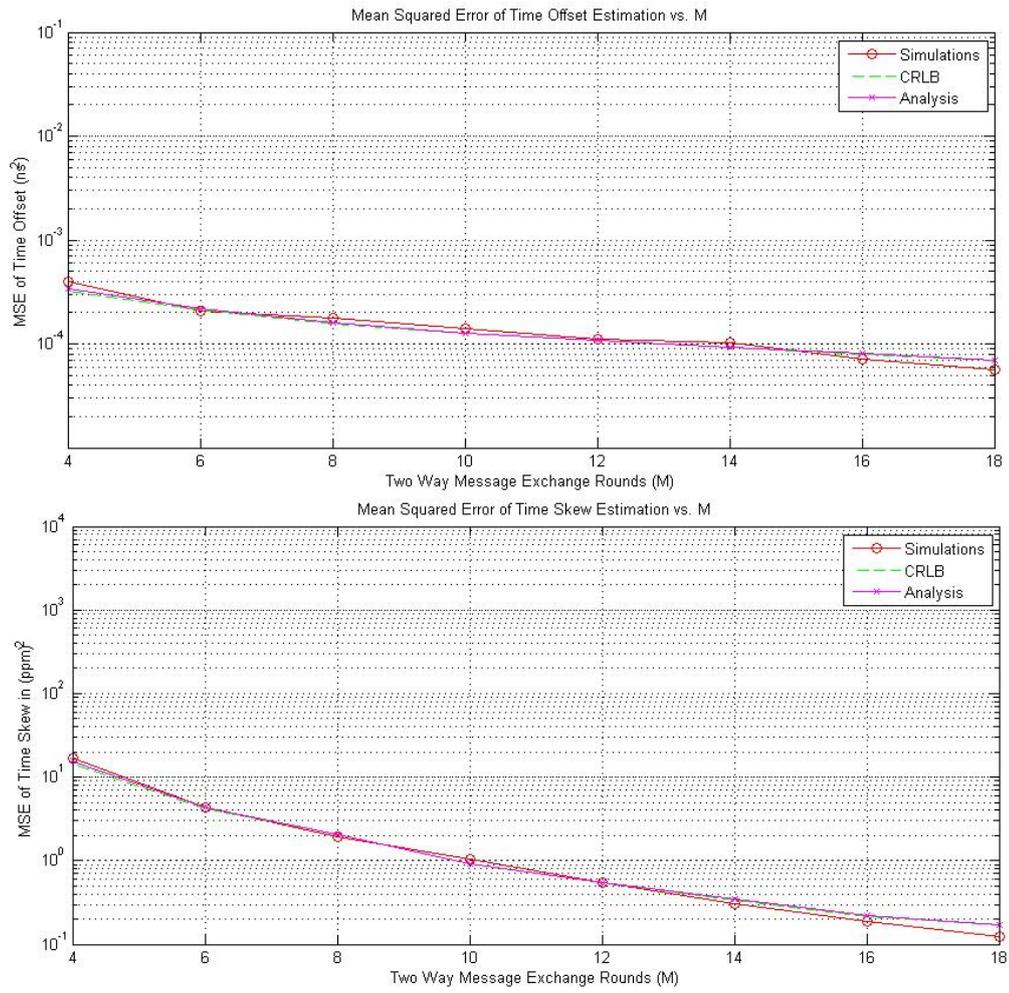Figure 4.11: MSE of location coordinates estimates vs M
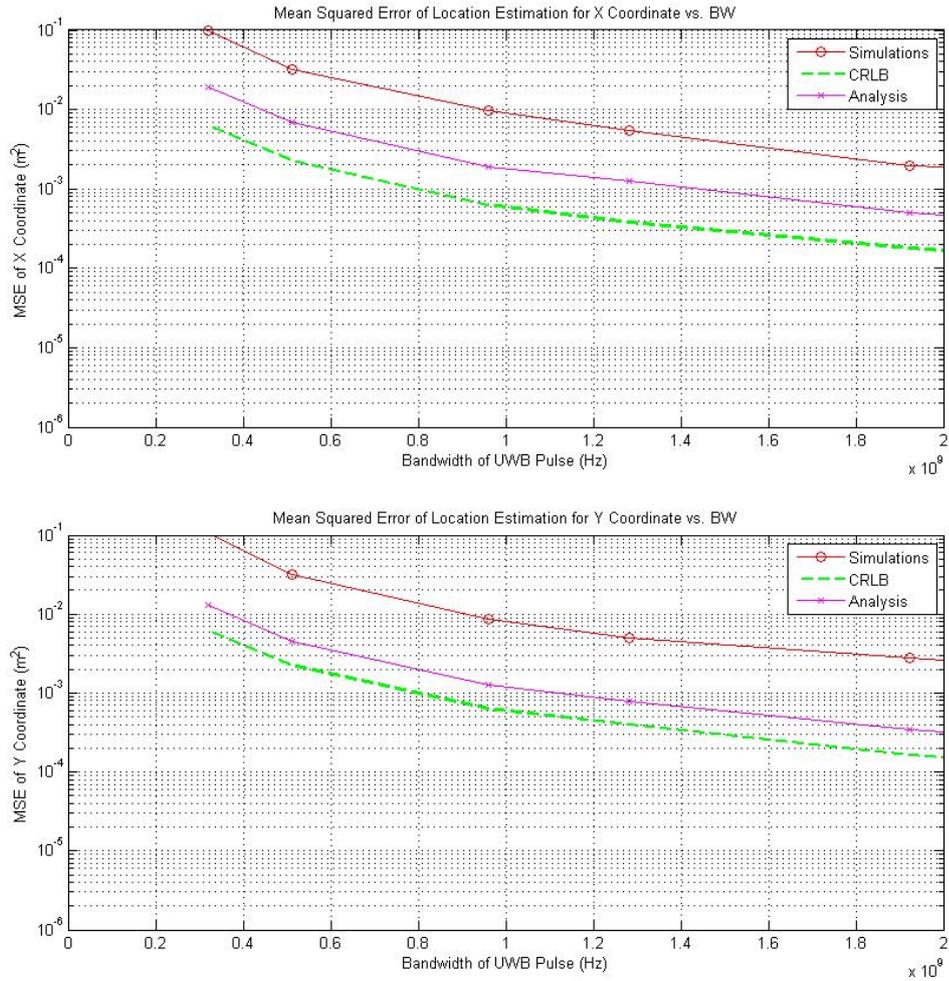
Figure 4.12: MSE of time parameters estimates vs. M

