TIMING SYNCHRONIZATION AND NODE LOCALIZATION IN WIRELESS SENSOR NETWORKS: EFFICIENT ESTIMATION APPROACHES AND PERFORMANCE BOUNDS

A Dissertation

by

AITZAZ AHMAD

Submitted to the Office of Graduate Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

Approved by:

Co-Chairs of Committee,	Erchin Serpedin
	Hazem Nounou
Committee Members,	Serap Savari
	Byung-Jun Yoon
	Anxiao (Andrew) Jiang
Department Head,	Chanan Singh

December 2012

Major Subject: Electrical Engineering

ABSTRACT

Wireless sensor networks (WSNs) consist of a large number of sensor nodes, capable of on-board sensing and data processing, that are employed to observe some phenomenon of interest. With their desirable properties of flexible deployment, resistance to harsh environment and lower implementation cost, WSNs envisage a plethora of applications in diverse areas such as industrial process control, battlefield surveillance, health monitoring, and target localization and tracking. Much of the sensing and communication paradigm in WSNs involves ensuring power efficient transmission and finding scalable algorithms that can deliver the desired performance objectives while minimizing overall energy utilization. Since power is primarily consumed in radio transmissions delivering timing information, clock synchronization represents an indispensable requirement to boost network lifetime. This dissertation focuses on deriving efficient estimators and performance bounds for the clock parameters in a classical frequentist inference approach as well as in a Bayesian estimation framework.

A unified approach to the maximum likelihood (ML) estimation of clock offset is presented for different network delay distributions. This constitutes an analytical alternative to prior works which rely on a graphical maximization of the likelihood function. In order to capture the imperfections in node oscillators, which may render a time-varying nature to the clock offset, a novel Bayesian approach to the clock offset estimation is proposed by using factor graphs. Message passing using the max-product algorithm yields an exact expression for the Bayesian inference problem. This extends the current literature to cases where the clock offset is not deterministic, but is in fact a random process.

A natural extension of pairwise synchronization is to develop algorithms for the more challenging case of network-wide synchronization. Assuming exponentially distributed random delays, a network-wide clock synchronization algorithm is proposed using a factor graph representation of the network. Message passing using the max-product algorithm is adopted to derive the update rules for the proposed iterative procedure. A closed form solution is obtained for each node's belief about its clock offset at each iteration.

Identifying the close connections between the problems of node localization and clock synchronization, we also address in this dissertation the problem of joint estimation of an unknown node's location and clock parameters by incorporating the effect of imperfections in node oscillators. In order to alleviate the computational complexity associated with the optimal maximum a-posteriori estimator, two iterative approaches are proposed as simpler alternatives. The first approach utilizes an Expectation-Maximization (EM) based algorithm which iteratively estimates the clock parameters and the location of the unknown node. The EM algorithm is further simplified by a non-linear processing of the data to obtain a closed form solution of the location estimation problem using the least squares (LS) approach. The performance of the estimation algorithms is benchmarked by deriving the Hybrid Cramér-Rao lower bound (HCRB) on the mean square error (MSE) of the estimators.

We also derive theoretical lower bounds on the MSE of an estimator in a classical frequentist inference approach as well as in a Bayesian estimation framework when the likelihood function is an arbitrary member of the exponential family. The lower bounds not only serve to compare various estimators in our work, but can also be useful in their own right in parameter estimation theory.

DEDICATION

To my mother, whose charming personality, remarkable intellect and unconditional love will always be a source of inspiration for me.

ACKNOWLEDGEMENTS

Praise be to God, the Cherisher and Sustainer of the worlds. I am very grateful for His infinite mercy and blessings.

I would like to thank my advisors, Dr. Erchin Serpedin and Dr. Hazem Nounou, for their guidance and encouragement. I am particularly indebted to Dr. Serpedin for his constant motivation and unflinching support. I have had much to learn from his immense dedication and exceptional commitment. I would also like to thank my committee members, Dr. Serap Savari, Dr. Byung-Jun Yoon and Dr. Anxiao (Andrew) Jiang, for serving on my committee and providing valuable feedback. In particular, I am grateful to Dr. Savari, with whom I have had a long association as a student, grader and teaching assistant. I have always been inspired by her total professional commitment. I would also like to thank Dr. Yoon whose lucid teaching style spurred my interest in graphical models. I would also like to express my gratitude to my colleagues Sabit, Sangwoo, Yi, Bilal, Davide and Aaron for many useful, and at times useless, discussions. My friend Qasim Chaudhari also deserves a lot of appreciation for his patience in listening to my often fantastic theories and ideas. I would like to take this opportunity to thank Dr. Mansoor Shafi for his guidance and support. He was extremely instrumental in keeping me motivated during this long journey. I am also grateful to Dr. Apostolos Papathanassiou for hosting me at Intel Corp., Santa Clara CA during Fall 2011 and serving as an excellent mentor.

I am truly indebted to my sisters, Farwah and Arfa, and my brother, Afraz, for their prayers and support. I am blessed to have such loving siblings. I cannot thank my father enough for all his prayers and constant support. I owe much of what I have achieved to his immeasurable sacrifices. I am also grateful to my mother-in-law

for her prayers.

I would also like to acknowledge the immense contribution of my wife, Amina, a very special person in my life. She has constantly been by my side during this arduous journey. Her remarkable patience during the not-so-bright days, and her cheerful exuberance during the sunny days have made my life very fulfilling.

Lastly, I would like to express my gratitude to the single most influential person in my life. My deceased mother. While I am deeply saddened that she is no more to enjoy this success with me, I also have a certain feeling of happiness on finishing this work, something which she had so dearly wished. No amount of gratitude can ever be enough to acknowledge what she did for me in her life!

TABLE OF CONTENTS

LIST OF FIGURES

1. INTRODUCTION

The fundamental problem of communication is that of reproducing at one point either exactly or approximately a message selected at another point.

-Claude E. Shannon.

1.1 Wireless Sensor Networks

Wireless sensor networks (WSNs) comprise a large number of inexpensive devices that are deployed for observing and initial processing of physical or environmental changes taking place in their vicinity. An on-board sensing equipment enables the sensors to summarize the useful information to be transmitted to a distant fusion center (FC), resulting in reduced communication requirements. The FC then aggregates this data to infer the desired information. In addition, these sensor nodes can also collaborate to accomplish common tasks. Recent technology breakthroughs in micro-electro-mechanical systems (MEMS) have enabled successful deployment of large scale WSNs. With their desirable properties of self-organization and meaningful data preprocessing, WSNs are expected to play a pivotal role in future wireless communications. A typical sensor network is depicted in Fig. 1.1.

WSNs offer several applications in numerous fields. Some of the target applications identified for WSNs are as follows [9].

1. Health care:

   Sensor networks can be useful in monitoring patients' health, drug administration in hospitals and keeping track of physiological changes in living organisms.

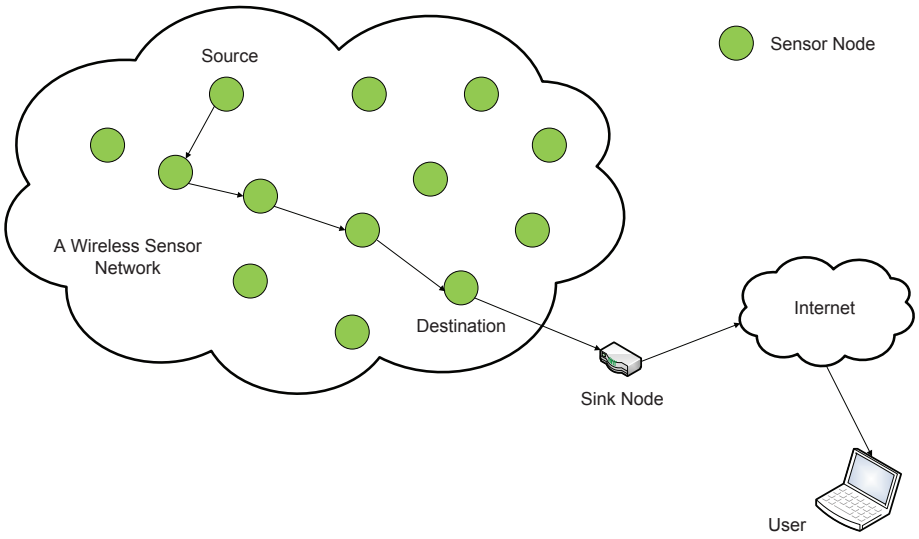

Figure 1.1: A typical wireless sensor network

2. Battlefield Surveillance:

   Sensor nodes can be deployed in harsh terrains and challenging environmental conditions to keep an eye on enemy movements. Their ability to summarize and transmit useful data can assist military authorities in drawing effective contingency plans.

3. Industrial Process Control:

   A typical application of WSNs is in monitoring the physical changes taking place in an industrial process. For instance, a sensor network can be deployed to sense the temperature and pressure of a particular chemical reaction.

4. Target Tracking:

   WSNs can also be deployed to observe and sense the movement of a target in

their vicinity. These observations are then used at the sink nodes to track the target.

While there are several attractive features of WSNs, deploying a sensor network to perform designated tasks prompts numerous design challenges. The scarcity of the energy resources available at a sensor perhaps accounts for the most stringent constraint. The sensor nodes are generally equipped with meagre power resources and often have to operate in harsh and unfriendly environmental conditions which render battery replacement impractical. Hence, it becomes extremely critical to use power efficiently to prolong the lifetime of sensors. Much of the sensing and communication paradigm in WSNs involves ensuring power efficient transmission and finding scalable algorithms that can deliver the desired performance objectives while minimizing the overall energy utilization. Since power is primarily consumed in radio transmissions delivering timing information [50], the network lifetime can be boosted by a careful management of the sleep and wake periods. Therefore, synchronization represents a fundamental requirement in WSN operation.

1.2 Clock Synchronization in Wireless Sensor Networks

Clock synchronization pertains to establishing a common notion of time across the whole network. A network remains synchronized as long as the clocks available at all the sensor nodes have the same offset and their frequency sources run at the same rate. However, practical sensors with cheap oscillators can drift apart seconds per day due to many factors such as manufacturing errors, environmental degradations and aging. This results in sensor nodes losing the notion of a common time scale and produces an adverse impact on power utilization. Efficient techniques for maintaining clock synchronization can lower the re-synchronization requests, which, in turn, can translate to significant energy savings. Clock synchronization has a direct impact on

several important WSN operations as described below [58].

1. Optimal Data Fusion:

   Sensor nodes are often used to deliver a summary of their observations to FC where this data is aggregated to infer meaningful information. Clock synchronization is mandatory to ensure optimal fusion of data coming from different sensors.

2. Efficient Duty Cycling:

   Clock synchronization can save energy resources by ensuring a coordinated wake and sleep operation for sensor nodes. In this way, a node activates its radio receiver only when it has to participate in the network activity, while turns it off otherwise.

3. Channel Access Schemes:

   Deterministic channel access is based on a firm time agreement between nodes, so that techniques such as time-division-multiple-access (TDMA) and frequency-division-multiple-access (FDMA) can be successfully applied.

4. Target Localization and Tracking:

   Target localization and tracking are other applications of WSNs in which clock synchronization is mandatory in order to obtain relevant results.

Clock synchronization has been actively studied over the past several years. Different data exchange protocols with varying degrees of accuracy and simplicity have been proposed that aim to synchronize clocks in a network. Data exchange protocols require the sensor nodes to exchange their timing information with each other and with the reference nodes. An excellent survey of the prevalent synchronization protocols is presented in [58]. Two common clock models that relate the clock at a

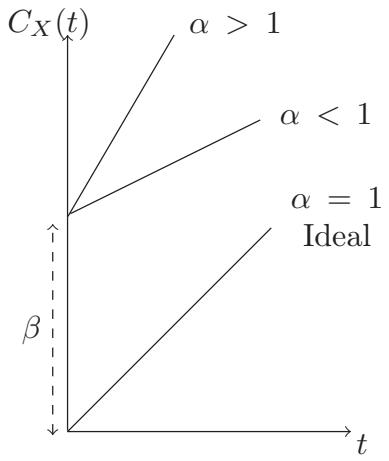

Figure 1.2: Ideal and non-ideal clocks with respect to the reference time t.

particular node X to the reference time t are as follows.

Offset-only Model:

$$C_X(t) = t + \beta \ ,$$

Offset and Skew Model:

$$C_X(t) = \alpha t + \beta \ ,$$

where α and β, referred to as the *clock parameters*, denote the *clock skew* and *clock offset*, respectively. The relationship between the ideal and non-ideal clocks with the reference time t is depicted in Fig. 1.2. An ideal clock has $\alpha = 1$ and $\beta = 0$, and runs at the same rate as the time t. In practice however, the clocks exhibit non-ideal behavior due to the factors mentioned above, so that α and β no longer assume ideal values.

The offset and skew model extends the simpler offset-only model to include the effects of skew, which is the difference in the frequencies of the clock at node X and the perfect clock. The inclusion of skew helps to achieve greater accuracy and also results in a significant reduction of re-synchronization requests. Much of the

literature on clock synchronization algorithms aims to obtain reliable estimates of the clock parameters so that their effects can be compensated and a node can be synchronized to the network. This dissertation studies both the aforementioned models for the purpose of clock synchronization.

The earlier approaches to clock synchronization presented data exchange protocols that can be used for communication among sensor nodes. The clock synchronization problem in a WSN offers a natural statistical signal processing framework. In this dissertation, our emphasis is mostly on using tools from parameter estimation theory to achieve clock synchronization by estimating the clock parameters. A comprehensive review of the prevalent statistical approaches for clock synchronization is presented below.

1.3 Related Work

A survey of the popular approaches employed for timing synchronization is presented in [58] and [53]. The *one-way* message exchange mechanism involves a reference node broadcasting its timing information to other nodes in a network. The receiver nodes record the arrival of these messages with respect to their own clock. After exchanging several time stamps, the nodes estimate their offsets based on these observations. A particular case is the flooding time synchronization protocol (FTSP) [40] which uses regression to estimate the clock offset. On the other hand, through a *two-way* timing exchange process, adjacent nodes aim to achieve pairwise synchronization by communicating their timing information with each other. After a round of N messages, each node estimates its own clock parameters. The timing-sync protocol for sensor networks (TPSNs) [36] uses this strategy in two phases to synchronize clocks in a network. The level discovery phase involves a spanning tree based representation of a WSN while nodes attempt to synchronize with their im-

mediate parents using a two-way message exchange process in the synchronization phase. In *receiver-receiver* synchronization, nodes collect time stamps sent from a common broadcasting node and utilize them to adjust their clocks. The reference broadcast synchronization (RBS) protocol [21] uses reference beacons sent from a master node to establish a common notion of time across a network. This can have an appreciable impact on saving power resources at a sensor node.

The *two-way* message exchange process, used in most of the synchronization protocols, involves a pair of nodes aiming to estimate their clock parameters by exchanging their timing information with each other [1]. There is an extensive literature that makes use of this data exchange process in order to derive expressions for clock offset and skew estimators. A critical component of the clock synchronization problem, which has received widespread interest in recent years, is the accurate modeling of the random network delays that contaminate the data exchange process. By considering the clock offset as the cause of time disagreement between sensors, and assuming exponentially distributed network delays, [30] reported the maximum likelihood estimator (MLE) for the clock offset while an earlier study [1] stated the non-uniqueness of the MLE when both the mean of the random delays and the fixed portion of the overall delay are known [11]. The expressions for both the best linear unbiased estimator using order statistics (BLUE-OS, [31]) and the minimum variance unbiased estimator (MVUE, [31]) are derived in [16] under the assumptions of asymmetric exponentially distributed delays between uplink and downlink. By recasting the clock parameter estimation problem as a linear program, joint ML estimators of clock offset and skew have been proposed in [39] for exponentially distributed network delays. The Gaussian distribution is assumed in [45] and [37] to jointly estimate the clock offset and skew for known and unknown propagation delay, respectively. A more general assumption is considered in [33,34], where the authors use

the Gaussian mixture Kalman particle filter to cope with the presence of a general delay model represented as a mixture of several distributions. Assuming the random delays to be Gaussian, the two-way data exchange has also been considered for designing clock synchronization algorithms in underwater acoustic networks [66]. The achievable synchronization performance for a two-way message exchange has been studied in [56] by deriving Cramér-Rao bounds. In general, the set of candidate distributions for modeling the network delays includes the Gaussian, exponential, Gamma [11] and Weibull distributions [3].

A natural extension of pairwise synchronization is to design algorithms for synchronizing sensor nodes across the whole network. Based on clock offset estimates, the authors in [10] proposed iterative distributed algorithms for establishing network-wide clock synchronization. A network-wide clock synchronization algorithm is proposed in [57] assuming both clock offset and fixed skew affecting the running behavior of the oscillators. The main novelty of the aforementioned work resides in exploiting the natural network constraint that the relative clock offsets in network loops sum to zero. A statistical analysis of the algorithm proposed in [57] is performed in [27]. The authors in [22] proposed a synchronization algorithm by assuming no initial clock offsets but time-varying skews among the oscillators in the network. The proposed algorithm therein aims at synchronization by estimating the logarithm of the clock skews, starting from measurements affected by Gaussian noise. Extending the results to a multi-hop sensor network, a distributed heuristic algorithm is presented in [17] which uses pairwise broadcast synchronization (PBS) for network-wide synchronization.

An alternative framework for network-wide distributed clock synchronization consists of recasting the problem of agreement on oscillation phases and/or frequencies as a consensus based recursive model in which only local message passing is re-

quired among nodes. By assuming a connected network, it is possible to design efficient distributed algorithms by carefully choosing the update function. Under this framework, [55] proposed a Laplacian-based algorithm for establishing agreement on oscillation frequencies all over the network based on standard consensus. A combined agreement over both clock phases and frequencies has been studied in [65], by making use of state-of-the-art fast consensus techniques. Scalable synchronization algorithms for large sensor networks are developed in [28] and [46] inspired by mathematical biology models justifying synchrony in the biological agents. The convergence of distributed consensus time synchronization algorithms is investigated in [63] assuming a Gaussian delay between sensor nodes.

More recently, there has been a focus on the application of graphical models and message passing to the clock synchronization problem. Both sum-product [38] and max-product [7] algorithms have been used in designing clock synchronization algorithms and for clock offset estimation purposes, respectively. Assuming Gaussian distributed network delays, the authors in [38] proposed the use of the sum-product algorithm in order to design a network-wide clock synchronization method, where clock offset is the only cause of time disagreement between sensors (i.e., no skew).

Node localization is an important aspect of several WSN applications that require location-awareness such as geographical routing, disaster rescue, etc., [23, 24, 44]. There is an extensive literature on location estimation algorithms in WSNs [47, 54]. In general, the range-based localization algorithms utilize the metrics of time of arrival (TOA) [18], time difference of arrival (TDOA) [29] and received signal strength (RSS) to determine the distance between the unknown node and the anchors. These distance-based measurements are then used for node localization. Distributed location estimation algorithms have also been studied for cooperative and passive sensors using the above mentioned metrics [25]. Using a hybrid TOA and TDOA approach,

positioning of multiple target nodes in a cooperative wireless network has been proposed in [26].

Since TOA and TDOA are time-based techniques, synchronization is an important prerequisite in node localization as well [49]. The close connection between the problems of localization and synchronization necessitates a joint estimation approach. Based on the two-way ranging protocol in IEEE 802.15.4a, a joint localization and synchronization algorithm was proposed in [20] using TOA measurements. Recently, several contributions have studied joint localization and synchronization from a statistical signal processing viewpoint. Optimal and sub-optimal algorithms for estimating an unknown node's position and clock parameters were derived in [67]. The performance of the estimation algorithms developed therein was also compared with the Cramér-Rao lower bound (CRB). A weighted least squares approach for joint estimation is devised in [68]. Robust algorithms for joint estimation that are resistant to target node's uncertainties were proposed in [62].

1.4 Main Contributions of this Research

Our main contributions in this research are as follows.

- In Section 2, a unified framework for ML estimation of clock offset is presented when the likelihood function of the observations is Gaussian, exponential or log-normally distributed [6]. The proposed framework recovers the already known results for Gaussian and exponentially distributed network delays and determines the ML estimate in case of log-normal distribution. Hence, the proposed analytical approach represents a simpler alternative, and a more general derivation of ML estimator, which bypasses the graphical analysis used in [30] to maximize the likelihood function.

- Since sensor nodes are often deployed in harsh environmental conditions, degra-

dations arise in quartz crystals over time, introducing temporal variations in clock parameters. In order to capture these time variations, a Bayesian framework is presented by considering the clock offset as a random Gauss-Markov process. Bayesian inference is performed using factor graphs and the max-product algorithm. A novel message passing strategy yields an exact solution for Gaussian, exponential and log-normally distributed likelihood functions [6]. This extends the current literature to cases where the clock offset may not be deterministic, but is in fact a random process. An extension of these results to pairwise broadcast synchronization, where an inactive node synchronizes by overhearing the communication between two active nodes is also discussed in Appendix C.

- Section 3 focuses on extending the factor graph approach developed for pairwise synchronization in Section 2 to the more challenging case of network-wide synchronization. Assuming exponentially distributed network delays, graphical models in conjunction with message passing is used to design a network-wide clock synchronization algorithm. Specifically, a factor graph representation of the network is used to design the message exchange rules between sensors, based on the max-product algorithm. This leads to an iterative update of the belief, whose maximization gives rise to the expression for the clock offset estimator. A simulation study is then performed to show that the proposed algorithm converges in various network topologies of interest [64].

- The idea of time variations in clock parameters is introduced for joint node localization and clock synchronization in Section 4. By incorporating these temporal variations in the system model, a joint maximum a-posteriori (MAP) estimation approach entails high computational load due to the need to in-

vert large matrices. Since sensor nodes are inexpensive devices with limited computational power, simpler alternatives are required. Towards this end, an Expectation-Maximization (EM) based joint localization and time-varying synchronization algorithm is proposed that iteratively determines the time-varying clock parameters using a Kalman smoother followed by a likelihood maximization of the location parameter. In order to alleviate the computational complexity that comes with the two-dimensional likelihood optimization required for localization, a linearization based least squares (LS) method is presented which yields a closed form solution and is therefore, a simpler alternative to the EM algorithm. Moreover, it is observed through numerical simulations that the performance of the LS based location estimator is fairly close to the EM algorithm for small to moderate measurement noise errors. Theoretical lower bounds on the MSE of an estimator are obtained by deriving the Hybrid Cramér-Rao bound (HCRB) in our estimation framework. This helps to compare the performance of the aforementioned estimators [4].

- The performance of an estimator can be gauged by comparing its mean square error (MSE) performance with the theoretical lower bounds suggested by parameter estimation theory. In Section 5, we derive lower bounds on the MSE of an estimator when the likelihood function is an arbitrary member of the exponential family of distributions in the classical frequentist inference approach as well as in a Bayesian estimation framework [6]. Hence, these results are fairly general and can be useful in their own right in parameter estimation theory, and at the same time serve as a stepping stone towards comparing the estimators derived in Section 2.

2. PAIRWISE SYNCHRONIZATION: UNIFIED MAXIMUM LIKELIHOOD AND BAYESIAN ESTIMATION*

2.1 Introduction

The ML estimate of the clock offset in case of exponentially distributed network delays was derived in [30] by using graphical arguments. The minimum variance unbiased estimate (MVUE) of the clock offset was presented in [16]. Most of the statistical approaches employed for clock synchronization assume *fixed* clock offsets. However, degradations in quartz crystals in sensor nodes due to environmental effects can give rise to temporal variations in clock parameters. In this section, a Bayesian approach is proposed to accurately track the time-varying clock offset and keep the sensor nodes synchronized for longer periods.

2.2 Main Contributions

The main contributions of this work can be summarized as follows.

1. A unified framework for ML estimation of clock offset is presented when the likelihood function of the observations is Gaussian, exponential or log-normally distributed. The resulting analytical approach represents a simpler alternative which bypasses the complex graphical analysis used in [30] to maximize the likelihood function.

2. A Bayesian solution to the time-varying clock offset estimation problem is presented by using a factor graph representing the factorization of the global probability density function.

*Reprinted with permission from "A factor graph approach to clock offset estimation in wireless sensor networks," Aitzaz Ahmad, Davide Zennaro, Erchin Serpedin, and Lorenzo Vangelista, 2012, IEEE Transactions on Information Theory, vol. 58, no. 7, pg. 4244-4260, Copyright 2012 by IEEE.

3. An exact solution to the Bayesian estimation problem is obtained using a novel message passing strategy. This extends the current literature by incorporating temporal variations that may arise over time in sensor nodes[1].

2.3 System Model

The process of pairwise synchronization between two nodes S and R is illustrated in Fig. 2.1. At the jth message exchange, node S sends the information about its current time through a message including time stamp T_j^1. Upon receipt of this message, node R records the reception time T_j^2 according to its own time scale. The two-way timing message exchange process is completed when node R replies with a synchronization packet containing time stamps T_j^2 and T_j^3, which is received at time T_j^4 by node S with respect to its own clock. After N such messages have been exchanged between nodes S and R, node S is equipped with time stamps $\{T_j^1, T_j^2, T_j^3, T_j^4\}_{j=1}^N$. The impairments in the signaling mechanism occur due to a *fixed* propagation delay, which accounts for the time required by the message to travel through the transmission medium, and a *variable* network delay, that arises due to queuing and other processing delays experienced by the messages during transmission and reception [21]. By assuming that the respective clocks of nodes S and R are related by $C_S(t) = \beta + C_R(t)$, the two-way timing message exchange model at the jth instant can be represented as

$$\begin{aligned} T_j^2 &= T_j^1 + d + \beta + X_j \\ T_j^4 &= T_j^3 + d - \beta + Y_j \,, \end{aligned} \tag{2.1}$$

[1] An extension of these results to pairwise broadcast synchronization where an inactive node synchronizes by over hearing the time-stamps exchanged between active nodes is discussed in Appendix C [2].

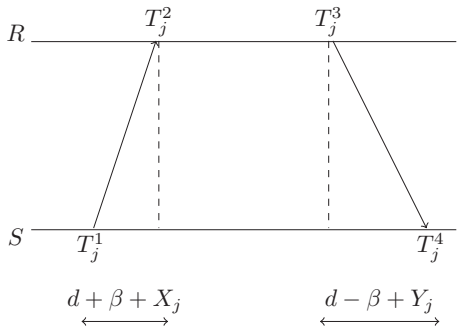

Figure 2.1: A two-way timing message exchange mechanism

where d represents the propagation delay, assumed symmetric in both directions, and β is the offset of the clock at node S relative to the clock at node R. X_j and Y_j are the independent and identically distributed variable network delays. By defining [1]

$$U_j \triangleq T_j^2 - T_j^1, \quad V_j \triangleq T_j^4 - T_j^3 \ ,$$

the system of equations in (2.1) can be equivalently expressed as

$$U_j = d + \beta + X_j, \quad V_j = d - \beta + Y_j \ . \tag{2.2}$$

By further defining

$$\xi \triangleq d + \beta \qquad \psi \triangleq d - \beta \ , \tag{2.3}$$

the model in (2.2) can be written as

$$U_j = \xi + X_j, \quad V_j = \psi + Y_j \tag{2.4}$$

for $j = 1, \ldots, N$. The goal is to determine precise estimates of ξ and ψ using observations $\{U_j, V_j\}_{j=1}^N$. An estimate of β can, in turn, be obtained using (2.3) as

follows

$$\beta = \frac{\xi - \psi}{2} \, . \tag{2.5}$$

Accurate modeling of the variable delays, X_j and Y_j, has been a topic of interest in recent years. Several distributions have been proposed that aim to capture the random effects caused by the queuing delays [11]. These distributions include exponential, gamma, log-normal and Weibull. In addition, the authors in [37] argued that X_j and Y_j might result from contributions of numerous independent random processes and can, therefore, be assumed Gaussian. The ML estimate of d and β for the case of exponentially distributed network delay was determined in [30]. Recently, the minimum variance unbiased estimate (MVUE) of the clock offset under an exponentially distributed network delay was proposed in [16]. In this work, instead of working with a specific distribution, a general framework of the clock synchronization problem is proposed that yields a parameterized solution of the clock offset estimation problem in the classical frequentist inference approach as well in a Bayesian estimation framework when the likelihood function of the observations, U_j and V_j, is Gaussian, exponential or log-normally distributed.

In particular, the general notation used when the likelihood function of the observations $\mathbf{U} \triangleq [U_1, \ldots, U_N]^T$ and $\mathbf{V} \triangleq [V_1, \ldots, V_N]^T$ is Gaussian or log-normally distributed is given below.

Unconstrained Likelihood:

$$f(\mathbf{U}; \xi) \propto \exp\left(\xi \sum_{j=1}^{N} \eta_\xi(U_j) - N\phi_\xi(\xi) \right) \tag{2.6}$$

$$f(\mathbf{V}; \psi) \propto \exp\left(\psi \sum_{j=1}^{N} \eta_\psi(V_j) - N\phi_\psi(\psi) \right) \, , \tag{2.7}$$

where $\eta_\xi(U_j)$ and $\eta_\psi(V_j)$ are sufficient statistics for estimating ξ and ψ, respectively. The log-partition functions $\phi_\xi(.)$ and $\phi_\psi(.)$ serve as normalization factors so that $f(\mathbf{U};\xi)$ and $f(\mathbf{V};\psi)$ are valid probability distributions. The likelihood function is called 'unconstrained' since its domain is independent of the parameters ξ and ψ.

