

Kalman Filter Based Ranging and Clock Synchronization for Ultra Wide Band Sensor Networks

by

©Nushen M. Senevirathna

A Thesis submitted to the School of Graduate Studies in partial fulfillment of the
requirements for the degree of

Master of Engineering

Memorial University of Newfoundland

October 2020

St. John's

Newfoundland

Abstract

This Thesis presents the design, implementation, and validation of a Kalman filter-based range estimation technique to precisely calculate the inter-node ranges of Ultra Wide Band (UWB) modules. In addition to that the development and validation of an improved global clock synchronization framework is presented.

Noise characteristics of relative time measurements of a stationary UWB anchor pair are first analyzed using an Allan deviation plot. To track the propagation of the imprecise clocks on low cost UWB transceiver platforms, Kalman filters are used in between every anchor pair. These filters track the variation of a remote anchor's hardware clock relative to it's own hardware clock, while estimating the time of flight between the anchor pair as a filter state. While adhering to a simple round robin transmission schedule, both inbound and outbound message timestamp data are used to update the filter. These measurements have made the time of flight observable in the chosen state space. A faster relative clock filter convergence has been achieved with the inclusion of the clock offset ratio as a measurement additional to the timestamps.

Furthermore, a modified gradient clock synchronization algorithm is used to achieve global clock synchronization throughout the network. A correction term is used in the gradient clock synchronization algorithm to enforce the global clock rate to converge at the average of individual clock rates while achieving asymptotic stability in clock rate error state. Stability of the original and modified methods for time invariant hardware clocks are compared using eigenvalue tests. Experiments are conducted to evaluate synchronization and ranging accuracy of the proposed range estimation approach.

keywords: Ultra Wide Band (UWB), UWB ranging, Wireless Clock Synchronization, Gradient Clock Synchronization, Range based Localization.

Acknowledgements

I wish to express my sincere appreciation to my supervisors Dr.Oscar De Silva, Dr.George Mann and Dr. Ray Gosine for guiding me through my Master of Engineering degree. Their knowledge, experience and encouragement was immensely helpful to successfully complete my research and write up the thesis.

I place on record, my sincere thanks to Natural Sciences and Engineering Research Council of Canada (NSERC), for funding my masters research and Memorial university fellowship program for the financial assistance provided.

I would also like to extend my gratitude to all personals at Is Lab namely, Dr.Thumeera R. Wanasinghe, Mr.Mihiran Galagedarage Don, Mr.Eranga Fernando, Mr.Ravingu Thalagala, Mr.Mahmoud Abd El Hakim, Mr.Kusal Tennakoon, Mr. Didula Disanayaka for their support, advise, and motivation throughout the course.

And not to mention my beloved wife Ms.Sachithra Attapattu who walk along with me throughout the journey, and being supportive all the time.

Table of Contents

Abstract	ii
Acknowledgments	iii
Table of Contents	v
List of Tables	vi
List of Figures	vii
1 Introduction	1
1.1 Motivation	1
1.2 Two way ranging	4
1.3 Two way ranging for anchor networks	6
1.4 Time difference of arrival and Clock synchronization	7
1.5 Sources of error	9
1.6 Problem Statement	11
1.6.1 Problem I: Noise associated with transmission and reception effecting time of flight calculation	11
1.6.2 Problem II: Chaotic behavior in gradient clock synchronization	12
1.7 Objective and Expected Contributions of the Research	12
1.8 Organization of the Thesis	14
2 Background	15
2.1 Enhanced Two way ranging	15
2.2 Synchronization in Wireless networks	18
2.3 Gradient clock synchronization	21
3 Relative clock filtering and Range Estimation	24
3.1 Introduction	24
3.2 Hardware clock analysis	25
3.3 Relative Clock Dynamics	27

3.4	State space model	29
3.5	Measurement model	30
3.6	Kalman filter	34
3.7	Process noise model	37
3.8	Implementation and Validation	37
4	Global Clock Synchronization	41
4.1	Gradient clock synchronization	41
4.2	Chaotic clock rate correction	44
4.3	Stability analysis	45
4.4	Implementation and validation	48
5	Ranging network and localization	51
5.1	Message content	51
5.2	Transmission schedule	52
5.3	Hardware platform	55
5.4	Ranging experiment	55
5.5	Localization	58
6	Conclusion and Future Directives	62
6.1	Summery of objective 1	62
6.2	Summery of objective 2	63
6.3	Summery of objective 3	64
6.4	Summery of conclusions	65
6.5	Publications	65
6.6	Future Directives	65
	Bibliography	xi

List of Tables

4.1	Synchronization errors	50
5.1	Initialization parameters	55
5.2	RMS Range errors	57

List of Figures

1.1	Decawave DWM1001 development board	3
1.2	Two way ranging	5
1.3	Time difference of arrival network	7
2.1	Filter performance	16
2.2	Double two way ranging	17
2.3	Symmetrical multi way ranging	18
2.4	Distributed Kalman filter for precise and robust clock synchronization in wireless networks	20
3.1	Relative hardware clock rate	25
3.2	Allan Deviation plot	26
3.3	Two way ranging	28
3.4	Filter update flow	34
3.5	Relative clock rate convergence comparison	38
3.6	Filter states	39
3.7	Measurement errors	40
4.1	Global clock rate parameters	49
4.2	Synchronization error distribution	50
5.1	Round robin Transmission schedule timing diagram	54
5.2	Experimental setup schematic	56
5.3	Experimental setup	56
5.4	experiment plots	57
5.5	Acceleration input to the localization filter	60
5.6	Position estimate with ranges	61
5.7	Position estimate without ranges	61

Abbreviations

DGPS Differential Global Positioning System

DTU Device Time Unit

FPU Floating Point Unit

FTSP Flooding Time Synchronization Protocol

GNSS Global Navigation Satellite System

GPS Global Positioning System

IC Integrated Circuit

IMU Inertial Measurement Unit

LIDAR Light Detection and Ranging

MAC Media Access Control

MAV Micro Aerial Vehicle

ppm parts per million

RMS Root Mean Squared

RMSE Root Mean Squared Error

SOC System on a Chip

TDOA Time Difference Of Arrival

TOF Time Of Flight

TPSN Timing-Sync Protocol for Sensor Networks

TWR Two Way Ranging

UWB Ultra Wide Band

WSN Wireless Sensor Network

Chapter 1

Introduction

In this chapter, first, the motivation for this study is presented, then followed by an overview of available ranging techniques along with the associated limitations. The problem statement of the thesis is then formulated, introducing the outstanding issues associated with the range calculations that will be addressed in this thesis. Finally, objectives and expected contributions will be highlighted along with the organization of the thesis.

1.1 Motivation

The state estimation has been the most challenging and vital component for implementing an autonomous navigation system. In mobile robot systems, this process narrows down mainly to the pose (position and orientation) estimation and its derivatives. The most practical way of obtaining pose is through inertial navigation systems using the embedded inertial measurement units (IMU). IMUs are usually composed of an accelerometer and a gyroscope. The gyroscope provides angular rate measurements while accelerometer provides accelerations, and these measurements are relative to the world frame. Obtaining orientation from angular rates and obtaining velocity

and position from acceleration measurements involve time integration of the measurements. Integrating the noise associated with the measurement signals causes drifts in the integrated states over time. Gravity vector measured through the accelerometer and the earth magnetic field measured through a compass sensor can be easily used to correct the drifts in the pose. Nevertheless, there is no such trivial measurement available to correct the drift in position. This has made any kind of a position measurement critical in IMU assisted state estimators on mobile robot platforms.

For outdoor applications, a position measurement can be obtained by GPS or broadly GNSS. Although position accuracy from GPS/GNSS is in the range of few meters, when integrated with an IMU, it provides state estimations with sufficient accuracy to navigate through outdoor environments for most of the applications such as delivery drones, video drones etc. For enhanced positioning, DGPS provides positioning accuracies in the range of few centimeters using the corrections from received signals at a ground station. When GPS signals are not available, such as in indoor spaces, obtaining position measurements has been a challenging research area.

Although various sensors, such as visual sensors, LIDAR, are available, these involve expensive hardware and require substantial computational power. Furthermore, bulky sizes of these sensors make them less suitable for small mobile robot applications such as MAVs. As an alternative, range based position estimation has been studied for indoor localization problem. During recent years, application of low cost ultra wide band (UWB) ranging devices became significantly popular [1–7] due to their fine time resolution and robustness against multi path propagation compared to other ranging systems [8]. Furthermore, their low cost, small form-factor, simplicity, and availability of higher data rates as compared to other expensive sensor suites make them more suitable for indoor mobile robot localization problem. Ranging using the time of propagation of radio signals require an accurate reference clock as the transmission

and reception of messages required to be timestamped. Since the speed of propagation of the signal is same as the speed of light, these reference clocks need to have a very high time resolution and a good precision. Although, these UWB transceivers are small low cost devices, hence they usually lack high accurate hardware clocks. DW1000 UWB chip by Decawave [9] has been popularly used in robotic research due to its relatively low cost and accurate time-stamping capabilities. The UWB transceiver (DWM1000) has an internal hardware clock, with a typical relative rate offsets around five ppm while IEEE 802.15.4 standards allow center rate offsets up to 40ppm [10]. Although this rate offset is insignificant for general microcontroller applications, it significantly affects range measurement applications due to the fact that a small time error can produce a higher ranging error when multiplied with the very high speed of propagation through the air. For example, the clock tick width of 15.65ps in the Decawave DWM1000 module represents a distance of 4mm in signal propagation. This implies that accurate modeling of clock dynamics can greatly improve the estimation accuracy of UWB ranging.

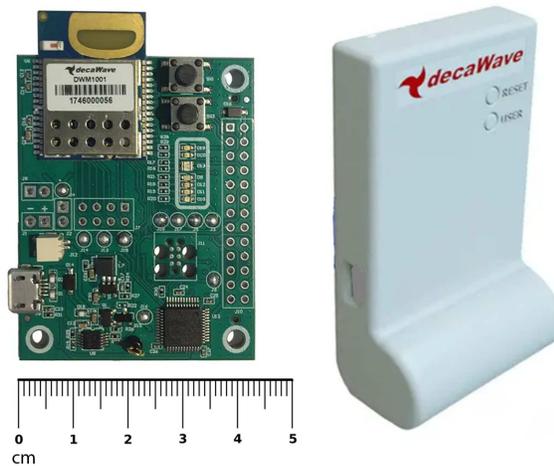


Figure 1.1: Decawave DWM1001 development board
image source: [9]

Other than the timestamp measurements, DWM1000 provides a direct measurement of the transmitter’s clock rate relative to the receiver [11]. This is estimated by integrating the carrier signal frequency of inbound messages. After a message reception, carrier integration value can be read from the device which can be used to calculate the relative clock rate. However, the experimental data logged from the anchors in Fig. 3.6(a) shows that this measurement is rather noisy.

The ability of DW1000 UWB transceiver IC to transmit a message at a given specific time and to time stamp reception of a message accurately allows to calculate the time of flight in order to estimate the range between transmission and reception devices. Calculating the time of flight using transmission and reception timestamps alone requires both the transmitter and receiver to agree on a common time. This is achieved through clock synchronization. If multiple transmitters are synchronized, Time difference of arrival (TDOA) method can be used, which incorporates the difference of arrival time as the measurement. In the absence of synchronization among clocks, range can be calculated using return trip time. This is known as two way ranging.

1.2 Two way ranging

In two way ranging (TWR), the initiating device (A) transmits a message, and the responder (B) replies after a known delay (d_r) from receiving it. When the initiator receives the reply, it can measure the round trip time (t_{rt}) and calculate the time of flight (tof) as follows.

$$tof = \frac{1}{2}(t_{rt} - d_r) \quad (1.1)$$

The reply delay can be a predetermined value or the responding device can embed this value in the reply message. In either case, the reply delay is calculated with

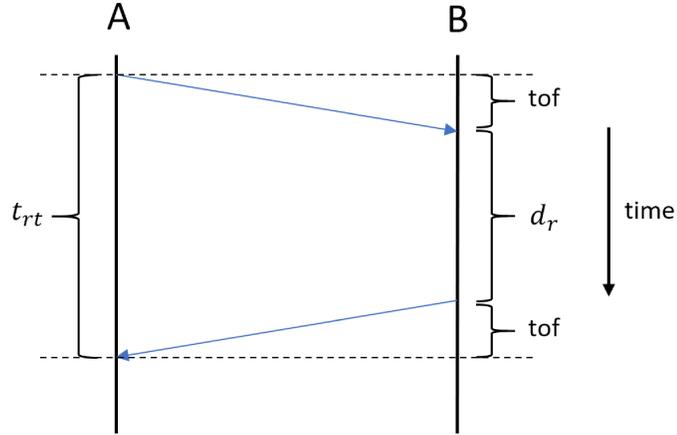


Figure 1.2: Two way ranging

respect to the responder's clock, where the round trip time is calculated with respect to the initiator's clock. Since these two clocks are not synchronized, they progress at slightly different rates. This leads to an error in time of flight calculation. Under the assumption that the initiator's clock is absolute, the range error δd can be estimated as;

$$\delta d = \frac{1}{2} c d_r \delta h_r \quad (1.2)$$

where δh_r is the relative clock progression rate and c being the speed of propagation. It can be seen that the error is proportional to the reply delay and the clock rate error. Practically there is a lower-bound for the reply delay, so this error cannot be completely eliminated. Alternatively, if the clock rate error can be estimated accurately, this error can be quantified, and two way ranging accuracy can be improved. Improved two way ranging techniques using clock rate corrections are discussed in section 2.1

1.3 Two way ranging for anchor networks

It is known that at least three range measurements are required to localize a point in 3D space with the ambiguity of a mirror solution. In order to obtain a unique solution it requires 4 range measurements from non plainer anchors. Usually, we would also want to provide infrastructure for multiple nodes to be localized at the same time. These requirements will lead to develop UWB networks with multiple transceiver nodes for localizing multiple agents in a given space. Adaptation of standard two way ranging protocol and its variants in a localization network is less desirable since several messages are involved in a single range measurement. Increased number of messages increases the power requirement for the devices as well as can cause possible interference with other networks operating at the same frequency range. Furthermore, in a repetitive ranging application, information from previous messages are not used in two way ranging. To overcome these issues, and utilize useful information from past messages, two way ranging can be performed over a transmission schedule. In this method, return trip time will be in the order of transmission cycle time, hence reply delays tend to be very long relative to standard two way ranging. This can cause substantial range errors due to the clock progression rate errors, and sometimes the errors can be magnified to the order of the measurements. If relative clock dynamics between every anchor pair is tracked on software, errors from the clock progression rate over long reply delays can be corrected, enabling us to use two way ranging over a transmission schedule.

Kalman filters are widely used in clock tracking filters that can be found in many literature [1, 3, 4, 7]. Linear behavior of the clock dynamics makes Kalman filter a perfect candidate to filter out the noise and estimate unmeasured states of the system.