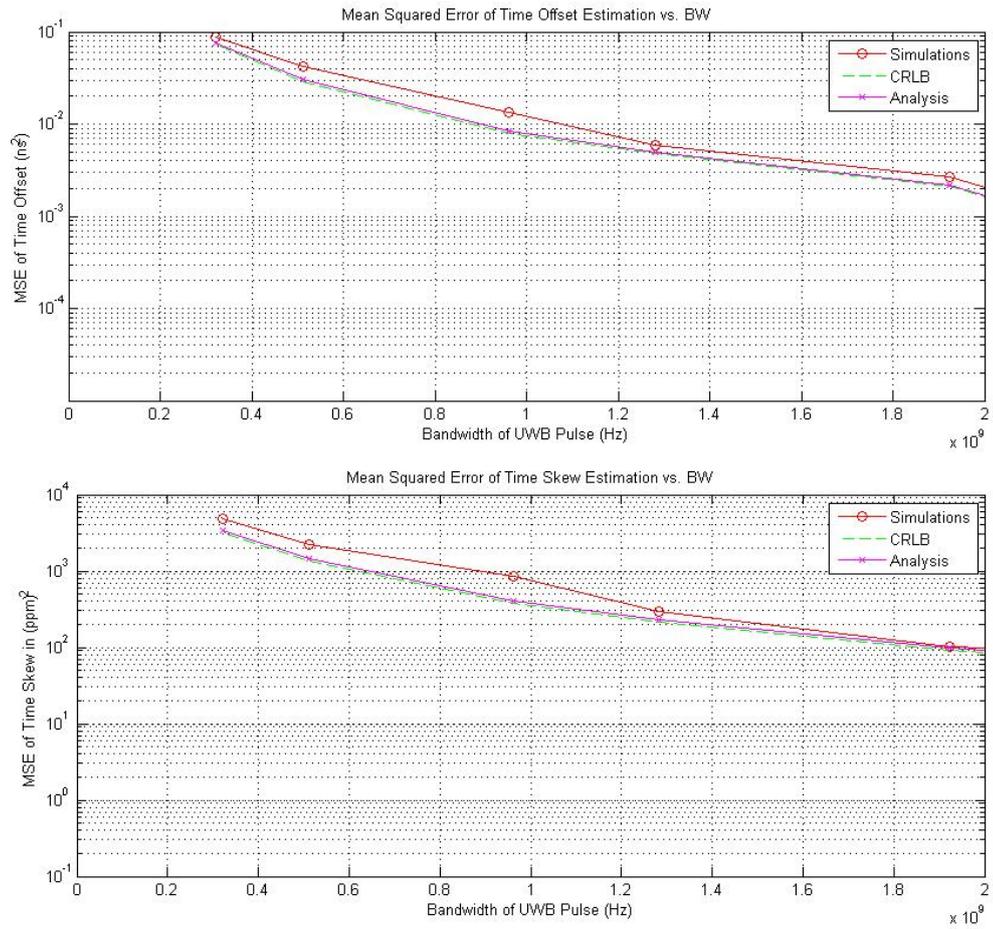Figure 4.13: MSE of location coordinates estimates vs. BW

Figure 4.14: MSE of time parameters vs. BW

Chapter 5

Localization in Perfectly Time Synchronized Network

In this chapter, we apply the introduced joint localization and synchronization algorithm in a network where all the nodes are perfectly time synchronized to a reference time. We adjust the simulation model, the model analysis and the CRLB accordingly and we reach well known results for location estimation. The next section states the assumptions under which we apply the algorithm, while Section 5.2 presents the algorithm under these assumptions. Afterwards, we proceed with the model analysis, the CRLB derivation and finally we present the simulation results. Since the generic form of the algorithm has been already presented, in this chapter we describe only the differences due to the assumptions.

5.1   Assumptions

In this chapter, we assume that all the nodes are time synchronized to a reference time. Thus, the time parameters skew and offset are $\theta_s = 1$ and $\theta_o = 0$ respectively.

5.2   Localization Algorithm Description

Since the algorithm remains the same, we use two way message communication for ranging. A sensor node is self-localized by exchanging M rounds of messages with each anchor l and based on the propagation delay t_1 it applies trilateration to determine its location. In more details, Figure 5.1 shows this bidirectional communication between sensor node B and anchor node A.

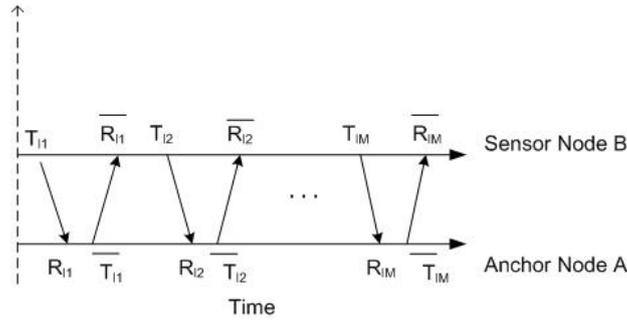- at time T sensor node B transmits a known pulse

Figure 5.1: Two way message exchange between a sensor node B and an anchor node A when they are time synchronized

- anchor node A detects and receives the pulse at time R

- node A transmits a known pulse at $\overline{T}$, and includes the timestamps $\overline{T}$ and R in the message that follows the known pulse

- node B detects the known pulse at $\overline{R}$ and receives the message.

We note that all all the timestamps are part of the same timing system. In the end of this message exchange, node B can calculate its distance from anchor node A. Since the measurements are noisy, node B repeats the message exchange M times with at least three anchors in its vicinity and using trilateration it estimates its location coordinates x and y. Thus, we seek for the solution with the minimum error.

5.3   Localization Model Analysis

The goal of this model analysis is to prove that the joint localization and synchronization algorithm described in the previous chapter yields a well known result for the localization problem.

In this case, due to perfect global time synchronization, all clocks run at the same frequency and they constantly display the same time. The equations that describe the two way message exchange depicted in Figure 5.1 are shown below

$$- R_{lm} + t_1 = -T_{lm} - n_{lm} \tag{5.1}$$

and

$$\overline{T}_{lm} + t_1 = \overline{R}_{lm} - \overline{n}_{lm}, \tag{5.2}$$

where n_{lm} and $\overline{n}_{lm}$ are the ToA detection errors which are assumed to be Gaussian distributed.