Similarly, the general notation used for an exponentially distributed likelihood function is given below.

Constrained Likelihood:

$$f(\mathbf{U};\xi) \propto \exp\left(\xi \sum_{j=1}^{N} \eta_\xi(U_j) - N\phi_\xi(\xi)\right) \prod_{j=1}^{N} \mathbb{I}(U_j - \xi) \tag{2.8}$$

$$f(\mathbf{V};\psi) \propto \exp\left(\psi \sum_{j=1}^{N} \eta_\psi(V_j) - N\phi_\psi(\psi)\right) \prod_{j=1}^{N} \mathbb{I}(V_j - \psi)\ , \tag{2.9}$$

where $\mathbb{I}(.)$ is the indicator function and the roles of $\eta_\xi(U_j)$, $\eta_\psi(V_j)$, $\phi_\xi(.)$ and $\phi_\psi(.)$ are similar to (2.6) and (2.7). The likelihood function is called constrained since its domain depends on the parameters ξ and ψ. It must be noted that the likelihood functions (2.6)-(2.9) are expressed in terms of general exponential family distributions. This approach helps to keep the exposition sufficiently general and also allows us to recover the known results for the ML estimation of clock offset for Gaussian and exponentially distributed likelihood functions [30] [45], and determine the ML estimator of the clock offset in case of log-normally distributed likelihood function. The proposed approach will also prove useful in investigating a unified novel framework for clock offset estimation in the Bayesian setting for Gaussian, exponential or log-normally distributed likelihood functions.

Some key ingredients of the proposed solution for the clock offset estimation problem, based on the properties of exponential family, can be summarized as follows

[61].

1. The mean and variance of the sufficient statistic $\eta_\xi(U_j)$ are expressed as

$$\mathbb{E}\left[\eta_\xi(U_j)\right] = \frac{\partial \phi_\xi(\xi)}{\partial \xi} \tag{2.10}$$

$$\sigma^2_{\eta_\xi} \triangleq \text{Var}\left[\eta_\xi(U_j)\right] = \frac{\partial^2 \phi_\xi(\xi)}{\partial \xi^2} \ . \tag{2.11}$$

2. The moment generating function (MGF) of the statistic $\eta_\xi(U_j)$ is given by

$$M_{\eta_\xi}(h) = \exp\left(\phi_\xi(\xi + h) - \phi_\xi(\xi)\right) \ . \tag{2.12}$$

3. The non-negativity of the variance $\sigma^2_{\eta_\xi}$ in (2.11) implies that the log-partition function $\phi_\xi(.)$ is convex.

4. For Gaussian, exponential and log-normally distributed likelihood functions, the log-partition function $\phi_\xi(\xi)$ can be expressed as a second degree polynomial given by

$$\phi_\xi(\xi) = a_\xi \xi^2 \ . \tag{2.13}$$

   The coefficient a_ξ in this approximation can be obtained using the variance of the statistic $\eta_\xi(U_j)$, which is assumed known. Using (2.11), a_ξ is given by

$$a_\xi = \frac{\sigma^2_{\eta_\xi}}{2} \ .$$

   If the statistical moment in (2.11) is not available, the empirical moment can be used.

Similar expressions can also be written for $\eta_\psi(V_j)$, $M_{\eta_\psi}(h)$ and $\phi_\psi(\psi)$, respectively.

2.4 A Unified Maximum Likelihood Estimation Approach

In this section, the ML estimates of β are obtained analytically. This approach differs from the graphical arguments used to maximize the likelihood in [30]. The specific cases of unconstrained and constrained likelihood functions are considered separately. Explicit expressions are provided for ξ only, since the analysis is analogous for ψ.

2.4.1 Unconstrained Likelihood

Using (2.6) and (2.13), the unconstrained likelihood function for data set $\mathbf{U}$ is given by

$$f(\mathbf{U};\xi) \propto \exp\left(\xi \sum_{j=1}^{N} \eta_\xi(U_j) - N\frac{\sigma_{\eta_\xi}^2}{2}\xi^2\right) . \tag{2.14}$$

The resulting ML estimate of ξ can be expressed as

$$\hat{\xi}_{\mathrm{ML}} = \frac{\sum_{j=1}^{N} \eta_\xi(U_j)}{N\sigma_{\eta_\xi}^2} . \tag{2.15}$$

Invoking the invariance principle [31], the ML estimator $\hat{\beta}_{\mathrm{ML}}$ for the clock offset is expressed as

$$\hat{\beta}_{\mathrm{ML}} = \frac{\hat{\xi}_{\mathrm{ML}} - \hat{\psi}_{\mathrm{ML}}}{2} . \tag{2.16}$$

2.4.1.1 Gaussian Distributed Likelihood Function

A particular application is the case when the likelihood functions $f(U_j;\xi)$ and $f(V_j;\psi)$ have a Gaussian distribution i.e., $f(U_j;\xi) \sim \mathcal{N}(\xi,\sigma_\xi^2)$ and $f(V_j;\psi) \sim \mathcal{N}(\psi,\sigma_\psi^2)$ [45]. Therefore,

$$f(\mathbf{U};\xi) = \frac{1}{(2\pi\sigma_\xi^2)^{\frac{N}{2}}} \exp\left(-\frac{\sum_{j=1}^{N}(U_j-\xi)^2}{2\sigma_\xi^2}\right) , \tag{2.17}$$

which can be rearranged as

$$f(\mathbf{U};\xi) \propto \exp\left(\xi\frac{\sum_{j=1}^{N} U_j}{\sigma_\xi^2} - \frac{N}{2\sigma_\xi^2}\xi^2\right) .$$

By comparing with (2.14), we have

$$\eta_\xi(U_j) = \frac{U_j}{\sigma_\xi^2}, \qquad \sigma_{\eta_\xi}^2 = \frac{1}{\sigma_\xi^2} \tag{2.18}$$

and the ML estimate using (2.15) is given by

$$\hat{\xi}_{\mathrm{ML}} = \frac{\sum_{j=1}^{N} U_j}{N} . \tag{2.19}$$

The ML estimate for the offset β follows from (2.16), and can be expressed as

$$\hat{\beta}_{\mathrm{ML}} = \frac{\sum_{j=1}^{N}(U_j - V_j)}{2N} . \tag{2.20}$$

The above estimate coincides with the one reported in [45].

2.4.1.2 Log-Normally Distributed Likelihood Function

When the samples U_j and V_j are log-normally distributed, we have

$$\begin{aligned} f(\mathbf{U};\xi) &= \frac{1}{\sqrt{2\pi\sigma_\xi^2}} \prod_{j=1}^{N} U_j^{-1} \exp\left(-\frac{\sum_{j=1}^{N}(\log U_j - \xi)^2}{2\sigma_\xi^2}\right) \\ &\propto \exp\left(\xi\frac{\sum_{j=1}^{N}\log U_j}{\sigma_\xi^2} - \frac{N}{2\sigma_\xi^2}\xi^2\right) . \end{aligned} \tag{2.21}$$

A comparison with (2.14) yields

$$\eta_\xi(U_j) = \frac{\log U_j}{\sigma_\xi^2}, \qquad \sigma_{\eta_\xi}^2 = \frac{1}{\sigma_\xi^2} .$$

The ML estimator for β can obtained from (2.16) using (2.15), and is given by

$$\hat{\beta}_{\mathrm{ML}} = \frac{\sum_{j=1}^{N} (\log U_j - \log V_j)}{N} \ . \tag{2.22}$$

2.4.2 Constrained Likelihood

Using (2.8) and (2.13), the constrained likelihood function for data set $\mathbf{U}$ is given by

$$f(\mathbf{U};\xi) \propto \exp\left(\xi \sum_{j=1}^{N} \eta_\xi(U_j) - N\frac{\sigma_{\eta_\xi}^2}{2}\xi^2\right) \prod_{j=1}^{N} \mathbb{I}(U_j - \xi) \ . \tag{2.23}$$

The resulting ML estimate of ξ can be obtained as

$$\begin{aligned} \hat{\xi}_{\mathrm{ML}} = \arg \ \max_{\xi} & \exp\left(\xi \sum_{j=1}^{N} \eta_\xi(U_j) - N\frac{\sigma_{\eta_\xi}^2}{2}\xi^2\right) \\ & \text{such that} \quad U_{(1)} \geq \xi \ , \end{aligned} \tag{2.24}$$

where $U_{(1)}$ denotes first order statistics of the samples U_j. The likelihood maximization problem (2.24) is strictly concave and the ML estimate can be expressed as

$$\hat{\xi}_{\mathrm{ML}} = \min\left(\frac{\sum_{j=1}^{N} \eta_\xi(U_j)}{N\sigma_{\eta_\xi}^2}, U_{(1)}\right) \ . \tag{2.25}$$

The ML estimator $\hat{\beta}_{\mathrm{ML}}$ for the clock offset, using the invariance principle, is given by

$$\hat{\beta}_{\mathrm{ML}} = \frac{\hat{\xi}_{\mathrm{ML}} - \hat{\psi}_{\mathrm{ML}}}{2} \ . \tag{2.26}$$

2.4.2.1 Exponentially Distributed Likelihood Function

For the case when the likelihood functions are exponentially distributed, the density function of the samples U_j can be written as

$$f(\mathbf{U};\xi) = \lambda_\xi^N \exp\left(-\lambda_\xi \sum_{j=1}^{N}(U_j - \xi)\right) \mathbb{I}(U_{(1)} - \xi) \;, \tag{2.27}$$

where λ_ξ^{-1} is the mean of the delays X_j. The density function can be rearranged as

$$f(\mathbf{U};\xi) \propto \exp\left(N\lambda_\xi \xi\right) \mathbb{I}(U_{(1)} - \xi) \;.$$

Comparing the above formulation with (2.23),

$$\eta_\xi(U_j) = \lambda_\xi, \qquad \sigma_\eta^2 = 0 \;. \tag{2.28}$$

Using (2.25), the ML estimate is given by

$$\hat{\xi}_{\mathrm{ML}} = U_{(1)} \;. \tag{2.29}$$

The ML estimate of β follows from (2.26), and can be expressed as

$$\hat{\beta}_{\mathrm{ML}} = \frac{U_{(1)} - V_{(1)}}{2} \;, \tag{2.30}$$

which coincides with the one reported in [30], where it is derived using graphical arguments.

2.5 Incorporating Temporal Variations: A Factor Graph Approach

The imperfections introduced by environmental conditions in the quartz oscillator in sensor nodes results in a time-varying clock offset between nodes in a WSN. To cater for such a temporal variation, a Bayesian approach to the clock synchronization problem is adopted by representing the a-posteriori density as a factor graph. The inference is performed on the factor graph by message passing using max-product algorithm. To ensure completeness, a brief description of factor graphs and the max-product algorithm is provided below.

A factor graph is a *bipartite* graph that represents a factorization of a global function as a product of local functions called *factors*, each factor being dependent on a subset of variables. Factor graphs are often used to produce a graphical model depicting inter-dependencies among a collection of interacting variables. Each factor is represented by a factor node and each variable has an edge or a half-edge. An edge connects a particular variable to a factor node only if it is an argument of the factor expressed by the factor node [35].

Inference can be performed by passing messages (sometimes called beliefs) along the edges of a factor graph. In particular, *max-product* algorithm is used to compute the messages exchanged between variables and factor nodes. These messages can be summarized as follows [5]

variable to factor node :

$$m_{x\to f}(x) = \prod_{h\in n(x)\setminus f} m_{h\to x}(x) \tag{2.31}$$

factor node to variable :

$$m_{f\to x}(x) = \max_{\setminus\{x\}} \left(f(Z) \prod_{z\in n(f)\setminus\{x\}} m_{z\to f}(z) \right) , \tag{2.32}$$

where $n(x)$ and $n(f)$ denote the set of neighbors of node x and the set of arguments of the local function f, respectively. The marginal distributions for each variable can be obtained by the product of all incoming messages on the variable.

In order to sufficiently capture the temporal variations, the parameters ξ and ψ are assumed to evolve through a Gauss-Markov process given by

$$\begin{aligned} \xi_k &= \xi_{k-1} + w_k \\ \psi_k &= \psi_{k-1} + v_k \quad \text{for } k = 1, \ldots, N , \end{aligned}$$

where w_k and v_k are *independent and identically distributed* (i.i.d) noise variables such that $w_k, v_k \sim \mathcal{N}(0, \sigma^2)$. The posterior pdf can be expressed as

$$\begin{aligned} f(\boldsymbol{\xi}, \boldsymbol{\psi}|\boldsymbol{U}, \boldsymbol{V}) &\propto f(\boldsymbol{\xi}, \boldsymbol{\psi}) f(\boldsymbol{U}, \boldsymbol{V}|\boldsymbol{\xi}, \boldsymbol{\psi}) \\ &= f(\xi_0) \prod_{k=1}^{N} f(\xi_k|\xi_{k-1}) f(\psi_0) \prod_{k=1}^{N} f(\psi_k|\psi_{k-1}) \prod_{k=1}^{N} f(U_k|\xi_k) f(V_k|\psi_k) \end{aligned} \tag{2.33}$$

where uniform priors $f(\xi_0)$ and $f(\psi_0)$ are assumed. Define $\delta_{k-1}^{k} \triangleq f(\xi_k|\xi_{k-1}) \sim \mathcal{N}(\xi_{k-1}, \sigma^2)$, $\nu_{k-1}^{k} \triangleq f(\psi_k|\psi_{k-1}) \sim \mathcal{N}(\psi_{k-1}, \sigma^2)$, $f_k \triangleq f(U_k|\xi_k)$, $h_k \triangleq f(V_k|\psi_k)$, where

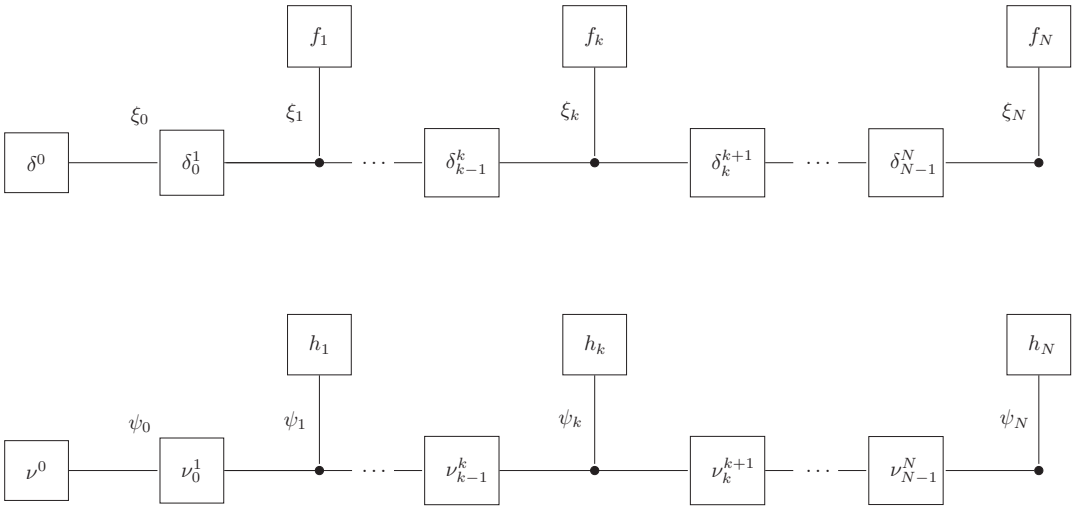

Figure 2.2: Factor graph representation of the posterior density (2.33)

the likelihood functions are given by

$$f(U_k|\xi_k) \propto \exp\left(\xi_k \eta_\xi(U_k) - \frac{\sigma^2_{\eta_{\xi,k}}}{2}\xi_k^2\right)$$
$$f(V_k|\psi_k) \propto \exp\left(\psi_k \eta_\psi(V_k) - \frac{\sigma^2_{\eta_{\psi,k}}}{2}\psi_k^2\right) \ , \tag{2.34}$$

based on (2.13). The resulting factor graph representation of the posterior pdf is shown in Fig. 2.2.

Remark 1. *Notice that the substitution in* (2.3) *renders a cycle-free nature to the factor graph. Therefore, inference by message passing on such a factor graph is indeed optimal [35].*

For the purpose of inference, only the case of constrained likelihood will be considered, since the case of an unconstrained likelihood is subsumed, as will be shown shortly. The clock offset estimator $\hat{\beta}_N$ can be obtained from $\hat{\xi}_N$ and $\hat{\psi}_N$ using (2.5).

By defining $\gamma_k \triangleq -\sigma^2_{\eta_{\xi,k}}/2$ and $\nu_k \triangleq \eta_\xi(U_k)$, the constrained likelihood function for the samples U_k can be written as

$$f_k \propto \exp\left(\gamma_k \xi_k^2 + \nu_k \xi_k\right) \mathbb{I}(U_k - \xi_k) \; . \tag{2.35}$$

The message passing strategy starts by sending a message from the factor node f_N to the variable ξ_N. The variable ξ_N relays this message to the factor node δ^N_{N-1}. The factor node computes the product of this message with the factor δ^N_{N-1} and sends the resulting message to the variable ξ_{N-1} after 'summarizing' over the variable ξ_N. In the max-product algorithm, a 'max' function is used as a summary propagation operator (cf. (2.32)). These messages are computed as follows

$$\begin{aligned}
m_{f_N \to \xi_N} &= f_N \\
m_{\xi_N \to \delta^N_{N-1}} &= f_N \\
m_{\delta^N_{N-1} \to \xi_{N-1}} &\propto \max_{\xi_N} \delta^N_{N-1} \cdot m_{\xi_N \to \delta^N_{N-1}} \\
&= \max_{\xi_N} \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(\frac{-(\xi_N - \xi_{N-1})^2}{2\sigma^2}\right) \exp\left(\gamma_N \xi_N^2 + \nu_N \xi_N\right) \mathbb{I}(U_N - \xi_N)
\end{aligned}$$

which can be rearranged as

$$m_{\delta^N_{N-1} \to \xi_{N-1}} \propto \max_{\xi_N \leq U_N} \exp\left(A_N \xi_N^2 + B_N \xi_{N-1}^2 + C_N \xi_N \xi_{N-1} + D_N \xi_N\right) , \tag{2.36}$$

where

$$A_N \triangleq -\frac{1}{2\sigma^2} + \gamma_N, \; B_N \triangleq -\frac{1}{2\sigma^2}, \; C_N \triangleq \frac{1}{\sigma^2}, \; D_N \triangleq \nu_N \; . \tag{2.37}$$

and $\gamma_N \triangleq -\sigma^2_{\eta_{\xi,N}}/2$ and $\nu_N \triangleq \eta_\xi(U_N)$. Let $\bar{\xi}_N$ be the unconstrained maximizer of

the exponent in the objective function above. This implies that

$$\bar{\xi}_N = -\frac{C_N \xi_{N-1} + D_N}{2A_N} \ . \tag{2.38}$$

Following a line of reasoning similar to Section 2.4.2 , it follows that

$$\hat{\xi}_N = \min\left(\bar{\xi}_N, U_N\right) \ .$$

However, $\bar{\xi}_N$ depends on ξ_{N-1}, which is undetermined at this stage. Hence, we need to further traverse the chain backwards. Assuming that $\bar{\xi}_N \leq U_N$, $\bar{\xi}_N$ from (2.38) can be plugged back in (2.36) which after some simplification yields

$$m_{\delta_{N-1}^N \to \xi_{N-1}} \propto \exp\left\{\left(B_N - \frac{C_N^2}{4A_N}\right)\xi_{N-1}^2 - \frac{C_N D_N}{2A_N}\xi_{N-1}\right\} \ . \tag{2.39}$$

The message passed from the variable ξ_{N-1} to the factor node δ_{N-2}^{N-1} is the product of the message (2.39) and the message received from the factor node f_{N-1}, i.e.,

$$m_{\xi_{N-1} \to \delta_{N-2}^{N-1}} = m_{\delta_{N-1}^N \to \xi_{N-1}} \cdot m_{f_{N-1} \to \xi_{N-1}} \ .$$

Upon receipt of this message, the factor node δ_{N-2}^{N-1} delivers a product of this message and the factor δ_{N-2}^{N-1} to the variable node ξ_{N-2} after maximizing over ξ_{N-1}. This message can be expressed as

$$\begin{aligned}
m_{\delta_{N-2}^{N-1} \to \xi_{N-2}} &\propto \max_{\xi_{N-1} \leq U_{N-1}} \delta_{N-2}^{N-1} \cdot m_{\xi_{N-1} \to \delta_{N-2}^{N-1}} \\
&= \max_{\xi_{N-1}} \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(\xi_{N-1} - \xi_{N-2})^2}{2\sigma^2}\right) \cdot \exp\left\{\left(B_N - \frac{C_N^2}{4A_N}\right)\xi_{N-1}^2 - \frac{C_N D_N}{2A_N}\xi_{N-1}\right\} \\
&\quad \cdot \exp\left(\gamma_{N-1}\xi_{N-1}^2 + \nu_{N-1}\xi_{N-1}\right) \mathbb{I}(U_{N-1} - \xi_{N-1}) \ .
\end{aligned}$$

After some algebraic steps, the message above can be compactly represented as

$$m_{\delta_{N-2}^{N-1}\to\xi_{N-2}} \propto \max_{\xi_{N-1}\leq U_{N-1}} \exp(A_{N-1}\xi_{N-1}^2 + B_{N-1}\xi_{N-2}^2 + C_{N-1}\xi_{N-1}\xi_{N-2} + D_{N-1}\xi_{N-1}) \,, \tag{2.40}$$

where

$$A_{N-1} \triangleq -\frac{1}{2\sigma^2} + \gamma_{N-1} + B_N - \frac{C_N^2}{4A_N}, \;\; B_{N-1} \triangleq -\frac{1}{2\sigma^2},$$
$$C_{N-1} \triangleq \frac{1}{\sigma^2}, \;\; D_{N-1} \triangleq \nu_{N-1} - \frac{C_N D_N}{2A_N} \,.$$

Proceeding as before, the unconstrained maximizer $\bar{\xi}_{N-1}$ of the objective function above is given by

$$\bar{\xi}_{N-1} = -\frac{C_{N-1}\xi_{N-2} + D_{N-1}}{2A_{N-1}}$$

and the solution to the maximization problem (2.40) is expressed as

$$\hat{\xi}_{N-1} = \min\left(\bar{\xi}_{N-1}, U_{N-1}\right) \,.$$

Again, $\bar{\xi}_{N-1}$ depends on ξ_{N-2} and therefore, the solution demands another traversal backwards on the factor graph representation in Fig. 2.2. By plugging $\bar{\xi}_{N-1}$ back in (2.40), it follows that

$$m_{\delta_{N-2}^{N-1}\to\xi_{N-2}} \propto \exp\left\{\left(B_{N-1} - \frac{C_{N-1}^2}{4A_{N-1}}\right)\xi_{N-2}^2 - \frac{C_{N-1}D_{N-1}}{2A_{N-1}}\xi_{N-2}\right\} \tag{2.41}$$

which has a form similar to (2.39). In general, for $i = 1, \ldots, N-1$, we can write

$$A_{N-i} \triangleq -\frac{1}{2\sigma^2} + \gamma_{N-i} + B_{N-i+1} - \frac{C_{N-i+1}^2}{4A_{N-i+1}},\ B_{N-i} \triangleq B_N,$$
$$C_{N-i} \triangleq C_N,\ D_{N-i} \triangleq \nu_{N-i} - \frac{C_{N-i+1}D_{N-i+1}}{2A_{N-i+1}}\ , \tag{2.42}$$

where $\gamma_{N-i} \triangleq -\sigma^2_{\eta_{\xi,N-i}}/2$, $\nu_{N-i} \triangleq \eta_\xi(U_{N-i})$ and

$$\bar{\xi}_{N-i} = -\frac{C_{N-i}\xi_{N-i-1} + D_{N-i}}{2A_{N-i}},\ \hat{\xi}_{N-i} = \min\left(\bar{\xi}_{N-i}, U_{N-i}\right). \tag{2.43}$$

Using (2.43) with $i = N-1$, it follows that

$$\bar{\xi}_1 = -\frac{C_1\xi_0 + D_1}{2A_1}, \quad \hat{\xi}_1 = \min\left(\bar{\xi}_1, U_1\right)\ . \tag{2.44}$$

Similarly, by observing the form of (2.39) and (2.41), it follows that

$$m_{\delta_0^1 \to \xi_0} \propto \exp\left\{\left(B_1 - \frac{C_1^2}{4A_1}\right)\xi_0^2 - \frac{C_1D_1}{2A_1}\xi_0\right\}\ . \tag{2.45}$$

The estimate $\hat{\xi}_0$ can be obtained by maximizing the received message in (2.45). It can be noticed from the structure of the factor graph that this maximization is inherently unconstrained i.e.,

$$\hat{\xi}_0 = \bar{\xi}_0 = \max_{\xi_0} m_{\delta_0^1 \to \xi_0} = \frac{C_1D_1}{4A_1B_1 - C_1^2}\ . \tag{2.46}$$

The estimate in (2.46) can now be substituted in (2.44) to yield $\bar{\xi}_1$, which can then be used to solve for $\hat{\xi}_1$. Clearly, this chain of calculations can be continued using recursions (2.42) and (2.43).

Define

$$g_k(x) \triangleq -\frac{C_k x + D_k}{2A_k} \ . \tag{2.47}$$

A key property of the function $g_k(.)$, which proves useful in the quest for an exact solution, can be summarized in the following lemma.

Lemma 1. *For real numbers a and b, the function $g_k(.)$ defined in (4.2) satisfies*

$$g_k\left(\min(a,b)\right) = \min\left(g_k(a), g_k(b)\right) \ .$$

Proof. See Appendix A. □

With the notation $g_k(.)$, the following chain of equalities can be conveniently written as

$$\bar{\xi}_1 = g_1\left(\hat{\xi}_0\right), \ \hat{\xi}_1 = \min\left(U_1, g_1\left(\hat{\xi}_0\right)\right)$$
$$\bar{\xi}_2 = g_2\left(\hat{\xi}_1\right), \ \hat{\xi}_2 = \min\left(U_2, g_2\left(\hat{\xi}_1\right)\right)$$

with

$$\begin{aligned} g_2\left(\hat{\xi}_1\right) &= g_2\left(\min\left(U_1, g_1\left(\hat{\xi}_0\right)\right)\right) \\ &= \min\left(g_2\left(U_1\right), g_2\left(g_1\left(\hat{\xi}_0\right)\right)\right) \ , \end{aligned} \tag{2.48}$$

where (2.48) follows from Lemma 1. The estimate $\hat{\xi}_2$ can be expressed as

$$\begin{aligned} \hat{\xi}_2 &= \min\left(U_2, \min\left(g_2\left(U_1\right), g_2\left(g_1\left(\hat{\xi}_0\right)\right)\right)\right) \\ &= \min\left(U_2, g_2\left(U_1\right), g_2\left(g_1\left(\hat{\xi}_0\right)\right)\right) \ . \end{aligned}$$

For $m \geq j$, define

$$G_j^m(.) \triangleq g_m\left(g_{m-1}(\ldots g_j\left(.\right)\right)\right) . \tag{2.49}$$

The estimate $\hat{\xi}_2$ can, therefore, be compactly represented as

$$\hat{\xi}_2 = \min\left(U_2, G_2^2\left(U_1\right), G_1^2\left(\hat{\xi}_0\right)\right) .$$

Hence, one can keep estimating $\hat{\xi}_k$ at each stage using this strategy. Note that the estimator only depends on functions of data and can be readily evaluated.

Generalizing this framework, the analytical expression for the clock offset estimate $\hat{\beta}_N$ is given by the following theorem.