1.4 Time difference of arrival and Clock synchronization

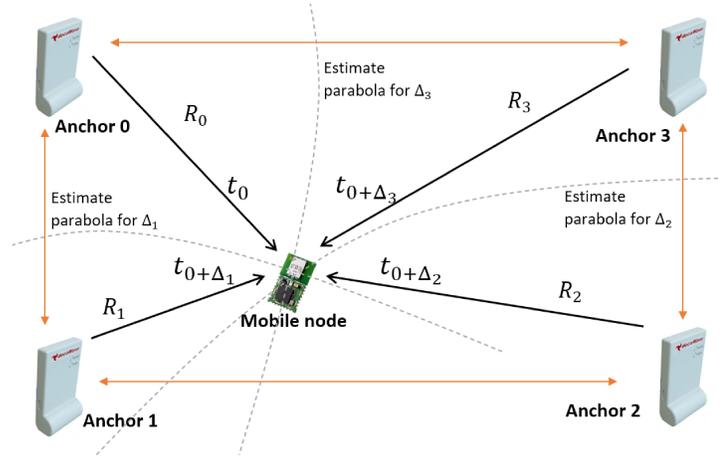


Figure 1.3: Time difference of arrival network

To provide localization information for a large number of nodes, time difference of arrival based localization can be implemented on a network. Since the receiver does not need to transmit in this method, an arbitrary number of nodes can be localized without adding any computational overhead to the network. This is especially useful when controlling a swarm of robots, or providing navigation information for a large number of people using mobile devices in an indoor environment. When a node wants to be localized, it listens for incoming messages and takes reception and transmit timestamps to calculate time of flight from transmitting anchors. Since the node is not synchronized with the network, it does not know its clock phase relative to the global time, which is required to calculate the reception timestamps in global time. Hence time differences are used as measurements. In order to obtain three range measurements for localization in 3D, the process requires three time differences, hence the receiving node is required to be in the vicinity of at least four anchors. Each time difference measurement corresponds to a location in a hyperbola as shown in

Fig.1.3. With multiple TDOA measurements, the location is usually estimated using numerical least square methods. [12] TDOA is the same basic methodology used in GPS based localization. In GPS positioning, TDOA outputs are just pseudo ranges and true ranges are calculated by applying several corrections to pseudo ranges. This discrepancy is due to clock errors and various delays occurred when the signal travels through the atmosphere.

TDOA based localization requires, all transmitting anchors to timestamp transmission events in a common global time, not with respect to their individual hardware clocks. This agreement on common time is achieved by global clock synchronization. Clock synchronization can be beneficial not only for localization networks but also in other WSN applications where accurate timestamping of an event or a measurement is required.

The traditional way of achieving this is by synchronizing to a master clock on a hierarchical structure. In the past several methods are proposed to adopt centralized synchronization for WSNs such as [13–15]. Yet in the domain of WSNs, distributed algorithms have many advantages over the centralized methods. They are robust against the failure of nodes and have better utilization of on-board computational resources. In the recent past, gradient clock synchronization as proposed in [2] has been quite popular in the field of robotics. This algorithm achieves the synchronization of a node by averaging the estimates for its own synchronization parameters using the global clock values received from an already synchronized anchor.

In the gradient clock synchronization framework, to convert a value of the hardware clock of anchor I at a time t , $h_I(t)$ to the corresponding global time $l_I(t)$, the global time is expressed as a affine function of the hardware clock as follows.

$$l_I(t) = \phi_I(h_I(t) + \theta_I). \quad (1.3)$$

Synchronization is achieved by estimating ϕ_I (scale) and θ_I (phase) parameters collectively .

1.5 Sources of error

We can summarise the various errors associated with range measurements as follow.

- timestamp noise: random noise associated with transmission and reception channels due to physical limitations.
- clock offset ratio noise: random noise associated with carrier frequency integration measurement.
- clock trim errors: steady clock rate being different to nominal value
- clock warm up: predictable variation of clock rate with time
- random variations in clock progression rate
- transmission and reception antenna delays: due to signal propagation delays in circuit
- triggering delays due to received power: signals with low reception power gets triggered delayed
- triggering delays due to received orientations: due to asymmetry in antenna radiation pattern
- multipath propagation: reflected signals are detected instead of direct path signals

The standard deviation of time of flight measurement caused by transmission and reception uncertainties is found to be around five clocks, which translates to a distance of 2-3cm. It could be observed that a coupled filter could effectively filter this noise while retaining information about low frequency variations.

The standard deviation of the noises associated with the clock offset ratio was found to be around 0.03ppm for warmed up clocks. Although this is quite small, for a 10ms reply delay which is practical in a network, this cause a range error of about 5cm. A better estimate can be achieved by incorporating the carrier frequency integration measurement in a relative clock rate tracking filter [3].

Clock trim errors can be corrected by calibration, and programming correct trim values. It has been found that the clock warm up variation follows a predictable curve [16]. However, together with the random variations, these errors can be estimated using relative clock tracking filters,

The antenna delays can lead to consistent ranging errors up to 30cm [1, 11, 17], This can be corrected by calibrating and programming the correct delays to hardware. Triggering delays are tackled by applying corrections based on received signal strength. Programmed look up table are used to store these calibrations.

Multipath propagation is when the receiver receives signal reflected from surfaces other than a straight path. In some cases reflected signal can have a higher strength. The early work have developed techniques to identify the preamble sequence in the first path. [8] However, in very complex environments, multipath propagation can result ambiguous measurements. In a localization framework, these can be filtered out using some kind of a gating technique based on a position estimate.

1.6 Problem Statement

This study presents a novel framework for UWB ranging and clock synchronization by addressing the following key challenges.

1.6.1 Problem I: Noise associated with transmission and reception effecting time of flight calculation

Timestamp resolution of DWM1000, 15.65ps does not necessarily mean that we can measure timestamps that accurately. Due to the physical limitations of components, there is a certain accuracy that we can expect from a measurement. If the measurement resolution is smaller than this achievable accuracy, we lose information in between measurement units. To avoid this undersampling from happening, usually the resolution of the measurement is designed to be higher than the expected accuracy. Oversampled data can be particularly useful when some of the errors can be corrected in post processing. Systematic errors can be modeled while random noises can be filtered out by averaging. Relative measurement timestamp precision of DWM1001 which is around 5 device time units (DTU), directly affects the standard two way range calculations. Assuming this noise is random, to obtain precise measurements, we need to incorporate a filtering technique. Although several implementations of filters are found in literature, they usually do not consider the time of flight as a state, although it is directly coupled with timestamp measurements. A filter with time of flight as a state will model the physical system more accurately and hence yield better performance.

1.6.2 Problem II: Chaotic behavior in gradient clock synchronization

Gradient clock synchronization algorithm [2] can be considered one of state of the art distributed synchronization methods for wireless sensor networks. It is used in many literature and have proven results. However, due to the nature of the algorithm used to update the scale parameter, there is no tie between the agreed global clock rate and the hardware clock rate values of the network anchors. This causes the settling value of the global clock rate to depend on the initial conditions and drift away from actual clock rates. This is referred to as the chaotic global clock rate phenomenon discussed in section 4.2. Although this error in the global clock rate is bounded under ideal conditions, this can cause very high errors at the event of an initialization error or when an inaccurate clock participating in early initialization as shown in the results of this thesis. Furthermore, after a temporal disturbance due to an unstable filter or due to high packet loss, scale parameters can settle with a high error scaling up the range measurements. As reported in some literature, this can lead to loss of synchronization. We would be rather interested in an algorithm that guarantees asymptotic stability in clock rate error states.

1.7 Objective and Expected Contributions of the Research

This thesis proposes a novel technique to estimate inter node ranges of a UWB network followed by an improved global synchronization framework. The main objectives of the proposed method and associated contributions are:

Objective 1 Propose a novel UWB ranging filter which uses time of flight as a state

- accurate modelling of coupled clock dynamics to effectively track variations in clock progression rates. To the best of authors knowledge incorporating time of flight as a state has not been performed previously for two ranging in filters.
- utilization of all available measurements to improve convergence, which includes reception timestamp of the outbound message, transmit timestamp of the inbound message and two clock offset ratio measurements at reception events.

Objective 2 Propose a novel distributed global clock synchronization algorithm

- handling the chaotic behaviour of global clock synchronization while retaining the distributed nature. This addresses the chaotic global clock phenomenon of the state of the art gradient clock sync algorithm proposed in [2]

Objective 3 Compare the performance and experimental validation of the proposed methods.

- comparison of the relative clock tracking filter convergence with and without the added redundant measurements.
- experimental validation of the modified clock synchronization algorithm.
- experimental validation of the Kalman filter based range estimator.

1.8 Organization of the Thesis

Chapter 1 presents an overview of the research area, highlights the research statement, and outlines the objectives and associated contributions of this study.

Chapter 2 presents the related work carried out in the area of the study and discuss their highlights and limitations.

Chapter 3 presents the novel Kalman filter based range estimation framework.

Chapter 4 presents the global clock synchronization algorithm and theoretical analyse of convergence.

Chapter 5 presents the experimental framework and results with a comparison with the state of the art methods, followed by the conclusion and future work.

Chapter 6 presents the conclusion and directions for future studies.

Chapter 2

Background

This chapter presents the previous work carried out in related areas. First section discusses several proposed methods to increase the two way ranging accuracy. Then general work related to wireless network time synchronization are presented, followed by discussing few variations of gradient clock synchronization algorithms.

2.1 Enhanced Two way ranging

As discussed in chapter 1, one of the main sources of errors in standard two way ranging is related to the relative clock progression rate. As a simple solution to estimate and correct for the difference in clock progression rates, double sided two way ranging algorithms are proposed. [18–23]. Due to the simplicity and the higher accuracy compared to the standard method, double sided methods have been popular among the researches and in many practical applications. Several studies are found in the literature proposing several variants of these methods [18–23]. Generally in all these methods, error due to relative clock progression rate is corrected by one or more additional messages exchanged between the ranging devices. These messages can be transmitted at different stages of the ranging protocol.

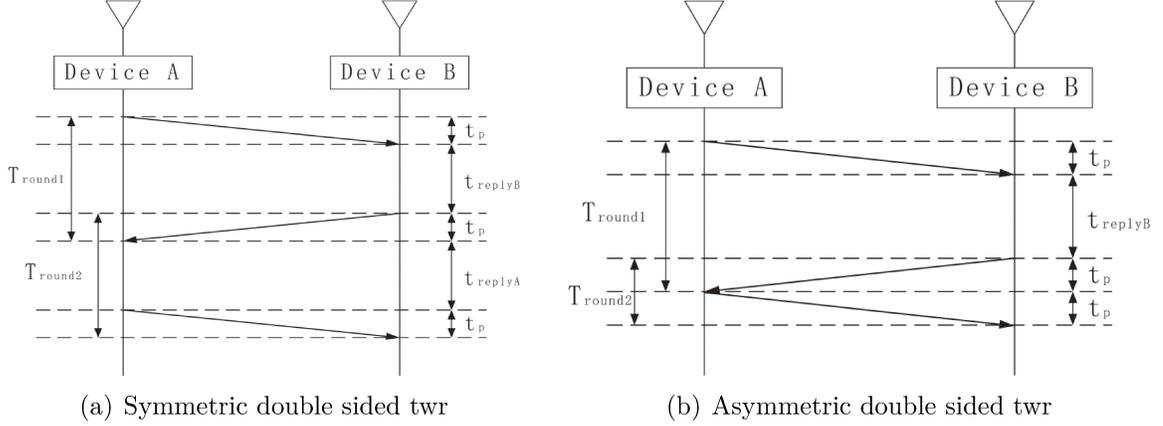


Figure 2.1: Filter performance
image source: [18]

Asymmetric and symmetric double sided two way ranging methods that are discussed in [18, 19, 22, 23] are shown in Fig.2.1(b) and 2.1(a) . These uses additional one or two messages after the standard two way ranging messages to correct the error caused by the relative clock progression rate. Additionally, [19] proposes a range calculation eliminating the symmetric constraint in symmetric double sided two way ranging. This is useful since it can be difficult to achieve a perfect symmetry in reply due to practical limitations of hardware. Asymmetric method shown in Fig.2.1(b) reduces the total ranging time by removing the second reply delay.

Work carried out in [20] proposes a one additional message transmitted by the initiator after a known delay from the first message as shown in Fig.2.2. Due to one less message required and overlapping transmissions, this reduces power consumption and overall ranging time compared to symmetric and asymmetric two way ranging methods. [23] proposes another variant where multiple reply messages after the first reception followed by a bidirectional data exchange packets.

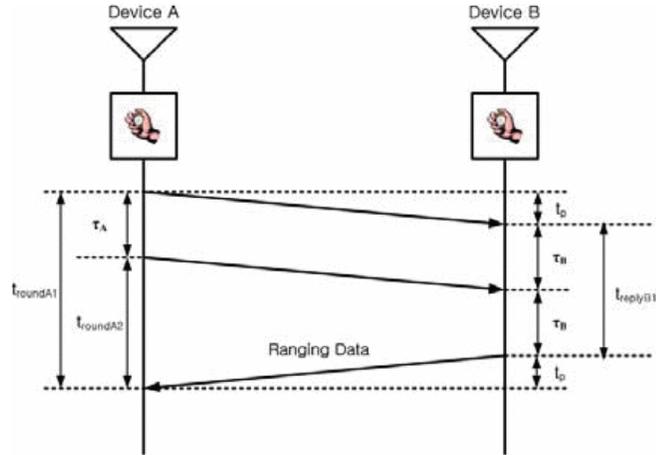


Figure 2.2: Double two way ranging
image source: [20]

Although these methods provide better accuracy with the relative clock rate corrections, the noise associated with transmission and reception timestamps directly affect the range calculations. Furthermore, requirement of multiple messages to be communicated between anchors for a single range measurement makes these less suitable to be used on a network of anchors. This also lead to an increased power consumption and possible interference with other devices.

A novel ranging scheme is proposed in [21] considering the pattern of the variation in return trip times in a sequence of two way ranging measurements due to the offset in hardware clock rates. Using the pattern of time of flight (TOF) variations, a precision higher than the clock resolution can be achieved. Although the theoretical contribution in this work is high, this method is not used in an experimental setup neither any continuation of this work was found. In most practical cases, the noise associated with the timestamps are higher than the timestamp resolution. Usually the limiting factor is the noise associated with the timestamps rather than the resolution.(not sure whether this work is relevant)

2.2 Synchronization in Wireless networks

Two way ranging algorithm can be used among nodes of a network to provide localization framework. However when used in a network, ranging messages between a pair of nodes can be received by other anchors as well. The network can be designed to use this redundant information in range calculations, minimizing the overall number of messages shared within a cycle. Symmetrical multi way ranging algorithm proposed in [24] is one such method. It is developed to perform ranging within a network while using a simple round robin transmission schedule as shown in Fig.2.3. Work in [24,25] illustrates numerical results of the symmetrical multi way ranging algorithm with a clock drift elimination. Two way ranging is performed twice in this two phase algorithm to calculate clock drift. Second phase being in the reverse order of the nodes. Transmission order in these two phases are illustrated in the top graphs in Fig. 2.3. However the large ranging errors resulted are expected due to the large return trip times for non adjacent nodes in the transmission order.