Equations 5.1 and 5.2 are repeated for M rounds per anchor node in the vicinity of the sensor node and $l \leq L$ times for all the anchors within communication

range. Following the same analysis as in the model analysis of the joint localization and time synchronization algorithm, we keep the parameter of interest t_1 in the left hand side, yielding

$$\frac{||\boldsymbol{x} - \boldsymbol{a}_l||}{c} = \overbrace{R_{lm} - T_{lm}}^{d_l/c} - n_{lm} \tag{5.3}$$

and

$$\frac{||\boldsymbol{x} - \boldsymbol{a}_l||}{c} = \overbrace{-\overline{T}_{lm} + \overline{R}_{lm}}^{d_l/c} - \overline{n}_{lm}, \tag{5.4}$$

where d_l is the distance between sensor node and anchor node A_l.

Furthermore, we rewrite the above set of equations after squaring both sides and expanding the euclidean distance $||\boldsymbol{x} - \boldsymbol{a}_l||$

$$\frac{2\boldsymbol{x}^T \boldsymbol{a}_l}{c^2} - \frac{||\boldsymbol{x}||^2}{c^2} = \frac{||\boldsymbol{a}_l||^2}{c^2} - \frac{d_l^2}{c^2} + e_{lm} \tag{5.5}$$

and

$$\frac{2\boldsymbol{x}^T \boldsymbol{a}_l}{c^2} - \frac{||\boldsymbol{x}||^2}{c^2} = \frac{||\boldsymbol{a}_l||^2}{c^2} - \frac{\overline{d}_l^2}{c^2} + \overline{e}_{lm}, \tag{5.6}$$

where e_{lm} and $\overline{e}_{lm}$ include all the error terms and are given by

$$e_{lm} = 2(-T_{lm}\theta_1 + R_{lm})n_{lm} - n_{lm}^2 \tag{5.7}$$

$$\overline{e}_{lm} = 2(\overline{R}_{lm}\theta_1 - \overline{T}_{lm})\overline{n}_{lm} - \overline{n}_{lm}^2. \tag{5.8}$$

As we observe, the error terms are no longer depending on unknown parameters.

Thus we have the following system of equations between sensor node B and anchor node A_1

$$\begin{cases} \frac{2\boldsymbol{x}^T \boldsymbol{a}_1}{c^2} - \frac{||\boldsymbol{x}||^2}{c^2} = \frac{||\boldsymbol{a}_1||^2}{c^2} - \frac{d_1^2}{c^2} + e_{11} \\ \frac{2\boldsymbol{x}^T \boldsymbol{a}_1}{c^2} - \frac{||\boldsymbol{x}||^2}{c^2} = \frac{||\boldsymbol{a}_1||^2}{c^2} - \frac{\overline{d}_1^2}{c^2} + \overline{e}_{11} \\ \vdots \\ \frac{2\boldsymbol{x}^T \boldsymbol{a}_1}{c^2} - \frac{||\boldsymbol{x}||^2}{c^2} = \frac{||\boldsymbol{a}_1||^2}{c^2} - \frac{d_1^2}{c^2} + e_{1M} \\ \frac{2\boldsymbol{x}^T \boldsymbol{a}_1}{c^2} - \frac{||\boldsymbol{x}||^2}{c^2} = \frac{||\boldsymbol{a}_1||^2}{c^2} - \frac{\overline{d}_1^2}{c^2} + \overline{e}_{1M} \end{cases}$$

The above message exchange is repeated between sensor node B and anchor node A_2, leading us to the following system

$$
\begin{cases}
\frac{2\boldsymbol{x}^T\boldsymbol{a}_2}{c^2} - \frac{||\boldsymbol{x}||^2}{c^2} &= \frac{||\boldsymbol{a}_2||^2}{c^2} - \frac{d_2^2}{c^2} + e_{21} \\
\frac{2\boldsymbol{x}^T\boldsymbol{a}_2}{c^2} - \frac{||\boldsymbol{x}||^2}{c^2} &= \frac{||\boldsymbol{a}_2||^2}{c^2} - \frac{\bar{d}_2^2}{c^2} + \bar{e}_{21} \\
\vdots \\
\frac{2\boldsymbol{x}^T\boldsymbol{a}_2}{c^2} - \frac{||\boldsymbol{x}||^2}{c^2} &= \frac{||\boldsymbol{a}_2||^2}{c^2} - \frac{d_2^2}{c^2} + e_{2M} \\
\frac{2\boldsymbol{x}^T\boldsymbol{a}_2}{c^2} - \frac{||\boldsymbol{x}||^2}{c^2} &= \frac{||\boldsymbol{a}_2||^2}{c^2} - \frac{\bar{d}_2^2}{c^2} + \bar{e}_{2M}
\end{cases}
$$

If we subtract the equations describing the messages exchanged between sensor node B and anchor node A_2 from those between sensor B and anchor node A_1 we reach to the following system of equations where the unknowns are the location coordinates $\boldsymbol{x} = [x, \ y]^T$

$$
\begin{cases}
\frac{2\boldsymbol{x}^T(\boldsymbol{a}_1-\boldsymbol{a}_2)}{c^2} &= \frac{||\boldsymbol{a}_1||^2-||\boldsymbol{a}_2||^2}{c^2} - \frac{d_1^2}{c^2} + \frac{d_2^2}{c^2} + e_{11} - e_{21} \\
\frac{2\boldsymbol{x}^T(\boldsymbol{a}_1-\boldsymbol{a}_2)}{c^2} &= \frac{||\boldsymbol{a}_1||^2-||\boldsymbol{a}_2||^2}{c^2} - \frac{\bar{d}_1^2}{c^2} + \frac{\bar{d}_2^2}{c^2} + \bar{e}_{11} - \bar{e}_{21} \\
\vdots \\
\frac{2\boldsymbol{x}^T(\boldsymbol{a}_1-\boldsymbol{a}_2)}{c^2} &= \frac{||\boldsymbol{a}_1||^2-||\boldsymbol{a}_2||^2}{c^2} - \frac{d_1^2}{c^2} + \frac{d_2^2}{c^2} + e_{1M} - e_{2M} \\
\frac{2\boldsymbol{x}^T(\boldsymbol{a}_1-\boldsymbol{a}_2)}{c^2} &= \frac{||\boldsymbol{a}_1||^2-||\boldsymbol{a}_2||^2}{c^2} - \frac{\bar{d}_1^2}{c^2} + \frac{\bar{d}_2^2}{c^2} + \bar{e}_{1M} - \bar{e}_{2M}.
\end{cases}
$$