Theorem 1. *The state estimates $\hat{\xi}_N$ and $\hat{\psi}_N$ for the posterior pdf in* (2.33) *can be expressed as*

$$\begin{aligned}\hat{\xi}_N &= \min\left(U_N, G_N^N\left(U_{N-1}\right), \ldots, G_2^N\left(U_1\right), G_1^N\left(\hat{\xi}_0\right)\right) \\ \hat{\psi}_N &= \min\left(V_N, \tilde{G}_N^N\left(V_{N-1}\right), \ldots, \tilde{G}_2^N\left(V_1\right), \tilde{G}_1^N\left(\hat{\psi}_0\right)\right) ,\end{aligned} \tag{2.50}$$

where $\tilde{G}(.)$ is the counterpart of $G(.)$ in (2.49) *defined for ψ. The factor graph based clock offset estimate (FGE) $\hat{\beta}_N$ is given by*

$$\hat{\beta}_N = \frac{\hat{\xi}_N - \hat{\psi}_N}{2} . \tag{2.51}$$

Proof. The proof follows from the discussion above and using (2.5). □

2.5.1 Gaussian Distributed Likelihood Function

In this case, we have

$$f\left(U_k|\xi_k\right) = \frac{1}{\sqrt{2\pi\sigma_{\xi,k}^2}} \exp\left\{-\frac{\left(U_k-\xi_k\right)^2}{2\sigma_{\xi,k}^2}\right\}$$
$$\propto \exp\left(\frac{\xi_k U_k}{2\sigma_{\xi,k}^2} - \frac{\xi_k^2}{2\sigma_{\xi,k}^2}\right) . \tag{2.52}$$

The aforementioned Gaussian distribution constitutes an unconstrained likelihood function, i.e., the domain of the pdf is independent of the unknown parameter ξ_k. It is clear from the message passing approach that at each stage k of the factor graph, the unconstrained maximizer $\bar{\xi}_k$ is the actual solution to the likelihood maximization problem

$$\max_{\xi_k} \ \exp\left(A_k\xi_k^2 + B_k\xi_{k-1}^2 + C_k\xi_k\xi_{k-1} + D_k\xi_k\right)$$

i.e., $\hat{\xi}_k = \bar{\xi}_k \ \forall k = 1, \ldots, N$. Hence, the unconstrained likelihood maximization problem is subsumed in the message passing framework for constrained likelihood maximization. It follows from Theorem 1 that $\hat{\xi}_N$ for Gaussian distributed observations U_k in (2.52) is given by

$$\hat{\xi}_N = G_1^N\left(\hat{\xi}_0\right) ,$$

where $\hat{\xi}_0$ and $G_1^N(.)$ are defined in (2.46) and (2.49), respectively. By comparing (2.52) with (2.35), we have

$$\gamma_k = -\frac{1}{2\sigma_{\xi,k}^2}, \quad \nu_k = \frac{U_k}{\sigma_{\xi,k}^2}, \tag{2.53}$$

so that the constants in (2.37) and (2.42) can be easily determined. It follows from (2.5) and (2.51) that the FGE, $\hat{\beta}_N$, can be expressed as

$$\hat{\beta}_N = \frac{G_1^N\left(\hat{\xi}_0\right) - \tilde{G}_1^N\left(\hat{\psi}_0\right)}{2} \,. \tag{2.54}$$

The behavior of $\hat{\beta}_N$ can be further investigated for the case when the noise variance σ^2 in the Gauss-Markov model goes to zero. Consider

$$g_N(\xi) = -\frac{C_N\xi + D_N}{2A_N} \,,$$

where the constants A_N, B_N, C_N and D_N are given by (2.37). After some algebraic steps, we have

$$g_N(\xi) = \frac{\sigma_\xi^2\xi + \sigma^2 U_N}{\sigma_\xi^2 + \sigma^2} \,.$$

As $\sigma^2 \to 0$, $g_N(\xi) \to \xi$. Similarly, it can be shown that $g_{N-1}(\xi) \to \xi$ as $\sigma^2 \to 0$. Hence, it follows that in the low system noise regime, as $\sigma^2 \to 0$

$$\hat{\xi}_N \to \hat{\xi}_0 = \frac{C_1 D_1}{4A_1B_1 - C_1^2} \,.$$

Therefore

$$\hat{\beta}_N \to \frac{\hat{\xi}_0 - \hat{\psi}_0}{2} \,,$$

which is equal to the ML estimator (2.20).

2.5.2 Log-Normally Distributed Likelihood Function

The log-normally distributed likelihood function in the Bayesian regime can be expressed as

$$f(U_k|\xi_k) = \frac{1}{U_k\sigma_{\xi,k}\sqrt{2\pi}} \exp\left(-\frac{(\log U_k - \xi_k)^2}{2\sigma_{\xi,k}^2}\right)$$
$$\propto \exp\left(\frac{\xi_k \log(U_k)}{2\sigma_{\xi,k}^2} - \frac{\xi_k^2}{2\sigma_{\xi,k}^2}\right) . \tag{2.55}$$

By comparing (2.55) and (2.35), we have

$$\gamma_k = -\frac{1}{2\sigma_{\xi,k}^2}, \qquad \nu_k = \frac{\log U_k}{\sigma_{\xi,k}^2},$$

so that the recursively evaluated constants in (2.37) and (2.42) can be easily determined. Clearly, the only difference here with the Gaussian distribution is a redefinition of ν_k. The expression of $\hat{\xi}_N$ in this case is again

$$\hat{\xi}_N = G_1^N\left(\hat{\xi}_0\right) ,$$

where $G_1^N(.)$ and $\hat{\xi}_0$ are given by (2.49) and (2.46), respectively. Hence, the FGE $\hat{\beta}_N$ can be expressed as

$$\hat{\beta}_N = \frac{G_1^N\left(\hat{\xi}_0\right) - \tilde{G}_1^N\left(\hat{\psi}_0\right)}{2} . \tag{2.56}$$

Again, as the Gauss-Markov system noise $\sigma^2 \to 0$, the above estimator approaches its ML counterpart (2.22).

2.5.3 Exponential Distribution

Theorem 1 can also be used to derive a Bayesian estimator $\hat{\xi}_N$ for the exponentially distributed network delay case considered in [30]. In this case, we have

$$\begin{aligned} f(U_k|\xi_k) &= \lambda_\xi \exp\left(-\lambda_\xi (U_k - \xi_k)\right) \mathbb{I}(U_k - \xi_k) \\ &\propto \exp(\lambda_\xi \xi_k) \mathbb{I}(U_k - \xi_k) \ , \end{aligned} \tag{2.57}$$

where λ_ξ^{-1} is the mean network delay of X_k. A comparison of (2.57) with (2.35) reveals that

$$\gamma_k = 0, \quad \nu_k = \lambda_\xi \ .$$

The constants A_k, B_k, C_k and D_k are thus given by

$$A_k = B_k = -\frac{1}{2\sigma^2}, \quad C_k = \frac{1}{\sigma^2}, \quad D_k = \lambda_\xi$$

for all $k = 1, \ldots, N$. Using Theorem 1 and (2.57), it can be easily verified that

$$G_{N-i}^N(U_{N-1-i}) = U_{N-1-i} + (i+1)\,\lambda_\xi \sigma^2 \ ,$$

for $i \in \{0, 1, \ldots, N-2\}$. The estimator $\hat{\xi}_0$ can be evaluated as

$$\hat{\xi}_0 = \frac{C_1 D_1}{4A_1 B_1 - C_1^2} = +\infty \ .$$

This implies that

$$G_1^N(\hat{\xi}_0) = +\infty \ . \tag{2.58}$$

Using (2.58) and Theorem 1, it readily follows that

$$\hat{\xi}_N = \min(U_N, U_{N-1} + \lambda_\xi \sigma^2, U_{N-2} + 2\lambda_\xi \sigma^2, \ldots, U_1 + (N-1)\lambda_\xi \sigma^2) \; . \tag{2.59}$$

A similar expression can also be written for $\hat{\psi}_N$. The estimate $\hat{\beta}_N$ can be expressed as

$$\begin{aligned} \hat{\beta}_N = & \frac{1}{2} \min(U_N, U_{N-1} + \lambda_\xi \sigma^2, U_{N-2} + 2\lambda_\xi \sigma^2, \ldots, U_1 + (N-1)\lambda_\xi \sigma^2) - \\ & \frac{1}{2} \min(V_N, V_{N-1} + \lambda_\psi \sigma^2, V_{N-2} + 2\lambda_\psi \sigma^2, \ldots, V_1 + (N-1)\lambda_\psi \sigma^2) \; . \end{aligned} \tag{2.60}$$

As the Gauss-Markov system noise $\sigma^2 \to 0$, (2.60) yields

$$\hat{\beta}_N \to \hat{\beta}_{\mathrm{ML}} = \frac{\min(U_N, \ldots, U_1) - \min(V_N, \ldots, V_1)}{2} \; ,$$

which is the ML estimator given by (2.30).

2.6 Simulation Results

This section aims to corroborate the theoretical results derived in preceding sections by conducting simulation studies in various scenarios. The measure of fidelity used to rate this performance is the MSE of the estimators for β and β_N. The parameter choice is $\sigma_\xi = \sigma_\psi = 0.1$ for both Gaussian and log-normally distributed likelihoods, while $\lambda_\xi = \lambda_\psi = 10$ for exponentially distributed likelihood functions. The performance of the estimators is compared by plotting them against the theoretical lower bounds. A general derivation of the lower bounds is presented in Section 5 where their relation with clock offset estimation is also discussed. For a detailed explanation of the lower bounds, see Section 5.

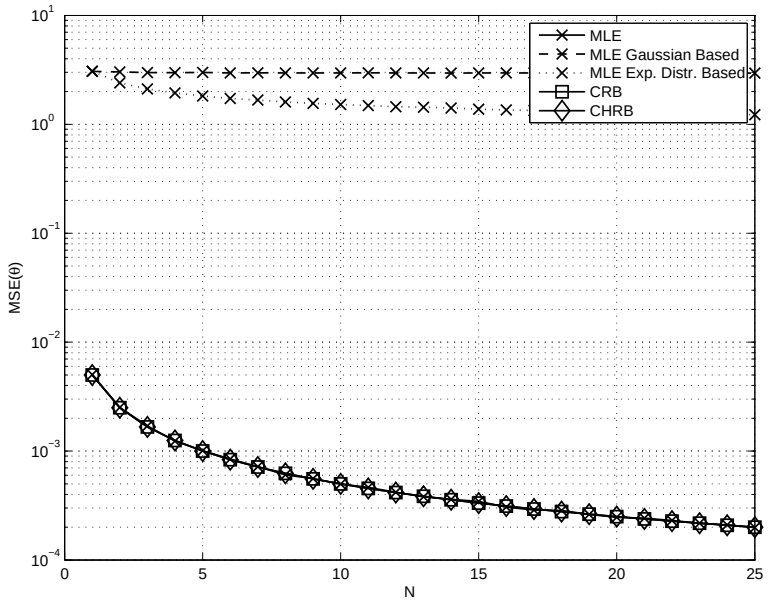

Figure 2.3: MSE and bounds for estimating β by using the MLE with log-normal likelihood.

2.6.1 Log-normal Distribution

The existing approaches in literature only consider the Gaussian and the exponential cases, therefore (2.22) is a new result in the state-of-the-art about clock offset estimation.

2.6.1.1 Classical Frequentist Inference Approach

Fig. 2.3 shows a comparison between the proposed ML estimator (MLE) (2.22) in case of a log-normally distributed likelihood (2.21) with MLEs which (wrongly) assume that the likelihood is Gaussian and exponentially distributed, respectively. The plot shows that the latter approaches are not robust with respect to the likelihood distribution, and their performance is extremely poor if their assumptions do not hold. In addition, Fig. 2.3 also shows that the proposed MLE (2.22) is efficient since it attains the CRB (as well as the CHRB).

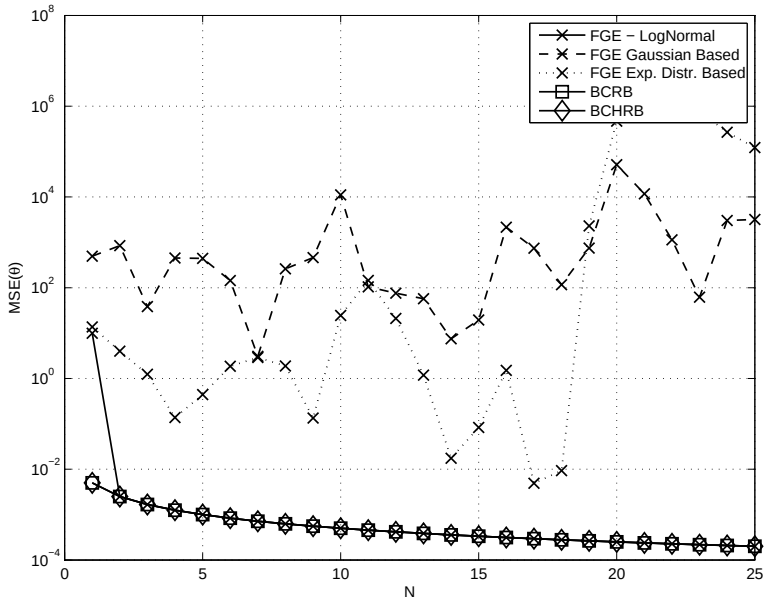

Figure 2.4: MSE and bounds for estimating β_N by using FGE with log-normal likelihood.

2.6.1.2 Bayesian Estimation Framework

Fig. 2.4 plots the MSE of the FGE (2.56) as well as the BCRB and the BCHRB when the likelihoods are log-normally distributed (2.55), and $\sigma = 10^{-4}$. Firstly, it can be seen that the MSE coincides with the estimation bounds. Secondly, as in the classical estimation case, if we were to (wrongly) assume a Gaussian or exponential distribution for the likelihoods (2.34), the resulting FGEs would perform poorly. This fact is evident in Fig. 2.4 by observing the unboundedness and unpredictability of the dashed curve (Gaussian assumption for the likelihoods) and the dotted curve (likelihoods assumed exponentially distributed). This clearly establishes that the FGE (2.56), obtained assuming log-normally distributed likelihoods, shows a marked improvement with respect to existing estimators.

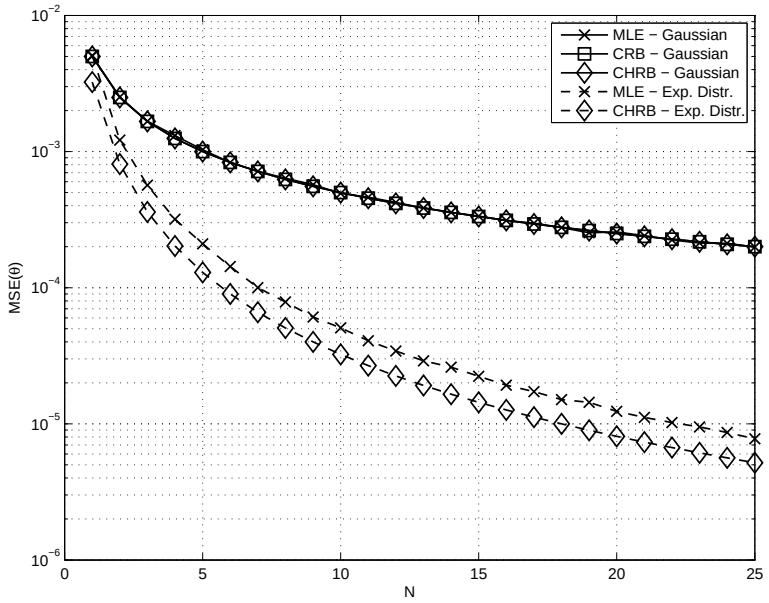

Figure 2.5: MSE and bounds for estimating β by using the MLE with Gaussian and exponentially distributed likelihood.

2.6.2 Estimator Performance vs Estimation Bounds

It will also be useful to asses the performance of the MLEs in Gaussian and exponential cases derived in Section 2.4 against the various benchmark estimation bounds derived in Section 5. Similarly, the FGEs for Gaussian and exponential distributions, proposed in Section 2.5, can also be compared with the Bayesian bounds to study their MSE performance.

2.6.2.1 Classical Frequentist Inference Approach

Fig. 2.5 shows the performance comparison between the MSE of the MLEs (2.20) and (2.30) for Gaussian and exponentially distributed likelihood functions against the CRB and the CHRB, respectively. Firstly, it is evident that in the case of Gaussian distribution, the CRB and the CHRB coincide. Moreover, the MSE of $\hat{\beta}_{\mathrm{ML}}$ also coincides with the aforementioned bounds. On the other hand, for an exponentially distributed likelihood function, due to its lack of regularity, the CRB cannot be

derived, thus only the CHRB is shown. It can be observed that the MSE of $\hat{\beta}_{\mathrm{ML}}$ is fairly close to the CHRB, even though it does not coincide with it. From Fig. 2.5 the MSE of the MLEs for the Gaussian and exponential distribution case can be also compared. In order to ensure a fair comparison, parameters are chosen to have the same variance of the observations for both distributions. From the MSE curves, one can infer that the MSE in case of an exponentially distributed likelihood is lower than the one for a Gaussian distribution as the number of observations N increases. This behavior is expected since the MSE decays proportionally to $1/N$ for a Gaussian distribution, while in the exponential distribution case this decay is proportional to $1/N^2$ (see Appendix B).

2.6.2.2 Bayesian Estimation Framework

In Fig. 2.6, the MSE performance of the FGEs $\hat{\beta}_N$ (2.54) and (2.60) is compared with the BCRB and the BCHRB for $\sigma = 10^{-4}$. As in the classical estimation scenario, it is evident that for Gaussian distributed likelihoods, the MSE using (2.54) for $\hat{\beta}_N$ coincides with the reported bounds. The MSE of the FGE derived assuming exponentially distributed likelihoods (2.60) is plotted against the BCHRB as well in Fig. 2.6. It is clear that the MSE is quite close to the BCHRB, although not coinciding with it, as exactly was the case in the classical estimation framework.

2.6.3 Comparing Classical Frequentist and Bayesian Frameworks

The estimators proposed in the classical frequentist and the Bayesian frameworks can also be compared with each other based on their MSE performance as the system noise decreases. The aim here is to show that the latter approaches the former as $\sigma \to 0$.

Fig. 2.7 depicts the MSE for the cases of Gaussian, exponential and log-normal distribution for the likelihoods with $N = 25$. In the plot, the horizontal lines rep-

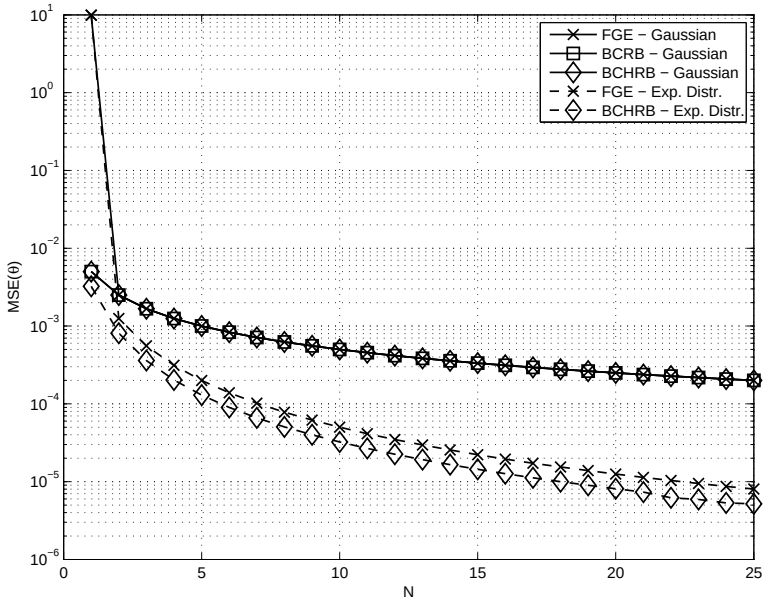

Figure 2.6: MSE and bounds for estimating β_N by using FGE with Gaussian and exponentially distributed likelihood.

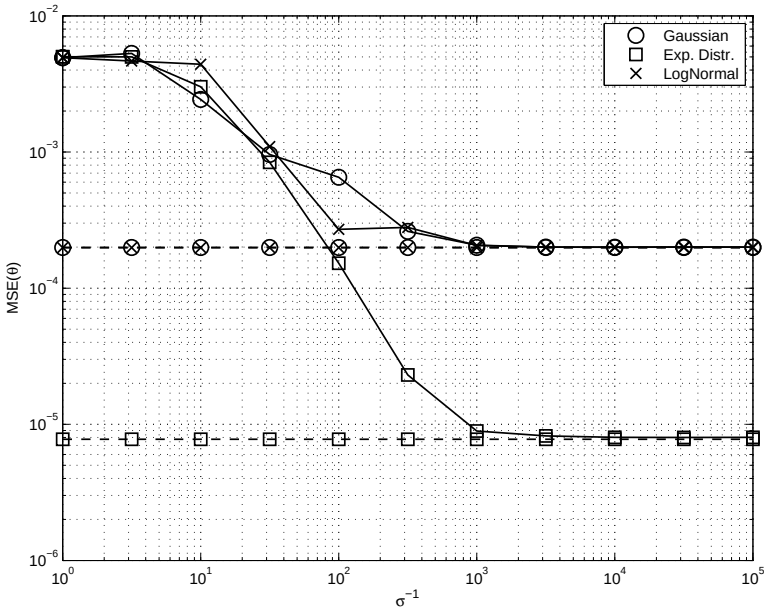

Figure 2.7: MSE in estimating of β_N vs σ. Solid lines: FGE; dashed lines: MLE (classical framework).

resent the MSEs in the classical framework, obtained with the MLEs, as shown in (B.1) and (B.2) in Appendix B. It can be observed that, for all the three considered distributions, the MSE obtained by using the FGE for estimating β approaches the MSE of the MLEs for $\sigma < 10^{-3}$.

2.7 Summary

Based on a two-way timing message exchange scenario, this section proposes a unified framework for the clock offset estimation problem when the likelihood function of the observation time stamps is Gaussian, exponential and log-normally distributed. An analytical approach for the ML estimation of clock offsets is presented. The results known thus far for Gaussian and exponentially distributed network delays are subsumed in the general approach while the ML estimator is derived when the likelihood function is log-normally distributed. In order to study the case of a possibly time-varying clock offset, a Bayesian approach is also studied using factor graphs. The novel message passing strategy results in an exact solution of the time-varying clock offset estimation problem. The theoretical findings are corroborated by simulation studies conducted in various scenarios.

3. DISTRIBUTED NETWORK-WIDE CLOCK SYNCHRONIZATION

3.1 Introduction

Developing a network-wide synchronization algorithm using message passing on factor graphs constitutes a natural extension of the Bayesian clock offset estimation approach for pairwise synchronization outlined in Section 2. A network-wide synchronization algorithm based on belief propagation assuming Gaussian distributed network delays is proposed in [38]. It was observed by experiments in [48] that the Minimum Link Delay algorithm proposed in [1] had vastly superior performance, and this algorithm was later derived independently in [30] assuming exponentially distributed network delays. Therefore, the assumption of an exponential distribution seems to be the best suited through experimental observations and it becomes imperative to study the network-wide clock synchronization problem in such a case. In this section, we aim to study network-wide clock synchronization for exponentially distributed network delays.

3.2 Main Contributions

Our main contributions in this section are as follows.

1. A network-wide clock synchronization algorithm is proposed in case of exponentially distributed network delays by representing the sensor network as a factor graph. Inference on the factor graph is performed by using max-product message passing. This yields an iterative update of the belief whose maximization provides the clock offset. This also extends the framework in [3] to the much more challenging case of network-wide synchronization.

2. The proposed algorithm is fully distributed since the clock offset of each node

is determined at the node itself, instead of centralized processing at a fusion center.

3. A simulation study is then performed to show that the proposed algorithm converges in various network topologies of interest.

3.3 System Model

Consider a strictly connected network of $N+1$ sensor nodes that share a wireless channel to communicate. Each sensor node bases its notion of time on an internal oscillator. It is assumed that clock inconsistency between different nodes is due to the presence of clock offset, so that the time measured by node i, denoted by T_i, can generally be expressed as [43, 58]

$$T_i = t + \beta_i \ ,$$

where t is the actual time at the reference node and β_i represents the clock offset of node i, $i = 1, \ldots, N$. Clock offsets are assumed mutually independent and their prior beliefs, denoted by $p_i(\beta_i)$, are considered uniform over the entire real axis. The clock of node 0 is assumed to provide the reference time to the network, therefore its clock offset is set to zero, i.e., $\beta_0 = 0$ and $p_0(\beta_0) = \delta(\beta_0)$.

The radio coverage area of each node is assumed circular with a specific radius so that communication between two nodes is possible if and only if they are in the coverage area of each other, i.e., their distance is less than the communication radius. The communication network topology is considered time invariant and described by the link set $L \triangleq \{(i,j) : \text{there is a link between nodes } i \text{ and } j\}$

The focus in this section will be on the message exchange rules between any two nodes, thus setting the stage for the network-wide synchronization algorithm

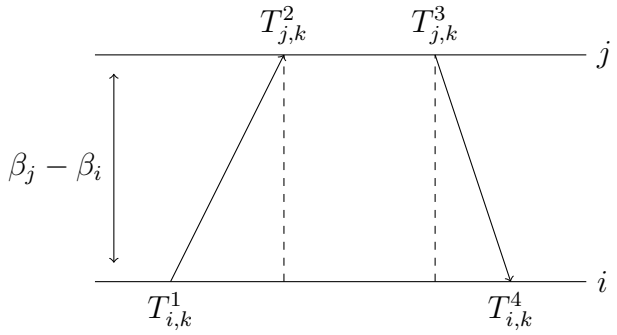

Figure 3.1: Two-way message exchange between nodes i and j.

described in Section 3.4.

3.3.1 Two-Way Message Exchange Mechanism

Assume that nodes i and j are in the coverage area of each other, i.e., $(i, j) \in L$. The process of two-way message exchange involves sending two messages, one in each link direction, and is depicted in Fig. 3.1 for the k-th round. The data transmission is started by node i, which sends a message to node j containing the timestamp $T^1_{i,k}$ with respect to the clock of node i. Node j records the time $T^2_{j,k}$ at which this message is received, based on its own clock. After a predetermined period of time, node j replies with a message carrying the timestamps $T^2_{j,k}$ and $T^3_{j,k}$, the latter being the second message sending timestamp. Finally, node i receives this message at time $T^4_{i,k}$, with respect to its own time scale. Since both nodes need all the timestamps for processing purposes, we assume that the first message in the $(k + 1)$-th round carries inside the timestamp $T^4_{i,k}$ as well. The message exchange is repeated for $k \in [1, \ldots, K]$, while node i finally sends an additional message to node j containing the timestamp $T^4_{i,K}$, so that both nodes have all the timestamps[1] .

The message transmission process described above suffers from different types of fixed and random network delays. A thorough description of such delays can be

[1]An additional message may not be necessary if piggybacking is adopted.

found in [21]. In this work, the fixed delay between nodes i and j, denoted by d_{ij}, is considered unknown but symmetric in uplink and downlink transmission. The random portion of delays in uplink and downlink are asymmetric and are denoted by $X_{ij,k}$ and $Y_{ij,k}$, respectively. The pair-wise message exchange between nodes i and j can be expressed as [38]

$$\begin{aligned} T^2_{j,k} &= T^1_{i,k} + \beta_j - \beta_i + d_{ij} + X_{ij,k} \\ T^4_{i,k} &= T^3_{j,k} + \beta_i - \beta_j + d_{ij} + Y_{ij,k} \, . \end{aligned} \tag{3.1}$$

There is an extensive literature about candidate distributions that can accurately model the random delays $X_{ij,k}$ and $Y_{ij,k}$, based on observing network traffic followed by data fitting [11]. In this section, the random delays are assumed to be independent and exponentially distributed with mean $1/\lambda$, $\lambda > 0$. This choice is motivated by the fact that among all distributions with support in $[0, +\infty)$ and mean $1/\lambda$, the exponential distribution has the maximum entropy. The fitness of the exponential distribution with the network traffic is also experimentally demonstrated in [48], based on the Minimum Link Delay algorithm proposed in [1] and its later derivation in [30] assuming exponentially distributed network delays.

By recalling the steps performed in [1], the timestamps are processed to obtain

$$\begin{aligned} U_{ij,k} &\triangleq T^2_{j,k} - T^1_{i,k} = d_{ij} + (\beta_j - \beta_i) + X_{ij,k} \\ V_{ij,k} &\triangleq T^4_{i,k} - T^3_{j,k} = d_{ij} - (\beta_j - \beta_i) + Y_{ij,k} \, . \end{aligned} \tag{3.2}$$

It was shown in [30] that the maximum likelihood estimator (MLE) for the clock

offset difference $(\beta_j - \beta_i)$ is given by

$$S_{ij} \triangleq \frac{U_{ij,(1)} - V_{ij,(1)}}{2} \ , \tag{3.3}$$

where $U_{ij,(1)}$ and $V_{ij,(1)}$ denote the first order statistics of the data sets $\{U_{ij,k}\}_{k=1}^{K}$ and $\{V_{ij,k}\}_{k=1}^{K}$, respectively. It was proved in [3] that $[U_{ij,(1)}, V_{ij,(1)}]^T$ constitutes a sufficient statistics for estimating $(\beta_j - \beta_i)$, and that (3.3) is also the uniformly minimum variance unbiased (UMVU) estimator for $(\beta_j - \beta_i)$. Based on the aforementioned properties, it suffices to work with pre-processed data S_{ij} in order to make the network-wide algorithm computationally sustainable and scalable.