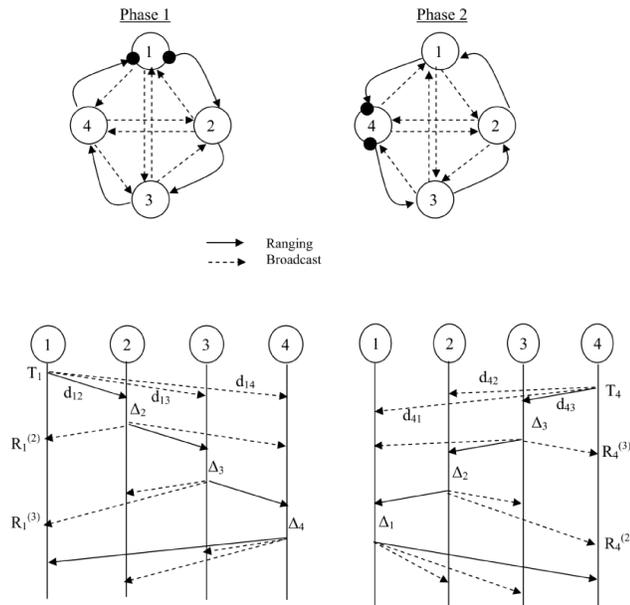


Figure 2.3: Symmetrical multi way ranging
image source: [24]

The work carried out in [3] proposes a simultaneous ranging and synchronization protocol using a centralized method. It chooses one anchor in the network as the master and all other anchors are considered slaves. Master's clock is taken as the reference, and network synchronization is achieved by pairwise synchronization with master anchor using two way ranging packets. In this work, the relation between the synchronized network clock which is same as the masters clock and a slave's clock is expressed using the parameter clock offset $\delta(t^S)$. The relation is shown below where t^M and t^s are master's and slave's clocks respectively, and $\hat{\cdot}$ indicates an estimation.

$$\hat{t}^M(t^S) = t^S - \hat{\delta}(t^S). \quad (2.1)$$

Clock offset is propagated by the clock drift parameter $\gamma(t_k^S)$, which is same as the relative clock progression rate. Here, t_k^S represent the time at the last synchronization event.

$$\hat{\delta}(t^S) \approx \hat{\delta}(t_k^S) + (t^S - t_k^S)\hat{\gamma}(t_k^S). \quad (2.2)$$

Clock offset and clock drift, are tracked using a Kalman filter, using the parameters themselves as the filter states. Clock offset calculated using the timestamps and the clock drift measured by carrier integration are used as the measurements when updating the filter.

To achieve clock synchronization in a WSN, [4] takes a distributed approach. Here, the global clock is expressed as a Taylor series of the local hardware clock values, and synchronization is achieved by estimating the offset and relative clock rate parameters. these parameters are estimated through a Kalman filter. The novelty in this work is that it controls the virtual master clock rate by a voting scheme. The overview of the process flow is shown in the Fig.2.4.

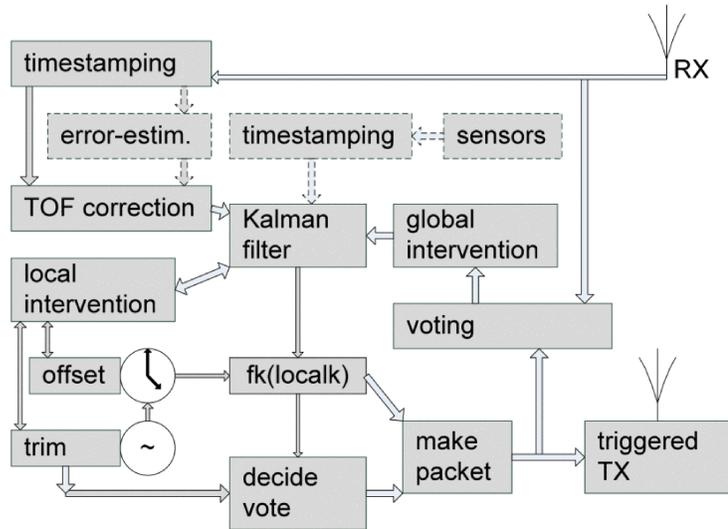


Figure 2.4: Distributed Kalman filter for precise and robust clock synchronization in wireless networks

image source: [4]

In this particular method [4], all nodes compare their hardware clock rate with the global clock rate tracked by the Kalman filter, then votes whether the global clock has to accelerate or decelerate to meet the hardware clock rate. These votes are shared throughout the network via messages. After a predetermined interval if there is a clear majority (e.g. $>2/3$) the global clock rate will be changed by a small step towards the voted direction. However the presented network operates with accuracies in the range of microseconds, using predetermined rough propagation delays. It can be observed that this work can be extended to achieve precise synchronization in UWB localization networks.

Several other implementations [7] have used Kalman filters to track the clock dynamics, using timestamps as measurements.

2.3 Gradient clock synchronization

Although there are several topological based synchronization protocols such as "Timing-Sync Protocol for Sensor Networks" (TPSN) [14] and "Flooding Time Synchronization Protocol" (FTSP) [15] proposed for wireless networks, when it comes to true distributed clock synchronization, the Gradient clock synchronization algorithms can be considered the state of the art.

The generic version of the gradient clock synchronization algorithm proposed in [2] defines absolute logical clock rate $x_i(t)$ of node i at time t as the product of hardware clock rate $h_i(t)$, and the relative logical clock rate $l_i(t)$.

$$x_i(t) = h_i(t) \cdot l_i(t) \quad (2.3)$$

This logical clock rate is updated after each transmission cycle of the network according to the following equation. where \mathcal{N}_i is the set of neighbouring anchors to i .

$$x_i(t_{k+1}) = \frac{\left(\sum_{j \in \mathcal{N}_i} x_j(t_k)\right) + x_i(t_k)}{|\mathcal{N}_i| + 1} \quad (2.4)$$

An clock offset θ_i is defined to compensate drift, and updated using a similar logic as follows.

$$\theta_i(t_{k+1}) = \theta_i(t_k) + \frac{\sum_{j \in \mathcal{N}_i} L_j(t_k) - L_i(t_k)}{|\mathcal{N}_i| + 1} \quad (2.5)$$

The convergence of rate is shown by arranging the system of update equations in the vector form as, $X(t_{k+1}) = A(t_k)X(t_k)$. here $X(t_k)$ is the vector containing all x_i values at time t_k . The authors have proven convergence by means of $A^n \rightarrow I$, as n becomes large, therefore $X(t_\infty)$ becomes steady.

Although this gradient clock synchronization algorithm guarantees convergence of logical clock rates in a neighbourhood, it pays less attention to the global logical clock

rate converging value. The convergence test guarantees only the marginal stability in the clock rate error states. The logical clock rates can settle at an arbitrary value scaling the range measurements of a synchronized network by an unknown amount. In an event of an initialization error or a participation of one inaccurate clock during the initialization, this scaling error of ranges can be significant. Due to these reasons, it is desirable to seek an algorithm whose converging values will be independent of the initial conditions and the rate will converge at the average of individual clock rates.

A different version of gradient clock synchronization is used in [26] to achieve WSN synchronization. Instead of averaging out the estimates for the logical clock rate estimates, a dynamic weighted average is taken to make the algorithm more robust. Used logical clock rate l_i update rule is as follows, where $\hat{l}_j(t_k)$ represents the estimate obtained from anchor j .

$$l_i(t_k) = l_i(t_{k-1})(1 - \rho(t_k)) + \hat{l}_j(t_k)\rho(t_k) \quad (2.6)$$

Here, the time varying weight $\rho(t_k)$ is calculated based on four cases depending on whether the logical clock is leading or lagging the estimate and whether the logical clock rate is greater or smaller than the estimate. If the logical clock of anchor i is leading and moving slower or lagging and moving faster, more weight is given towards the logical clock of anchor i and less towards the estimate. In other two cases the estimate is weighted more.

Hammer and D'Andrea, in their seminal paper [1] have presented a time difference of arrival based UWB localization network. It uses a Kalman filter to track the relative clock dynamics and uses gradient clock synchronization algorithm as proposed in [2] to

synchronise individual anchor clocks. The Kalman filter used, models hardware clock as a 3rd order linear system and incorporates transmission and reception time-stamps of inbound messages as measurements.

With its ability to localize arbitrary number of tags passively, the proposed method is ideally suitable for providing infrastructure for robot Swarm localization. While this has proven accurate results, filter convergence time can be improved by incorporating the clock offset ratio as an additional measurement in the relative clock filter. Furthermore, This work uses two way ranging over the transmission schedule to estimate inter anchor ranges in the self calibrating procedure. Two way ranging calculation uses direct time-stamps which makes the time of flight independent from the Kalman filter. Low pass filtering proposed to smooth out noise associated with the time of flight is sub-optimal for non stationary anchors, while a Kalman filter based time of flight estimation can converge much faster and perform better even with anchor movements. Synchronization of clocks between moving nodes can be helpful in many WSN applications.

Chapter 3

Relative clock filtering and Range Estimation

This chapter presents the novel Kalman filter used to track relative clock dynamics and range estimation. First, clock dynamics analysis is performed to identify the model, then filter is developed to satisfy the observability conditions while utilizing the available measurements.

3.1 Introduction

Basic two way ranging calculations can be performed using transmit and reception timestamps of inbound and outbound messages, and the error due to clock offset can be corrected using relative clock rate measurements. Yet, the noise associated with timestamps directly effects the calculation in this method. Furthermore, available measurements for relative clock rate are rather noisy and cannot be used to achieve precise two way range calculations as shown in Fig. 3.6(b).

In order to get better estimations for the timestamps and relative clock rate from

noisy data, a novel Kalman filter is proposed. This allows to track remote clocks over a longer periods of time [1] and also allows to correct the error in long reply delays in two way ranging, facilitating two way ranging following a transmission schedule. This method will reduce the power consumption of active ranging to a minimum.

3.2 Hardware clock analysis

Each UWB anchor processes a hardware clock, of which exact value can be read at an event. To be particular, events we are interested here are transmission and reception events. Since these clocks does not progress at the same exact rate we cannot use these to measure longer periods of time to the level of accuracy required in ranging applications. Usually when a clock is powered on i.e. at a cold start the clock rate is few ppms lower than the steady state value and will increase gradually as the device warms up. Even the steady state clock rate can depend on the environmental factors such as ambient temperature and can vary from device to device. More information about device variation and other factors affecting the clock rate variation can be found in [16].

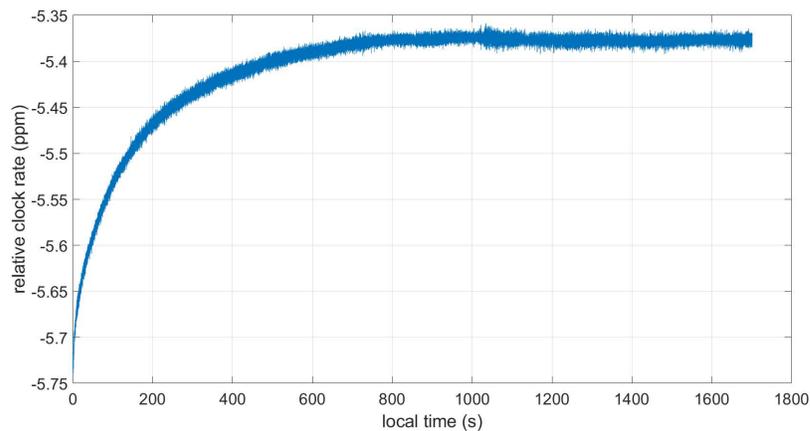


Figure 3.1: Relative hardware clock rate

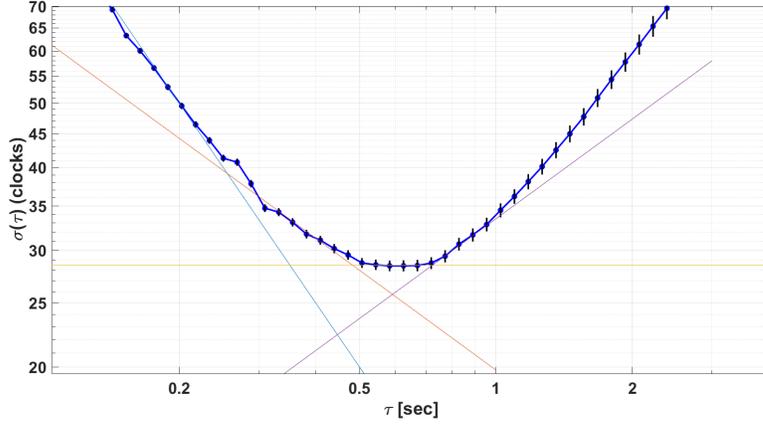


Figure 3.2: Allan Deviation plot

Quantization noise is found to be 5.8clocks , phase random walk noise is $19.8\text{clocks}/\sqrt{s}$, clock phase instability is found to be $28.5\text{clocks}/s$, and rate random walk noise is $58\text{clocks}/s/\sqrt{s}$.

To understand the dynamics of clocks we collected a set of relative clock measurements using two stationary anchors transmitting periodically. Value of the remote hardware clock at remote transmission event is recorded against the local clock value at the local reception event. Since the anchors are stationary, both clock values are considered as from the same event neglecting the constant offset due to propagation delay. It should be noted that there is no acceptable method to record absolute time of an event without having any absolute time reference. Hence relative measurements were taken. Relative clock rate calculated from logged data over a span of 30 minutes are shown in Fig.3.1. In order to achieve relatively a steady reference time, the logging anchor was kept turned on for several minutes before turning the remote anchor on. The clock progression rate here is calculated using the timestamps. hence we can observe that the calculated clock rate has been affected by the timestamp noise. To further investigate the noise characteristics at different frequencies, an Allen deviation plot is created using the same logged data. Packages provided in [27] are

used in this evaluation. This Allan plot gives valuable quantitative information about different noise parameters of derivatives and justifies the 3rd order linear clock model found in many literature. Blue, orange, yellow and purple tangent lines in Fig.3.2 represent f^{-1}, f^{-2}, f^{-3} and f^{-4} power noises of relative clock phase.

3.3 Relative Clock Dynamics

As proposed in many literature, The hardware clock model used is a third order linear model. When modelling, each anchor clock value is considered as a sample of a continuous time process t . Sampled value at event ϵ of anchor I 's clock is represented as $t_I[\epsilon]$. When representing a general event, the notation $[\epsilon]$ is sometimes omitted for simplicity. Since we only observe relative measurements, associated notation is introduced as follows.