Hence, the derived system of equations for nodes B, A_1 and A_2 can now be described by the following linear system:

$$
\boldsymbol{A}_2\boldsymbol{x} = \boldsymbol{b}_2 + \boldsymbol{e}_2 \tag{5.9}
$$

where

$$
\boldsymbol{A_2} = 2 \begin{bmatrix}
(a_{1x} - a_{2x}) & (a_{1y} - a_{2y}) \\
(a_{1x} - a_{2x}) & (a_{1y} - a_{2y}) \\
\vdots \\
(a_{1x} - a_{2x}) & (a_{1y} - a_{2y}) \\
(a_{1x} - a_{2x}) & (a_{1y} - a_{2y})
\end{bmatrix}_{2M\times 2}
$$

$$
\boldsymbol{x} = \begin{bmatrix} x \\ y \end{bmatrix}_{2\times 1}
$$

$$\boldsymbol{b_2} = \begin{bmatrix} (\|\boldsymbol{a}_1\|^2 - \|\boldsymbol{a}_2\|^2) - d_1^2 + d_2^2 \\[1em] (\|\boldsymbol{a}_1\|^2 - \|\boldsymbol{a}_2\|^2) - d_1^2 + d_2^2 \\[1em] \vdots \\[1em] (\|\boldsymbol{a}_1\|^2 - \|\boldsymbol{a}_2\|^2) - d_1^2 + d_2^2 \\[1em] (\|\boldsymbol{a}_1\|^2 - \|\boldsymbol{a}_2\|^2) - d_1^2 + d_2^2 \end{bmatrix}_{2M \times 1}$$

$$\boldsymbol{e}_2 = c^2 [e_{11} - e_{21}, \ \bar{e}_{11} - \bar{e}_{21}, \ ..., \ e_{11} - e_{21}, \ \bar{e}_{11} - \bar{e}_{21}]_{2M \times 1}^T$$

Since the sensor node exchanges messages with all the anchor nodes in its vicinity and assuming that the equations between sensor node and anchor nodes are subtracted from those between sensor node and the anchor node A_1, the matrices $\boldsymbol{A}$, $\boldsymbol{b}$ and $\boldsymbol{e}$ are as follows $\boldsymbol{A} = [A_2^T, ..., A_L^T]^T$, $\boldsymbol{b} = [b_2^T, ..., b_L^T]^T$ and $\boldsymbol{e} = [e_2^T, ..., e_L^T]^T$.

We choose the LSE Estimator to minimize the error $(\boldsymbol{Ax} - \boldsymbol{b})^T (\boldsymbol{Ax} - \boldsymbol{b})$. The location coordinates estimates are given by

$$\boldsymbol{x} = (\boldsymbol{A}^T \boldsymbol{A})^{-1} \boldsymbol{A}^T \boldsymbol{b}, \tag{5.10}$$

which is a well known solution for the localization problem in research.

5.4 Cramer Rao Lower Bound of Location Estimate with Perfect Timing

The CRLB for this special case of the generic algorithm has been already derived in Section 4.4.1.

5.5 Simulation Parameters

For the simulations of the localization algorithm, we used the simulator introduced in Section 4.5. We adjusted it to incorporate the perfect time synchronization of all the nodes.

5.6 Simulation Results

During the simulation runs, the following parameters are fixed:

- two way message exchange rounds $M = 4$
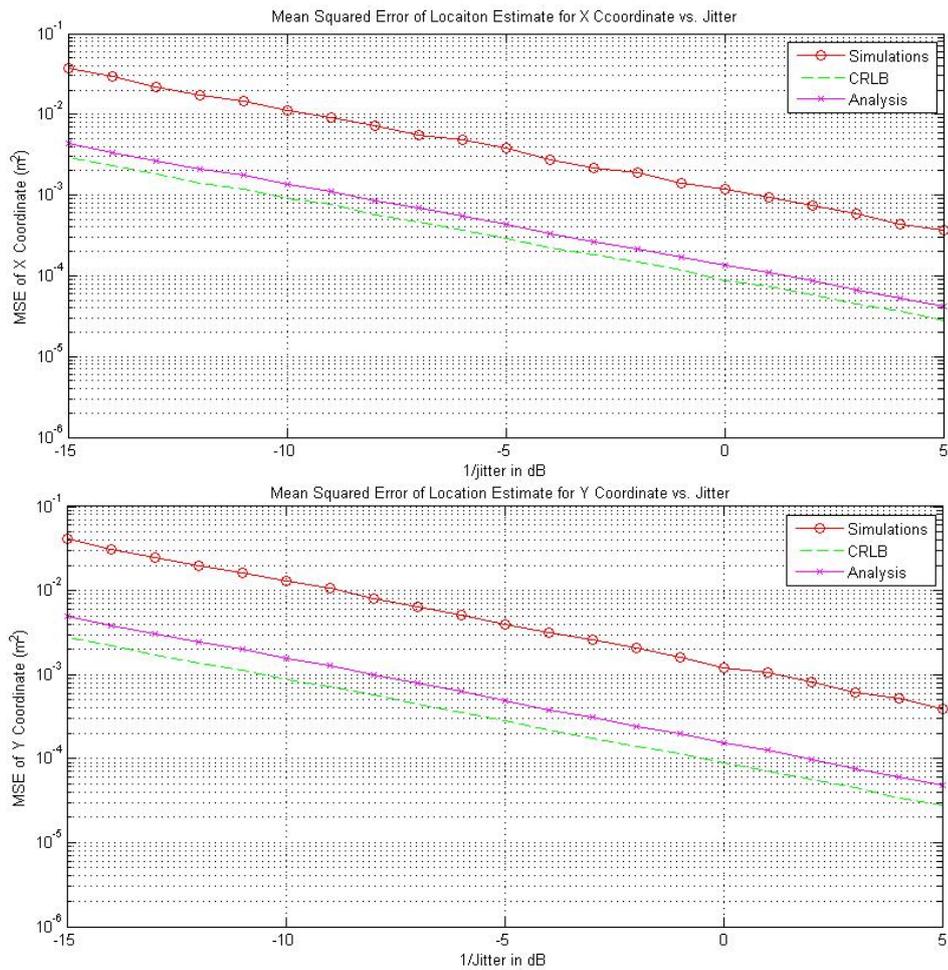
- ultra wideband bandwidth of known pulse $BW = 1.5 \ GHz$

Figure 5.2: MSE of location coordinates estimates vs jitter when all nodes are perfectly time-synchronized

- $SNR = 40 \ dB$.