Using the definition of S_{ij} in (3.3) and the system model expressed in (3.2), we can write

$$S_{ij} = (\beta_j - \beta_i) + Z_{ij} \ , \tag{3.4}$$

where $Z_{ij} \stackrel{\Delta}{=} \left(X_{ij,(1)} - Y_{ij,(1)}\right)/2$ is a zero mean Laplace random variable with parameter $1/(2K\lambda)$, i.e., $Z_{ij} \sim \mathscr{L}\left(0, 1/(2K\lambda)\right)$. The factor $1/K$ comes from computing the minimum of K independent exponential random variables. Consequently, $S_{ij} \sim \mathscr{L}\left(\beta_j - \beta_i, 1/(2K\lambda)\right)$, so that its probability density function (pdf) can be expressed as

$$p\left(S_{ij}|\beta_i, \beta_j\right) = K\lambda \exp\left(-2K\lambda|S_{ij} - \beta_j + \beta_i|\right) \ . \tag{3.5}$$

The goal of this work is to infer the value of the clock offset β_i for all i, using data S_{ij} gathered from the message exchanges between pairs of nodes $(i, j) \in L$. Clearly, an estimator for β_i does not depend solely on the data exchanged between nodes i and $j \in \mathcal{N}_i$, with $\mathcal{N}_i$ being the set of neighbors of node i. Rather, all nodes in the network play a role through data exchanges between pairs of neighbors. Inference

about β_i can be obtained by

$$\hat{\beta}_i \triangleq \arg\max_{\beta_i} p_i\left(\beta_i|\boldsymbol{S}\right)$$

where the *a-posteriori* pdf, $p_i\left(\beta_i|\boldsymbol{S}\right)$, is given by

$$p_i\left(\beta_i|\boldsymbol{S}\right) = \int_{\tilde{\boldsymbol{\beta}}_i} p\left(\boldsymbol{\beta}|\boldsymbol{S}\right) \mathrm{d}\tilde{\boldsymbol{\beta}}_i \tag{3.6}$$

where $\boldsymbol{\beta} \triangleq [\beta_1, \ldots, \beta_N]$, the symbol $\tilde{\boldsymbol{\beta}}_i$ denotes the vector composed of all the offset variables except β_i, and $\boldsymbol{S}$ is the (antisymmetric) $(N+1)\times(N+1)$ matrix composed of entries S_{ij} given by (3.3) if $(i,j) \in L$ and zero otherwise. The next section shows how the problem of inferring the clock offsets β_i is solved via belief propagation on a suitably defined factor graph.

3.4 The Belief Propagation Algorithm

This section describes the graphical approach adopted to solve the problem of network-wide clock synchronization and outlines the message exchange rules between neighboring nodes in order to derive an iterative algorithm for clock offset estimation.

The overall *a-posteriori* pdf in (3.6) can be factorized using Bayes' rule and the independence of the link delays in (3.1) as well as the independence of the clock offsets, as follows

$$p\left(\boldsymbol{\beta}|\boldsymbol{S}\right) \propto \prod_{(i,j)\in L} h_{ij}\left(\beta_i, \beta_j\right) \cdot \prod_{i=0}^{N} p_i\left(\beta_i\right) \ , \tag{3.7}$$

where

$$h_{ij}\left(\beta_i, \beta_j\right) \triangleq p\left(S_{ij}|\beta_i, \beta_j\right) \ .$$

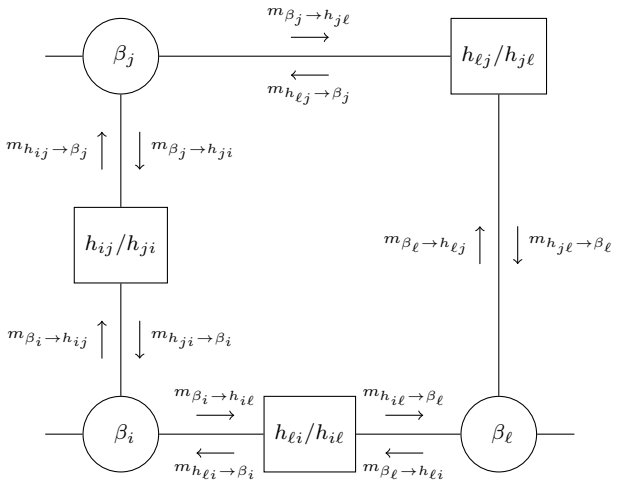

Figure 3.2: Example of propagation of messages (3.8) and (3.9) in a factor graph.

The factorization in (3.7) naturally leads to a factor graph representation of the problem. By representing random variables with circles and factors with squares, a factor graph such as the one in Fig. 3.2 can be easily obtained [35]. Message passing on the resulting factor graph is then applied to obtain, either exactly or approximately (depending on whether the graph has cycles), the posterior pdf $p_i\left(\beta_i|\boldsymbol{S}\right)$, which can then be maximized to yield the clock offset β_i. We use max-product message passing algorithm to infer β_i in (3.6), primarily due to its greater analytical tractability in our synchronization problem, as compared to sum-product message passing algorithm.

In a bipartite graph, belief propagation algorithms require the exchange of messages both from a variable node to a connected factor node and from a factor node to a variable node connected to it. In the max-product algorithm, the message $m_{\beta_i \to h_{i\ell}}\left(\beta_i\right)$ sent from the variable node β_i to the factor node $h_{i\ell}$ is given by the product of the prior $p_i(\beta_i)$ and the incoming messages from all factor nodes except

the l-th one, i.e.,

$$m_{\beta_i \to h_{i\ell}}(\beta_i) = p_i(\beta_i) \cdot \prod_{j \in \mathcal{N}_i, j \neq \ell} m_{h_{ji} \to \beta_i}(\beta_i) \ . \tag{3.8}$$

At the factor node, the marginalization process is performed using the 'max' operator, so that the message $m_{h_{i\ell} \to \beta_\ell}(\beta_\ell)$ from the factor node $h_{i\ell}$ to the variable node β_ℓ is given by

$$m_{h_{i\ell} \to \beta_\ell}(\beta_\ell) = \max_{\beta_i} \left[m_{\beta_i \to h_{i\ell}}(\beta_i) \, h_{i\ell}(\beta_i, \beta_\ell) \right] \ . \tag{3.9}$$

Fig. 3.2 also shows an example of message exchange in a factor graph. At node i, the belief about β_i, obtained from all the connected factor nodes as well as the prior, is given by

$$b_i(\beta_i) = p_i(\beta_i) \prod_{j \in \mathcal{N}_i} m_{h_{ji} \to \beta_i}(\beta_i) \ . \tag{3.10}$$

This belief $b_i(\beta_i)$ represents the posterior pdf $p_i(\beta_i | \boldsymbol{S})$ in (3.6). An estimate of β_i is then obtained as

$$\hat{\beta}_i = \arg\max_{\beta_i} b_i(\beta_i) \ . \tag{3.11}$$

3.4.1 Message Computation

The message circulation begins with messages sent from the leaf variable nodes. At the first iteration, denoted by $t = 0$ in the superscript, for all non-reference nodes, i.e., $i \neq 0$, (3.8) is simply equal to the prior, since no messages have arrived from

neighboring nodes yet. This implies that

$$m^{(0)}_{\beta_i \to h_{i\ell}}(\beta_i) = p_i(\beta_i) \ ,$$

while the subsequent message from the factor node $h_{i\ell}$ to the variable node β_ℓ is given by

$$\begin{aligned} m^{(0)}_{h_{i\ell} \to \beta_\ell}(\beta_\ell) &= \max_{\beta_i} \left[m^{(0)}_{\beta_i \to h_{i\ell}}(\beta_i)\, h_{i\ell}(\beta_i, \beta_\ell) \right] \\ &= \max_{\beta_i} \left[p_i(\beta_i)\, p(S_{i\ell}|\beta_i, \beta_\ell) \right] \\ &\propto 1 \ , \ . \end{aligned} \tag{3.12}$$

where (3.12) follows from (3.5) and the fact that $p_i(\beta_i)$ is uniform. On the other hand, for the reference node $i = 0$, the message sent by the variable node β_0 to the factor node $h_{0\ell}$, $\ell \in \mathcal{N}_0$, can be written as

$$m^{(0)}_{\beta_0 \to h_{0\ell}}(\beta_0) = p_0(\beta_0) = \delta(\beta_0) \ .$$

The subsequent message from the factor node $h_{0\ell}$ to the variable node β_ℓ can be expressed as (cf. (3.9))

$$\begin{aligned} m^{(0)}_{h_{0\ell} \to \beta_\ell}(\beta_\ell) &\propto \max_{\beta_0} \left[m^{(0)}_{\beta_0 \to h_{0\ell}}(\beta_0)\, h_{0\ell}(\beta_0, \beta_\ell) \right] \\ &\propto \max_{\beta_0} \left[\delta(\beta_0)\, p(S_{0\ell}|\beta_0, \beta_\ell) \right] \\ &\propto \exp\left(-2K\lambda|\beta_\ell - S_{0\ell}|\right) \ , \end{aligned} \tag{3.13}$$

where (3.13) follows from (3.5). This implies that the message $m^{(0)}_{h_{0\ell} \to \beta_\ell}$ is proportional to a Laplace pdf with mean $S_{0\ell}$ and parameter $1/(2K\lambda)$. In other words, at the first

iteration, the messages coming to the variable node β_ℓ from the factor nodes other than $h_{0\ell}$, (i.e., $m^{(0)}_{h_{q\ell}\to\beta_\ell}$, with $q \in \mathcal{N}_\ell$, $q \neq 0$) are just constants given by (3.12).

At the second iteration $t = 1$, the variable node β_l uses (3.8) to compute the message

$$\begin{aligned} m^{(1)}_{\beta_\ell\to h_{\ell j}}(\beta_\ell) &= p_\ell(\beta_\ell) \cdot \prod_{q\in\mathcal{N}_\ell, q\neq j} m^{(0)}_{h_{q\ell}\to\beta_\ell}(\beta_\ell) \\ &\propto m^{(0)}_{h_{0\ell}\to\beta_\ell}(\beta_\ell) \ . \end{aligned} \tag{3.14}$$

As a consequence, the message from the factor node $h_{\ell j}$ to the variable node β_j can be expressed as

$$\begin{aligned} m^{(1)}_{h_{\ell j}\to\beta_j}(\beta_j) &= \max_{\beta_\ell}\left[m^{(1)}_{\beta_\ell\to h_{\ell j}}(\beta_\ell)\, h_{\ell j}(\beta_j,\beta_\ell)\right] \\ &= \max_{\beta_\ell}\left[m^{(0)}_{h_{0\ell}\to\beta_\ell}(\beta_\ell)\, p(S_{\ell j}|\beta_\ell,\beta_j)\right] \ , \end{aligned} \tag{3.15}$$

where (3.15) follows from (3.7) and (3.14). The following lemma provides a closed form expression for the message $m^{(1)}_{h_{\ell j}\to\beta_j}$, and also serves as a stepping stone for the computation of the generic message sent by a factor node to a variable node at the general iteration index t.

Lemma 2. *The message $m^{(1)}_{h_{\ell j}\to\beta_j}$ is proportional to a Laplace pdf with mean $W^{(1)}_{\ell j}$ and parameter $1/(2K\lambda)$, i.e.,*

$$m^{(1)}_{h_{\ell j}\to\beta_j}(\beta_j) \propto \exp\left(-2K\lambda|\beta_j - W^{(1)}_{\ell j}|\right) \ . \tag{3.16}$$

where $W^{(1)}_{\ell j} \triangleq S_{\ell j} + S_{0\ell}$ is to be interpreted as the message received by the node β_ℓ from its neighbors except the node β_j at iteration $t = 1$.

Proof. Using its definition, the message can be written as

$$m^{(1)}_{h_{\ell j}\to\beta_j}(\beta_j) \propto$$
$$\max_{\beta_\ell} \exp\left[-2K\lambda\left(|\beta_\ell - S_{0\ell}| + |\beta_\ell - \beta_j + S_{\ell j}|\right)\right] ,$$

and the problem reduces to the minimization with respect to β_ℓ of the term

$$\psi(\beta_\ell) = |\beta_\ell - S_{0\ell}| + |\beta_\ell - \beta_j + S_{\ell j}| .$$

By inspection, it follows that $\psi(\beta_\ell)$ is a piece-wise linear function on β_ℓ, with slopes

$$\begin{aligned}
-2, &\qquad \text{when } \beta_\ell \le \min\{S_{0\ell}, \beta_j - S_{\ell j}\} \\
0, &\qquad \text{when } \min\{S_{0\ell}, \beta_j - S_{\ell j}\} < \beta_\ell \le \max\{S_{0\ell}, \beta_j - S_{\ell j}\} \\
2, &\qquad \text{when } \beta_\ell > \max\{S_{0\ell}, \beta_j - S_{\ell j}\} .
\end{aligned}$$

Therefore, choosing any β_ℓ inside the interval $[\min\{S_{0\ell}, \beta_j - S_{\ell j}\}, \max\{S_{0\ell}, \beta_j - S_{\ell j}\}]$ provides the solution (3.16). □

At a general iteration t, it is clear that several neighboring nodes of a particular node i that have updated their beliefs after receiving some communication from the reference node, either directly or indirectly, will be sending messages similar to (3.16). Messages from such nodes will be termed as *non-constant messages* in the sequel. Hence, it follows that at a general iteration t, (3.8) is a product of a number of Laplace distributions, taking the form

$$m^{(t)}_{\beta_i\to h_{i\ell}}(\beta_i) \propto \exp\left(-2K\lambda \sum_{n=1}^{r^{(t)}_{i,\ell}-1} \left|\beta_i - W^{(t)}_{i\ell}(n)\right|\right) , \tag{3.17}$$

where $\left(r_{i,\ell}^{(t)}-1\right)$ is the number of neighbors of node i, other than node ℓ, that have sent non-constant messages at iteration $(t-1)$. The sequence $\{W_{i\ell}^{(t)}(n)\}$ denotes the non-constant messages sent by neighbors of node i other than node ℓ. Without loss of generality, $\{W_{i\ell}^{(t)}(n)\}_n$ are assumed sorted in an increasing order, i.e., $-\infty < W_{i\ell}^{(t)}(1) \leq W_{i\ell}^{(t)}(2) \leq \ldots \leq W_{i\ell}^{(t)}\left(r_{i,\ell}^{(t)}-1\right) < +\infty$. It must be remarked that $\{W_{i\ell}^{(t)}(n)\}$ are fully determined by data (cf. (3.16) for $t=1$).

Using (3.5), (3.7) and (3.17), the message (3.9) at iteration t can be written as

$$\begin{aligned} m_{h_{i\ell}\to\beta_\ell}^{(t)}(\beta_\ell) &= \max_{\beta_i}\left[m_{\beta_i\to h_{i\ell}}^{(t)}(\beta_i)\, h_{i\ell}(\beta_i,\beta_\ell)\right] \\ &\propto \max_{\beta_i}\exp\left[-2K\lambda\left(|\beta_i-\beta_\ell+S_{i\ell}|+\sum_{n=1}^{r_{i,\ell}^{(t)}-1}\left|\beta_i-W_{i\ell}^{(t)}(n)\right|\right)\right]. \end{aligned} \tag{3.18}$$

The following lemma provides an approximate closed form expression for the message $m_{h_{i\ell}\to\beta_\ell}^{(t)}$ at iteration t.

Lemma 3. *The message $m_{h_{i\ell}\to\beta_\ell}^{(t)}$ can be approximated by a Laplace pdf with mean $S_{i\ell}+C_{i\to\ell}^{(t)}$ and parameter $1/(2K\lambda)$, and can be expressed as*

$$m_{h_{i\ell}\to\beta_\ell}^{(t)}(\beta_\ell) \,\widetilde{\propto}\, \exp\left(-2K\lambda\left|\beta_\ell-\left(S_{i\ell}+C_{i\to\ell}^{(t)}\right)\right|\right), \tag{3.19}$$

where $\widetilde{\propto}$ should be read "approximately proportional to", and with

$$C_{i\to\ell}^{(t)} = \begin{cases} W_{i\ell}^{(t)}\left(\frac{r_{i,\ell}^{(t)}}{2}\right), & \text{even } r_{i,\ell}^{(t)}, \\ \frac{1}{2}\left[W_{i\ell}^{(t)}\left(\frac{r_{i,\ell}^{(t)}-1}{2}\right)+W_{i\ell}^{(t)}\left(\frac{r_{i,\ell}^{(t)}+1}{2}\right)\right], & \\ & \text{odd } r_{i,\ell}^{(t)}. \end{cases} \tag{3.20}$$

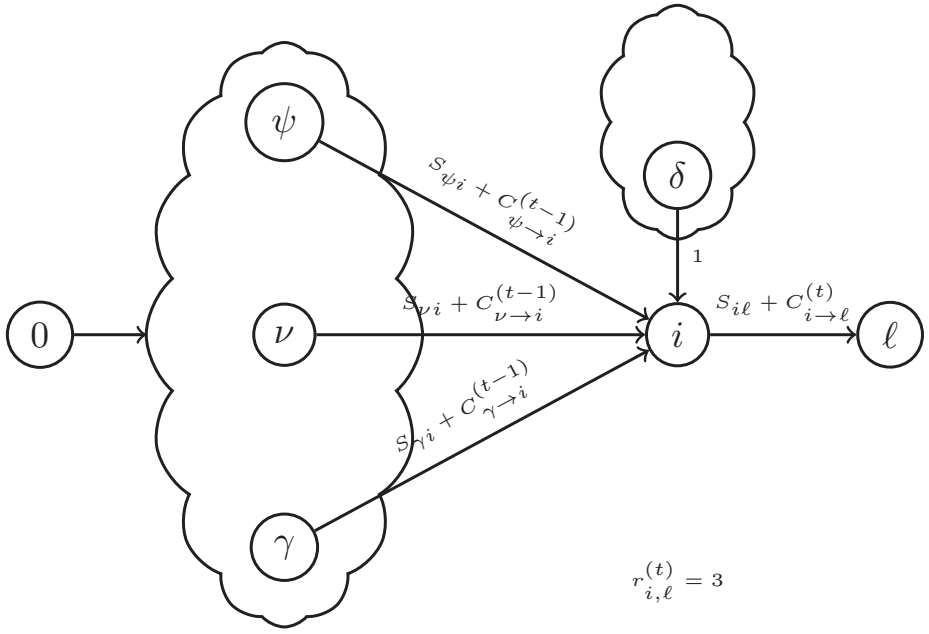

Figure 3.3: Example of message circulation between sensor nodes: node i computes the message to send to the node ℓ at time t based on the information messages received at time $t-1$ from the nodes γ, ν and ψ and the information-less constant message received from node δ.

Proof. See Appendix D. □

An example of message propagation between sensor nodes is shown in Fig. 3.3, in which it is assumed that the messages from the variable nodes to the factor nodes are not actually transmitted as they only serve as intermediate steps for computing the messages (3.19), which are then exchanged among neighboring nodes. In the example shown in Fig. 3.3, the sequence $\{W^{(t)}_{i\ell}(n)\}$ is given by

$$\{W^{(t)}_{i\ell}(n)\} = \left\{S_{\gamma i} + C^{(t-1)}_{\gamma \to i}, S_{\nu i} + C^{(t-1)}_{\nu \to i}, S_{\psi i} + C^{(t-1)}_{\psi \to i}\right\} .$$

This sequence is then sorted to obtain $W^{(t)}_{i\ell}(1)$, $W^{(t)}_{i\ell}(2)$ and $W^{(t)}_{i\ell}(3)$. The quantity

$C_{i\to\ell}^{(t)}$ is then determined using (3.20). Equation (3.20) essentially requires calculating the exact or approximate median of $\{W_{i\ell}^{(t)}(n)\}$, depending on $r_{i,\ell}^{(t)}$. This explains the need for sorting this sequence as mentioned earlier.

3.4.2 A Synchronization Algorithm

A synchronization algorithm which bases its functioning on the max-product message passing procedure described above will now be introduced. It has been remarked in the previous section that no actual factor nodes are present in the physical sensor network, and the computation resides entirely in the variable nodes, being one-to-one mapped to the physical nodes in the WSN. Therefore, the focus for describing the synchronization algorithm will be on the messages from factor nodes to variable nodes (3.19) only.

At each iteration, node 0 sends to its neighbors the mean of the distribution described by (3.13), which is $S_{0\ell}$ with $\ell \in \mathcal{N}_0$. On the other hand, from iteration 0 onward, the other nodes send no data to its neighbors unless their beliefs are different from the uniform priors.

At the beginning of step t, node $i \neq 0$ receives[1] $r_i^{(t)}$ non-constant values $S_{ji} + C_{j\to i}^{(t-1)}$ $\forall j \in \mathcal{N}_i$ (if i is not receiving anything from a particular neighbor j, it will be assumed that the message $m_{h_{ji}\to\beta_i}^{(t)}$ is constant, and therefore useless) and it sorts all these values to obtain a sequence[2] $\{W_i^{(t)}(n)\}$. It then updates its belief $b_i^{(t)}$ according

[1]It must be noted that $r_{i,\ell}^{(t)} - 1$ refers to the number of neighbors of node i, other than node ℓ, that send non-constant messsages at iteration t, while $r_i^{(t)}$ indicates the number of neighbors of node i that send non constant messages.

[2]It must be noted that $\{W_{i,\ell}^{(t)}(n)\}$ refers to the sorted sequence of messages received from neighbors of node i, other than node ℓ, at iteration t, while $\{W_i^{(t)}(n)\}$ indicates the sorted sequence of messages received from neighbors of node i.

to (3.10), as

$$b_i^{(t)}(\beta_i) = p_i(\beta_i) \cdot \prod_{j \in \mathcal{N}_i} m_{h_{ji} \to \beta_i}^{(t)}(\beta_i)$$
$$\propto \exp\left(-2K\lambda \sum_{n=1}^{r_i^{(t)}} \left| \beta_i - W_i^{(t)}(n) \right| \right), \tag{3.21}$$

where (3.21) is due to the prior pdf $p_i(\beta_i)$ being uniform. At this point, by maximizing the belief (3.21), node i can compute the estimate (3.11) of β_i as follows

$$\begin{aligned} \hat{\beta}_i^{(t)} &= \arg\max_{\beta_i} b_i^{(t)}(\beta_i) \\ &= \arg\max_{\beta_i} \exp\left(-2K\lambda \sum_{n=1}^{r_i^{(t)}} \left| \beta_i - W_i^{(t)}(n) \right| \right) \\ &= W_i^{(t)}\left(\left\lceil \frac{r_i^{(t)}}{2} \right\rceil \right), \end{aligned} \tag{3.22}$$

where $\lceil x \rceil$ is the smallest integer not less than x. The maximizer has been computed through techniques similar to those used for proving Lemma 3. Lastly, node i transmits all the messages $S_{i\ell} + C_{i \to \ell}^{(t)}$, computed according to (3.19) and (3.20), to its neighboring nodes ℓ.

The update of the estimate using (3.22) is performed until no more relevant data is available. Formally, $\hat{\beta}_i^{(t)}$ is updated according to (3.22) until the first update of less than $\varepsilon\%$ is encountered, where ε is the desired error margin. This time instant is denoted by t_i^* at node i, and the estimate $\hat{\beta}_i^{(t)}$ is kept unchanged for all $t > t_i^*$. After this instant, node i stops sending messages to its neighbors. The steps of the algorithm are summarized in Algorithm 1.

Hence, the aforementioned algorithm computes the clock offset estimates $\hat{\beta}_i^{(t)}$ for each node i in the network. Moreover, it must be emphasized that the algorithm is

Algorithm 1 Max-Product Algorithm.

$\mathcal{F} = \{0\}$.
for t from 0 on **do**
  Node $i = 0$ sends to its neighbors a packet with means $S_{0\ell}$, $\ell \in \mathcal{N}_0$, according to (3.13).
  for each node $i \notin \mathcal{F}$ in parallel to node 0 **do**
    Node i computes the belief $b_i^{(t)}(\beta_i)$ according to (3.21) from data $W_i^{(t)}(n)$ received from its neighbors.
    Node i computes the estimate $\hat{\beta}_i^*$ according to (3.22).
    if $|\hat{\beta}_i^* - \hat{\beta}_i^{(t-1)}|/\hat{\beta}_i^{(t-1)} > \varepsilon$ **then**
      $\hat{\beta}_i^{(t)} = \hat{\beta}_i^*$.
      Node i transmits all the messages $S_{i\ell} + C_{i\to\ell}^{(t)}$, computed according to (3.19) and (3.20), to its neighbors ℓ.
    else
      $\hat{\beta}_i^{(t)} = \hat{\beta}_i^{(t-1)}$
      $\mathcal{F} = \mathcal{F} \cup \{i\}$.
    end if
  end for
end for
return

completely distributed, since the clock offset of the node i is determined at the node itself, as opposed to employing a central processing unit to compute all offsets. The estimator $\hat{\beta}_i^{(t)}$, in general, produces biased estimates of β_i. However, it can be shown that the estimator is unbiased under certain assumptions as follows.

Consider the estimate (3.22) at node i and at time t. Assume that node i has received at least one non-constant message from its neighbors; therefore, it is straightforward to notice that a π_i-long node chain $j_0, j_1, \ldots, j_{\pi_i}$ from the reference node $j_0 = 0$ to node $j_{\pi_i} = i$ can be identified (see Fig. 3.4). The node chain could also contain loops. Moreover, here it is assumed that

$$r_{j_{n-1},j_n}^{(\tau)} \quad \text{are even numbers, for all } 1 \le n \le \pi_i \text{ and } \tau \le t\,. \tag{3.23}$$

This assumption implies that there is an odd number of non-constant messages

received at every node in the chain, so that the approximation in Lemma 2 becomes exact. The following discussion establishes a recursive relation for computing the values W. First, note that

$$W_i^{(t)}\left(\left\lceil \frac{r_i^{(t)}}{2} \right\rceil\right) = S_{ji} + C_{j\to i}^{(t-1)} \quad \text{for some } j \in \mathcal{N}_i \ ,$$
$$W_{i\ell}^{(t)}\left(\frac{r_{i,\ell}^{(t)}}{2}\right) = S_{ji} + C_{j\to i}^{(t-1)} \quad \text{for some } j \in \mathcal{N}_i, j \neq \ell \ .$$

By recursively applying the definition of $C_{i\to\ell}^{(t)}$ (3.20), it follows that

$$W_i^{(t)}\left(\left\lceil \frac{r_i^{(t)}}{2} \right\rceil\right) = S_{ji} + W_{ji}^{(t-1)}\left(\frac{r_{j,i}^{(t-1)}}{2}\right) = \sum_{n=1}^{\pi_i} S_{j_{n-1}j_n} \ , \tag{3.24}$$

which means that $W_i^{(t)}\left(\left\lceil r_i^{(t)}/2 \right\rceil\right)$ is the sum of π_i Laplace random variables, each one being the median of the messages coming from a node's neighborhood.

Lemma 4. *Estimator* (3.22) *is unbiased for all* i *and for all* t *under assumption* (3.23).

Proof. Under assumption (3.23) and recalling (3.4), it holds that

$$\begin{aligned}\mathbb{E}\left[W_i^{(t)}\left(\left\lceil \frac{r_i^{(t)}}{2} \right\rceil\right)\right] &= \sum_{n=1}^{\pi_i} \mathbb{E}\left[S_{j_{n-1}j_n}\right] \\ &= \sum_{n=1}^{\pi_i} \left(\beta_{j_n} - \beta_{j_{n-1}}\right) \\ &= \beta_i \ .\end{aligned}$$

□

Given the nature of the clock synchronization problem (due to exponentially distributed network delays), an iterative expression for the MSE or a linear update

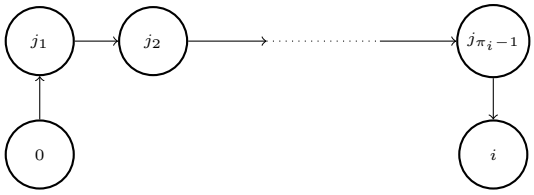

Figure 3.4: Node chain representing the information flow from the reference node 0 to node i.

equation for the estimator (3.22) are mathematically intractable. In fact, the computation of the estimator involves ordering operations which renders a non-linear nature to the estimator, and therefore an eigenvalue-based analysis of convergence is not possible. We, therefore, use numerical simulations to study the MSE performance of the algorithm. It is observed that the estimator exhibits high fidelity under various network topologies.

3.5 Simulation Results

This section presents numerical simulation results to assess the performance of the proposed algorithm in structured as well as random topologies. A WSN consisting of $N+1$ nodes is considered. The local estimate of clock offset difference (3.3) is based on $K = 4$ message exchanges. Results are averaged both with respect to the instances of the random delays and with respect to the clock offsets, generated uniformly in the interval $[-30, 30]$. The algorithm runs for a maximum of 150 iterations but the update of $\hat{\beta}_i^{(t)}$ (3.22) is stopped when estimate updates of less than $\varepsilon = 5\%$ are encountered. The fidelity criterion employed is the MSE given by

$$\begin{aligned} \mathrm{MSE}(t) &= \frac{\sum_{i=1}^{N} \mathrm{MSE}_i(t)}{N} \\ \mathrm{MSE}_i(t) &= \mathbb{E}\left[\left(\hat{\beta}_i^{(t)} - \beta_i\right)^2\right] . \end{aligned}$$

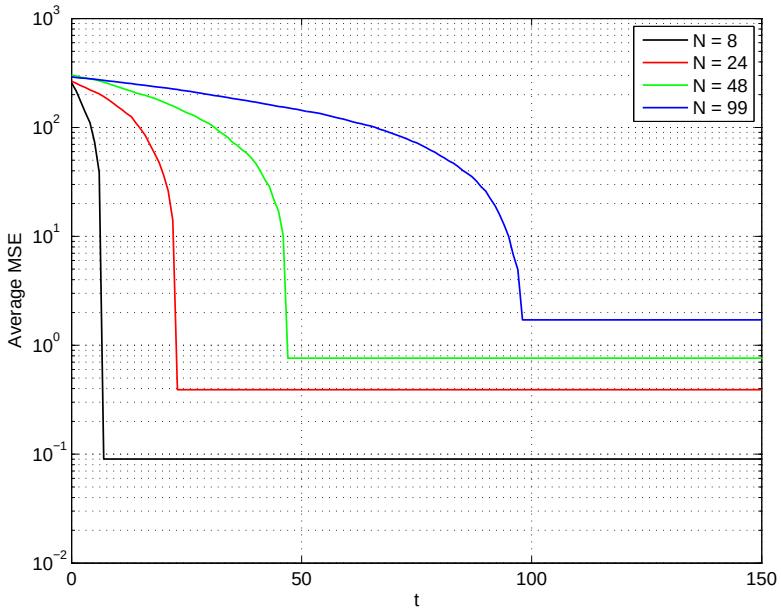

Figure 3.5: MSE versus t for different values of N with nodes forming a CG topology.