The relative clock rate of anchor J with respect to anchor I is expressed as,

$$\dot{t}_J^I[\epsilon] = \frac{dt_J}{dt_I}[\epsilon] \quad (3.1)$$

and the relative clock acceleration as,

$$\ddot{t}_J^I[\epsilon] = \frac{d^2t_J}{dt_I^2}[\epsilon] \quad (3.2)$$

then the jerk as,

$$\ddot{\dot{t}}_J^I[\epsilon] = \frac{d^3t_J}{dt_I^3}[\epsilon] \quad (3.3)$$

Since the relative clock dynamics system is composed of two linear hardware clocks and the derivative terms are quite small, we can consider the relative time (phase) between anchor clocks as a third order linear system with the third derivative of the relative time is considered to be driven by noise.

$$\ddot{t}_J^I[\epsilon] = \nu_c(dt) \quad (3.4)$$

where $\nu_c(dt) \sim N(0, \sigma_c^2 dt)$ is the Gaussian noise driving the system. [1]

The relative time (phase) is expected to propagate between two events ϵ_k and ϵ_{k+1} according to the following equation.

$$t_J^I[\epsilon_{k+1}] = t_J^I[\epsilon_k] + \dot{t}_J^I[\epsilon_k] dt^I + \frac{1}{2} \ddot{t}_J^I[\epsilon_k] dt^{I^2} + \nu_c(dt^I) \quad (3.5)$$

where,

$$dt^I = t_I[\epsilon_{k+1}] - t_I[\epsilon_k] \quad (3.6)$$

A transmission event is denoted as tx and a reception event is denoted as rx . The event of transmitting k^{th} message by anchor J is denoted as tx_k^J and an event of receiving that message by anchor I is denoted as rx_k^{IJ} . To clear out the notations used henceforth, a transmission and a reply between a pair of anchors is expressed in the following timing graph Fig.3.3.

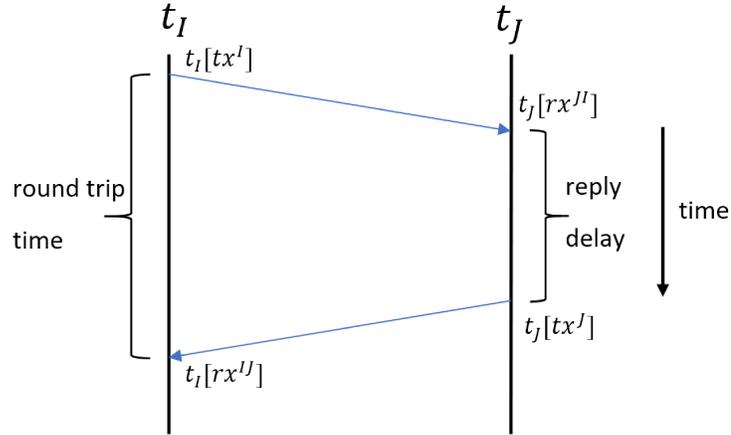


Figure 3.3: Two way ranging

The delayed transmission mode in DWM1000 allows anchor I to write the transmit timestamp $t_I[tx_k^I]$ to the outbound message data before the actual transmission is performed. Anchor J embeds the reception timestamp of that message $t_J[rx_k^J]$, and the transmission timestamp of the reply $t_J[tx_k^J]$ in the reply message data. Anchor I can use these data to track the hardware clock of anchor J .

3.4 State space model

With the assumption that there is underlying exact dynamics driving the clock phase, a discrete domain state space model is proposed to simulate the assumed relative clock dynamics between remote anchors. Here, the local clock is considered absolute and the progression of remote anchor's clock is tracked relative to the local clock. To track the variations smoothly, remote anchor J 's clock value (phase) and its first two derivatives with respect to an anchor I 's own clock are chosen to be the filter states. Since time of flight is directly coupled with the estimated clock value, it is also included as a filter state. Here superscript I in the time of flight δ_{IJ}^I , represents that it is as measured relative to anchor I 's clock.

$$X_J^I = \begin{bmatrix} t_J^I[t] \\ \dot{t}_J^I[t] \\ \ddot{t}_J^I[t] \\ \delta_{IJ}^I \end{bmatrix} \quad (3.7)$$

Time of flight is assumed to be noise-driven since no higher derivatives of position or ranges are tracked in this work.

$$\dot{\delta}_{IJ}^I = \nu_\delta(dt) \quad (3.8)$$

where $\nu_\delta(dt) \sim N(0, \sigma_\delta^2 dt)$ is a hypothetical noise associated with anchor movement. It is assumed that there is no correlation between the ν_c and ν_δ . Since the second and third states are by definition first and second derivatives of the first state, the discrete domain system matrix turns as follows. We do not need to keep track of the third derivative since it is considered purely noise.

$$\Phi = \begin{bmatrix} 1 & dt & \frac{1}{2}dt^2 & 0 \\ 0 & 1 & dt & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3.9)$$

Complete discrete domain system model can be written as:

$$X_J^I[t_{k+1}] = \Phi(dt)X_J^I[t_k] + Q\nu \quad (3.10)$$

where,

$$\nu = \begin{bmatrix} \nu_c[t_{k+1}] \\ \nu_\delta[t_{k+1}] \end{bmatrix} \quad (3.11)$$

3.5 Measurement model

We can track the variation of a remote clock by just listening to periodically transmitted messages. Although, in addition to the transmit timestamps, we have access to the clock offset ratio as an additional measurement. We would assume these measurements are corrupted by an additive noise. We can express the measurements obtained from inbound messages as follows.

$$Y_{1k} = \begin{bmatrix} y_{1k} \\ y_{2k} \end{bmatrix} = \begin{bmatrix} t_J[t x_k^J] \\ \dot{t}_J^I[r x_k^{IJ}] \end{bmatrix} + \eta \quad (3.12)$$

$$\eta = \begin{bmatrix} \eta_t \\ \eta_r \end{bmatrix} \quad (3.13)$$

where η_t and η_r are the measurement noise associated with timestamp and clock offset ratio, which are assumed to be Gaussian distributed.

Since these measurements are available at the local reception event, we shall discretize the state space system at this event. We need to express the measurements in terms of tracked filter states at the discretized event. For this, we can use the following relationship.

$$t_J[tx_k^J] = t_J[rx_k^{IJ}] - \delta_{JI}^J \quad (3.14)$$

We can neglect the variation of clock rates between transmission and reception events ($\dot{t}_J[tx_k^J] \approx \dot{t}_J[rx_k^{IJ}]$) as this dynamics is very slow compared to the propagation delay. Also neglecting the difference propagation delays measured with respect to two clocks δ_{JI}^I and δ_{JI}^J , we can derive the corresponding output equations is as follows.

$$Y_{1k} = H_1 X_J^I[rx_k^{IJ}] \quad (3.15)$$

$$H_1 = \begin{bmatrix} 1 & 0 & 0 & -1 \\ 0 & 1 & 0 & 0 \end{bmatrix} \quad (3.16)$$

The observably matrix shown below, which has a rank 3, confirms that these measurements are sufficient to estimate the first three states.

$$Obsv\left(F(1:3), H(1:3)\right) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & dt & dt^2 \\ 0 & 0 & dt \\ 0 & 0 & dt^2 \\ 0 & 0 & 0 \end{bmatrix} \quad (3.17)$$

Yet, we can see the fourth state, which is time of flight appears in the measurement equation. To use the correct measurement model, an estimation for time of flight is required. We can use two way ranging calculations to obtain this estimation or use a constant offset if true phase is not important. But considering the time of flight as a state captures the true system behaviour without modelling errors. Using same measurements for the full state space yields following observability matrix which has rank 3. Time of flight state is not observable here.

$$Obsv\left(F, H\right) = \begin{bmatrix} 1 & 0 & 0 & -1 \\ 0 & 1 & 0 & 0 \\ 0 & dt & dt^2 & 0 \\ 0 & 0 & dt & 0 \\ 0 & 0 & dt^2 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \quad (3.18)$$

Since time of flight is not observable with only one way transmission time-stamps, additional two measurements are obtained using the time-stamps of the outbound messages. Reception time-stamp of the outbound messages (y_3) and the clock offset ratio at the reception (y_4) are shared through the reply message transmitted by the remote anchor.

$$Y_{2k} = \begin{bmatrix} y_{3k} \\ y_{4k} \end{bmatrix} = \begin{bmatrix} t_J[rx_k^{JJ}] \\ t_I^J[rx_k^{JJ}]^{-1} \end{bmatrix} + \eta \quad (3.19)$$

This time, the system is discretized at the local transmission event, gives us the following relationship.

$$t_J[rx_k^{JJ}] = t_J[tx_k^I] + \delta_{IJ}^J \quad (3.20)$$

Similarly, corresponding output equation is as follows.

$$Y_{2k} = H_2 X_J^I[tx_k^I] \quad (3.21)$$

$$H_2 = \begin{bmatrix} 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 \end{bmatrix} \quad (3.22)$$

Timestamp measurements in Y_1 and Y_2 make the time of flight observable. Following observability matrix is derived for the full state space using a concatenated H matrix removing the redundant row. and it has rank 4.

$$Obsv\left(F, [H_1; H_2]\right) = \begin{bmatrix} 1 & 0 & 0 & -1 \\ 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 \\ 0 & dt & dt^2 & 0 \\ 0 & dt & dt^2 & 0 \\ 0 & 0 & dt & 0 \\ 0 & 0 & dt^2 & 0 \\ 0 & 0 & dt^2 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \quad (3.23)$$

3.6 Kalman filter

A Kalman filter is a perfect candidate to track the propagation of this one dimensional linear model of the relative clock dynamics. The advantages of using a Kalman filter are:

- It gives the optimal estimate for the assumed noise model.
- All available measurements can be utilized effectively.
- Smooth variations in estimates with reduced noise.

Usually, transmission and reception timestamps alone are sufficient to update the relative clock tracking filter and we see several published work have achieved good results with it [1]. Although, having clock offset ratio calculated by carrier integration [11] available as a measurement, we incorporated it to the filter expecting faster convergence. Fig. 3.5 shows the improvement of the filter convergence time when this measurement is used.

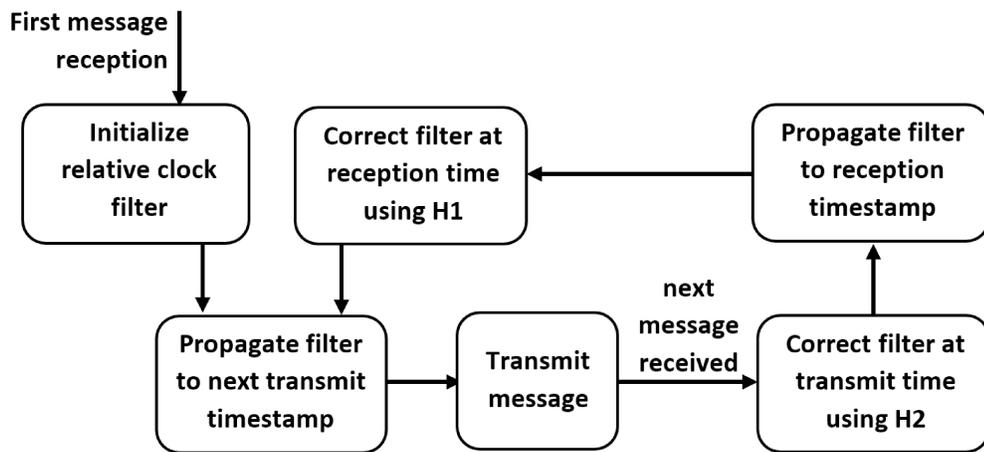


Figure 3.4: Filter update flow

Since measurements are only available at local transmission and reception events, the system is discretized at these events. Standard Kalman filter equations are used to propagate states and apply measurement updates at discretized events. Once a transmission time is computed by adding the transmission interval to the most recent reception event, states are propagated to the local transmission event. This is performed before the transmission since current filter states are required to calculate message data. Then When the reply message is received, a correction is applied to the filter using H_2 and y_3, y_4 measurements, which are remote reception timestamp and relative rate measured at the remote reception. These are communicated back through the reply message. Then the states are propagated to the local reception event by time interval dt_1^I .

$$dt_1^I = t_I[rx_k^{IJ}] - t_I[tx_k^I] \quad (3.24)$$

Then the filter is corrected by H_1 and measurements corresponding to the remote transmission and local reception. When the next transmission is scheduled, the states are propagated to the next transmission event tx_{k+1}^I by time interval dt_2^I ,

$$dt_2^I = t_I[tx_{k+1}^I] - t_I[rx_k^{IJ}] \quad (3.25)$$

Kalman filter algorithm used to update the filter can be summarized as follows.

Algorithm 3.1 Relative clock tracking and ranging filter

```

1: Initialize filter states and state covariance matrix:
2:    $t_I[rx_0^{IJ}] = t_0$ 
3:    $X_J^I[rx_0^{IJ}] = X_0$ 
4:    $P_J^I[rx_0^{IJ}] = P_0$ 
5: for  $k = 1, \dots, \infty$  do
6:   Propagate to next local transmit event:
7:      $dt_{1k}^I = t_I[tx_k^I] - t_I[rx_{k-1}^{IJ}]$ 
8:      $X_J^I[tx_k^I]^- = \Phi(dt_{1k}^I)X_J^I[rx_{k-1}^{IJ}]$ 
9:      $P_J^I[tx_k^I]^- = \Phi(dt_{1k}^I)P_J^I[rx_{k-1}^{IJ}]\Phi(dt_{1k}^I)' + Q\nu(dt_{1k}^I)$ 
10:  wait until a reply or timeout
11:  if reply received then
12:    Measurement update at local transmit event:
13:     $E_{1k} = Y_{1k} - H_1X_J^I[tx_k^I]^-$ 
14:     $K_{1k} = P_J^I[tx_k^I]^-H_1'(H_1P_J^I[tx_k^I]^-H_1' + R)^{-1}$ 
15:     $X_J^I[tx_k^I] = X_J^I[tx_k^I]^- + K_{1k}E_{1k}$ 
16:     $P_J^I[tx_k^I] = (I - K_{1k}H_1)P_J^I[tx_k^I]^-$ 
17:    Propagate to reception event
18:     $dt_{2k}^I = t_I[rx_k^{IJ}] - t_I[tx_k^I]$ 
19:     $X_J^I[rx_k^{IJ}]^- = \Phi(dt_{2k}^I)X_J^I[tx_k^I]$ 
20:     $P_J^I[rx_k^{IJ}]^- = \Phi(dt_{2k}^I)P_J^I[tx_k^I]\Phi(dt_{2k}^I)' + Q\nu(dt_{2k}^I)$ 
21:    Measurement update at local reception event:
22:     $E_{2k} = Y_{2k} - H_2X_J^I[rx_k^{IJ}]^-$ 
23:     $K_{2k} = P_J^I[rx_k^{IJ}]^-H_2'(H_2P_J^I[rx_k^{IJ}]^-H_2' + R)^{-1}$ 
24:     $X_J^I[rx_k^{IJ}] = X_J^I[rx_k^{IJ}]^- + K_{2k}E_{2k}$ 
25:     $P_J^I[rx_k^{IJ}] = (I - K_{2k}H_2)P_J^I[rx_k^{IJ}]^-$ 
26:  else
27:    Set variables for next transmission
28:     $t_I[rx_k^{IJ}] = t_I[tx_k^I]$ 
29:     $X_J^I[rx_k^{IJ}] = X_J^I[tx_k^I]^-$ 
30:     $P_J^I[rx_k^{IJ}] = P_J^I[tx_k^I]^-$ 
31:  end if
32: end for