In Figure 5.2, we plot the MSE of the location coordinates estimates versus $1/jitter$. We see that the MSE of the joint localization and time synchronization algorithm is of the order of a few centimeters regardless whether the nodes are perfectly synchronized or not.

Chapter 6

Time Synchronization Algorithm of Localized Nodes

In this chapter, we apply the joint localization and synchronization algorithm introduced in Chapter 4 in a network, where all nodes know their location coordinates and only the anchors are perfectly time synchronized to a reference clock. We adjust the simulation model, the model analysis and the CRLB by removing the coordinate vector $\boldsymbol{x}$ from the unknown vector $\boldsymbol{\xi}$ reaching well known results for time parameters estimation. The next section states the assumptions under which we apply the algorithm, while Section 6.2 presents the algorithm under these assumptions. Then we proceed with the model analysis, the CRLB derivation and finally we present the simulation results. Since the generic form of the algorithm has been already presented, in this chapter we only describe the differences.

6.1 Assumptions

We assume that all nodes are aware of their location coordinates x and y and that only the anchor nodes are time synchronized with a reference time.

6.2 Time Synchronization Algorithm Description

The goal of this algorithm is to estimate the time skew and offset of the sensors' clocks. Therefore sensor nodes exchange messages with the anchors within communication range as explained in Section 4.2. The exchanged messages with the included timestamps are used by the sensor nodes to estimate their time skew θ_s and time offset θ_o from the reference clock.

6.3   Time Synchronization Model Analysis

We use the same notation as in Section 4.3 for the sending and receiving times-
tamps to describe the relation between an anchor node A_l and a sensor node B
when they exchange M rounds of messages

$$- R_{lm} + t_1 = -T_{lm}\theta_1 + \theta_2 - n_{lm} \tag{6.1}$$

and

$$\overline{T}_{lm} + t_1 = \overline{R}_{lm}\theta_1 - \theta_2 - n_{lm}. \tag{6.2}$$

The unknown vector here is

$$\boldsymbol{\theta} = [\theta_1, \ \theta_2]^T. \tag{6.3}$$

Since the nodes know their location coordinates, the propagation delay t_1 is
a constant number describing the time needed by the RF signal to traverse the
distance between the two nodes and is computed by Equation 4.4. We transfer
the unknown terms of Equations 6.1 and 6.2 to the left and we have

$$T_{lm}\theta_1 + \theta_2 = -R_{lm} + t_1 + n_{lm} \tag{6.4}$$

and

$$\overline{R}_{lm}\theta_1 - \theta_2 = \overline{T}_{lm} + t_1 + n_{lm} \tag{6.5}$$

Then, we group all these equations in matrices as shown below

$$\boldsymbol{R}_l = \begin{bmatrix} -T_{l1} & 1 \\ \overline{R}_{l1} & -1 \\ -T_{l2} & 1 \\ \overline{R}_{l2} & -1 \\ \vdots & \\ -T_{lM} & 1 \\ \overline{R}_{lM} & -1 \end{bmatrix}_{2M \times 2}$$

$$\boldsymbol{b}_l = \begin{bmatrix} -R_{l1} + t_1 \\ \overline{T}_{l1} + t_1 \\ -R_{l2} + t_1 \\ \overline{T}_{l2} + t_1 \\ \vdots \\ -R_{lM} + t_1 \\ \overline{T}_{lM} + t_1 \end{bmatrix}_{2M \times 1}$$

$$\text{and } \boldsymbol{n}_l = \begin{bmatrix} n_{l1} \\ \overline{n}_{l1} \\ n_{l2} \\ \overline{n}_{l2} \\ \vdots \\ n_{lM} \\ \overline{n}_{lM} \end{bmatrix}_{2M \times 1}$$

and we represent the system of Equations 6.4 and 6.5 in vectors in the following form

$$\boldsymbol{R}_l \boldsymbol{\theta} = \boldsymbol{b}_l + \boldsymbol{n}_l. \tag{6.6}$$

As we observe, the error terms have the same variance and they contribute the same to the minimizing function

$$\boldsymbol{\theta}_{LS} = \arg \min_{\boldsymbol{\theta}}[(\boldsymbol{R}\boldsymbol{\theta} - \boldsymbol{b})^T (\boldsymbol{R}\boldsymbol{\theta} - \boldsymbol{b})]. \tag{6.7}$$

Thus the overdetermined linear system of equations can be solved by the LSE method. The estimation vector is given by the formula

$$\boldsymbol{\theta}_{LS} = (\boldsymbol{R}^T \boldsymbol{R})^{-1} \boldsymbol{R}^T \boldsymbol{b}. \tag{6.8}$$

6.4 Cramer Rao Lower Bound of Timing Estimate with Perfect Location

The CRLB for this special case of the generic algorithm has been calculated in section 4.4.2.

6.5 Simulation Parameters

For the simulations of the time synchronization algorithm, we used the simulator introduced in Section 4.5. We adjusted it to incorporate the known location coordinates of all the nodes.