The plots in this section show the MSE in several network topologies for a varying number of nodes N and the exponential parameter λ.

3.5.1 Varying N

Figs. 3.5 and 3.6 show the averaged MSE incurred in the estimation of the clock offset versus the number of iterations t in chain graphs (CGs) and mesh grids (MGs), respectively. The performance is plotted for various values of N and $\lambda = 1$. For the CG and the MG, the averaging of the MSE is performed with respect to both the values of the random delays and the instances of the clock offsets. For initial iterations, it can be observed that in both the topologies, the average MSE slightly decreases with the number of iterations, implying that the information coming from the reference node is circulating across the network and it is being correctly exploited by the sensor nodes for clock offset estimation. The MSE abruptly decreases at a

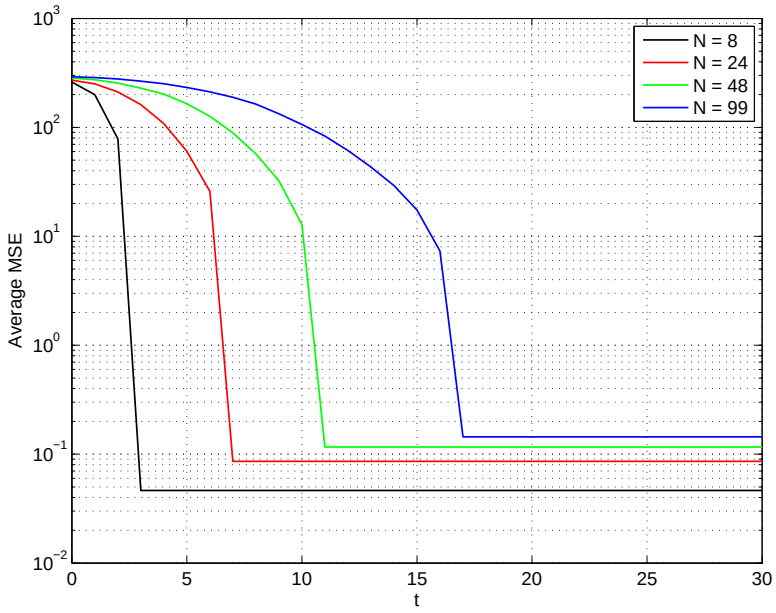

Figure 3.6: MSE versus t for different values of N with nodes forming a MG topology.

certain number of iterations and then it reaches a floor at $t = t^*$, where

$$t^* \geq \max_i \{t_i^*\} \ , \tag{3.25}$$

The value t^* depends on the type of graph and the number of nodes N. One of the purposes of showing the MSE averaged with respect to all the nodes is to ensure that the estimation process is successfully performed at each sensor node, rather than picking up a representative node which provides limited information on the MSE across the whole network. Moreover, while the result in cycle-free graphs is not surprising, from Fig. 3.6 it can be seen that the proposed algorithm estimates the clock offset even in loopy graphs.

For the CGs of Fig. 3.5, the reference node is assumed positioned at one of the two ends of the chain. The plot shows that the MSE decreases to the floor value after $t^* = N$, which is quite expected since for each node in the chain, even the

farthest one, a meaningful clock offset estimate is only derived after the original information coming from the reference node has traveled across the whole chain. In addition, it can be clearly observed that the floor value is increasing with the number of nodes N. This is due to the fact that when the number of nodes increase in a chain topology, the message transmitted from the reference node becomes more and more contaminated with random noise as it traverses through the chain to reach the farthest node.

Similarly, for the MGs of Fig. 3.6, the reference node is positioned at one corner of the grid. Also this plot shows that the number of iterations t^* at which the MSE decreases to the floor value is dependent on the specific value of N. It can be easily seen that $t^* = 2\sqrt{N+1} - 1$, which is the number of iterations needed by the information coming from the reference node at one corner to reach the farthest node lying at the opposite corner of the grid. The dependence of the MSE floor value on the number of nodes N is also clear in this case: the more nodes the network has, the higher is the floor value.

Fig. 3.7 depicts a comparison of MSE for CG, MG and random geometric graphs (RGG) with $N = 99$. For RGGs, the nodes are assumed uniformly distributed on a disc of area 1, and a communication radius of $\zeta = 0.25$ is chosen in order to allocate a considerable number of neighbors, N_{neg}, for each node, where $N_{neg} = 0.25^2\pi N$. The purpose of this plot is to compare the MSE for different graphs keeping the number of nodes N fixed. The plot clearly shows that the MSE decreases at the fastest rate when the underlying graph is random geometric. This comes from the better communication properties of the RGG, since the average number of neighbors is considerable and the information coming from the reference node can quickly disseminate across all the nodes in the network. Secondly, as expected, the number of iterations needed for getting a uniformly precise estimate in the CG is bigger in

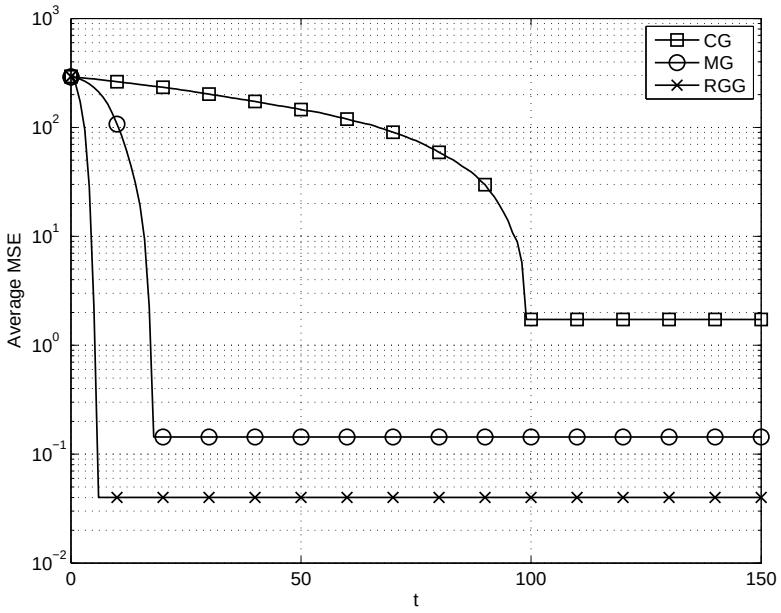

Figure 3.7: Comparison of MSE versus t for different network topologies, with $N = 99$.

comparison with MG.

3.5.2 Varying λ

Fig. 3.8 shows a comparison of the MSE of the proposed algorithm with the one in [38] (dashed lines) for a MG by varying the parameter λ of the random delays distribution, while keeping $N = 99$. The algorithm in [38] was proposed for Gaussian noise. The plot shows the comparison when the random delays are exponentially distributed. It can be observed that our proposed algorithm yields lower MSE for all values of λ. It can also be seen how the floor value of the MSE decreases as λ increases. The same observation holds for the CG and the RGG, whose plots are omitted to keep the discussion concise. To further investigate this dependence, Fig. 3.9 plots the variance of a single Laplace random variable (3.4), $1/(2K^2\lambda^2)$ on the x-axis, and the average MSE floor value normalized with respect to this variance on the y-axis for different network topologies. The purpose is to determine if a

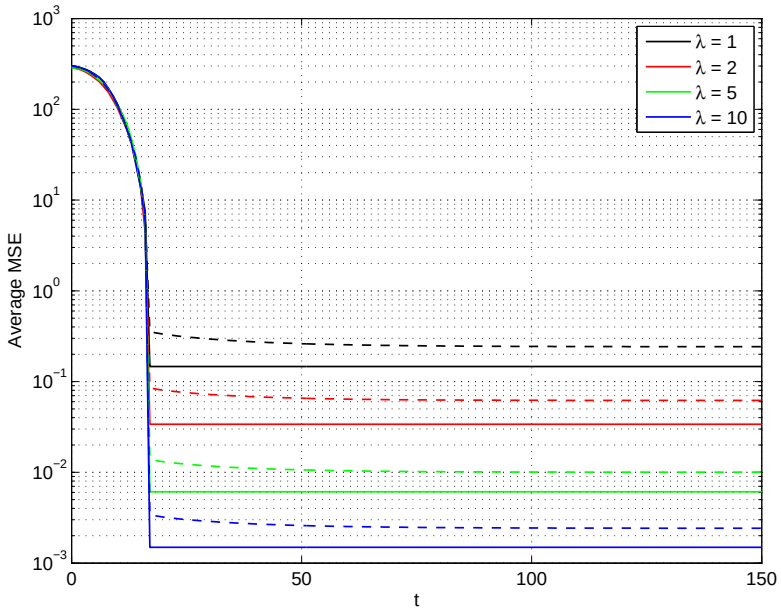

Figure 3.8: MSE versus t for different values of λ with nodes forming an MG topology. Solid curves show proposed algorithm, while the dashed curves represent the algorithm in [38].

relationship exists between the steady state value of the MSE and the variance of a single Laplace random variable S_{ij}. The plot shows that such a relationship indeed exists, showing that the variance of the estimator (3.22) can be well approximated by a proportionality factor times the variance of a single S_{ij}. For instance, for the chain topology, it can be observed that this proportionality factor is almost equal to the average (since the MSE is averaged with respect to all the nodes in the network) of the path lengths between the reference node and each node in the chain, which is $\frac{\sum_{i=0}^{N} i}{N+1} = \frac{N(N+1)}{2(N+1)} = \frac{N}{2}$

Following the same reasoning, for MGs the average MSE should take the form

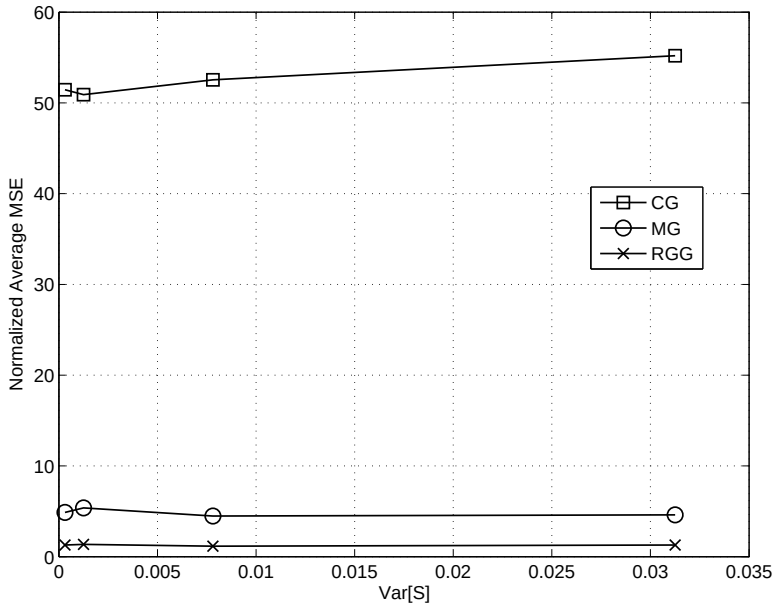

Figure 3.9: Floor value of the MSE versus Var[S] for different network topologies.

$(M = \sqrt{N+1})$

$$\frac{1}{N+1}\left[\sum_{i=0}^{M-1} i(i+1) + \sum_{i=0}^{M-2}(i+M)(M-1-i)\right] =$$
$$\frac{1}{N+1}\left[\frac{(M-1)M}{2}\left(1+\frac{2M-1}{3}\right)+\right.$$
$$\left.(M-1)\left(M(M-1)-\frac{M-2}{2}-\frac{(M-2)(2M-3)}{6}\right)\right]$$
$$= M-1 = \sqrt{N+1}-1\ .$$

On the other hand, it can be observed from Fig. 3.9 that the corresponding value on the plot is considerably less than $\sqrt{N+1}-1=9$. The main reason for this discrepancy is the multiple routes that the information coming from the reference node can take, so that the resulting clock offset estimate is not simply the addition of a number of Laplace random variables, but the ordering operations actually play

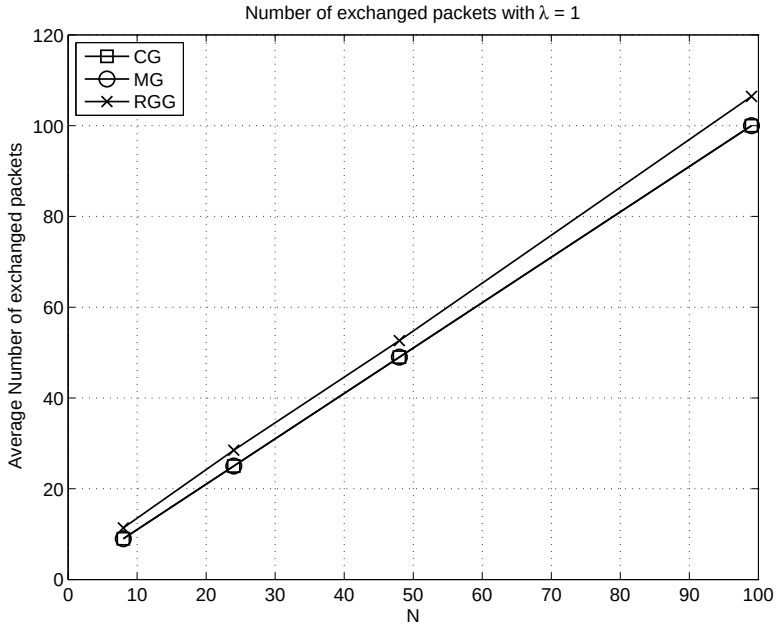

Figure 3.10: Number of packets exchanged among the nodes in the network versus N for different network topologies with $\lambda = 1$.

an active role in determining the final value of the estimate, reducing the average floor value of the MSE.

To assess the power consumption of the proposed algorithm, Fig. 3.10 shows the number of packets exchanged between nodes from the first iteration until the instant at which the algorithm stops running for all nodes in the network. The so-called *communication overhead* is plotted versus N for different network topologies. It can be observed that the complexity of the algorithm is almost linear in terms of number of packets exchanged for all three topologies considered, thus making it well suited for implementation in a real WSN.

3.6 Summary

This section studied the problem of network-wide clock synchronization in a WSN. Assuming that the variable portion of the link delays is exponentially dis-

tributed, an iterative factor graph-based algorithm is proposed for network-wide clock synchronization. The algorithm is completely distributed since the clock offset of each node is estimated at the node itself, as opposed to a non-distributed architecture where a central unit determines the offsets for all nodes in the sensor network. The update rules of the proposed algorithm are derived by message passing using the max-product algorithm. The MSE performance of the algorithm is studied for structured as well as random network topologies. It is observed that the RGG exhibits faster convergence rates compared to the CG and MG. In order to characterize the communication overhead of the algorithm, a study of the number of transmitted packets is also performed. It is observed that the number of packets exchanged increase linearly with the network size, therefore, the algorithm is well suited for clock synchronization of a sensor network.

4. JOINT NODE LOCALIZATION AND TIME-VARYING CLOCK SYNCHRONIZATION

4.1 Introduction

The problems of node localization and clock synchronization are closely tied. Recently, several contributions have proposed joint estimation approaches for localization and synchronization [62,67,68]. However, a common theme in these contributions is the assumption of fixed clock parameters. Sensor nodes are often deployed in harsh environmental conditions which can introduce degradations in the quartz crystals over time. Failure to cope with the temporal variations can result in frequent re-synchronization requests. Since power is primarily consumed in radio transmissions delivering timing information [50], exchanging time-stamps for re-synchronization can quickly drain a sensor's already meagre energy resources. Accurately tracking the drifts in clock parameters can boost the network lifetime by minimizing energy consumption. Recently, several synchronization-only approaches have considered time-variations in clock parameters in WSNs. A Bayesian approach for clock offset estimation was presented in [14] and [34], while a factor graph approach was utilized in [6] to obtain a closed form solution of the clock offset estimator when the likelihood function of the network delays is Gaussian, exponential or log-normally distributed. Time-variations in clock skew were incorporated in [22] for clock synchronization. A Kalman filter based approach was used for tracking clock skew variations in [32]. In this section, we aim to adopt the notion of temporal variation in clock parameters in the realm of joint node localization and clock synchronization in WSNs. We develop two iterative estimation algorithms which have varying degrees of accuracy and simplicity. The performance of the estimators is also benchmarked by deriving

the theoretical lower bounds on the MSE of the estimators.

4.2 Main Contributions

Our main contributions in this section are summarized as follows.

1. In order to ease the computational complexity associated with the MAP estimator, an EM based joint localization and time-varying synchronization algorithm is proposed that iteratively determines the time-varying clock parameters using a Kalman smoother followed by a likelihood maximization for the location estimation.

2. The location estimation process is further simplified by a linearization based LS method which yields a closed form solution and is therefore, a simpler alternative to the EM algorithm.

3. The performance of the aforementioned estimators is compared in various scenarios by deriving the HCRB in our estimation framework.

The notation used in this section is as follows. Upper (lower) case bold faced letters denote matrices (vectors). The matrices $\mathbf{0}_{M\times N}$ and $\mathbf{1}_{M\times 1}$ denote the $M \times N$ matrix of zeros and the $M \times 1$ matrix of ones, respectively. The notations $\mathrm{diag}(\boldsymbol{x})$ and $\mathrm{blkdiag}\left(\boldsymbol{X}, \boldsymbol{Y}\right)$ denote a diagonal matrix with the vector $\boldsymbol{x}$ on its main diagonal and a block diagonal matrix with matrices $\boldsymbol{X}$ and $\boldsymbol{Y}$ on its main diagonal, respectively. The operators Tr, $\otimes$ and $\odot$ denote the trace of a matrix, the Kronecker product and the Hadamard product between matrices, respectively. $\mathbb{E}_{\boldsymbol{x},\boldsymbol{y}}\left[\cdot\right]$ stands for the expectation with respect to the joint distribution $f\left(\boldsymbol{x}, \boldsymbol{y}\right)$. The partial derivative of a function $h\left(\boldsymbol{x}\right)$ with respect to $\boldsymbol{x}$, $\frac{\partial h(\boldsymbol{x})}{\partial \boldsymbol{x}}$, is the column vector $\left[\frac{\partial h(\boldsymbol{x})}{\partial x_1}, \ldots, \frac{\partial h(\boldsymbol{x})}{\partial x_N}\right]^T$.

4.3 System Model

Consider a network composed of N anchor nodes and an unknown node, denoted as Node X, that needs to be synchronized and localized to the anchors. The locations of the j^{th} anchor and Node X are given by $\boldsymbol{s}_j = \left[s_x^j, s_y^j\right]^T$ and $\boldsymbol{x} = [x_1, x_2]^T$, respectively. It is assumed that the anchors are synchronized with the same reference time t and their locations are accurately known. The process of joint localization and synchronization proceeds by exchanging time-stamps between Node X and the anchors using a two-way message exchange mechanism as shown in Fig. 4.1.

At the k^{th} message exchange, Node X transmits its current timing information to the j^{th} anchor through time-stamp $S_{j,k}$. The anchor records the time $R_{j,k}$ at which this message is received according to its own time scale. After some time has elapsed, the j^{th} anchor replies at time $\bar{S}_{j,k}$ and transmits a synchronization packet containing both the time-stamps $R_{j,k}$ and $\bar{S}_{j,k}$ to Node X. This message is received at time $\bar{R}_{j,k}$ by Node X according to its own clock. Therefore, after K exchanges with the j^{th} anchor, Node X is equipped with time-stamps $\left\{S_{j,k}, R_{j,k}, \bar{S}_{j,k}, \bar{R}_{j,k}\right\}_{k=1}^{K}$ which are to be used to ascertain its location and clock parameters. In this work, it is assumed that the clock of Node X is related to the reference time t as follows:

$$C_X(t) = \alpha t + \beta \ ,$$

where α and β denote the clock skew and clock offset with respect to the reference time, respectively. Hence, the aforementioned two-way timing exchange process can

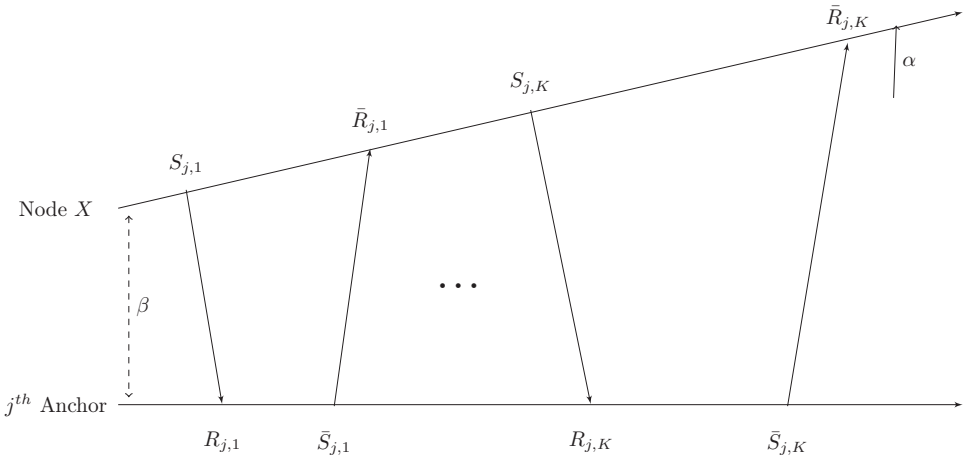

Figure 4.1: A two-way timing message exchange mechanism

be expressed as [15]

$$\begin{aligned} S_{j,k} &= \alpha \left(R_{j,k} - d_j - w_{j,k} \right) + \beta \\ \bar{R}_{j,k} &= \alpha \left(\bar{S}_{j,k} + d_j + \bar{w}_{j,k} \right) + \beta \ , \end{aligned} \tag{4.1}$$

where the measurement noise errors $w_{j,k}$ and $\bar{w}_{j,k}$ are assumed *i.i.d.* Gaussian with zero mean and variance σ_w^2. The *fixed* line-of-sight propagation delay, denoted by d_j, is given by[1] $d_j = \|\boldsymbol{x} - \boldsymbol{s}_j\|$. By defining

$$\theta_1 \triangleq \frac{1}{\alpha}, \qquad \theta_2 \triangleq \frac{\beta}{\alpha} \ , \tag{4.2}$$

we can equivalently express (4.1) as

$$\begin{aligned} R_{j,k} - d_j &= S_{j,k}\theta_1 - \theta_2 + w_{j,k} \\ -\bar{S}_{j,k} - d_j &= -\bar{R}_{j,k}\theta_1 + \theta_2 + \bar{w}_{j,k} \ . \end{aligned} \tag{4.3}$$

[1]The speed of light constant c is omitted for brevity.

By stacking data from all N anchors at the k^{th} message exchange, the system model in (4.3) can be compactly expressed as

$$\boldsymbol{y}_k - \boldsymbol{d}\left(\boldsymbol{x}\right) = \boldsymbol{H}_k\boldsymbol{\theta} + \boldsymbol{w}_k \ , \tag{4.4}$$

where $\boldsymbol{y}_k = \left[R_{1,k}, -\bar{S}_{1,k}, \ldots, R_{N,k}, -\bar{S}_{N,k}\right]^T$, the parameter vector $\boldsymbol{\theta} \triangleq [\theta_1 \ \theta_2]^T$, measurement noise $\boldsymbol{w}_k = [w_{1,k}, \bar{w}_{1,k}, \ldots, w_{N,k}, \bar{w}_{N,k}]$, $\boldsymbol{d}\left(\boldsymbol{x}\right) = \left[d_1\mathbf{1}_{2\times1}^T, \ldots, d_N\mathbf{1}_{2\times1}^T\right]^T$, and the $2N \times 2$ matrix $\boldsymbol{H}_k$ is given by

$$\boldsymbol{H}_k = \begin{bmatrix} S_{1,k} & -1 \\ -\bar{R}_{1,k} & 1 \\ \vdots & \vdots \\ S_{N,k} & -1 \\ -\bar{R}_{N,k} & 1 \end{bmatrix} .$$

Since sensor nodes are usually deployed in harsh environmental conditions, degradations in quartz oscillators render a time-varying nature to the clock skew and offset of Node X. Several recent contributions have proposed clock synchronization schemes by considering temporal variations in the clock parameters [6], [14], [32]. In this work, it is assumed that the variations in the clock parameters induce a Gauss-Markov evolution model for $\boldsymbol{\theta}$ at the k^{th} message exchange, i.e.,

$$\boldsymbol{\theta}_k = \boldsymbol{\theta}_{k-1} + \boldsymbol{n}_k \ , \tag{4.5}$$

where $\boldsymbol{n}_k$ is *i.i.d* zero mean Gaussian noise such that $\mathbb{E}\left(\boldsymbol{n}_k\boldsymbol{n}_k^T\right) = \sigma_n^2\boldsymbol{I}$. This model helps to capture time variations and also lends mathematical simplicity to gain a theoretical insight into the problem of joint localization and time-varying clock syn-

chronization. Using (4.5), the two-way message exchange model (4.4) at the k^{th} round is now expressed as

$$\boldsymbol{y}_k - \boldsymbol{d}(\boldsymbol{x}) = \boldsymbol{H}_k \boldsymbol{\theta}_k + \boldsymbol{w}_k \ . \tag{4.6}$$

By collecting data for all $k = 1, \ldots, K$, it follows that

$$\boldsymbol{y} - \boldsymbol{d}(\boldsymbol{x}) \otimes \mathbf{1}_{M\times 1} = \boldsymbol{H}\boldsymbol{\Theta} + \boldsymbol{w} \ , \tag{4.7}$$

where $\boldsymbol{y} = \left[\boldsymbol{y}_1^T, \ldots, \boldsymbol{y}_K^T\right]^T$, $\boldsymbol{\Theta} = \left[\boldsymbol{\theta}_1^T, \ldots, \boldsymbol{\theta}_K^T\right]^T$, $\boldsymbol{w} = \left[\boldsymbol{w}_1^T, \ldots, \boldsymbol{w}_K^T\right]^T$ and the matrix $\boldsymbol{H} = \text{diag}\left\{\boldsymbol{H}_1, \ldots, \boldsymbol{H}_K\right\}$.

The joint distribution of $\{\boldsymbol{y}, \boldsymbol{\Theta}\}$, parameterized by $\boldsymbol{x}$, can be expressed as

$$\begin{aligned} f(\boldsymbol{y}, \boldsymbol{\Theta}; \boldsymbol{x}) &= f(\boldsymbol{\Theta})\, f(\boldsymbol{y}|\boldsymbol{\Theta}; \boldsymbol{x}) \\ &= f(\boldsymbol{\theta}_0) \prod_{k=1}^{K} f(\boldsymbol{\theta}_k|\boldsymbol{\theta}_{k-1}) \prod_{k=1}^{K} f(\boldsymbol{y}_k|\boldsymbol{\theta}_k; \boldsymbol{x}) \ . \end{aligned} \tag{4.8}$$

The conditional pdfs $f(\boldsymbol{\theta}_k|\boldsymbol{\theta}_{k-1})$ and $f(\boldsymbol{y}_k|\boldsymbol{\theta}_k; \boldsymbol{x})$ are given by

$$f(\boldsymbol{\theta}_k|\boldsymbol{\theta}_{k-1}) = C_1 \exp\left(-\frac{(\boldsymbol{\theta}_k - \boldsymbol{\theta}_{k-1})^T (\boldsymbol{\theta}_k - \boldsymbol{\theta}_{k-1})}{2\sigma_n^2}\right) \tag{4.9}$$

$$f(\boldsymbol{y}_k|\boldsymbol{\theta}_k; \boldsymbol{x}) = C_2 \exp\left(-\frac{(\boldsymbol{y}_k - \boldsymbol{d}(\boldsymbol{x}) - \boldsymbol{H}_k\boldsymbol{\theta}_k)^T (\boldsymbol{y}_k - \boldsymbol{d}(\boldsymbol{x}) - \boldsymbol{H}_k\boldsymbol{\theta}_k)}{2\sigma_w^2}\right) \ , \tag{4.10}$$

where C_1 and C_2 are constants. Our goal is to jointly estimate $\boldsymbol{\xi} = \left[\boldsymbol{\Theta}^T, \boldsymbol{x}^T\right]^T$ using the time-stamps $\left\{S_{j,k}, R_{j,k}, \bar{S}_{j,k}, \bar{R}_{j,k}\right\}_{k=1}^{K}$ as well as the known anchor locations $\boldsymbol{s}_j$, for $j = 1, \ldots, N$.