```

3.7 Process noise model

Although it is desirable to transmit at a fixed delay after a reception, DWM1000 discards the least significant 9 bits of the transmission time in delayed transmit mode. Because of this, transmission period cannot be kept constant. Furthermore, there is no guarantee that all the transmitted packets will be successfully received by all other anchors. To handle this variation in the sampling time, a time-dependent process noise matrix (Q) is defined similar to what's suggested in [1]. By making the filter sampling time-dependent, it will become robust against packet losses and irregularities in transmission schedule [1].

$$Q\nu = \begin{bmatrix} \frac{\sigma_c^2}{20} dt^5 & \frac{\sigma_c^2}{8} dt^4 & \frac{\sigma_c^2}{6} dt^3 & 0 \\ \frac{\sigma_c^2}{8} dt^4 & \frac{\sigma_c^2}{3} dt^3 & \frac{\sigma_c^2}{2} dt^2 & 0 \\ \frac{\sigma_c^2}{6} dt^3 & \frac{\sigma_c^2}{2} dt^2 & \sigma_c^2 dt & 0 \\ 0 & 0 & 0 & \sigma_s^2 dt \end{bmatrix} \quad (3.26)$$

3.8 Implementation and Validation

Since the time-stamps from the DWM1000 module are 40-bit unsigned integers, they wrap around in about 17s. Due to this, the difference between the current and last update events and reception time-stamps are used to update the filter. This eliminated the need to handle a reception time-stamp wrap event explicitly since the difference between two unsigned integers is robust to wrap around. Furthermore, to avoid relative clock phase state ($X(0)$) accumulating time differences and causing numerical instabilities, and to handle variable overflowing, integer part of the $X(0)$ is cleared and stored separately after each filter update. This eliminated the need of integrating dt and maintaining a continuous time scale.

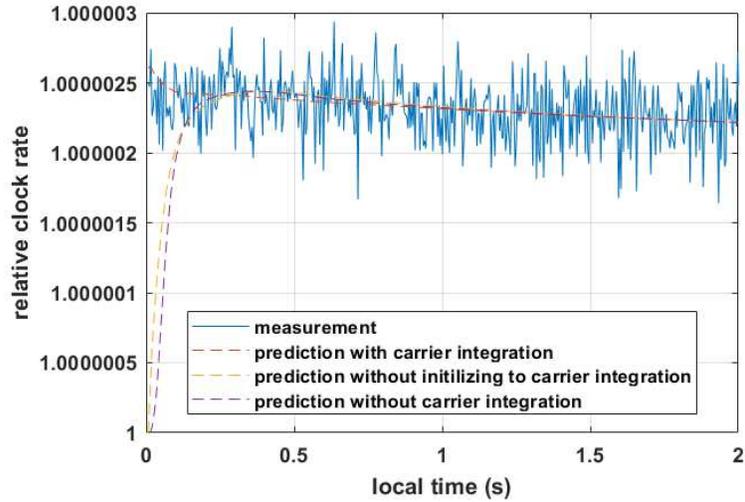
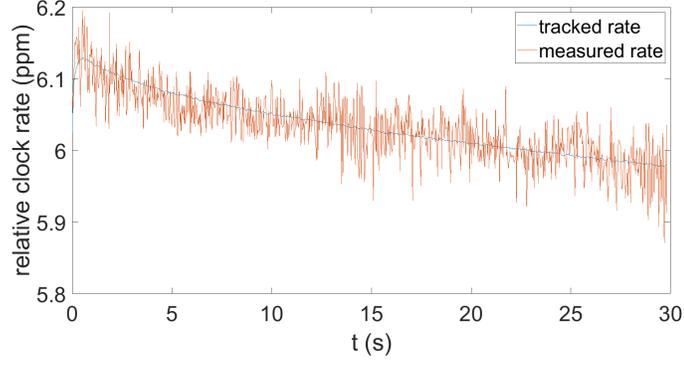


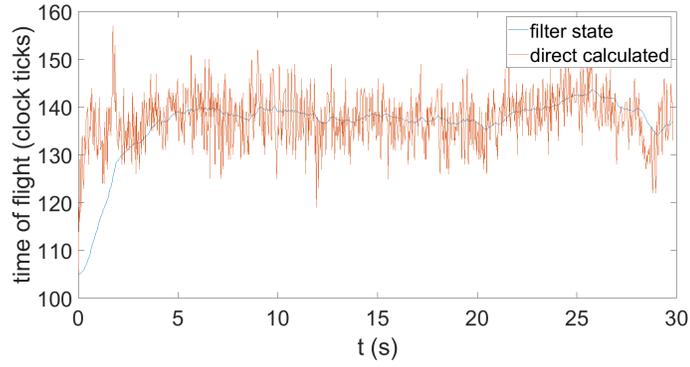
Figure 3.5: Relative clock rate convergence comparison

To justify the use of clock offset ratio measurement in relative clock tracking filter, offline tracked relative clock rate convergence plots with and without the measurement are shown in Fig.3.5, for comparison, relative clock rate estimations without initializing to the measured rate is also included. To visualize the difference clearly, only the first two seconds are included in the plot. It can be seen that the convergence time of tracked relative clock rate is significantly improved by incorporating the clock offset ratio measurement. Although all estimates converge to the same value within few seconds, in low power applications where an anchor sleeps between transmissions this convergence time improvement is significant.

Real-time tracking performance of the filter was evaluated by logging filter states and measurements for a stationary anchor pair. Noise figures were tuned to achieve a smooth relative clock rate that would correctly track the noisy clock offset ratio measurement. Data was logged at 50Hz and the variation of the filter states with measurements are shown in Fig.3.6. When collecting data, the logging anchor (I) was turned on for few minutes in order to make its clock stable, and then the remote



(a) Relative clock rate



(b) Time of flight

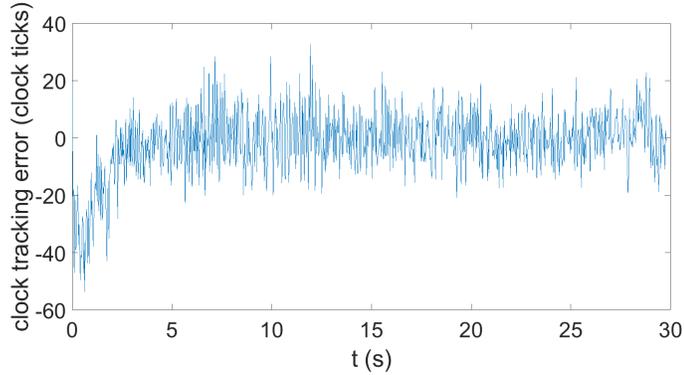
Figure 3.6: Filter states

anchor (J) whose clock is tracked by the logging anchor was turned on. Since anchor J is at a cold start, initially its clock rate error is high, and as time progresses it can be observed that the clock rate becomes stable as in Fig. 3.6(a).

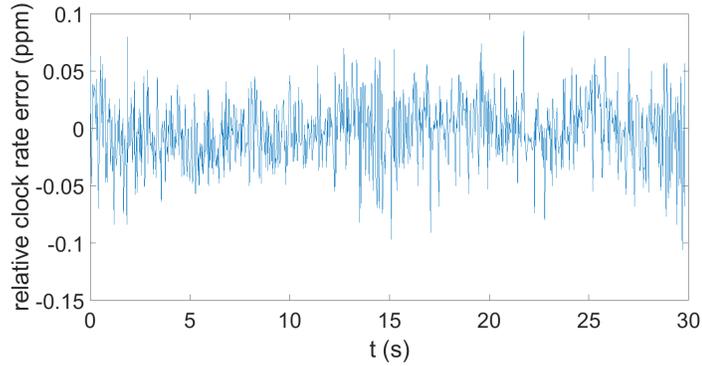
For comparison, the best estimate that can be achieved for the time of flight without filtering, has been plotted with tracked time of flight state in Fig. 3.6(b). It should be noted that this calculated unfiltered time of flight value (tof_{UF}) has not been used as a filter measurement rather calculated as follows. It should be noted that the \dot{t}_J^I value used here is the clock offset ratio calculated using carrier integration.

$$tof_{UF} = \frac{1}{2}((t_I[rx_k^{IJ}] - t_I[tx_{k-1}^I]) - \dot{t}_J^I(t_J[tx_k^J] - t_J[rx_{k-1}^{JI}])) \quad (3.27)$$

Kalman filter cloak rate measurement errors and direct calculated time of flight errors compared to the filter states are shown in Fig. 3.7



(a) Direct calculated time of flight error compared to filter



(b) Clock rate measurement error in filter

Figure 3.7: Measurement errors

Measurement error standard deviation of relative clock rate was found to be 0.0281ppm, and the standard deviation of the direct calculated time of flight compared to the filter was found to be around 11 clocks in this experiment, which converts to a distance error of 5.2cm.

Comparing the noisy direct calculated corrected two way ranging measurement with the filter state, in Fig. 3.6(b), it is apparent that the filter has reduced the range noise efficiently while tracking the low frequency variations. It is important to note that the time of flight noise covariance (σ_δ) can be further tuned to match the dynamics of the tracking agent, to achieve stable progression at the expense of the response time or vice versa.

Chapter 4

Global Clock Synchronization

This chapter first presents the generic gradient clock synchronization algorithm and its limitations. Then we will discuss the methodology followed to apply corrections to the identified issue. Then an analysis of the algorithm with a stability proof. followed by results.

4.1 Gradient clock synchronization

The ability of Decawave UWB modules to transmit a message at a given specific time allows to process information associated with the transmission timestamp and embed in transmit message before transmission is performed. This allows to calculate a global logical transmit timestamp corresponding to the transmission event and transmit message with that information allowing synchronized transmissions for time difference of arrival (TDOA) based localization.

With the tracked relative clock rate, a global logical clock is defined so that all anchors can agree on it. Gradient clock synchronization algorithm estimates this global clock by a distributed gradient descent manner. The relation between each anchor's hardware clock to the global logical clock is expressed as a second order Taylor series.

$$l^I(t_I) = t_0^I + d_1^I(t_I - t_0^I) + d_2^I(t_I - t_0^I)^2 \quad (4.1)$$

where, $l^I(t_I)$ is the global logical time estimated by anchor I . Parameters t_0^I , d_1^I and d_2^I are global time at the synchronization event, first derivative and 2 times the second derivative of global time with respect to anchor I 's clock. The synchronization is achieved by estimating these parameters. To derive equations for parameter estimates, we can equate the global time at a particular event computed by two synchronized clocks.

$$t_0^I + d_1^I(t_I - t_0^I) + d_2^I(t_I - t_0^I)^2 = t_0^J + d_1^J(t_J - t_0^J) + d_2^J(t_J - t_0^J)^2 \quad (4.2)$$

Partially differentiating this once and twice with respect to anchor I 's time yields another two equations.

$$d_1^I + 2d_2^I(t_I - t_0^I) = d_1^J \frac{\partial t_J}{\partial t_I} + 2d_2^J(t_J - t_0^J) \frac{\partial t_J}{\partial t_I} \quad (4.3)$$

$$2d_2^I = d_1^J \frac{\partial^2 t_J}{\partial t_I^2} + 2d_2^J(t_J - t_0^J) \frac{\partial^2 t_J}{\partial t_I^2} + 2d_2^J \left(\frac{\partial t_J}{\partial t_I} \right)^2 \quad (4.4)$$

From these equations we can get estimations for anchor I 's parameters using anchor J 's parameters and relative clock filter states. It should be noted that at the time of synchronization, t_I is equal to t_0^I . Hence the higher derivative terms on the left side becomes zero.

$$\hat{t}_{0_{k+1}}^I = t_0^J + d_1^J(t_J - t_0^J) + d_2^J(t_J - t_0^J)^2 \quad (4.5)$$

$$\hat{d}_{1_{k+1}}^I = d_1^J \frac{\partial t_J}{\partial t_I} + 2d_2^J(t_J - t_0^J) \frac{\partial t_J}{\partial t_I} \quad (4.6)$$

$$\hat{d}_{2_{k+1}}^I = \frac{1}{2} d_1^J \frac{\partial^2 t_J}{\partial t_I^2} + d_2^J(t_J - t_0^J) \frac{\partial^2 t_J}{\partial t_I^2} + d_2^J \left(\frac{\partial t_J}{\partial t_I} \right)^2 \quad (4.7)$$

When using these estimates to update synchronization parameters it was found that the second order terms contributes marginally on the improvement with very small values of d_2 in the range of $10^{-20}s^{-1}$. Also the values of d_2 were oscillating rapidly without converging. This lead us to believe d_2 values were fitted to the noise of the filter estimates. Hence the proposed implementation was reduced to the first order approximation. By setting d_2 to zero, we obtain following estimate equations.

$$\hat{t}_0^J = t_0^J + d_1^J(t_J - t_0^J) \quad (4.8)$$

$$d_1^{\hat{I}J} = d_1^J \frac{\partial t_J}{\partial t_I} \quad (4.9)$$

Here the additional superscript J in the estimate indicate that the it is calculated using anchor J 's parameters.

The update rule is then established to correct t_0^I and d_1^I based on the estimations made using the parameters of remote anchor J . Although the parameterization is slightly different, the method is similar to that in gradient clock synchronization algorithm [2]. The parameterization step has made the algorithm simpler since it does not require explicit overflow handling which is needed in the case of [1,2,26]. The generic gradient clock synchronization update rule is given below which is similar to whats suggested in [2].

$$t_{0k+1}^I = t_{0k}^I + \frac{\sum(t_{0k}^{\hat{I}J} - t_{0k}^I)}{n+1} \quad (4.10)$$

$$d_{1k+1}^I = d_{1k}^I + \frac{\sum(d_{1k}^{\hat{I}J} - d_{1k}^I)}{n+1} \quad (4.11)$$

4.2 Chaotic clock rate correction

It can be observed that these update rules causes chaotic behaviour during clock rate convergence. Although the global rates calculated by each anchor using (18), has shown to be converging, the row stochasticity test used in [2], can guarantee only marginal stability. Since the global clock rate does not have any real tie to the physical clock rates, the global clock rate can be scaled by an arbitrary amount. Since there is no absolute time reference in the network, we do not have a one true target for the for the global clock rate to converge. Each anchors hardware clock is not consistent from the beginning and can have trim errors that leads to different steady state clock rates. However, assuming this inter device variation to be random, we can have a more stable target by averaging out all the hardware cloak rates.