6.6 Simulation Results

During the simulation runs, the following parameters are fixed:

- two way message exchange rounds $M = 4$

- ultra wideband bandwidth of known pulse $BW = 1.5 \ GHz$
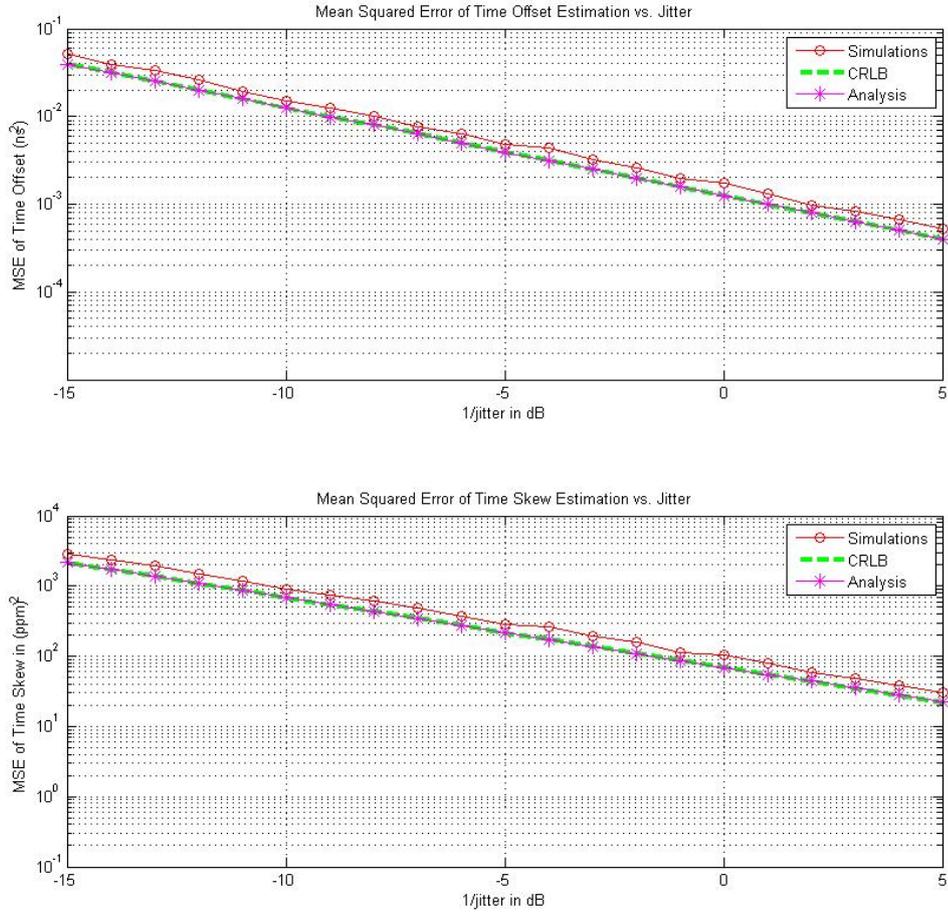
- $SNR = 40 \ dB$.

Figure 6.1: MSE of time parameters estimates vs. jitter when location coordinates are known

In Figure 6.1, we plot the MSE of the time offset and time skew estimates versus $1/jitter$. We observe that the model analysis estimate is efficient because it achieves the CRLB. This is justified by the fact that we assumed Gaussian distributed errors and the data information are linearly dependent on the variables that we estimate.

Chapter 7

Implementation Feasibility of Joint Localization and Time Synchronization Approach

In this section, we describe how expensive in terms of computation the joint algorithm is. In order to measure this cost in literature the floating point operations (flops) are used, which count for an addition, subtraction, multiplication or division. However, since division and multiplication are more expensive than addition and subtraction we separately calculate them.

7.1 Least Squares Computational Complexity

The joint localization and time synchronization algorithm computes the estimates for the location and the time in two steps: the first one is a rough estimation produced by the least squares algorithm and the second step refines this solution with the use of weighted least squares. For the first step, the set of equations and the required calculations are presented below. The LS solution that was given in previous section is repeated before the calculation of the complexity:

$$\boldsymbol{x}_{LS} = (\boldsymbol{G_1}^T \boldsymbol{A}^T \boldsymbol{A} \boldsymbol{G_1})^{-1} \boldsymbol{G_1}^T \boldsymbol{A}^T \boldsymbol{b} \tag{7.1}$$

- Multiply the [n x k] matrix $\boldsymbol{A}$ with the [k x p] matrix $\boldsymbol{G_1}$:
  Additions: $np(k-1)$
  Multiplications: npk

- Multiply the [p x n] matrix $(\boldsymbol{AG}_1)^T$ with the [n x p] matrix $\boldsymbol{AG}_1$:
  Additions: $p^2(n-1)$
  Multiplications: $p^2 n$

- Inverse the [p x p] matrix $((\boldsymbol{AG}_1)^T \boldsymbol{AG}_1)$:
  Additions: $p^3 - 2p^2 + p$
  Multiplications/divisions: p^3

- Multiply the [p x p] matrix $[(\boldsymbol{AG}_1)^T \boldsymbol{AG}_1]^{-1}$ with the [p x n] matrix $(\boldsymbol{AG}_1)^T$:
  Additions: $pn(p-1)$
  Multiplications: $p^2 n$

- Multiply the [p x n] matrix $[(\boldsymbol{AG}_1)^T \boldsymbol{AG}_1]^{-1} (\boldsymbol{AG}_1)^T$ with the [n x 1] vector $\boldsymbol{b}$
  Additions: $p(n-1)$
  Multiplications: pn

In the above calculations, we made the assumption that we can store an outcome and reuse it many times and also we assume that if we have already multiplied two matrices the transpose matrix of their product is available without further operations. Until this point, if we sum the operations need, we see that we need to perform $p^3 + 2p^2 n - 3p^2 - np + npk$ additions and $p^3 + 2p^2 + pn + pnk$ multiplications. In our case the actual number of operations is calculated if we substitute $p = 4$, $k = 7$, $n = 2ML$. Thus we have in total $64 + 64ML$ additions and $80 + 64ML$ multiplications.

7.2 Weighted Least Squares Computational Complexity

For the second step, to refine the solution of the first step we perform the same approach once again, but taking into consideration the covariance of the error vector. Thus for the Weighted Least Squares approach, we need all the operations described above and also the calculation of the inverse of the error covariance matrix. In more detail, we present below the required calculations and the WLS solution.

$$\boldsymbol{\omega}_{CWLS} = (\boldsymbol{G}_2^T \boldsymbol{A}^T \boldsymbol{C}_e^{-1} \boldsymbol{AG}_2)^{-1} \boldsymbol{G}_2^T \boldsymbol{A}^T \boldsymbol{C}_e^{-1} \boldsymbol{b} \qquad (7.2)$$

- Multiply the [n x k] matrix $\boldsymbol{A}$ with the [k x p] matrix $\boldsymbol{G}_2$:
  Additions: $np(k-1)$

Multiplications: npk

- Inverse the [n x n] covariance matrix $\boldsymbol{C}_e$:
  Additions: $n^3 - n^2 + n$
  Multiplications: n^3