The joint estimates of $\boldsymbol{\Theta}$ and $\boldsymbol{x}$ can be obtained as

$$\left\{\hat{\boldsymbol{\Theta}}, \hat{\boldsymbol{x}}\right\} = \arg\max_{\boldsymbol{\Theta},\boldsymbol{x}} \ln f(\boldsymbol{y}, \boldsymbol{\Theta}; \boldsymbol{x}) \,. \tag{4.11}$$

Solving (4.11) requires inverting large matrices which is computationally demanding. The nodes in a WSN are generally inexpensive devices characterized by limited capabilities of computation and communication. This computational complexity necessitates the development of simpler alternative algorithms that lower the computational burden while maintaining a desired performance level. In the next section, two iterative methods are explored for joint localization and timing synchronization.

4.4 Iterative Approaches

In this section, two iterative estimation algorithms are proposed which differ mainly in their approach to determine the location of the unknown node.

4.4.1 The EM Algorithm

The EM algorithm is an iterative method used to determine the ML estimate of the parameters of a given distribution from incomplete data [19]. The EM algorithm offers a simpler alternative to an otherwise intractable ML estimation problem by assuming additional unobserved parameters in the underlying distribution. The ML estimates are then computed by iterating between the *Expectation* and *Maximization* steps. Due to its analytical tractability, the EM algorithm finds numerous applications in diverse fields [41].

Assuming that the data $\boldsymbol{y}$ is incomplete, the complete data vector is defined as $\boldsymbol{z} \triangleq \left[\boldsymbol{y}^T, \boldsymbol{\Theta}^T\right]^T$. The expectation and maximization steps in the EM algorithm can be described as follows.

E-Step:

Given an estimate $\hat{\boldsymbol{x}}^{(i)}$ of the unknown node's location at iteration i, and the observed data $\boldsymbol{y}$, determine the likelihood function

$$Q\left(\boldsymbol{x}, \hat{\boldsymbol{x}}^{(i)}\right) \triangleq \mathbb{E}_{\boldsymbol{\Theta}|\boldsymbol{y},\hat{\boldsymbol{x}}^{(i)}}\left[\ln f\left(\boldsymbol{z};\boldsymbol{x}\right)\right] \ . \tag{4.12}$$

M-Step:

Obtain an estimate of $\boldsymbol{x}$ at iteration index $i+1$ by maximizing $Q\left(\boldsymbol{x}, \hat{\boldsymbol{x}}^{(i)}\right)$, i.e.,

$$\hat{\boldsymbol{x}}^{(i+1)} = \arg\max_{\boldsymbol{x}} Q\left(\boldsymbol{x}, \hat{\boldsymbol{x}}^{(i)}\right) \ . \tag{4.13}$$

The E-Step and M-Step are repeated until convergence. After each iteration, we are guaranteed to converge towards a local maximum [19].

Using (4.8), (4.9) and (4.10), it follows that

$$\ln f\left(\boldsymbol{z};\boldsymbol{x}\right) = C - \frac{1}{2\sigma_w^2}\sum_{k=1}^{K}\left(\tilde{\boldsymbol{y}}_k\left(\boldsymbol{x}\right) - \boldsymbol{H}_k\boldsymbol{\theta}_k\right)^T\left(\tilde{\boldsymbol{y}}_k\left(\boldsymbol{x}\right) - \boldsymbol{H}_k\boldsymbol{\theta}_k\right) \ , \tag{4.14}$$

where $\tilde{\boldsymbol{y}}_k\left(\boldsymbol{x}\right) \triangleq \boldsymbol{y}_k - \boldsymbol{d}\left(\boldsymbol{x}\right)$ and the terms that do not depend on $\boldsymbol{x}$ are collected in the constant C . The likelihood function at the i^{th} iteration, $Q\left(\boldsymbol{x}, \hat{\boldsymbol{x}}^{(i)}\right)$, can be evaluated as

$$\begin{aligned} Q\left(\boldsymbol{x}, \hat{\boldsymbol{x}}^{(i)}\right) &= \mathbb{E}_{\boldsymbol{\Theta}|\boldsymbol{y},\hat{\boldsymbol{x}}^{(i)}}\left[-\frac{1}{2\sigma_w^2}\sum_{k=1}^{K}\left(\tilde{\boldsymbol{y}}_k\left(\boldsymbol{x}\right) - \boldsymbol{H}_k\boldsymbol{\theta}_k\right)^T\left(\tilde{\boldsymbol{y}}_k\left(\boldsymbol{x}\right) - \boldsymbol{H}_k\boldsymbol{\theta}_k\right)\right] \\ &= -\frac{1}{2\sigma_w^2}\sum_{k=1}^{K}\mathrm{Tr}\left\{\mathbb{E}_{\boldsymbol{\Theta}|\boldsymbol{y},\hat{\boldsymbol{x}}^{(i)}}\left[\left(\tilde{\boldsymbol{y}}_k\left(\boldsymbol{x}\right) - \boldsymbol{H}_k\boldsymbol{\theta}_k\right)\left(\tilde{\boldsymbol{y}}_k\left(\boldsymbol{x}\right) - \boldsymbol{H}_k\boldsymbol{\theta}_k\right)^T\right]\right\} . \end{aligned} \tag{4.15}$$

By defining

$$\hat{\boldsymbol{\theta}}_{k|K}^{(i)} \triangleq \mathbb{E}_{\boldsymbol{\Theta}|\boldsymbol{y},\hat{\boldsymbol{x}}^{(i)}}\left[\boldsymbol{\theta}_k\right], \quad \hat{\boldsymbol{R}}_{k|K}^{(i)} \triangleq \mathbb{E}_{\boldsymbol{\Theta}|\boldsymbol{y},\hat{\boldsymbol{x}}^{(i)}}\left[\boldsymbol{\theta}_k\boldsymbol{\theta}_k^T\right] ,$$

the likelihood function in (4.15) can be expressed as

$$Q\left(\boldsymbol{x},\hat{\boldsymbol{x}}^{(i)}\right) = -\frac{1}{2\sigma_w^2}\sum_{k=1}^{K}\mathrm{Tr}\Big\{\tilde{\boldsymbol{y}}_k\left(\boldsymbol{x}\right)\tilde{\boldsymbol{y}}_k^T\left(\boldsymbol{x}\right) + \boldsymbol{H}_k\hat{\boldsymbol{R}}_{k|K}^{(i)}\boldsymbol{H}_k^T - \tilde{\boldsymbol{y}}_k\left(\boldsymbol{x}\right)\hat{\boldsymbol{\theta}}_{k|K}^{(i)^T}\boldsymbol{H}_k^T - \boldsymbol{H}_k\hat{\boldsymbol{\theta}}_{k|K}^{(i)}\tilde{\boldsymbol{y}}_k^T\left(\boldsymbol{x}\right)\Big\} . \tag{4.16}$$

After some algebraic steps, (4.16) can be equivalently written as

$$Q\left(\boldsymbol{x},\hat{\boldsymbol{x}}^{(i)}\right) = -\frac{1}{2\sigma_w^2}\sum_{k=1}^{K}\mathrm{Tr}\left\{\boldsymbol{H}_k\hat{\boldsymbol{\Sigma}}_{k|K}^{(i)}\boldsymbol{H}_k^T + \left(\tilde{\boldsymbol{y}}_k\left(\boldsymbol{x}\right) - \boldsymbol{H}_k\hat{\boldsymbol{\theta}}_{k|K}^{(i)}\right)\left(\tilde{\boldsymbol{y}}_k\left(\boldsymbol{x}\right) - \boldsymbol{H}_k\hat{\boldsymbol{\theta}}_{k|K}^{(i)}\right)^T\right\} , \tag{4.17}$$

where

$$\hat{\boldsymbol{\Sigma}}_{k|K}^{(i)} \triangleq \hat{\boldsymbol{R}}_{k|K}^{(i)} - \hat{\boldsymbol{\theta}}_{k|K}^{(i)}\hat{\boldsymbol{\theta}}_{k|K}^{(i)^T} .$$

Given an estimate $\hat{\boldsymbol{x}}^{(i)}$, it can be observed that (4.5) and (4.6) represent a linear Gaussian model. The minimum mean square error (MMSE) estimator $\hat{\boldsymbol{\theta}}_{k|K}^{(i)}$ can be obtained from a standard Kalman smoother. The forward recursion for obtaining $\hat{\boldsymbol{\theta}}_{k|k}^{(i)}$ can be expressed as follows [31].

Forward Recursion

Prediction:

$$\begin{aligned}\hat{\boldsymbol{\theta}}_{k|k-1}^{(i)} &= \hat{\boldsymbol{\theta}}_{k-1|k-1}^{(i)} \\ \hat{\boldsymbol{\Sigma}}_{k|k-1}^{(i)} &= \hat{\boldsymbol{\Sigma}}_{k-1|k-1}^{(i)} + \sigma_n^2\boldsymbol{I}\end{aligned} \tag{4.18}$$

Correction:

$$\begin{aligned}
\boldsymbol{\kappa}_k &= \hat{\boldsymbol{\Sigma}}_{k|k-1}^{(i)} \boldsymbol{H}_k^T \left(\boldsymbol{H}_k \hat{\boldsymbol{\Sigma}}_{k|k-1}^{(i)} \boldsymbol{H}_k^T + \sigma_w^2 \boldsymbol{I} \right) \\
\hat{\boldsymbol{\theta}}_{k|k}^{(i)} &= \hat{\boldsymbol{\theta}}_{k|k-1}^{(i)} + \boldsymbol{\kappa}_k \left(\tilde{\boldsymbol{y}}_k \left(\hat{\boldsymbol{x}}^{(i)} \right) - \boldsymbol{H}_k \hat{\boldsymbol{\theta}}_{k|k-1}^{(i)} \right) \\
\hat{\boldsymbol{\Sigma}}_{k|k}^{(i)} &= \left(\boldsymbol{I} - \boldsymbol{\kappa}_k \boldsymbol{H}_k \right) \hat{\boldsymbol{\Sigma}}_{k|k-1}^{(i)} ,
\end{aligned} \tag{4.19}$$

where $\hat{\boldsymbol{\Sigma}}_{k|k}^{(i)}$ is the MMSE matrix.

The operation of a smoother is completed by employing a backward sweep that produces the smoothed estimates $\hat{\boldsymbol{\theta}}_{k|K}^{(i)}$ and $\hat{\boldsymbol{\Sigma}}_{k|K}^{(i)}$. The recursions of the Rauch-Tung-Striebel (RTS) smoother are given as follows [51].

Backward Recursion

$$\begin{aligned}
\boldsymbol{B}_{k-1} &= \hat{\boldsymbol{\Sigma}}_{k-1|k-1}^{(i)} \hat{\boldsymbol{\Sigma}}_{k|k-1}^{(i)^{-1}} \\
\hat{\boldsymbol{\theta}}_{k-1|K}^{(i)} &= \hat{\boldsymbol{\theta}}_{k-1|k-1}^{(i)} + \boldsymbol{B}_{k-1} \left(\hat{\boldsymbol{\theta}}_{k|K}^{(i)} - \hat{\boldsymbol{\theta}}_{k|k-1}^{(i)} \right) \\
\hat{\boldsymbol{\Sigma}}_{k-1|K}^{(i)} &= \hat{\boldsymbol{\Sigma}}_{k-1|k-1}^{(i)} + \boldsymbol{B}_{k-1} \left(\hat{\boldsymbol{\Sigma}}_{k|K}^{(i)} - \hat{\boldsymbol{\Sigma}}_{k|k-1}^{(i)} \right) \boldsymbol{B}_{k-1}^T .
\end{aligned} \tag{4.20}$$

Remark 2. *Intuitively, the backward step yields an improvement in the forward step-only approach since it uses the entire data sequence to smooth out the estimates of* $\boldsymbol{\Theta}$*. This improvement comes at the cost of some additional processing. The extent of this improvement is quantified through simulations in Section 4.6.*

Therefore, the E-step of the EM algorithm (4.12) yields an MMSE (equivalently, MAP) estimate of $\boldsymbol{\theta}_k$ for $k = 1, \ldots, K$ as a by-product while calculating $Q\left(\boldsymbol{x}, \hat{\boldsymbol{x}}^{(i)}\right)$. The estimates of $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ can, in turn, be obtained by using the transformation in (4.2). The resulting estimates are sub-optimal since, in general, the MAP estima-

tor does not commute over non-linear transformations. However, the sub-optimal estimators show good fidelity performance and closely match the theoretical lower bounds derived in Section 4.5.

The M-step can now be expressed using (4.13) as

$$\hat{\boldsymbol{x}}^{(i+1)} = \arg\max_{\boldsymbol{x}} \frac{-1}{2\sigma_w^2} \sum_{k=1}^{K} \mathrm{Tr}\Bigg\{ \boldsymbol{H}_k \hat{\boldsymbol{\Sigma}}_{k|K}^{(i)} \boldsymbol{H}_k^T + \left(\tilde{\boldsymbol{y}}_k(\boldsymbol{x}) - \boldsymbol{H}_k \hat{\boldsymbol{\theta}}_{k|K}^{(i)}\right) \left(\tilde{\boldsymbol{y}}_k(\boldsymbol{x}) - \boldsymbol{H}_k \hat{\boldsymbol{\theta}}_{k|K}^{(i)}\right)^T \Bigg\}.$$

After some simplifications, the estimate $\hat{\boldsymbol{x}}^{(i+1)}$ is given as the solution of a 2-D norm minimization problem

$$\hat{\boldsymbol{x}}^{(i+1)} = \arg\min_{\boldsymbol{x}} \sum_{k=1}^{K} \left\| \tilde{\boldsymbol{y}}_k(\boldsymbol{x}) - \boldsymbol{H}_k \hat{\boldsymbol{\theta}}_{k|K}^{(i)} \right\|^2 . \tag{4.21}$$

It can be shown using convex optimization techniques that the objective function in the aforementioned minimization problem is a strictly convex function and hence, has a unique minimum. However, a closed form solution of the optimization problem in (4.21) does not exist. The interior point methods can be used to obtain the estimates $\hat{\boldsymbol{x}}^{(i+1)}$ [12].

The EM algorithm, therefore, provides estimates of $\boldsymbol{\Theta}$ and $\boldsymbol{x}$ by alternating between the E and M-steps, respectively. The algorithm is terminated when the sequence $\hat{\boldsymbol{x}}^{(1)}, \hat{\boldsymbol{x}}^{(2)}, \hat{\boldsymbol{x}}^{(3)}, \ldots$ converges. The EM algorithm for joint localization and time-varying clock synchronization of Node X is summarized in Algorithm 2.

4.4.2 The LS estimator

The location estimator in (4.21) requires a costly 2-D norm minimization that may be computationally infeasible for a sensor node. Therefore, it becomes imperative to

Algorithm 2 `The EM Algorithm`

1: Input time-stamps $\left\{S_{j,k}, R_{j,k}, \bar{S}_{j,k}, \bar{R}_{j,k}\right\}_{k=1}^{K}$ and known anchor locations $\boldsymbol{s}_j$, for $j = 1, \ldots, N$.
2: Initialize $\hat{\boldsymbol{x}}^{(0)}$.
3: **for** $k = 1, \ldots, K$ **do**
4: Determine $Q\left(\boldsymbol{x}, \boldsymbol{x}^{(i)}\right)$ in (4.17) using the MMSE estimator $\hat{\boldsymbol{\theta}}_{k|K}^{(i)}$ from (4.19) and (4.20).
5: **end for**
6: Obtain the ML estimate $\hat{\boldsymbol{x}}^{(i+1)}$ by solving the optimization problem (4.21).
7: **return**

simplify the location estimation method in Section 4.4.1.

Similar to the above discussion, it can be noticed that with an estimate $\hat{\boldsymbol{x}}^{(i)}$ available at iteration i, the parameters $\boldsymbol{\theta}_k$ evolve according to a linear state space Gaussian model and can be efficiently estimated by the Kalman smoother described in Section 4.4.1. Using estimates $\hat{\boldsymbol{\theta}}_k$, (4.3) can be expressed at iteration i as

$$R_{j,k} - d_j = S_{j,k}\hat{\theta}_{1,k}^{(i)} - \hat{\theta}_{2,k}^{(i)} + w_{j,k} \tag{4.22}$$

$$-\bar{S}_{j,k} - d_j = -\bar{R}_{j,k}\hat{\theta}_{1,k}^{(i)} + \hat{\theta}_{2,k}^{(i)} + \bar{w}_{j,k} \, . \tag{4.23}$$

Define

$$t_{j,k}^{(i)} \triangleq R_{j,k} - S_{j,k}\hat{\theta}_{1,k}^{(i)} + \hat{\theta}_{2,k}^{(i)} \, . \tag{4.24}$$

Squaring (4.22), yields

$$\|\boldsymbol{x} - \boldsymbol{s}_j\|^2 = {t_{j,k}^{(i)}}^2 - 2t_{j,k}^{(i)} w_{j,k} + w_{j,k}^2 \, ,$$

which can be re-written as

$$2\boldsymbol{s}_j^T\boldsymbol{x} - \|\boldsymbol{x}\|^2 = \|\boldsymbol{s}_j\|^2 - t_{j,k}^{(i)^2} + u_{j,k}^{(i)} \,, \tag{4.25}$$

where $u_{j,k}^{(i)} \triangleq 2t_{j,k}^{(i)}w_{j,k} - w_{j,k}^2$. Similarly, define

$$\bar{t}_{j,k}^{(i)} \triangleq \bar{R}_{j,k}\hat{\theta}_{1,k}^{(i)} - \hat{\theta}_{2,k}^{(i)} - \bar{S}_{j,k} \,. \tag{4.26}$$

Using $\bar{t}_{j,k}^{(i)}$ and squaring (4.23), we have

$$2\boldsymbol{s}_j^T\boldsymbol{x} - \|\boldsymbol{x}\|^2 = \|\boldsymbol{s}_j\|^2 - \bar{t}_{j,k}^{(i)^2} + \bar{u}_{j,k}^{(i)} \,, \tag{4.27}$$

where $\bar{u}_{j,k}^{(i)} \triangleq 2\bar{t}_{j,k}^{(i)}\bar{w}_{j,k} - \bar{w}_{j,k}^2$. By stacking all K observations for the j^{th} anchor, (4.25) and (4.27) can be written in matrix form as

$$\boldsymbol{A}_j\boldsymbol{\eta} = \boldsymbol{p}_j^{(i)} + \boldsymbol{u}_j^{(i)} \,, \tag{4.28}$$

where

$$\boldsymbol{A}_j = \begin{bmatrix} 2\boldsymbol{s}_j^T & -1 \\ 2\boldsymbol{s}_j^T & -1 \\ \vdots & \vdots \\ 2\boldsymbol{s}_j^T & -1 \\ 2\boldsymbol{s}_j^T & -1 \end{bmatrix}, \quad \boldsymbol{p}_j^{(i)} = \begin{bmatrix} \|\boldsymbol{s}_j\|^2 - t_{j,1}^{(i)^2} \\ \|\boldsymbol{s}_j\|^2 - \bar{t}_{j,1}^{(i)^2} \\ \vdots\vdots \\ \|\boldsymbol{s}_j\|^2 - t_{j,M}^{(i)^2} \\ \|\boldsymbol{s}_j\|^2 - \bar{t}_{j,M}^{(i)^2} \end{bmatrix},$$

$$\boldsymbol{u}_j^{(i)} = [u_{j,1}^{(i)}, \bar{u}_{j,k}^{(i)}, \ldots, u_{j,M}^{(i)}, \bar{u}_{j,M}^{(i)}]^T \,, \tag{4.29}$$

and $\boldsymbol{\eta} = [\boldsymbol{x}^T, \|\boldsymbol{x}\|^2]^T$. By augmenting data from all N anchor nodes, (4.29) can be

written as

$$\boldsymbol{A}\boldsymbol{\eta} = \boldsymbol{p}^{(i)} + \boldsymbol{u}^{(i)} \ , \tag{4.30}$$

where $\boldsymbol{A} = [\boldsymbol{A}_1^T, \ldots, \boldsymbol{A}_N^T]^T$, $\boldsymbol{p}^{(i)} = [\boldsymbol{p}_1^{(i)^T}, \ldots, \boldsymbol{p}_N^{(i)^T}]^T$, and $\boldsymbol{u}^{(i)} = [\boldsymbol{u}_1^{(i)^T}, \ldots, \boldsymbol{u}_N^{(i)^T}]^T$. It can be observed that (4.30) represents a linear matrix equation for the estimation of $\boldsymbol{x}$.

By neglecting the second order noise terms, $w_{j,k}^2$ and $\bar{w}_{j,k}^2$, in $\boldsymbol{u}^{(i)}$ and letting

$$\boldsymbol{q}^{(i)} = [2t_{1,1}^{(i)}, 2\bar{t}_{1,1}^{(i)}, \ldots, 2t_{1,K}^{(i)}, 2\bar{t}_{1,K}^{(i)}, \ldots, 2t_{N,1}^{(i)}, 2\bar{t}_{N,1}^{(i)}, \ldots, 2t_{N,K}^{(i)}, 2\bar{t}_{N,K}^{(i)}]^T \ ,$$

the noise covariance matrix $\boldsymbol{\Sigma}_{\boldsymbol{u}}^{(i)}$ can be obtained as

$$\begin{aligned} \boldsymbol{\Sigma}_{\boldsymbol{u}}^{(i)} &\approx \mathbb{E}\left[\boldsymbol{u}^{(i)}\boldsymbol{u}^{(i)^T}\right] = \mathbb{E}\left[\left(\boldsymbol{q}^{(i)} \odot \tilde{\boldsymbol{w}}\right)\left(\boldsymbol{q}^{(i)} \odot \tilde{\boldsymbol{w}}\right)^T\right] \\ &= \sigma_w^2 \mathrm{diag}(\boldsymbol{q}^{(i)} \odot \boldsymbol{q}^{(i)}) \ , \end{aligned} \tag{4.31}$$

where $\tilde{\boldsymbol{w}} = [w_{1,1}, \bar{w}_{1,1}, \ldots, w_{1,K}, \bar{w}_{1,K}, , \ldots, w_{N,K}, \bar{w}_{N,K}]^T$.

The LS solution for the estimation of $\boldsymbol{\eta}$ at iteration $i+1$ can now be expressed as

$$\hat{\boldsymbol{\eta}}^{(i+1)} = \left(\boldsymbol{A}^T \boldsymbol{\Sigma}_{\boldsymbol{u}}^{(i)^{-1}} \boldsymbol{A}\right)^{-1} \boldsymbol{A}^T \boldsymbol{\Sigma}_{\boldsymbol{u}}^{(i)^{-1}} \boldsymbol{p}^{(i)} \ . \tag{4.32}$$

The Kalman smoother stage employed at iteration i yielding the estimates $\hat{\boldsymbol{\theta}}_k, k = 1, \ldots, K$, can be used to compute $\boldsymbol{\Sigma}_{\boldsymbol{u}}^{(i)}$ and $\boldsymbol{p}^{(i)}$ (cf. (4.29) and (4.31)).

The estimate can be further refined by exploiting the relationship between elements of $\boldsymbol{\eta}$ [67]. We have

$$\bar{\boldsymbol{M}}\hat{\boldsymbol{x}}^{(i+1)} = \hat{\boldsymbol{\eta}}^{(i+1)} + \boldsymbol{w}_{\mathrm{LS}} \ , \tag{4.33}$$

where

$$\bar{\boldsymbol{M}} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ \hat{x}_1 & \hat{x}_2 \end{bmatrix}$$

and $\boldsymbol{w}_{\text{LS}}$ indicates the estimation error in estimating $\hat{\boldsymbol{\eta}}^{(i+1)}$. The estimates $\{\hat{x}_1, \hat{x}_2\}$ are available from (4.32). Finally the refined LS location estimator $\hat{\boldsymbol{x}}^{(i+1)}$ at iteration $i+1$ is given by

$$\begin{aligned} \hat{\boldsymbol{x}}_{\text{LS}}^{(i+1)} &= \left(\bar{\boldsymbol{M}}^T \boldsymbol{A}^T {\boldsymbol{\Sigma}_{\boldsymbol{u}}^{(i)}}^{-1} \boldsymbol{A} \bar{\boldsymbol{M}}\right)^{-1} \bar{\boldsymbol{M}}^T \boldsymbol{A}^T {\boldsymbol{\Sigma}_{\boldsymbol{u}}^{(i)}}^{-1} \boldsymbol{A} \hat{\boldsymbol{\eta}}^{(i+1)} \\ &= \left(\bar{\boldsymbol{M}}^T \boldsymbol{A}^T {\boldsymbol{\Sigma}_{\boldsymbol{u}}^{(i)}}^{-1} \boldsymbol{A} \bar{\boldsymbol{M}}\right)^{-1} \bar{\boldsymbol{M}}^T \boldsymbol{A}^T {\boldsymbol{\Sigma}_{\boldsymbol{u}}^{(i)}}^{-1} \boldsymbol{p}^{(i)} \ . \end{aligned} \tag{4.34}$$

Remark 3. *The LS location estimator in* (4.34) *is derived by neglecting the second order measurement noise errors. Since the measurement noise errors are usually small, it is expected that the performance of the LS estimator will closely match that of the EM algorithm.*

The aforementioned LS-based approach presents a simpler closed form alternative to the potentially costly 2-D norm minimization problem for location estimation in Section 4.4.1. The steps of the algorithm are summarized in Algorithm 3.

Algorithm 3 `The LS Algorithm`

1: Input time-stamps $\left\{S_{j,k}, R_{j,k}, \bar{S}_{j,k}, \bar{R}_{j,k}\right\}_{k=1}^{K}$ and known anchor locations $\boldsymbol{s}_j$, for $j = 1, \ldots, N$.
2: Initialize $\hat{\boldsymbol{x}}^{(0)}$.
3: **for** $k = 1, \ldots, K$ **do**
4: Determine the MMSE estimator $\hat{\boldsymbol{\theta}}_{k|K}^{(i)}$ from the Kalman smoother in (4.19) and (4.20).
5: **end for**
6: Obtain the LS estimate $\hat{\boldsymbol{x}}^{(i+1)}$ using (4.34).
7: **return**

4.5 Hybrid Cramér-Rao Bound

The HCRB, proposed in [52], is useful in cases where the complete parameter vector comprises deterministic as well as random parameters. This fits our estimation scenario since the location $\boldsymbol{x}$ is deterministic while $\boldsymbol{\Theta}$ is a random process.

The HCRB states that the covariance matrix of an estimator $\hat{\boldsymbol{\xi}}$ is lower bounded as

$$\mathbb{E}\left[\left(\hat{\boldsymbol{\xi}}-\boldsymbol{\xi}\right)\left(\hat{\boldsymbol{\xi}}-\boldsymbol{\xi}\right)^{T}\right] \succeq \left[\boldsymbol{\mathcal{H}}(\boldsymbol{\Theta},\boldsymbol{x})\right]^{-1} , \tag{4.35}$$

where the matrix inequality $\succeq$ is to be interpreted in the positive semi-definite sense. The $2(M+1)\times 2(M+1)$ hybrid information matrix (HIM) is given by

$$\boldsymbol{\mathcal{H}}(\boldsymbol{\Theta},\boldsymbol{x}) = \mathbb{E}_{\boldsymbol{\Theta}}\left[\boldsymbol{F}\left(\boldsymbol{\Theta},\boldsymbol{x}\right)\right] + \mathbb{E}_{\boldsymbol{\Theta}|\boldsymbol{x}}\left[\left(\frac{\partial \ln f\left(\boldsymbol{\Theta}|\boldsymbol{x}\right)}{\partial \boldsymbol{\xi}}\right)\left(\frac{\partial \ln f\left(\boldsymbol{\Theta}|\boldsymbol{x}\right)}{\partial \boldsymbol{\xi}}\right)^{T}\right] , \tag{4.36}$$

where the Fisher information matrix (FIM) $\boldsymbol{F}\left(\boldsymbol{\Theta},\boldsymbol{x}\right)$ is given by

$$\boldsymbol{F}(\boldsymbol{\Theta},\boldsymbol{x}) = \mathbb{E}_{\boldsymbol{y}|\boldsymbol{\Theta},\boldsymbol{x}}\left[\left(\frac{\partial \ln f\left(\boldsymbol{y}|\boldsymbol{\Theta},\boldsymbol{x}\right)}{\partial \boldsymbol{\xi}}\right)\left(\frac{\partial \ln f\left(\boldsymbol{y}|\boldsymbol{\Theta},\boldsymbol{x}\right)}{\partial \boldsymbol{\xi}}\right)^{T}\right] . \tag{4.37}$$

The HIM can be calculated as shown in the following theorem.