In order to make the global logical clock rate converging to the average of the rates of individual anchors, a correction term has been added to the proposed update rule for the term d_1^I . To derive this, first we will define the global cloak rate (\dot{i}) to be the average of individual clock rates.

$$\dot{i} = \frac{\sum \dot{t}^I}{n} \quad (4.12)$$

desired rate synchronization parameter for each anchor can now be expressed as,

$$\begin{aligned} d_1^I &= \frac{\dot{i}}{\dot{t}^I} \approx i - (t^I - 1) \\ \dot{t}^I &\approx i - (d_1^I - 1) \end{aligned} \quad (4.13)$$

Using the eqn. 4.2 we can obtain the following constraint.

$$\begin{aligned} i &= \frac{\sum (i - (d_1^I - 1))}{n} \\ ni &= \sum i - \sum (d_1^I - 1) \\ \sum (d_1^I - 1) &= 0 \end{aligned} \quad (4.14)$$

This constraint allows to calculate another estimate for the d_1^I parameter as shown below.

$$\begin{aligned} \sum (d_{1k}^I - 1) &= 0 \\ \hat{d}_{1k}^I &= 1 - \sum_{J \neq I} (d_{1k}^J - 1) \end{aligned} \quad (4.15)$$

By combining this estimate with a weight K we obtain the following update rule.

$$d_{1k+1}^I = d_{1k}^I + \frac{\sum (d_{1k}^J - d_{1k}^I)}{n+1} + \frac{K}{n+1} \left(1 - d_{1k}^I - \sum_{J \neq I} (d_{1k}^J - 1) \right) \quad (4.16)$$

4.3 Stability analysis

To further investigate this rule, assuming time invariant absolute individual clock rates we define an error state as the difference between each estimated global clock rate parameter and the true value. Error in a rate synchronization parameter can be calculated as,

$$\begin{aligned} \Delta d_1^I &= \hat{d}_1^I - d_1^I \\ \hat{d}_1^I &= d_1^I + \Delta d_1^I \end{aligned} \quad (4.17)$$

Here, d_1^I represent the true non varying value and \hat{d}_1^I represents the estimated values collectively by the network. from the original update rule, the estimates can be expressed as, 3

$$\begin{aligned} \hat{d}_1^{IJ} &= d_1^J \frac{\partial t_J}{\partial t_I} \\ d_1^I + \Delta \hat{d}_1^{IJ} &= (d_1^J + \Delta d_1^J) \frac{d_1^I}{d_1^J} \\ \Delta \hat{d}_1^{IJ} &= \Delta d_1^J \frac{d_1^I}{d_1^J} \end{aligned} \quad (4.18)$$

$$\begin{aligned}
d_{1k+1}^I &= d_{1k}^I + \frac{\sum(d_{1k}^{\hat{I}^J} - d_{1k}^I)}{n+1} \\
d_1^I + \Delta d_{1k+1}^I &= d_1^I + \Delta d_{1k}^I + \frac{\sum\left(d_1^I + \Delta \hat{d}_{1k}^{\hat{I}^J} - (d_1^I + \Delta d_{1k}^I)\right)}{n+1} \\
\Delta d_{1k+1}^I &= \Delta d_{1k}^I + \frac{\sum(\Delta \hat{d}_{1k}^{\hat{I}^J} - \Delta d_{1k}^I)}{n+1} \\
\Delta d_{1k+1}^I &= \frac{\sum(\Delta \hat{d}_{1k}^{\hat{I}^J}) + \Delta d_{1k}^I}{n+1} \tag{4.19} \\
\Delta d_{1k+1}^I &= \frac{\sum(\Delta d_{1k}^J \frac{d_1^I}{d_1^J}) + \Delta d_{1k}^I}{n+1} \\
\Delta d_{1k+1}^I &= \frac{\sum(\Delta d_{1k}^J \frac{d_1^I}{d_1^J})}{n+1} \\
\Delta d_{1k+1}^I &= \sum\left(\left(\frac{1}{n+1} \frac{d_1^I}{d_1^J}\right) \Delta d_{1k}^J\right)
\end{aligned}$$

This yields the discrete domain update matrix for error states, $A1 = a_{1ij}$ where,

$$\begin{bmatrix} \vdots \\ \Delta d_{1k+1}^I \\ \vdots \end{bmatrix} = A1 \begin{bmatrix} \vdots \\ \Delta d_{1k}^I \\ \vdots \end{bmatrix} \tag{4.20}$$

$$a_{1ij} = \frac{1}{n+1} \frac{d_1^I}{d_1^J} \tag{4.21}$$

With accurate clock trims, $\frac{d_1^I}{d_1^J} \approx 1$. Then, $A1$ becomes,

$$A1 = \begin{bmatrix} \frac{1}{n+1} & \frac{1}{n+1} & \cdots & \frac{1}{n+1} \\ \frac{1}{n+1} & \frac{1}{n+1} & \cdots & \frac{1}{n+1} \\ \vdots & \vdots & & \vdots \\ \frac{1}{n+1} & \frac{1}{n+1} & \cdots & \frac{1}{n+1} \end{bmatrix} \tag{4.22}$$

discrete domain update matrix of the error state was tested for eigenvalues. The original update rule has resulted a matrix with a unity eigenvalue,

For the modified update rule, first we will analyze the correction term.

$$\begin{aligned}
\frac{K}{n+1} \left(1 - d_{1k}^I \sum_{J \neq I} (d_{1k}^J - 1) \right) &= \frac{K}{n+1} \left(1 - (d_{1k}^I + \Delta d_{1k}^I) - \sum_{J \neq I} (d_{1k}^J + \Delta d_{1k}^J - 1) \right) \\
&= \frac{K}{n+1} \left(1 - (d_{1k}^I + \Delta d_{1k}^I) - \sum_{J \neq I} (d_{1k}^J - 1) - \sum_{J \neq I} \Delta d_{1k}^J \right) \\
&= - \sum \frac{K}{n+1} \Delta d_{1k}^J
\end{aligned} \tag{4.23}$$

with the correction term, the error state update matrix becomes,

$$a_{2ij} = \frac{1}{n+1} \frac{d_1^I}{d_1^J} - \frac{K}{n+1} \tag{4.24}$$

whereas the the modified version has resulted in all eigenvalues to be within the unit circle. As an example, a fully connected network with n nodes will result in the following error state update matrices $A1$ and $A2$ for original and modified rules respectively.

$$A2 = \begin{bmatrix} \frac{1-K}{n+1} & \frac{1-K}{n+1} & \cdots & \frac{1-K}{n+1} \\ \frac{1-K}{n+1} & \frac{1-K}{n+1} & \cdots & \frac{1-K}{n+1} \\ \vdots & \vdots & & \vdots \\ \frac{1-K}{n+1} & \frac{1-K}{n+1} & \cdots & \frac{1-K}{n+1} \end{bmatrix} \tag{4.25}$$

$A1$ has only one unity eigenvalue, which is for the original update rule and $A2$ for the modified case, has one eigenvalue with the value $1 - K$. It can be seen that the additional term does not change the row sum of the rate parameter update matrix as the term will evaluate to zero for a correctly estimated global clock rate. Hence, the row stochasticity test for convergence shown in [2] is also still valid.

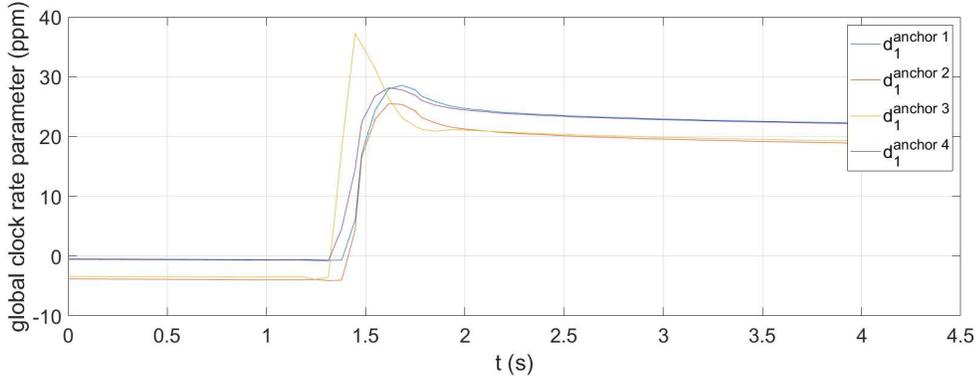
4.4 Implementation and validation

In the implementation, each anchor updates their individual parameters using the described update rules, before performing a transmission. However, having a drifting global clock phase compared to hardware clocks does not affect the ranging accuracy in any means. Since there is no particular advantage having a global time converging to the average of individual clock values, the above correction term has not been used in the t_0^I parameter update rule.

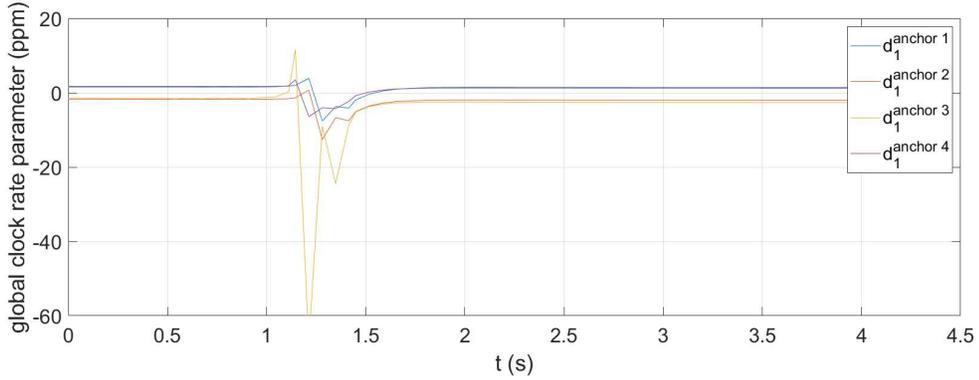
Robustness of this update rule against a disturbance in the network is tested on the hardware platform. To simulate an unstable clock, we increased the initial state covariances of the filter in one anchor. When that particular anchor joined the network, global clock rate parameters of all anchors experienced a disturbance. Settlement of these parameters can be seen in Fig. 4.1(a) and Fig. 4.1(b).

As a result of the marginal stability of the original update rule the global clock rate parameters settle at a higher error resulting a scaling in the global time progression rate. Besides the higher disturbance occurred in parameters, the modified update rule has the ability to converge the parameters back to the original values within a relatively very short period of time. This can happen in practical scenarios due to a filter instability caused by high packet loss or due to a poor hardware clock. During our experiments we experienced this type of behaviours occasionally. Authors in [1] have also reported occasional loss of synchronization during their experiments.

To evaluate the synchronization accuracy and repeatability of the proposed method, anchor network was initialized multiple times and synchronization error was recorded. After 30s from the power-up, data has been collected for a duration of one minute. In order to accommodate cold start for all clocks, experiments were conducted keeping about 10 minute down time in between. Here, the synchronization error is calculated as the difference between the global time calculated locally and the global time cal-



(a) Original update rule



(b) Modified update rule

Figure 4.1: Global clock rate parameters

culated by a remotely located anchor. This is calculated at a local transmission event and before the synchronization update. Results shown in table 4.1 indicates clear improvement as compared to the results shown in [1]. This is expected since the coupled dynamics of time of flight and timestamps are tracked better with the Kalman filter. It should be noted that the experimental network of this work uses a longer cycle time to accommodate the low power microcontroller. With a faster microcontroller it is possible to achieve a faster cycle time with even less synchronization errors. Standard deviation for all measurements was found to be 2.594 clocks, which translates into a time error of 40ns. This is about 20% better performance as compared to that is presented in [3 Fig.6].

Table 4.1: Synchronization errors

Standard deviations (clocks)	σ_{12}	σ_{13}	σ_{14}
test 1	2.258	2.562	2.295
test 2	2.480	3.122	2.646
test 3	2.482	2.659	2.073
test 4	2.439	3.549	2.378
test 5	2.093	2.922	2.052

For a better visualization of the distribution of these errors, an error histogram is presented in Fig.4.2 for the data from all the experiments. The red line represents the fitted Gaussian distribution.

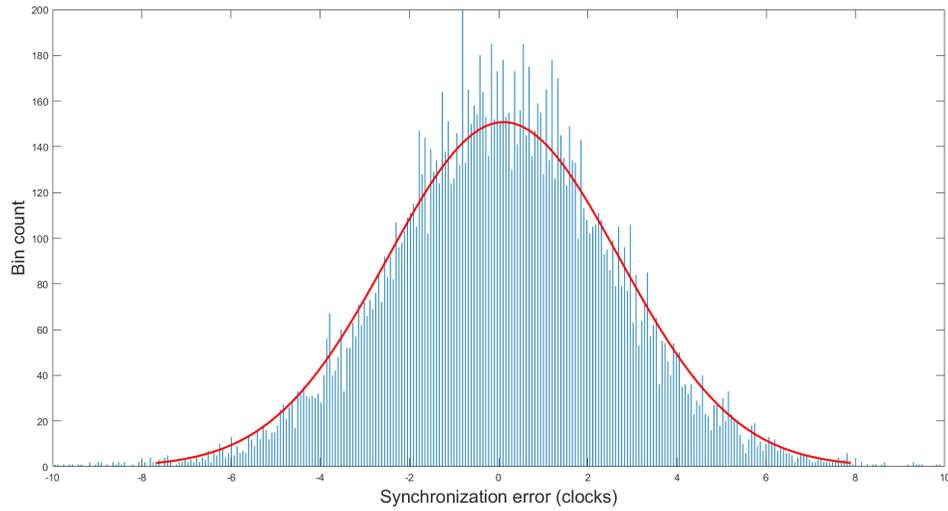


Figure 4.2: Synchronization error distribution

Chapter 5

Ranging network and localization

In this chapter, first we present the implementation level details of the network, then the proposed range based localization framework is presented. Then the ranging experiments and localization simulation are presented.

5.1 Message content

Messages used in this ranging application are composed of two parts, header and payload. The first 12 bytes of a message is encoded with header information. This header is compliant with IEEE 202.15.4 MAC standards. The content is listed as follows.

- byte 0-1: frame control.
- byte 2: message sequence number.
- byte 3-4: network ID.
- byte 5-6: destination address.
- byte 7-8: source address.

- byte 9: message type.
- byte 10-11: node count.

Message payload contains all the following information.

- anchor position x,y,x
- synchronization parameters, t_0,d_1
- local transmit timestamp
- last reception timestamps from all other anchors
- last clock offset ratio measurement from all other anchors

After the data payload, a checksum of two bytes are added by DWM1000 to ensure a successful reception. In this implementation, the network ID is programmed with firmware while device addresses are allocated dynamically in the order the nodes join the network.