- Multiply the [p x n] matrix $(\boldsymbol{G}_2 \boldsymbol{A})^T$ with the [n x n] inverse covariance matrix $\boldsymbol{C}_e^{-1}$:
  Additions: $pn(n-1)$
  Multiplications: pn^2

- Multiply the [p x n] matrix $(\boldsymbol{A}\boldsymbol{G}_2)^T \boldsymbol{C}_e^{-1}$ with the [n x p] matrix $\boldsymbol{A}\boldsymbol{G}_2$:
  Additions: $p^2(n-1)$
  Multiplications: $p^2 n$

- Inverse the [p x p] matrix $(\boldsymbol{A}\boldsymbol{G}_2)^T \boldsymbol{C}_e^{-1} \boldsymbol{A}\boldsymbol{G}_2$:
  Additions: $p^3 - 2p^2 + p$
  Multiplications/divisions: p^3

- Multiply the [p x p] matrix $(\boldsymbol{A}\boldsymbol{G}_2)^T \boldsymbol{C}_e^{-1} \boldsymbol{A}\boldsymbol{G}_2^{-1}$ with the [p x n] matrix $(\boldsymbol{A}\boldsymbol{G}_2)^T \boldsymbol{C}_e^{-1}$:
  Additions: $pn(p-1)$
  Multiplications: $p^2 n$

- Multiply the [p x n] matrix $(\boldsymbol{A}\boldsymbol{G}_2)^T \boldsymbol{C}_e^{-1} \boldsymbol{A}\boldsymbol{G}_2^{-1}(\boldsymbol{A}\boldsymbol{G}_2)^T \boldsymbol{C}_e^{-1}$ with the [n x 1] vector $\boldsymbol{b}$
  Additions: $p(n-1)$
  Multiplications: pn

For the refinement step the amount of operations that are required sum up to $(npk - 2np + n^3 - n^2 + n + 2np^2 + n^2p + p^3 - 3p^2)$ additions and $(n^3 + n^2p + npk + p^3 + 2p^2n + pn)$ multiplications. If we substitute the sizes of the matrices ($p = 4$, $k = 7$, $n = 2ML$), we see that the WLS need $8M^3L^3 + 106ML + 12M^2L^2 + 16$ additions and $8M^3L^3 + 4M^2L^2 + 128ML + 64$ multiplications.

7.3 Matrix Acquisition

So far we have explained how many operations are needed for the LS and the WLS approaches without however mentioning how many operations are needed to fill the matrices that are used in these methods. In this section, we explain how many multiplications (or divisions) and additions (or subtractions) are needed to fill those matrices.

- For [n x k] matrix $\boldsymbol{A}$: $6n + kn$ multiplications

- For [n x 1] vector $\boldsymbol{b}$: $4n$ multiplications and n subtractions

- For [k x p] matrix $\boldsymbol{G}_1$: $kp + 4$ multiplications

- For [n x n] covariance matrix $\boldsymbol{C}_e$: $5n + 2n^2$ multiplications and $2n^2 + 2n$ additions

- For [k x p] matrix $\boldsymbol{G}_2$: $kp + 4$ multiplications

Below we summarize in a table how expensive the joint approach is when we have M rounds of two way message exchange and L anchor nodes in the vicinity of the node that applies this algorithm.

Operation Type	Multiplication/Division	Addition/Subtraction
Least Squares	80+64ML	64+64ML
Weighted Least Squares	64+128ML+$4M^2L^2 + 8M^3L^3$	16+106ML+$12M^2L^2 + 8M^3L^3$
Matrices Filling	64+34ML	6ML+$8M^2L^2$
Total Sum	208+226ML+$4M^2L^2 + 8M^3L^3$	80+176ML+$20M^2L^2 + 8M^3L^3$

Table 7.1: Computational Complexity of Joint Localization and Time Synchronization Algorithm

Chapter 8

Conclusion

In this master thesis, we have addressed the challenges of localization and time synchronization of wireless sensor nodes in the context of land seismic exploration, where the accurate location estimation and time synchronization of the sensor nodes is essential for generating a good quality three dimensional seismic image.

We have introduced a new joint localization and time synchronization algorithm for WSNs. A model has been derived for jointly estimating the location and time skew and offset of sensor nodes by exchanging short communication messages between sensors and anchors. We assessed the estimation model by comparing it with the Cramer Rao Lower bound of the algorithm. We developed a simulator which allowed us to vary model parameters such as the noise, the SNR, the bandwidth of the communication signals, and visualize the performance of the algorithm in these scenarios. Finally, we investigated the implementation feasibility and the computational cost of the proposed algorithm. According to the results, the estimation accuracy is of the order of centimeters for location coordinates and of the order of sub-nanoseconds for the time parameters when the bandwidth of the communication signals and the rounds of the bidirectional communication between the sensor nodes and the anchor nodes increase.

A deeper investigation on the computational cost of the algorithm is left for future work, along with ways to decrease the the amount of operations required by the algorithm. Furthermore due to the limited amount of time of this thesis, we leave the implementation and evaluation on the target hardware platform of the suggested algorithm as future work.

Bibliography

[1] S. Savazzi and U. Spagnolini, "Synchronous Ultra-Wide Band Wireless Sensors Networks for oil and gas exploration," in *IEEE Symposium on Computers and Commpnications*, pp. 907 – 912, July 2009. [cited at p. 3, 12, 16]

[2] N. Patwari, I. Hero, A. O., M. Perkins, N. S. Correal, and R. J. O'Dea, "Relative location estimation in wireless sensor networks," vol. 51, no. 8, pp. 2137–2148, 2003. [cited at p. 6, 15]

[3] Y. Kwon, S. Sundresh, and K. Mechitov, "Resilient Localization for Sensor Networks in Outdoor Environments," *TOSN*, vol. 7, pp. 3:1–3:30, August 2010. [cited at p. 7, 8, 14]

[4] D. Niculescu and B. Nath, "Ad hoc positioning system (APS) using AOA," in *Proc. INFOCOM 2003. Twenty-Second Annual Joint Conf. of the IEEE Computer and Communications. IEEE Societies*, vol. 3, pp. 1734–1743, 2003. [cited at p. 8]