Theorem 2. *The sub-matrices in the HIM corresponding to* $\boldsymbol{\Theta}$ *and* $\boldsymbol{x}$ *can be expressed respectively as*

$$\boldsymbol{\mathcal{H}}_{11} = blkdiag\left(\frac{\boldsymbol{H}_1^T\boldsymbol{H}_1}{\sigma_w^2},\ldots,\frac{\boldsymbol{H}_K^T\boldsymbol{H}_K}{\sigma_w^2}\right) + \boldsymbol{\Upsilon} \tag{4.38}$$

$$\boldsymbol{\mathcal{H}}_{22} = \frac{2K}{\sigma_w^2}\sum_{j=1}^{N}\frac{\left(\boldsymbol{x}-\boldsymbol{s}_j\right)\left(\boldsymbol{x}-\boldsymbol{s}_j\right)^{T}}{\left\|\boldsymbol{x}-\boldsymbol{s}_j\right\|^{2}} , \tag{4.39}$$

where the tri-diagonal matrix $\boldsymbol{\Upsilon}$ *is*

$$\boldsymbol{\Upsilon} = \frac{1}{\sigma_n^2} \begin{bmatrix} 2\boldsymbol{I} & -\boldsymbol{I} & & & \mathbf{0} \\ -\boldsymbol{I} & 2\boldsymbol{I} & -\boldsymbol{I} & & \\ & -\boldsymbol{I} & \ddots & \ddots & \\ & & \ddots & & -\boldsymbol{I} \\ \mathbf{0} & & & -\boldsymbol{I} & \boldsymbol{I} \end{bmatrix} . \tag{4.40}$$

Similarly, the cross term is given by

$$\boldsymbol{\mathcal{H}}_{12} = \boldsymbol{\mathcal{H}}_{21}^T = \left[\left(\frac{\boldsymbol{H}_1^T \boldsymbol{d}'(\boldsymbol{x})}{\sigma_w^2} \right)^T, \ldots, \left(\frac{\boldsymbol{H}_K^T \boldsymbol{d}'(\boldsymbol{x})}{\sigma_w^2} \right)^T \right]^T , \tag{4.41}$$

where $\boldsymbol{d}'(\boldsymbol{x})$ *is defined in* (E.9).

Proof. See Appendix E. □

Using the sub-matrices derived above, the following result is immediate.

Lemma 5. *The covariance matrices of estimators* $\hat{\boldsymbol{\Theta}}$ *and* $\hat{\boldsymbol{x}}$ *can be lower bounded as*

$$\mathbb{E}\left[\left(\hat{\boldsymbol{\Theta}} - \boldsymbol{\Theta} \right) \left(\hat{\boldsymbol{\Theta}} - \boldsymbol{\Theta} \right)^T \right] \succeq \boldsymbol{HCRB}_{\boldsymbol{\Theta}} \tag{4.42}$$

$$\mathbb{E}\left[(\hat{\boldsymbol{x}} - \boldsymbol{x}) (\hat{\boldsymbol{x}} - \boldsymbol{x})^T \right] \succeq \boldsymbol{HCRB}_{\boldsymbol{x}} , \tag{4.43}$$

where $\boldsymbol{HCRB}_{\boldsymbol{\Theta}} = \left(\boldsymbol{\mathcal{H}}_{11} - \boldsymbol{\mathcal{H}}_{12} \boldsymbol{\mathcal{H}}_{22}^{-1} \boldsymbol{\mathcal{H}}_{21} \right)^{-1}$ *and* $\boldsymbol{HCRB}_{\boldsymbol{x}} = \left(\boldsymbol{\mathcal{H}}_{22} - \boldsymbol{\mathcal{H}}_{21} \boldsymbol{\mathcal{H}}_{11}^{-1} \boldsymbol{\mathcal{H}}_{12} \right)^{-1}$.

Proof. The proof simply follows from Theorem 2 and the inversion properties of a block matrix . □

Lemma 5 can be used to benchmark the estimation performance of the EM and the LS algorithms discussed in Section 4.4. Two special cases when $\boldsymbol{x}$ or $\boldsymbol{\Theta}$ may be known are described below.

Corollary 1. *In case when the location $\boldsymbol{x}$ may be known, the covariance matrix of $\hat{\boldsymbol{\Theta}}$ can be lower bounded as*

$$\mathbb{E}\left[\left(\hat{\boldsymbol{\Theta}}-\boldsymbol{\Theta}\right)\left(\hat{\boldsymbol{\Theta}}-\boldsymbol{\Theta}\right)^{T}\right] \succeq \boldsymbol{\mathcal{H}}_{11}^{-1} \,. \tag{4.44}$$

Similarly, if the parameter $\boldsymbol{\Theta}$ is known, we have

$$\mathbb{E}\left[\left(\hat{\boldsymbol{x}}-\boldsymbol{x}\right)\left(\hat{\boldsymbol{x}}-\boldsymbol{x}\right)^{T}\right] \succeq \boldsymbol{\mathcal{H}}_{22}^{-1} \,. \tag{4.45}$$

Using the lower bound for $\hat{\boldsymbol{\Theta}}$ in (4.44), an expression for the corresponding lower bound on $\hat{\boldsymbol{\Psi}}$ is given by [31]

$$\mathbb{E}\left[\left(\hat{\boldsymbol{\Psi}}-\boldsymbol{\Psi}\right)\left(\hat{\boldsymbol{\Psi}}-\boldsymbol{\Psi}\right)^{T}\right] \succeq \left(\frac{\partial \mathbb{E}\left[\boldsymbol{g}\left(\boldsymbol{\Theta}\right)\right]}{\partial \boldsymbol{\Theta}}\right)^{T} \mathbf{HCRB}_{\boldsymbol{\Theta}} \left(\frac{\partial \mathbb{E}\left[\boldsymbol{g}\left(\boldsymbol{\Theta}\right)\right]}{\partial \boldsymbol{\Theta}}\right) \,, \tag{4.46}$$

where $\boldsymbol{g}\left(\boldsymbol{\Theta}\right)$ is the transformation defined in (4.2) and $\mathbf{HCRB}_{\boldsymbol{\Theta}}$ is given by Lemma 5. Due to the non-linear function $\boldsymbol{g}$, an exact expression for (4.46) is mathematically intractable. Hence, we use numerical methods to evaluate the lower bound on $\hat{\boldsymbol{\Psi}}$.

4.6 Simulation Results

In this section, we present simulation results to corroborate our findings in the earlier sections. In particular, we compare the relative performance of the estimators proposed above against the theoretical lower bounds.

4.6.1 Simulation Setup

The anchors are located at (14, 21), (6, -8) and (24, 4). The clock skew is randomly drawn from $[0.998, 1.002]$ and the offset is drawn randomly from $[1, 10]$. Unless stated otherwise, the location of Node X is generated by drawing x_1 and x_2 randomly from $[1, 10]$. The Gauss-Markov process noise variance σ_n^2 is set to 10^{-4}.

The measure of accuracy used to benchmark different estimators is the MSE. The MSE of the location estimator is given by $\mathbb{E}\left[(\hat{x}_1 - x_1)^2 + (\hat{x}_2 - x_2)^2\right]$. The average MSEs of $\hat{\boldsymbol{\alpha}}$ and $\hat{\boldsymbol{\beta}}$ are given by

$$\begin{aligned}
\mathrm{MSE}_{\boldsymbol{\alpha}} &= \frac{1}{K}\mathbb{E}\left[(\hat{\boldsymbol{\alpha}} - \boldsymbol{\alpha})^T (\hat{\boldsymbol{\alpha}} - \boldsymbol{\alpha})\right] \\
\mathrm{MSE}_{\boldsymbol{\beta}} &= \frac{1}{K}\mathbb{E}\left[\left(\hat{\boldsymbol{\beta}} - \boldsymbol{\beta}\right)^T \left(\hat{\boldsymbol{\beta}} - \boldsymbol{\beta}\right)\right] .
\end{aligned}$$

4.6.2 Convergence Analysis of the EM Algorithm

An important issue in the formulation of the EM algorithm is the impact of the increasing iterations on the recursive estimate of the location of Node X. In order to show the updates in the EM algorithm, the simulation results are averaged over 100 realizations of $\boldsymbol{x} = (2, 4)$ with $\sigma_w^2 = 10^{-1}$ and $K = 2$. Fig. 4.2 shows the updates in the likelihood function $Q\left(\boldsymbol{x}, \hat{\boldsymbol{x}}^{(i)}\right)$ as a function of $\boldsymbol{x}$ as the number of iterations i increases. It can be noticed that at each iteration, $Q\left(\boldsymbol{x}, \hat{\boldsymbol{x}}^{(i)}\right)$ is a strictly concave function and hence, does not present any local maxima. This allows the algorithm to converge at the solution uniquely. Initially, for $i = 1$, $Q\left(\boldsymbol{x}, \hat{\boldsymbol{x}}^{(i)}\right)$ is located away from the actual location $\boldsymbol{x}$. As i increases, $Q\left(\boldsymbol{x}, \hat{\boldsymbol{x}}^{(i)}\right)$ starts to move towards the left, finally settling at coordinates $(2, 4)$ at about $i = 12$. Hence, the plot illustrates the improvement obtained in estimates of $\hat{\boldsymbol{x}}^{(i)}$ with each iteration of the EM algorithm.

Fig. 4.3 shows the improvement in the recursive estimate $\hat{\boldsymbol{x}}^{(i)}$ as the number of iterations increases for different values of K. It is observed that the EM algorithm quickly settles at the true location coordinates in around 12 iterations for $K = 2$. Moreover, the convergence is faster with $K = 4$ compared to $K = 2$. Hence, the proposed iterative location estimation algorithm performs efficiently even with few message exchanges.

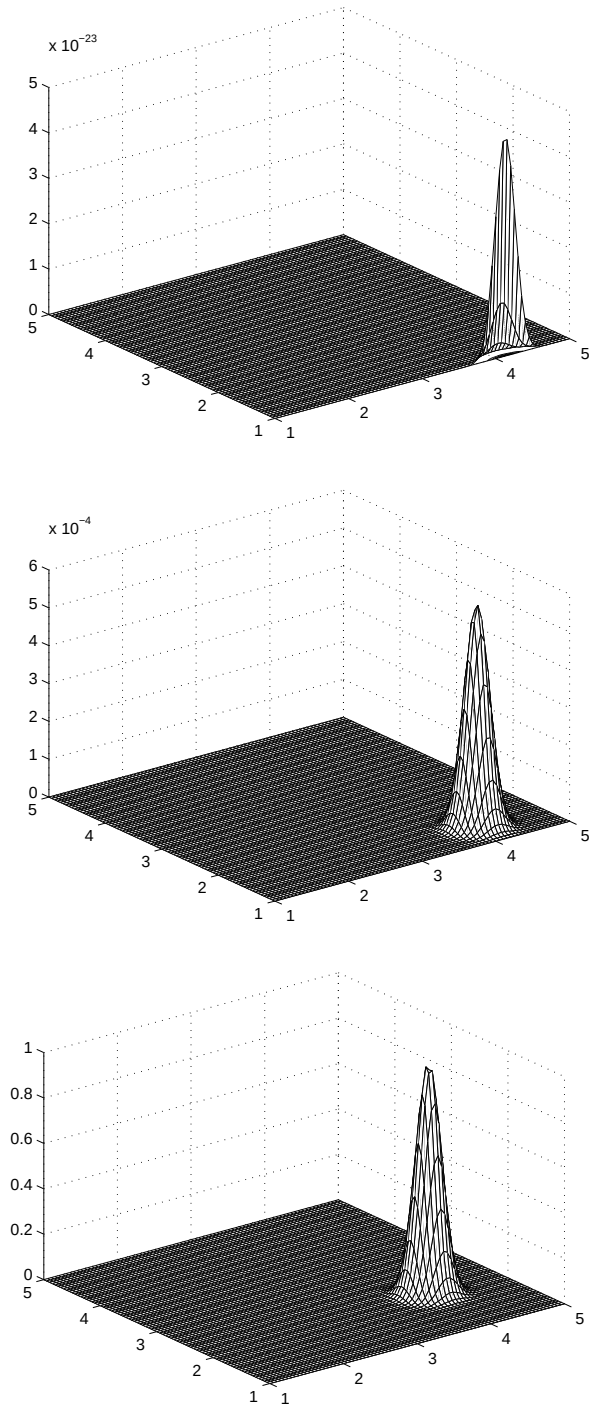

Figure 4.2: The updates of $Q\left(\boldsymbol{x}, \hat{\boldsymbol{x}}^{(i)}\right)$ versus the number of iterations i for $i = 1$, $i = 4$ and $i = 12$ with number of message exchanges $K = 2$ and measurement noise variance $\sigma_w^2 = 10^{-1}$.

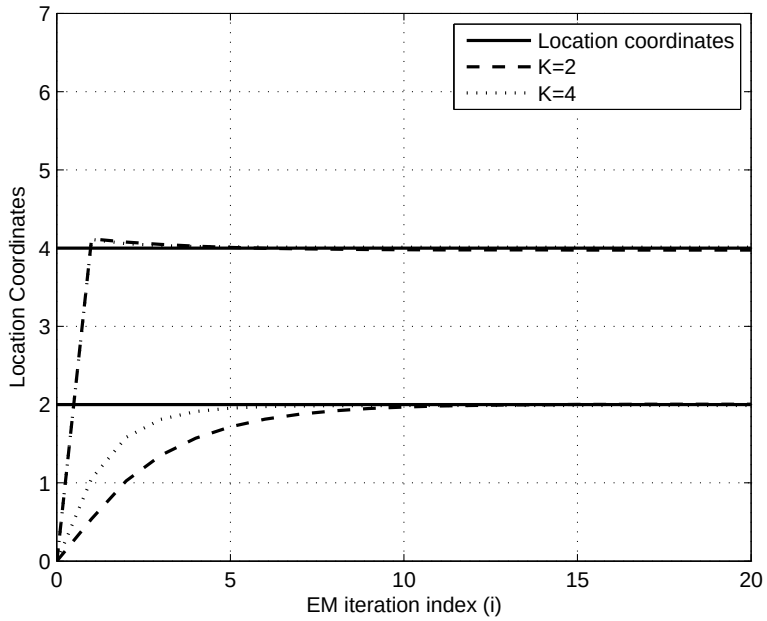

Figure 4.3: Updates of the recursive estimate $\hat{\boldsymbol{x}}^{(i)}$ versus the number of iterations i in the EM algorithm for $K = 2$ and $K = 4$ with measurement noise variance $\sigma_w^2 = 10^{-1}$. The location coordinates are set to $\boldsymbol{x} = (2, 4)$.

The convergence of the EM algorithm with increasing iterations for the two cases of forward step-only EM and the backward step EM is also illustrated in Fig. 4.4. It is evident that the backward step converges at the true locations coordinates in fewer iterations compared to a forward step-only approach. This is a direct consequence of the smoothing of the random parameters using a backward sweep. However, this improvement comes at the cost of additional processing required to carry out the time-series smoothing.

4.6.3 Backward Step versus Forward Step

It is commonly known that smoothing the time series yields improvements in the estimator of the random parameters. Fig. 4.5 plots a comparison of the MSE performance of the estimators of clock skew and offset for the forward step-only and backward step EM with measurement noise variance $\sigma_w^2 = 10^{-1}$. It is indeed

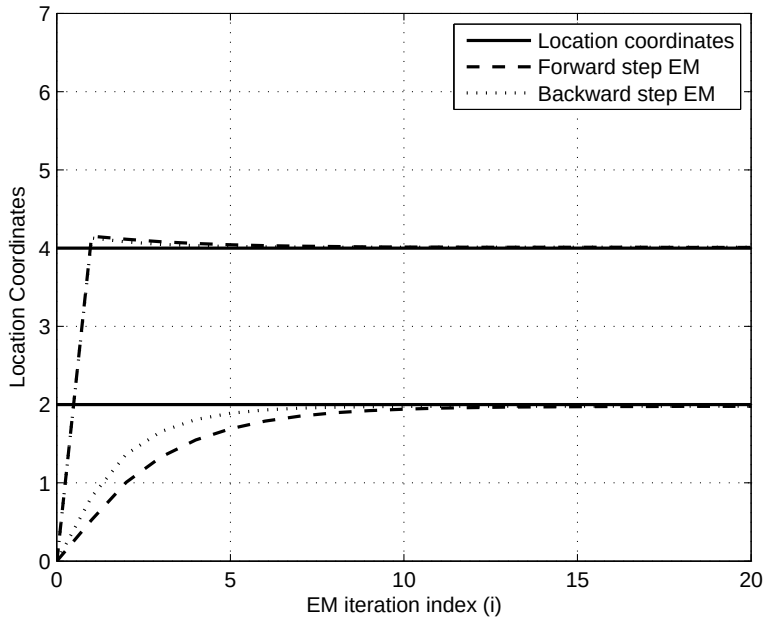

Figure 4.4: A comparison of the updates of the recursive estimate $\hat{\boldsymbol{x}}^{(i)}$ with the iteration i for the forward step-only and backward step EM algorithm for message exchanges $K = 2$ and measurement noise variance $\sigma_w^2 = 10^{-1}$.

observed that the backward step approach consistently outperforms the forward-only step. Moreover, the performance gap increases as the number of message exchanges K increase. This is intuitively satisfying since a higher value of K ensures that the smoother has a greater room for improving the estimates by using the complete data sequence. In addition, it can be observed that the performance improvement of the backward step is more pronounced for the clock skew $\boldsymbol{\alpha}$ as compared to the offset $\boldsymbol{\beta}$ and the MSE incurred in estimating the clock skew is lower than the corresponding MSE for the offset. The improvement using the backward step suggests that it would be useful to smooth the estimates of the clock parameters, since it yields higher accuracy and can result in a reduction of energy spent on exchanging timing information for re-synchronization. Similar MSE curves are also obtained for the iterative LS algorithm which are not shown here for brevity.

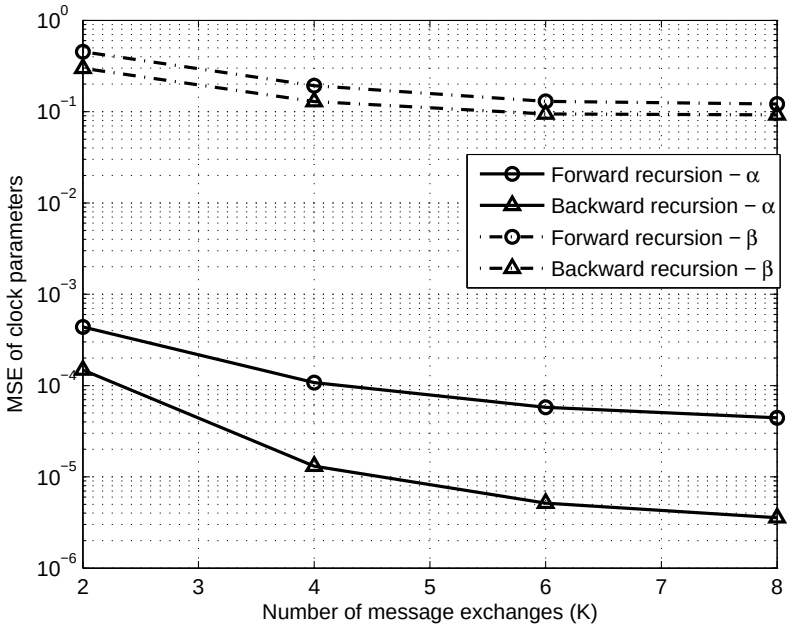

Figure 4.5: MSE performance comparison of clock parameters for the forward step-only and the backward step EM with increasing number of message exchanges K for measurement noise variance $\sigma_w^2 = 10^{-1}$.

4.6.4 Estimator Performance Analysis

In this section, we present simulation results for comparing the MSE performance with the theoretical HCRB derived in Section 4.5. The MSE plots are provided for the estimates of location as well as the clock parameters of the unknown node to observe the impact of measurement noise and the number of message exchanges.

Fig. 4.6 illustrates the MSE incurred in estimating the location as the variance σ_w^2 decreases. It can be observed that the MSE decays with a decrease in σ_w^2. Moreover, the MSE decay also increases as K increases from $K = 4$ to $K = 8$. The EM and the LS algorithm have the same performance for low to moderate noise variances. In addition, the MSE remains fairly close to the theoretical HCRB but does not attain it. This could potentially be due to the reason that HCRB is known to be less tight for the non-random part of the parameters [42].

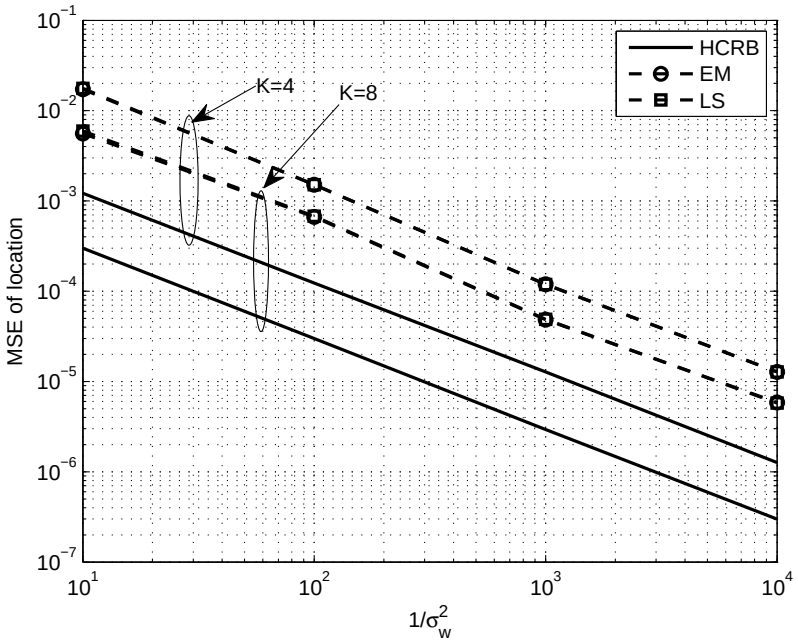

Figure 4.6: A comparison of MSE of location estimates versus measurement noise variance σ_w^2 for $K = 4$ and $K = 8$.

The MSE performance of the skew estimates provided by the EM and LS algorithms is illustrated in Fig. 4.7 as the measurement noise decreases. It is observed that the proposed estimator has high fidelity and matches closely with the theoretical lower bound provided by HCRB. Moreover, the EM and LS approaches have similar performance. As expected, the performance improves as more messages are exchanged so that K increases.

Similarly, Fig. 4.8 depicts the MSE curves for the estimation of clock offset versus measurement noise for $K = 4$ and $K = 8$. In this case, the MSE incurred in estimation is higher than the corresponding values for skew. However, the MSE performance is still close to the lower bounds and the gap diminishes as more messages are exchanged. Similar to the aforementioned observations, the EM and LS algorithms exhibit similar performance.

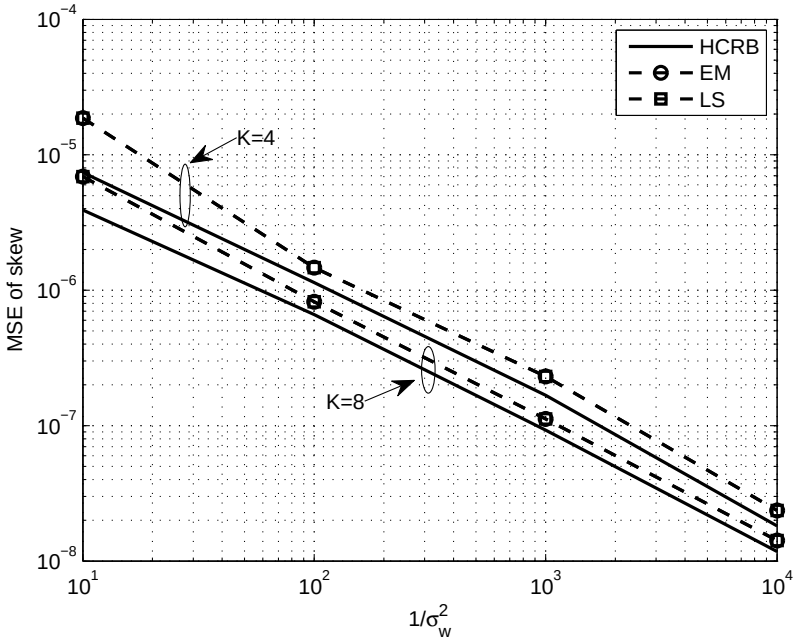

Figure 4.7: A comparison of MSE of skew estimates versus measurement noise variance σ_w^2 for $K = 4$ and $K = 8$.

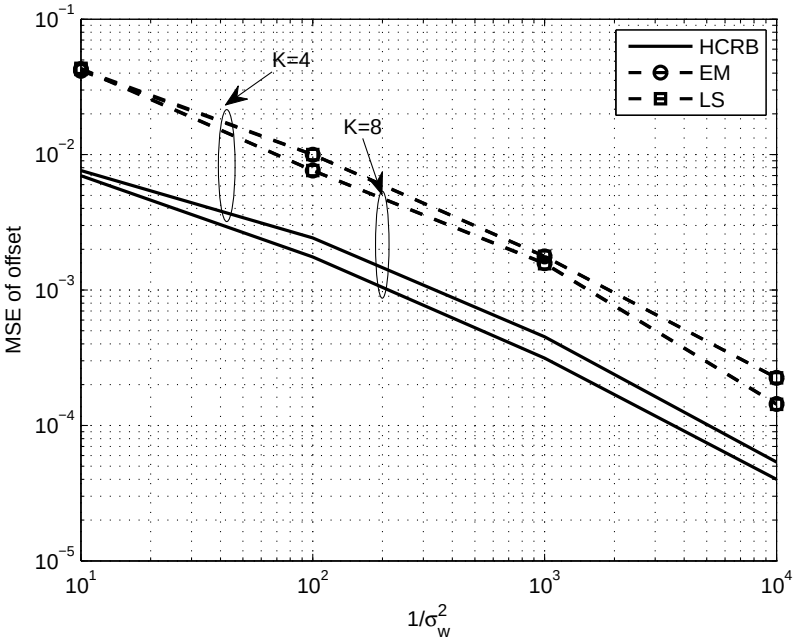

Figure 4.8: A comparison of MSE of offset estimates versus measurement noise variance σ_w^2 for $K = 4$ and $K = 8$.

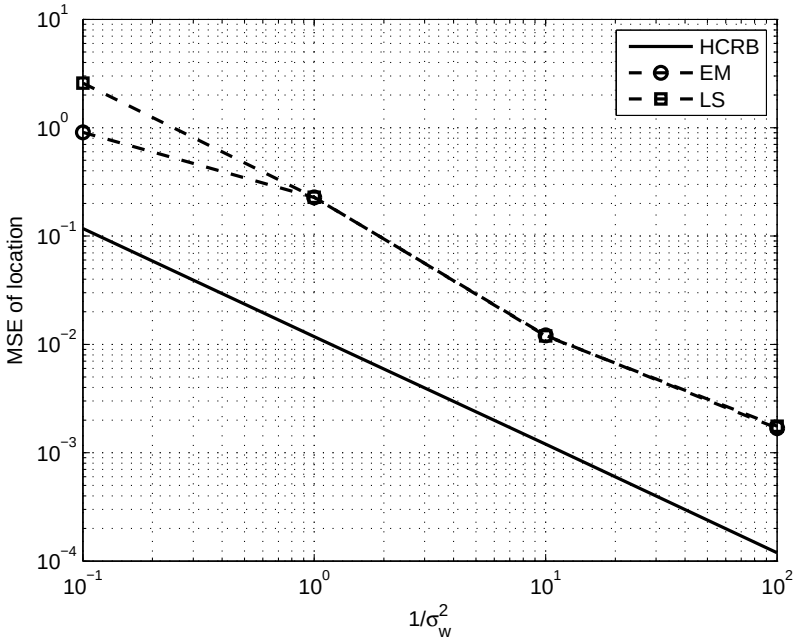

Figure 4.9: MSE comparison of EM and LS algorithms versus measurement noise variance σ_w^2 for $K = 4$.

4.6.5 LS Estimator Degrades Only at Large Measurement Noise

The LS estimator is obtained by a non-linear processing of the data and ignoring the second order terms in measurement noise. Hence, we expect to see a performance degradation in the LS estimator when σ_w^2 becomes large. This observation is illustrated in Fig. 4.9 where the MSE in location estimation is plotted for various values of σ_w^2 and $K = 4$. Performance starts to degrade as σ_w^2 increases. Since in practical sensor networks, measurement noise is usually small, the LS approach is a viable alternative to the potentially costly EM algorithm which requires a 2-D norm minimization for location estimation in each iteration.

4.7 Summary

The problems of node localization and clock synchronization in a wireless sensor network are closely tied. In this section, the joint estimation of an unknown node's

location and its time-varying clock parameters is considered. Accurately tracking the temporal variation in the clock parameters can lower the re-synchronization requirements and result in significant communication savings. The data exchange with the known anchors follows a two-way message exchange mechanism. In order to alleviate the computational complexity associated with the MAP estimator, two iterative approaches are proposed for joint localization and timing synchronization. The first approach utilizes the EM algorithm which iteratively estimates the unknown node's location by considering the clock parameters as hidden variables and estimates the location as the solution to a 2-D norm minimization. In order to further simplify the EM algorithm, an LS based location estimation algorithm is presented. This algorithm results in closed form expressions for the joint estimation problem and is particularly suited in scenarios where the computational resources come at a high premium. The MSE of the various estimators is lower bounded by deriving the HCRB. Simulation results corroborate our theoretical findings and demonstrate the high accuracy of the iterative algorithms. It is observed that the performance of the EM and LS algorithms are comparable and hence, LS can be used in low-cost sensor networks.