5.2 Transmission schedule

When multiple anchors are operating on the same network, it is necessary that only one anchor is transmitting at a time. During a transmission, all the other anchors should be in listening mode to successfully receive the transmitted message. This is collectively achieved by agreeing upon a transmission schedule. The simplest version of a transmission schedule is dividing the transmission cycle into a number of slots which is equal to the number of nodes and allocate each slot to each node. This is known as a round robin transmission schedule since the authority to transmit is passed around among the nodes in a cyclic manner.

In order to allow easy deployment, a node should have the ability to join the network while initializing itself. A simplified version of this process is described in Algorithm 5.1 shown below.

Algorithm 5.1 Anchor initialization and transmit algorithm

```

1: Start:
2:   reset filter variables
3:   timeout = timeout long
4: for  $k = 1, \dots, \infty$  do
5:   wait until a message received or timeout
6:   if timeout then
7:     self initialize:
8:       set network id
9:       set anchor count to 1
10:      timeout = timeout short
11:  else if reply received then
12:    if network not initialized then
13:      initialize:
14:        set network id
15:        set anchor count
16:        timeout = timeout short
17:    end if
18:    if filter not initialized then
19:      initialize filter
20:    else
21:      update filter
22:    end if
23:  end if
24:  if my transmission turn then
25:    transmit:
26:      compute transmit time
27:      compute synchronization
28:      transmit
29:  end if
30: end for

```

Once an anchor is turned on, first, it will listen for incoming messages from other anchors having the same programmed network ID. If not received within a specified period, it will start transmitting, assigning itself the address "1". If received, the

anchor will assign itself a device address based on the node count which is the number of nodes already in the network, and join the network adhering to the round robin transmission schedule. It will wait for a complete transmission cycle after the first reception to initialize filters for anchors in the vicinity. At its transmission turn, it will compute its initial synchronization and start transmitting. At the end of each transmission cycle, an empty slot is allocated in order to allow new anchors joining the network to transmit. All existing network anchors will be listening during this period. If a message is received, other anchors will increase the anchor count in the network and allocate variables accordingly. Anchor count will be shared to the other anchors who are not in the vicinity of the new anchor via subsequent messages. When the number of anchors in the network is predetermined, the empty slot is removed when the anchor count reaches the limit, thus reduce the cycle time slightly.

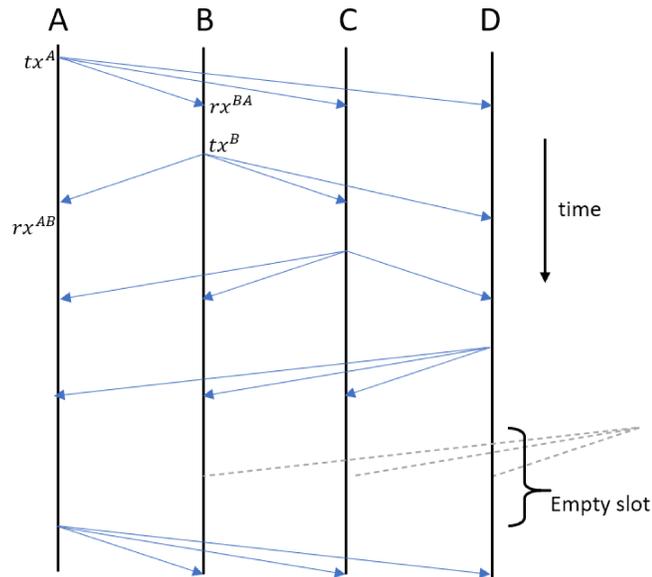


Figure 5.1: Round robin Transmission schedule timing diagram

5.3 Hardware platform

MDEK1001 Development kit module by Decawave is used as the hardware platform for this implementation. MDEK1001 module is a DWM 1001 module with header pin out and a battery housing. DWM 1001 module is the DWM1000 transceiver with a nRF52832 SOC, which has a cortex m4 microcontroller. Firmware is developed to flash the embedded microcontroller using Kiel and provided application development guide. During the initial tests, offline implementation of the filter has shown that the precision provided by single precision floating point variables is insufficient to converge the state covariance matrix. Hence double precision variables were used for the calculations in the hardware implementation. The particular hardware platform only has a single precision hardware floating point unit. Therefore, double floating point calculations are handled by the run-time libraries. Since most of the filter calculations are floating point operations, this adds substantial computational overhead to the processor, thus leads to a poor cycle time when several anchors are in the network. Default settings that were used to initialize the DWM1000 module are listed in table 5.1.

Table 5.1: Initialization parameters

Channel	2
Pulse repetition frequency	64MHz
Preamble length	128
Preamble acquisition chunk size	8
Data rate	6.8 Mb/s

5.4 Ranging experiment

Performance of range estimation of the filter for a typical robot localization scenario was experimentally evaluated by recording the estimated ranges between stationary anchors and a mobile node. As shown in Fig. 5.2, three stationary anchors were placed

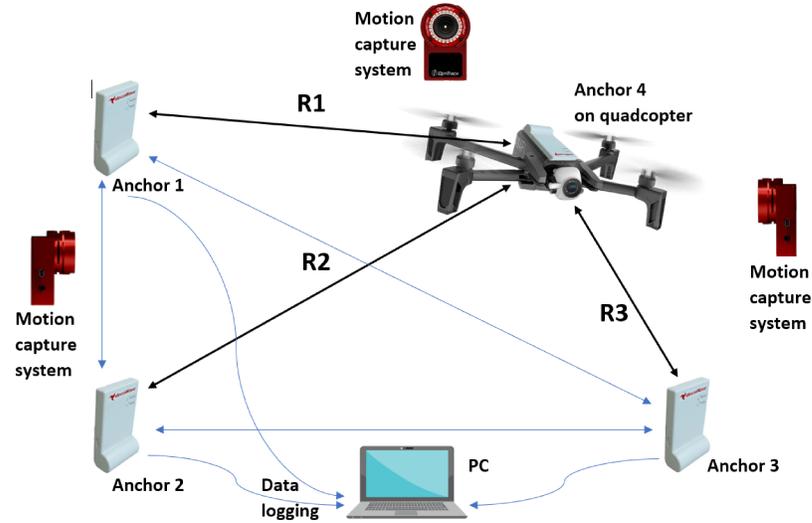


Figure 5.2: Experimental setup schematic



Figure 5.3: Experimental setup

at vertices of a 3m by 3m square, and a fourth anchor is attached to a quadcopter which is manually flown following an arbitrary flight path. Ranges calculated by the time of flight state were logged and compared with the ground truth taken from the motion capture system. For comparison, standard two way ranging measurements corrected using tracked relative clock rate, and corrected using carrier integration measurement are also plotted. Since antenna biases are now involved in the range measurements, the fixed bias component has been removed from the logged data. The range error is calculated as the difference between the measured range and ground truth obtained from motion capture system. Root mean square error (RMSE) of the

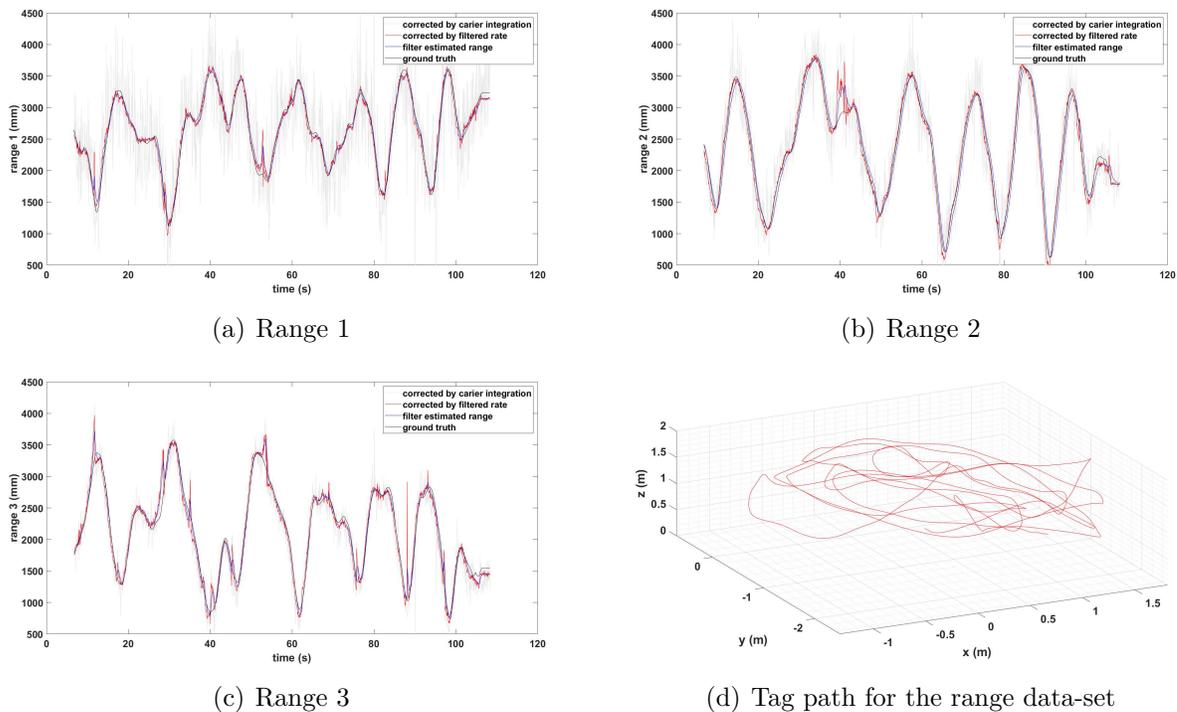


Figure 5.4: experiment plots

range tracked by the filter turned out to be around 10cm, over range measurements up to 4m. It can be clearly seen that with long return trip durations in a transmission schedule, carrier integration correction alone is insufficient as the RMS errors can be large as 1m. Estimated range data can be used to update a localization filter to track the location. For reference, the path that the quadcopter flew along, captured by the motion capture system is presented in the Fig.5.4(d).

Table 5.2: RMS Range errors

RMSE (mm)	$R1$	$R2$	$R3$
Filter estimate	68.8	115.8	94.5
Corrected with tracked relative clock rate	106.4	130.6	162.3
Corrected with carrier integration	491.5	1083.5	1358.0

5.5 Localization

Having range measurements from three or four anchors, a node can localize itself within the network coordinate frame. While having three ranges, a close form solution can be obtained for the position. However, such a solution cannot be implemented for indoor localization due to the mirror solution ambiguity and the noise associated with the ranges. Moreover, the noise associated with ranges will be amplified in an undesirable manner when the range values are converted in to the Cartesian frame.

With four or more ranges, a position estimate \hat{P} , can be obtained through minimizing the error between the measured and estimated ranges. This can be achieved by minimizing the following cost function.

$$C = \sum_{i=0}^n \left((x - x_i)^2 + (y - y_i)^2 + (z - z_i)^2 - R_i^2 \right) \quad (5.1)$$

$$\hat{P} = \text{argmin}(C, [xyz]) \quad (5.2)$$

Here, $P = [x, y, z]$ are the coordinates of the mobile node and x_i, y_i, z_i are the known coordinates of stationary anchors. Iterative least square methods and maximum likelihood methods are widely used to estimate the position with a known initial estimate [12, 28]. However, for indoor robot localization, better estimates can be achieved through IMU assisted filters. We demonstrate the position estimation using this method with the ranges recorded in the above experiment. This filter was implemented offline using simulated acceleration measurements because we couldn't conduct further experiments due to the restricted lab access caused by the COVID-19. The filter used is briefly explained here with the usual notation.

$$X = \begin{bmatrix} p_x & p_y & p_z & v_x & v_y & v_z \end{bmatrix}^T \quad (5.3)$$

$$F(dt) = \begin{bmatrix} 1 & 0 & 0 & dt & 0 & 0 \\ 0 & 1 & 0 & 0 & dt & 0 \\ 0 & 0 & 1 & 0 & 0 & dt \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \quad (5.4)$$

With the assumption of an available world frame acceleration, it is considered as the control input to the system. Accelerometer bias is neglected for simplicity since this accelerometer measurement is simulated.

$$U = \begin{bmatrix} a_x & a_y & a_z \end{bmatrix}^T \quad (5.5)$$

$$G(dt) = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ dt & 0 & 0 \\ 0 & dt & 0 \\ 0 & 0 & dt \end{bmatrix} \quad (5.6)$$

$$X_{k+1} = F(dt_k)X_k + G(dt_k)U_k \quad (5.7)$$

Acceleration for this simulation is obtained using a state estimator on motion capture system data. Used world frame acceleration signals are presented in Fig. 5.5.

Ranges are used as measurements in the filter.

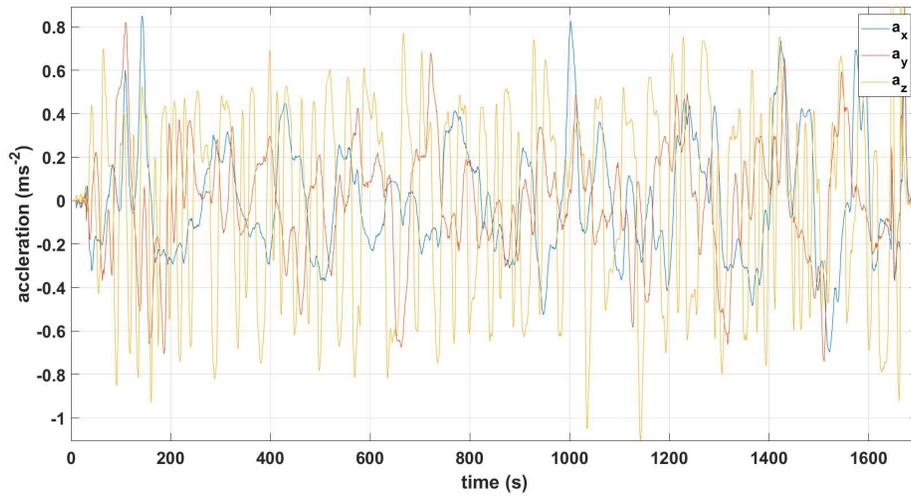


Figure 5.5: Acceleration input to the localization filter

$$Y = \begin{bmatrix} R_1 & R_2 & R_3 \end{bmatrix}^T \quad (5.8)$$

where,

$$R_i = \sqrt{(p_x - x_i)^2 + (p_y - y_i)^2 + (p_z - z_i)^2} \quad (5.9)$$

linearized measurement update matrix H is obtained as the jacobian (J) of the measurement vector, Y with respect to the state vector X . This was calculated using MATLAB symbolic tools and haven't included here due to being very large and low relevance to the thesis topic.

$$H = J(Y, X) \quad (5.10)$$

Standard Kalman filter equations are used to propagate and correct the system. Position estimated through this offline filter together with the ground truth is shown below. For comparison estimate without measurement correction is also presented.