[5] J. Hightower and G. Borriello, "Location Systems for Ubiquitous Computing," in *IEEE Computer*, p. 34(8):5766, August 2001. [cited at p. 9]

[6] Y. Kwon and G. Agha, "Passive Localization: Large Size Sensor Network Localization Based on Environmental Events," in *IPSN '08 Proceedings of the 7th international conference on Information processing in sensor networks* (I. C. Society, ed.), pp. 3–14, April 2008. [cited at p. 9]

[7] "IEEE Draft Standard for Information Technology - Telecommunications and Information Exchange Between Systems - Local and Metropolitan Area Networks - Specific Requirements - Part 15.4: Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low Rate Wireless Personal Area Networks (WPANs)," July 2006. [cited at p. 11]

[8] S. Gezici, Z. Tian, G. Giannakis, H. Kobayashi, A. Molisch, V. Poor, and Z. Sahinoglou, "Localization via Ultra-Wideband Radios: A look at positioning aspects for future sensor networks," *IEEE Signal Processing Magazine*, vol. 22, pp. 70–85, July 2005. [cited at p. 12]

[9] G. Sun, J. Chen, W. Guo, and K. J. R. Liu, "Signal processing techniques in network-aided positioning: a survey of state-of-the-art positioning designs," vol. 22, no. 4, pp. 12–23, 2005. [cited at p. 13]

[10] Z. You, M. Q.-H. Meng, H. Liang, S. Li, Y. Li, W. Chen, Y. Zhou, S. Miao, K. Jiang, and Q. Guo, "A Localization Algorithm nin Wireless Sensor Networks Using a Mobile Beacon Node," in *Proc. Int. Conf. Information Acquisition ICIA '07*, pp. 420–426, 2007. [cited at p. 13]

[11] N. Patwari, J. N. Ash, S. Kyperountas, A. O. Hero, R. L. Moses, and N. S. Correal, "Locating the Nodes: Cooperative localization in wireless sensor networks," *IEEE Signal Processing Magazine*, vol. 22, pp. 54–69, July 2005. [cited at p. 13]

[12] M. Maroti, P. Volgyesi, S. Dora, B. Kusy, A. Nadas, A. Ledeczi, G. Balogh, and K. Molnar, "Radio Interferometric Geolocation," in *ACM SenSys* (S. 2005, ed.), pp. 1–12, November 2005. [cited at p. 14]

[13] Y. Shang and W. Ruml, "Improved MDS-Based Localization," in *Proceedings of the 23rd Conference of the IEEE Communications Society* (I. Infocom, ed.), vol. 4, pp. 2640–2651, March 2004. [cited at p. 15]

[14] L. Girod, V. Bychkovskiy, J. Elson, and D. Erstin, "Locating tiny sensors in time and space: A case Study," in *Proceedings of the 2002 IEEE Conference on Computer Design*, pp. 214–219, 2002. [cited at p. 15]

[15] L. Lazos and R. Poovendran, "SeRLoc: Secure Range-Independent Localization for Wireless Sensor Networks," in *WiSe '04 Proceedings of the 3rd ACM workshop on Wireless security*, pp. 21–30, October 2004. [cited at p. 15]

[16] R. J. Kozick and B. M. Sadler, "Sensor Localization Using Acoustic Doppler Shift With a Mobile Access Point," in *IEEE Statistical Signal Processing*, pp. 1256–1261, IEEE/SP 13th Workshop on Statistical Signal Processing 2005, July 2005. [cited at p. 16]

[17] S. Savazzi and U. Spagnolin, "Synchronous ultra-wide band wireless sensors networks for oil and gas exploration," in *Proc. IEEE Symp. Computers and Communications ISCC 2009*, pp. 907–912, 2009. [cited at p. 16]

[18] F. Ren and C. Lin, "Self-Correcting Time Synchronization Using Reference Broadcast in Wireless Sensor Network," *IEEE Wireless Communication*, vol. 18, pp. 79–85, 2008. [cited at p. 16]

[19] R. Solis, V. S. Borkar, and P. R. Kumar, "A New Distributed Time Synchronization Protocol for Multihop Wireless Networks," tech. rep., University of Illinois, Apr 2005. [cited at p. 17]

[20] A. Giridhar and P. R. Kumar, "Distributed Clock Synchronization over Wireless NetworksL Algorithms and Analysis," in *Proceedings of the 45th IEEE Conference on Decision and Control*, pp. 4915 – 4920, Dec. 13-15 2006. [cited at p. 17, 38]

[21] S. Graham and P. R. Kumar, "Time in general-purpose control systems: the Control Time Protocol and an experimental evaluation," in *Proc. CDC Decision and Control 43rd IEEE Conf*, vol. 4, pp. 4004–4009, 2004. [cited at p. 17]

[22] J. Zheng and Y. C. Wu, "Joint Time Synchronization and Localization of an Unknown Node in Wireless Sensor Networks," *IEEE Transactions on Signal Processing*, vol. 58, pp. 1309–1320, March 2010. [cited at p. 17, 18]

[23] P. Ranganathan and K. Nygard, "Time Synchronization in Wireless Sensor Networks: A Survey," *IJU*, vol. 1, pp. 92–102, April 2010. [cited at p. 18]

[24] M. Maroti, B. Kusy, G. Simon, and A. Ledeczi, "The Flooding Time Synchronization Protocol," in *Proceedings of the 2nd internation conference on Embedded networked sensor systems SenSys 04*, 2004. [cited at p. 18]

[25] J. Elson, L. Girod, and D. Estrin, "Fine-grained network time synchronization using reference broadcasts," in *In Proceedings of the Fifth Symposium on Operating Systems Design and Implementation*, ACM OSDI, December 2002. [cited at p. 18]

[26] S. Ganeriwal, R. Kumar, and M.Srivastava, "Timing Sync Protocol for Sensor Networks," in *ACM SenSys*, 2003. [cited at p. 18]

[27] F. Ferrari, M. Zimmerling, L. Thiele, and O. Saukh, "Efficient network flooding and time synchronization with Glossy," in *Proc. 10th Int Information Processing in Sensor Networks (IPSN) Conf*, pp. 73–84, 2011. [cited at p. 18]

[28] E. B. Andersen, "Asymptotic Properties of Conditional Maximum-Likelihood Estimators," *Journal of the Royal Statistical Society*, vol. 32, pp. 283–301, 1970. [cited at p. 36]

List of Figures

List of Tables