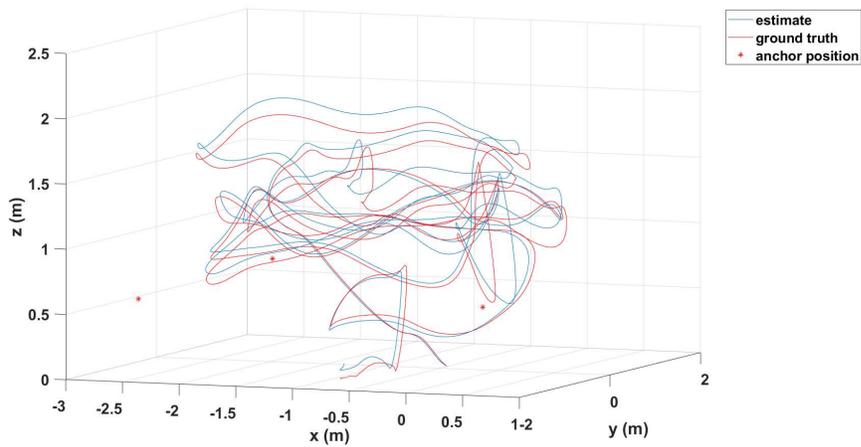


Figure 5.6: Position estimate with ranges

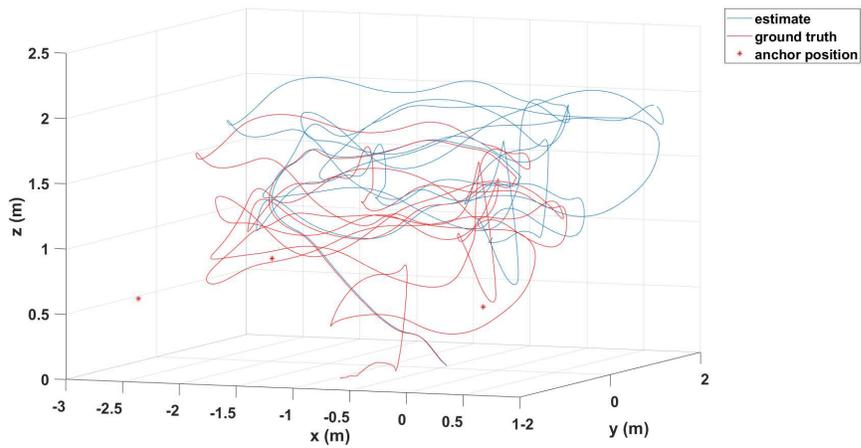


Figure 5.7: Position estimate without ranges

It can be seen that the drift in position estimate can be well controlled using range measurements in an IMU driven position estimation filter. However, in a realtime system, orientation is also needed to be tracked since there is no world frame acceleration measurement available.

Chapter 6

Conclusion and Future Directives

This thesis focuses on implementing a UWB based ranging and localization framework addressing the problem of indoor localization of autonomous vehicles. This chapter summarizes the work carried out to achieve objectives highlighting the conclusions. Future directives are then presented based on the observations and conclusions.

6.1 Summary of objective 1

Chapter 3 has presented an improved Kalman filter based method to estimate the time of flight between a set of UWB anchors for range estimation purposes. The time of flight is considered as a state in a Kalman filter due to the fact that it is coupled with the measurements. The filter uses transmission and reception timestamps of inbound and outbound messages together with the clock offset ratios calculated using carrier integration at the reception of inbound and outbound messages as measurements. Since ranging is performed while adhering to a transmission schedule, the power consumption for transmission and interference can be kept to a minimum.

The results indicated that this method can estimate the range between two nodes with higher precision and accuracy as compared to the traditional two way ranging

method. It has the ability to filter out higher frequency noise while capturing low frequency variations efficiently in time of flight calculations. It is important to note that the filtering can also be applied directly on the range measurements. However, the proposed method of estimation is more intuitive and suitable since it uses the low-level timestamps as measurements and models coupled clock dynamics accurately. It is important to note that the added state of the Kalman filter has introduced an additional computational complexity to the algorithm. With the low power microcontroller platform used in this study, the implementation has achieved a 15ms transmission period for an anchor pair, while without the time of flight state in the filter, the same could be achieved within 5ms periods. Since the overhead is mostly on double floating point calculations, a device with a compatible hardware FPU would be able to produce faster and improved updates.

6.2 Summery of objective 2

Chapter 4 has presented the modified gradient clock synchronization algorithm addressing the chaotic global clock rate parameter update rule. Derivation of the new update rule, the intuition behind it, and stability analysis of the error state is also presented. With the synchronized clocks and transmission timestamps in global clock, this network can provide infrastructure for TDOA based localization.

Authors of [1] have stated that they have experienced occasional loss of synchronization due to high packet losses. While conducting experiments, we have also experienced occasional loss of synchronization when using the generic clock rate parameter update rule in gradient clock synchronization. However, the modified global clock rate parameter update rule used in this paper is able to converge global clock rate parameters to original values even after a temporary disturbance. It guarantees the

convergence of the global clock rate at the average of individual clock rates, thus eliminates this chaotic behaviour.

Having a localization filter will allow to obtain predictions for the tof state, thus minimizing the lag in the tracked state. This will allow to have accurate synchronization even among non stationary anchors. Since the current implementation does not have an onboard localization filter and does not keep track of higher derivatives of the tof state, during fast movements, the lag in tof state will act as a bias in synchronization.

6.3 Summery of objective 3

Chapter 5 presents the ranging framework and experiments carried out to validate the proposed methods. However, due to unavoidable circumstances caused by COVID19, experiments for TDOA and localization couldn't be carried out. Hence a simulation is presented to demonstrate the range assisted localization performance, with simulated acceleration measurements. The results from the simulation shows that the drift in position estimation caused by integrating noisy world frame accelerations can be well controlled by incorporating the range measurements in the Kalman filter.

In the ranging experiment, closely observing the tracked range plots, it could be noted that the resulting RMS errors are mainly due to the lag in the time of flight tracking with fast movements. This can be corrected by adding a derivative of the time of flight as a filter state.

6.4 Summery of conclusions

This thesis presented a novel Kalman filter based range estimation technique considering the time of flight as a filter state. Furthermore, the chaotic global clock rate phenomenon in gradient clock synchronization is addressed by proposing a novel update rule. Experimental results proved that there is a considerable improvement in range estimation accuracy compared to traditional methods. The detailed proof provided and logged data confirms that the novel update rule is capable of handling chaotic behaviour achieving asymptotic stability in clock error state.

6.5 Publications

A portion of this work is presented at the Newfoundland Electrical and Computer Engineering Conference (NECEC) 2019 under the title "Wireless Clock Synchronization for UWB Ranging". Another paper from this work is submitted to the International conference of Intelligent Robots and Systems (IROS) 2020 with the title "Kalman Filter based Range Estimation and Clock Synchronization for Ultra Wide Band Networks", and it is being accepted to be published at the time of submitting this thesis.

6.6 Future Directives

Having the ability to use both TDOA and Kalman filter based ranging, this network can be designed to adapt both methods based on the available anchor count. When the number of anchors in the vicinity of a tag is less than four, the tag can request for information required to Kalman filter based ranging. The empty slot at the end of the transmission cycle can be used for this purpose. According to recent observability analysis [29], even less than three ranges can be used effectively with a range assisted

inertial navigation systems.

It is also suggested to further evaluate the performance due to the higher update rate when the clock offset ratio measurement is used just for initializing but not for filter update. In Fig.3.5, the filter initialized to the clock offset ratio performed relatively well. The marginal improvement in convergence when using clock offset ratio in filter update may be outperformed by the increased update rate due to less computational overhead when not using it. This can be significant particularly when using low power hardware. It is worth to note that in this implementation, the update rate is limited by the computational time.

The network self localization can be achieved using the methods proposed in [1]. When adding more anchors to the network, the firmware should be updated to use extended message length, which supports a payload up to 1023 bytes. In contrast, the regular mode maxes out at 127 bytes, which allows only four anchors with the used data structure. To further reduce the message length, the clock offset ratio calculated at the reception can be omitted from reply messages, which will eliminate the second row in the H_2 matrix.

Ideally, the relative clock tracking filter can be coupled with a localization filter in order to track the dynamics of a moving anchor to obtain predictions over the time of flight state. Better computational hardware needs to be incorporated to handle the added computational load.

For further improvements, antenna delays can also be correctly modelled or calibrated to remove the inherent biases in range measurements. A received signal power and orientation based correction for antenna biases may further improve the ranging accuracy as proposed in some literature [11, 16]. Future work will target extending the network to use both time difference of arrival and scheduled ranging methods based on the availability of the anchors, and implement the self calibration and localization.

Bibliography

- [1] M. Hamer and R. D’Andrea, “Self-calibrating ultra-wideband network supporting multi-robot localization,” *IEEE Access*, vol. 6, pp. 22 292–22 304, 2018.
- [2] P. Sommer and R. Wattenhofer, “Gradient clock synchronization in wireless sensor networks,” in *2009 International Conference on Information Processing in Sensor Networks*. IEEE, 2009, pp. 37–48.
- [3] J. Cano, S. Chidami, and J. Le Ny, “A kalman filter-based algorithm for simultaneous time synchronization and localization in uwb networks,” in *2019 International Conference on Robotics and Automation (ICRA)*. IEEE, 2019, pp. 1431–1437.
- [4] F. Kirsch and M. Vossiek, “Distributed kalman filter for precise and robust clock synchronization in wireless networks,” in *2009 IEEE Radio and Wireless Symposium*. IEEE, 2009, pp. 482–485.
- [5] A. Prorok, L. Gonon, and A. Martinoli, “Online model estimation of ultra-wideband tdoa measurements for mobile robot localization,” in *2012 IEEE International Conference on Robotics and Automation*. Ieee, 2012, pp. 807–814.
- [6] A. Prorok and A. Martinoli, “Accurate indoor localization with ultra-wideband using spatial models and collaboration,” *The International Journal of Robotics Research*, vol. 33, no. 4, pp. 547–568, 2014.
- [7] K. Liu and Z. Li, “Adaptive kalman filtering for uwb positioning in following luggage,” in *2019 34rd Youth Academic Annual Conference of Chinese Association of Automation (YAC)*. IEEE, 2019, pp. 574–578.
- [8] D. Dardari, A. Conti, U. Ferner, A. Giorgetti, and M. Z. Win, “Ranging with ultrawide bandwidth signals in multipath environments,” *Proceedings of the IEEE*, vol. 97, no. 2, pp. 404–426, 2009.
- [9] “Decawave.” [Online]. Available: <https://www.decawave.com/>
- [10] “Ieee standard for information technology– 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): Amendment 1: Add alternate phys,” *IEEE Std 802.15.4a-2007 (Amendment to IEEE Std 802.15.4-2006)*, pp. 1–210, Aug 2007.
- [11] “Dw1000 user manual.” [Online]. Available: <https://www.decawave.com/dw1000/usermanual/>
- [12] J. Smith and J. Abel, “Closed-form least-squares source location estimation from range-difference measurements,” *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 35, no. 12, pp. 1661–1669, 1987.
- [13] J. Elson, L. Girod, and D. Estrin, “Fine-grained network time synchronization using reference broadcasts,” *ACM SIGOPS Operating Systems Review*, vol. 36, no. SI, pp. 147–163, 2002.
- [14] S. Ganeriwal, R. Kumar, and M. B. Srivastava, “Timing-sync protocol for sensor networks,” in *Proceedings of the 1st international conference on Embedded networked sensor systems*, 2003, pp. 138–149.
- [15] M. Maróti, B. Kusy, G. Simon, and Á. Lédeczi, “The flooding time synchronization protocol,” in *Proceedings of the 2nd international conference on Embedded networked sensor systems*, 2004, pp. 39–49.
- [16] J. Sidorenko, V. Schatz, N. Scherer-Negenborn, M. Arens, and U. Hugentobler, “Decawave uwb clock drift correction and power self-calibration,” *Sensors*, vol. 19, no. 13, p. 2942, 2019.
- [17] M. W. Mueller, M. Hamer, and R. D’Andrea, “Fusing ultra-wideband range measurements with accelerometers and rate gyroscopes for quadcopter state estimation,” in *2015 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2015, pp. 1730–1736.
- [18] Y. Jiang and V. C. Leung, “An asymmetric double sided two-way ranging for crystal offset,” in *2007 International Symposium on Signals, Systems and Electronics*. IEEE, 2007, pp. 525–528.
- [19] D. Neiryneck, E. Luk, and M. McLaughlin, “An alternative double-sided two-way ranging method,” in *2016 13th workshop on positioning, navigation and communications (WPNC)*. IEEE, 2016, pp. 1–4.
- [20] M. Kwak and J. Chong, “A new double two-way ranging algorithm for ranging system,” in *2010 2nd IEEE International Conference on Network Infrastructure and Digital Content*. IEEE, 2010, pp. 470–473.
- [21] Y. Saigo, S.-i. Ko, J.-y. Takayama, and S. Ohyama, “Precise asynchronous rf tof measurement based on two-way-ranging using heterogeneous clocks,” in *2012 Proceedings of SICE Annual Conference (SICE)*. IEEE, 2012, pp. 1437–1442.

- [22] B. Al-Qudsi, M. El-Shennawy, N. Joram, and F. Ellinger, “Crystal oscillator frequency offset compensation for accurate fmcw radar ranging,” in *2016 German Microwave Conference (GeMiC)*. IEEE, 2016, pp. 405–408.
- [23] H. Kim, “Double-sided two-way ranging algorithm to reduce ranging time,” *IEEE Communications letters*, vol. 13, no. 7, pp. 486–488, 2009.
- [24] F. C. P. Shin, L. Zhiwei, and L. J. Xing, “Symmetric multi-way ranging for wireless sensor network,” in *2007 IEEE 18th International Symposium on Personal, Indoor and Mobile Radio Communications*. IEEE, 2007, pp. 1–5.
- [25] J. Lee, Z. Lin, P. Chin, and C. L. Law, “The use of symmetric multi-way two phase ranging to compensate time drift in wireless sensor network,” *IEEE transactions on wireless communications*, vol. 8, no. 2, pp. 613–616, 2009.
- [26] A. C. Pinho, D. R. Figueiredo, and F. M. França, “A robust gradient clock synchronization algorithm for wireless sensor networks,” in *2012 Fourth International Conference on Communication Systems and Networks (COMSNETS 2012)*. IEEE, 2012, pp. 1–10.
- [27] “The calculation of time domain frequency stability.” [Online]. Available: <https://ieee-uffc.org/frequency-control/educational-resources/the-calculation-of-time-domain-frequency-stability>
- [28] Yiu-Tong Chan, H. Yau Chin Hang, and Pak-chung Ching, “Exact and approximate maximum likelihood localization algorithms,” *IEEE Transactions on Vehicular Technology*, vol. 55, no. 1, pp. 10–16, 2006.
- [29] E. Fernando, O. D. Silva, G. K. I. Mann, and R. G. Gosine, “Observability analysis of position estimation for quadrotors with modified dynamics and range measurements,” in *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2019, pp. 2783–2788.