5. CLASSICAL FREQUENTIST AND BAYESIAN BOUNDS FOR EXPONENTIAL FAMILY DISTRIBUTIONS*

It is of significant theoretical interest to establish the best achievable estimation performance by placing lower bounds on an estimator. The placement of a lower bound also allows us to compare different estimators.

5.1 Main Contributions

Our main contributions in this section are as follows.

1. Theoretical lower bounds on the MSE of an estimator are derived in the classical frequentist inference approach as well as in the Bayesian estimation framework by considering the likelihood function as an arbitrary member of the exponential family of distributions, which constitutes a wide class. Hence, the results are quite general, and can be used in their own right in parameter estimation theory.

2. The implications of these results on the MSE of estimators derived in Section 2 are also discussed.

We consider the likelihood function as an arbitrary member of the exponential family of distributions. In addition, depending on whether the domain of the likelihood depends on the parameter to be estimated, both cases of unconstrained as well as constrained likelihood functions are discussed to maintain full generality. The general expressions for the unconstrained and constrained likelihood functions for observations $\boldsymbol{Z} \triangleq [Z_1, \ldots, Z_N]^T$ are given by

*Reprinted with permission from "A factor graph approach to clock offset estimation in wireless sensor networks," Aitzaz Ahmad, Davide Zennaro, Erchin Serpedin, and Lorenzo Vangelista, 2012, IEEE Transactions on Information Theory, vol. 58, no. 7, pg. 4244-4260, Copyright 2012 by IEEE.

Unconstrained Likelihood:

$$f(\boldsymbol{Z};\rho) \propto \exp\left(\rho \sum_{j=1}^{N} \eta(Z_j) - N\phi(\rho)\right) \tag{5.1}$$

Constrained Likelihood:

$$f(\boldsymbol{Z};\rho) \propto \exp\left(\rho \sum_{j=1}^{N} \eta(Z_j) - N\phi(\rho)\right) \prod_{j=1}^{N} \mathbb{I}(Z_j - \rho)\ , \tag{5.2}$$

where ρ is the scalar parameter to be estimated. The goal is to derive lower bounds on the variance of estimators of ρ.

5.2 Classical Frequentist Bounds

For the case of classical frequentist inference, the Cramér-Rao and the Chapman-Robbins bounds are considered.

5.2.1 Cramér-Rao Lower Bound

The Cramér-Rao lower bound (CRB) is a lower bound on the variance of an unbiased estimator of a deterministic parameter [31]. It is useful primarily because it is relatively simple to compute. However, it relies on certain 'regularity conditions' which are not satisfied by constrained likelihood functions when the domain of the likelihood depends on the unknown parameter (cf. (5.2)). Hence, the CRB is determined for the case of unconstrained likelihood functions only.

In particular, the CRB states that the variance of an unbiased estimator of ρ is lower bounded by

$$\mathrm{Var}(\hat{\rho}) \geq \frac{-1}{\mathbb{E}\left[\frac{\partial^2 \ln f(\boldsymbol{Z};\rho)}{\partial \rho^2}\right]}\ . \tag{5.3}$$

Lemma 6. *The CRB for ρ in the unconstrained likelihood function in* (5.1) *is given*

by

$$\mathrm{Var}(\hat{\rho}) \geq \frac{1}{N\sigma_\eta^2} \tag{5.4}$$

where

$$\sigma_\eta^2 = \frac{\partial^2 \phi\left(\rho\right)}{\partial \rho^2} \, .$$

Proof. The Fisher information for the likelihood function is given by

$$I(\rho) \triangleq \mathbb{E}\left[\frac{\partial^2 \ln f\left(\boldsymbol{Z};\rho\right)}{\partial \rho^2}\right] = -N\frac{\partial^2 \phi\left(\rho\right)}{\partial \rho^2}$$

and the proof readily follows. □

5.2.2 Chapman-Robbins Bound

The Chapman-Robbins bound (CHRB), proposed in [13], sets a lower bound on the variance of an estimator of a deterministic parameter. The CHRB does not make any assumptions on the differentiability of the likelihood function and regularity conditions that often constrain the use of the CRB, and is substantially tighter than the CRB in many situations. Hence, the CHRB is employed to determine a lower bound on the variance of an unbiased estimator of ρ for constrained likelihood functions.

In general for a parameter ρ, the CHRB is given by

$$\mathrm{Var}(\hat{\rho}) \geq \left[\inf_h \frac{1}{h^2}\left\{\mathbb{E}\left(\frac{f(\boldsymbol{Z};\rho+h)}{f(\boldsymbol{Z};\rho)}\right)^2 - 1\right\}\right]^{-1} , \tag{5.5}$$

which can be evaluated as shown below.

Lemma 7. *The CHRB for the parameter ρ given the likelihood function* (5.2) *can*

be expressed as

$$\mathrm{Var}(\hat{\rho}) \geq \left[\inf_{h} \frac{\left\{ (M_{\eta}(h))^{-2N} \cdot \zeta^{N}(h) - 1 \right\}}{h^2} \right]^{-1}, \tag{5.6}$$

where $M_{\eta}(h)$ *is the MGF of the statistic* $\eta(Z_j)$ *and*

$$\zeta(h) \triangleq \mathbb{E}\left[\exp\left(2h\eta(Z_j)\right) \mathbb{I}\left(Z_j - \rho - h\right) \right] \tag{5.7}$$

with the expectation taken with respect to any Z_j*.*

Proof. The ratio of the likelihood functions can be expressed as

$$\begin{aligned}
\frac{f(\boldsymbol{Z};\rho+h)}{f(\boldsymbol{Z};\rho)} =& \frac{\exp\left((\rho+h)\sum_{j=1}^{N}\eta(Z_j) - N\phi(\rho+h)\right)\prod_{j=1}^{N}\mathbb{I}(Z_j-\rho-h)}{\exp\left(\rho\sum_{j=1}^{N}\eta(Z_j) - N\phi(\rho)\right)\prod_{j=1}^{N}\mathbb{I}(Z_j-\rho)} \\
=& \exp\left(h\sum_{j=1}^{N}\eta(Z_j) - N\phi(\rho+h) + N\phi(\rho)\right)\prod_{j=1}^{N}\mathbb{I}(Z_j-\rho-h) \\
=& \exp\left(h\sum_{j=1}^{N}\eta(Z_j)\right)e^{(-N(\phi(\rho+h)+\phi(\rho)))}\prod_{j=1}^{N}\mathbb{I}(Z_j-\rho-h)\ .
\end{aligned}$$

The expectation of the ratio of the likelihood functions can now be calculated as

$$\begin{aligned}
&\mathbb{E}\left(\frac{f(\boldsymbol{Z};\rho+h)}{f(\boldsymbol{Z};\rho)}\right)^2 \\
&= \mathbb{E}\left[\exp\left(2h\sum_{j=1}^{N}\eta(Z_j)\right)\exp\left(-2N(\phi(\rho+h)+\phi(\rho))\right)\prod_{j=1}^{N}\mathbb{I}(Z_j-\rho-h)\right] \\
&= \exp\left(-2N(\phi(\rho+h)+\phi(\rho))\right)\mathbb{E}\left[\exp\left(2h\sum_{j=1}^{N}\eta(Z_j)\right)\prod_{j=1}^{N}\mathbb{I}(Z_j-\rho-h)\right] \\
&= \left(M_{\eta(Z)}(h)\right)^{-2N}\mathbb{E}\left[\exp\left(2h\sum_{j=1}^{N}\eta(Z_j)\right)\prod_{j=1}^{N}\mathbb{I}(Z_j-\rho-h)\right],
\end{aligned}$$

where it follows from (2.12) that

$$\left(M_{\eta(Z)}(h)\right)^{-2N} = \exp -2N\left(\phi(\rho+h) - \phi(\rho)\right) .$$

Since the samples Z_j are *i.i.d*,

$$\mathbb{E}\left[\exp\left(2h\sum_{j=1}^{N}\eta(Z_j)\right)\prod_{j=1}^{N}\mathbb{I}(Z_j-\rho-h)\right] = \left(\mathbb{E}\left[\exp\left(2h\eta(Z_j)\right)\mathbb{I}(Z_j-\rho-h)\right]\right)^N .$$

With $\zeta(.)$ defined in the theorem, the proof is complete. □

5.3 Bayesian Bounds

The Bayesian Cramér-Rao bound and a Bayesian version of the Chapman-Robbins bound are derived for the Bayesian paradigm.

5.3.1 Bayesian Cramér-Rao Lower Bound

The Bayesian Cramér-Rao bound (BCRB) is a lower bound on the variance of an unbiased estimator when the parameter assumes a prior density [60]. It requires the same regularity conditions to be satisfied as its classical counterpart.

For an estimator $\hat{\rho}_k$ of ρ_k, the BCRB states that the variance of the estimator is bounded below by the lower-right sub-matrix of the inverse of the Bayesian information matrix, $J_{\mathrm{CR}}^{-1}(k)$ [60], i.e.,

$$\mathrm{Var}\left(\hat{\rho}_k\right) \geq J_{\mathrm{CR}}^{-1}(k) \stackrel{\Delta}{=} \left[\boldsymbol{J}_{\mathrm{CR}}^{-1}(k)\right]_{kk} . \tag{5.8}$$

The Bayesian information matrix is given by

$$[\boldsymbol{J}_{\mathrm{CR}}(k)]_{ij} \triangleq \mathbb{E}\left[\frac{\partial \log f(\boldsymbol{Z}_k, \boldsymbol{\rho}_k)}{\partial \rho_i}\frac{\partial \log f(\boldsymbol{Z}_k, \boldsymbol{\rho}_k)}{\partial \rho_j}\right]$$
$$= -\mathbb{E}\left[\frac{\partial^2 \log f(\boldsymbol{Z}_k, \boldsymbol{\rho}_k)}{\partial \rho_i \partial \rho_j}\right] ,$$

where the expectation is taken with respect to the joint pdf and

$$\boldsymbol{Z}_k \triangleq [Z_1, \ldots, Z_k]^T$$
$$\boldsymbol{\rho}_k \triangleq [\rho_0, \rho_1, \ldots, \rho_k]^T$$
$$f(Z_k|\rho_k) \propto \exp\left(\eta(Z_k)\rho_k - \phi_k(\rho_k)\right) . \tag{5.9}$$

It is assumed that the parameter ρ_k evolves through a Gauss-Markov model given by

$$f(\rho_k|\rho_{k-1}) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(\rho_k - \rho_{k-1})^2}{2\sigma^2}\right) . \tag{5.10}$$

A recursive formula to evaluate the Bayesian sub-matrix, derived in [59], is given by

$$J_{\mathrm{CR}}(k+1) = -E^{(2)}_{\mathrm{CR}}(k)\left(J_{\mathrm{CR}}(k) + E^{(1)}_{\mathrm{CR}}(k)\right)^{-1} E^{(2)}_{\mathrm{CR}}(k) + E^{(3A)}_{\mathrm{CR}}(k) + E^{(3B)}_{\mathrm{CR}}(k) , \tag{5.11}$$

where

$$E^{(1)}_{\mathrm{CR}}(k) \triangleq \mathbb{E}\left[-\frac{\partial^2}{\partial \rho_k^2}\log f(\rho_{k+1}|\rho_k)\right]$$
$$E^{(2)}_{\mathrm{CR}}(k) \triangleq \mathbb{E}\left[-\frac{\partial^2}{\partial \rho_k \partial \rho_{k+1}}\log f(\rho_{k+1}|\rho_k)\right]$$
$$E^{(3A)}_{\mathrm{CR}}(k) \triangleq \mathbb{E}\left[-\frac{\partial^2}{\partial \rho_{k+1}^2}\log f(\rho_{k+1}|\rho_k)\right]$$
$$E^{(3B)}_{\mathrm{CR}}(k) \triangleq \mathbb{E}\left[-\frac{\partial^2}{\partial \rho_{k+1}^2}\log f(Z_{k+1}|\rho_{k+1})\right] ,$$

and the expectation is again with respect to the joint pdf.

Lemma 8. *For the Bayesian framework in* (5.9) *and* (5.10)*, the recursive Bayesian information matrix in* (5.11) *is given by*

$$J_{\mathrm{CR}}(k+1) = \left(\sigma^2 + J_{\mathrm{CR}}^{-1}(k)\right)^{-1} + \sigma_{\eta_k}^2 \tag{5.12}$$

with $J_{\mathrm{CR}}(0) = 0$.

Proof. For the density functions, $f(\rho_k|\rho_{k-1})$ and $f(Z_k|\rho_k)$ in (5.9), it can be verified that

$$E_{\mathrm{CR}}^{(1)}(k) = \frac{1}{\sigma^2}\,, \quad E_{\mathrm{CR}}^{(2)}(k) = -\frac{1}{\sigma^2}\,, \quad E_{\mathrm{CR}}^{(3A)}(k) = \frac{1}{\sigma^2}\,,$$

and

$$\begin{aligned} E_{\mathrm{CR}}^{(3B)}(k) &= \int\int \frac{\partial^2 \phi_k(\rho_{k+1})}{\partial \rho_{k+1}^2} f(\rho_{k+1}, Z_{k+1}) d\rho_{k+1} dZ_{k+1} \\ &= \sigma_{\eta_k}^2\,. \end{aligned}$$

The proof follows by plugging these quantities in (5.11). □

5.3.2 Bayesian Chapman-Robbins Bound

A Bayesian version of the Chapman-Robbins bound (BCHRB) can be used to provide a lower bound on the variance of an estimator of ρ_k when there are no regularity assumptions on the likelihood [8].

The BCHRB states that the variance of an estimator $\hat{\boldsymbol{\rho}}_k$ of $\boldsymbol{\rho}_k$ is lower bounded as

$$\mathrm{Var}(\hat{\boldsymbol{\rho}}_k) - \left[T_k(\mathbf{h}_k) - 1\right]^{-1} \mathbf{h}_k \mathbf{h}_k^T \succeq \mathbf{0}$$

with $\succeq$ in the positive semi-definite sense,

$$T_k(\mathbf{h}_k) \triangleq \mathbb{E}\left[\left(\frac{f(\mathbf{Z}_k, \boldsymbol{\rho}_k + \mathbf{h}_k)}{f(\mathbf{Z}_k, \boldsymbol{\rho}_k)}\right)^2\right] ,$$

and $\mathbf{h}_k \triangleq [0, h_1, \ldots, h_k]^T$.

Theorem 3. *The BCHRB for the parameter ρ_k can be expressed as*

$$\mathrm{Var}(\hat{\rho}_k) \geq \frac{1}{J_{CH,k}} ,$$

where

$$J_{CH,k} = \inf_{\mathbf{h}_k} \frac{T_k(\mathbf{h}_k) - 1}{h_k^2} ,$$

and

$$T_k(\mathbf{h}_k) = \left(\prod_{j=1}^{k} M_\eta^{-2}(h_j) M_\eta(2h_j)\right) \exp\left[\sum_{j=1}^{k} \frac{(h_j - h_{j-1})^2}{\sigma^2}\right] . \tag{5.13}$$

Proof. We have

$$\begin{aligned} T_k(\mathbf{h}_k) \triangleq \mathbb{E}\left[\left(\frac{f(\mathbf{Z}_k, \boldsymbol{\rho}_k + \mathbf{h}_k)}{f(\mathbf{Z}_k, \boldsymbol{\rho}_k)}\right)^2\right] &= \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} \left(\frac{f(\mathbf{Z}_k, \boldsymbol{\rho}_k + \mathbf{h}_k)}{f(\mathbf{Z}_k, \boldsymbol{\rho}_k)}\right)^2 f(\mathbf{Z}_k, \boldsymbol{\rho}_k) d\mathbf{Z}_k d\boldsymbol{\rho}_k \\ &= S(\mathbf{h}_k) \int_{-\infty}^{+\infty} \frac{f(\boldsymbol{\rho}_k + \mathbf{h}_k)^2}{f(\boldsymbol{\rho}_k)} d\boldsymbol{\rho}_k , \end{aligned}$$

where

$$S(\mathbf{h}_k) \triangleq \int_{-\infty}^{+\infty} \left(\frac{f(\mathbf{Z}_k|\boldsymbol{\rho}_k + \mathbf{h}_k)}{f(\mathbf{Z}_k|\boldsymbol{\rho}_k)}\right)^2 f(\mathbf{Z}_k|\boldsymbol{\rho}_k) d\mathbf{Z}_k . \tag{5.14}$$

Continuing with the calculations

$$\begin{aligned} T_h(\mathbf{h}_k) =& S(\mathbf{h}_k) \int_{-\infty}^{+\infty} \frac{f(\boldsymbol{\rho}_k + \mathbf{h}_k)^2}{f(\boldsymbol{\rho}_k)} d\boldsymbol{\rho}_k \\ =& S(\mathbf{h}_k) \int_{-\infty}^{+\infty} \frac{f^2(\rho_0 + h_0)}{f(\rho_0)} \prod_{j=1}^{k} \frac{f^2(\rho_j + h_j | \rho_{j-1} + h_{j-1})}{f(\rho_j | \rho_{j-1})} d\boldsymbol{\rho}_k \\ =& S(\mathbf{h}_k) \int_{-\infty}^{+\infty} \prod_{j=1}^{k} \frac{f^2(\rho_j + h_j | \rho_{j-1} + h_{j-1})}{f(\rho_j | \rho_{j-1})} d\boldsymbol{\rho}_k \ . \end{aligned}$$

Since $\frac{f^2(\rho_j + h_j | \rho_{j-1} + h_{j-1})}{f(\rho_j | \rho_{j-1})}$ can be verified to be equal to $(j = 1, \ldots, k)$

$$\begin{aligned} &\frac{1}{\sigma\sqrt{2\pi}} \exp\left[-\frac{\rho_{j-1}^2}{2\sigma^2} + \frac{\rho_j + 2(h_j - h_{j-1})}{\sigma^2} \rho_{j-1} \right] \times \\ &\exp\left[-\frac{(h_j - h_{j-1})^2}{\sigma^2} \right] \exp\left[-\frac{\rho_j^2}{2\sigma^2} - \frac{2\rho_j (h_j - h_{j-1})}{\sigma^2} \right] \end{aligned}$$

it turns out that

$$\int_{-\infty}^{+\infty} \frac{f^2(\rho_j + h_j | \rho_{j-1} + h_{j-1})}{f(\rho_j | \rho_{j-1})} d\rho_{j-1} = \exp\left[\frac{(h_j - h_{j-1})^2}{\sigma^2} \right]$$

and

$$T_k(\mathbf{h}_k) = S(\mathbf{h}_k) \exp\left[\frac{1}{\sigma^2} \sum_{j=1}^{k} (h_j - h_{j-1})^2 \right] \ .$$

It can be easily verified that (5.14) can be written as

$$S(\mathbf{h}_k) = \prod_{j=1}^{k} \int_{-\infty}^{+\infty} \left(\frac{f(Z_j | \rho_j + h_j)}{f(Z_j | \rho_j)} \right)^2 f(Z_j | \rho_j) dZ_j \ .$$

Moreover, it can be noted that

$$\left(\frac{f(Z_j | \rho_j + h_j)}{f(Z_j | \rho_j)} \right)^2 = \exp\left[-2 \left(\phi_\rho(\rho_j + h_j) - \phi_\rho(\rho_j) \right) \right] \times \exp\left(2 h_j \eta_\rho(Z_j) \right)$$

and therefore

$$\int_{-\infty}^{+\infty} \left(\frac{f(Z_j|\rho_j + h_j)}{f(Z_j|\rho_j)} \right)^2 f(Z_j|\rho_j) dZ_j = M_{\eta_\rho}^{-2}(h_j) \times \mathbb{E}\left[\exp\left(2h_j \eta_\rho(Z_j)\right)\right] \ .$$

Then, since

$$\mathbb{E}\left[\exp\left(2h_j \eta_\rho(Z_j)\right)\right] = \exp\left(\phi_\rho(\rho_j + 2h_j) - \phi_\rho(\rho_j)\right) \tag{5.15}$$

it can be easily seen that

$$\int_{-\infty}^{+\infty} \left(\frac{f(Z_j|\rho_j + h_j)}{f(Z_j|\rho_j)} \right)^2 f(Z_j|\rho_j) dZ_j = M_{\eta_\rho}^{-2}(h_j) M_{\eta_\rho}(2h_j) \ ,$$

thus getting

$$S(\mathbf{h}_k) = \prod_{j=1}^{k} M_{\eta_\rho}^{-2}(h_j) M_{\eta_\rho}(2h_j) \ .$$

□

5.4 Relation to Clock Offset Estimation

Using (2.5), the following result is immediate.

Proposition 1. *The MSE of any estimator of β can be expressed as*

$$\mathrm{MSE}\left(\hat{\beta}\right) = \frac{1}{4}\left(Var\left(\hat{\xi}\right) + \mathrm{Var}\left(\hat{\psi}\right)\right) + \frac{1}{4}\left(b_\xi - b_\psi\right)^2 \ ,$$

where b_ξ and b_ψ are the biases of the estimators $\hat{\xi}$ and $\hat{\psi}$, respectively.

5.4.1 Gaussian Distribution - CRB

If the likelihood function for ξ is Gaussian distributed, then using (2.18) and (5.4), it is straightforward to see that

$$\mathrm{MSE}\left(\hat{\beta}\right) \geq \frac{\sigma_\xi^2 + \sigma_\psi^2}{4N} \ . \tag{5.16}$$

As a remark, it is evident in this case that $\hat{\beta}_{\mathrm{ML}}$ (2.20) is efficient in the sense that its MSE achieves (5.16) with equality (cf. Appendix B).

5.4.2 Exponential Distribution - CHRB

If the likelihood for ξ is exponentially distributed, using (2.12) and (2.28), it can be easily verified that

$$M_{\eta_\xi(U)}(h) = 1$$

and (5.7) becomes

$$\zeta(h) = \exp\left(\lambda_\xi h\right) \ ,$$

so that the statement of the CHRB (5.6) can be rewritten as

$$\mathrm{Var}\left(\hat{\xi}\right) \geq \left[\inf_h \frac{\exp\left(\lambda_\xi h N\right) - 1}{h^2}\right]^{-1} = \frac{0.6476}{\lambda_\xi^2 N^2} \ ,$$

and similarly for $\hat{\psi}$. Using Proposition 1, it follows that

$$\begin{aligned} \mathrm{MSE}\left(\hat{\beta}\right) &= \frac{1}{4}\left(\mathrm{Var}\left(\hat{\xi}\right) + \mathrm{Var}\left(\hat{\psi}\right)\right) + \frac{1}{4}\left(b_\xi - b_\psi\right)^2 \\ &\geq \frac{0.162}{N^2}\left(\frac{1}{\lambda_\xi^2} + \frac{1}{\lambda_\psi^2}\right) + \frac{1}{4}\left(b_\xi - b_\psi\right)^2 \ . \end{aligned} \tag{5.17}$$

5.4.3 Gaussian Distribution - BCRB

In the Bayesian regime, if the likelihood function for ξ is Gaussian distributed, by using (2.53) and (5.12), it can be seen that

$$J_{\mathrm{CR},\xi}\left(k+1\right) = \left(\sigma^2 + J_{\mathrm{CR},\xi}^{-1}\left(k\right)\right)^{-1} + \frac{1}{\sigma_{\xi,k}^2} \ ,$$

with $J_{\mathrm{CR},\xi}(0) = 0$. A similar line of reasoning can be followed to derive an analogous recursion for $J_{\mathrm{CR},\psi}(k)$. The MSE of β can be now be lower bounded as

$$\mathrm{Var}(\hat{\beta}_k) \geq \frac{1}{4}\left(\frac{1}{J_{\mathrm{CR},\xi}(k)} + \frac{1}{J_{\mathrm{CR},\psi}(k)}\right) . \tag{5.18}$$

5.4.4 Exponential Distribution - BCHRB

If the likelihood for ξ_k is exponentially distributed, (5.13) turns out to be

$$T_k(\mathbf{h}_k) = \exp\left(\lambda_\xi \sum_{j=1}^{N} h_j\right) \exp\left[\frac{1}{\sigma^2}\sum_{j=1}^{k}(h_j - h_{j-1})^2\right] .$$

In fact, we just have to notice that $\phi_\xi(\xi_k)$ is a constant function over ξ_k and $\eta_\xi(U_j) = \lambda_\xi$, so that (5.15) becomes

$$\mathbb{E}\left[\exp\left(2h_j\eta_\xi(U_j)\right)\right] = \exp\left(\lambda_\xi h_j\right) .$$

Therefore $S(\mathbf{h}_k) = \exp\left(\lambda_\xi \sum_{j=1}^{N} h_j\right)$.

5.5 Summary

In order to compare various estimators, several lower bounds on the variance of an estimator have been derived in the classical frequentist inference approach as well as in the Bayesian estimation framework for likelihood functions which are arbitrary members of the exponential family, a wide class containing several distributions of interest.

6. CONCLUSIONS AND FUTURE WORK

Clock synchronization is an indispensable requirement for several wireless sensor network operations such as optimal data fusion, efficient duty cycling, target localization and tracking, and deterministic channel access schemes. Since sensors have inherent constraints of meagre power resources, the clock synchronization algorithms must be kept simple while maintaining desired accuracy. This dissertation proposes efficient clock synchronization algorithms in classical as well as Bayesian regimes in a two-way message exchange mechanism between sensor nodes.

First, a unified ML estimation approach is presented for different network delay distributions. This constitutes a simpler analytical alternative to the cumbersome graphical approaches used in prior contributions to maximize the likelihood function of the observations. An important extension to the existing clock synchronization approaches is proposed by incorporating the effects of temporal variations in clock parameters. A factor graph approach is used and inference is performed using the max-product algorithm. This idea is then extended to the more general and challenging case of network-wide clock synchronization. A synchronization algorithm using factor graphs is proposed for network-wide clock synchronization and its performance is tested for various network topologies of interest. By identifying the close statistical connections between the problems of node localization and clock synchronization, two iterative approaches are proposed for jointly estimating an unknown node's location and time-varying clock parameters. Finally, theoretical lower bounds on the MSE of an estimator are derived in classical as well as Bayesian regimes. These lower bounds are fairly general and can be useful in their own right in parameter estimation theory.

This dissertation opens several avenues for further research. It would be useful to study the extension of these results for multi-hop sensor networks. In addition, using statistical sampling approaches, developing synchronization algorithms that do not assume a specific distribution of the network delays is another idea worth exploring. The effect of clock skew can also be incorporated in the belief propagation algorithm for network-wide synchronization. This can result in a reduction in re-synchronization requests. The problem of joint estimation of a node's location and clock parameters can be extended by incorporating the effects of node mobility. Several mobility models have been proposed in the literature and it would be interesting to devise a joint tracking approach for localization and clock synchronization.

REFERENCES

[1] H.S. Abdel-Ghaffar. Analysis of synchronization algorithms with time-out control over networks with exponentially symmetric delays. *IEEE Transactions on Communications*, 50(10):1652–1661, Oct. 2002.

[2] A. Ahmad, A. Noor, and E. Serpedin. Joint clock offset and skew estimation for inactive nodes in wireless sensor networks. In *Conference on Information Sciences and Systems (CISS)*, Mar. 2011.

[3] A. Ahmad, A. Noor, E. Serpedin, H. Nounou, and M. Nounou. On clock offset estimation in wireless sensor networks with Weibull distributed network delays. In *20th International Conference on Pattern Recognition (ICPR)*, pages 2322–2325, Aug. 2010.

[4] A. Ahmad, E. Serpedin, H. Nounou, and M. Nounou. Joint node localization and time-varying clock synchronization in wireless sensor networks. *submitted to IEEE Transactions on Communications*.

[5] A. Ahmad, E. Serpedin, and K. Qaraqe. Factor graphs and message passing algorithms. In E. Serpedin, T. Chen, and D. Rajan, editors, *Mathematical Foundations for Signal Processing, Communications and Networking*. CRC Press, Boca Raton, FL, 2012.

[6] A. Ahmad, D. Zennaro, E. Serpedin, and L. Vangelista. A factor graph approach to clock offset estimation in wireless sensor networks. *IEEE Transactions on Information Theory*, 58(7):4244–4260, July 2012.

[7] A. Ahmad, D. Zennaro, E. Serpedin, and L. Vangelista. Time-varying clock offset estimation in two-way timing message exchange in wireless sensor networks using factor graphs. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP 2012)*, Mar. 2012.

[8] M. Akahira and N. Ohyauchi. A bayesian view of the hammersley-chapman-robbins-type inequality. *Statistics*, 41(2), 2007.

[9] I. F. Akyildiz, W. Su, Y. Sankarasubramanium, and E. Cayirci. A survey on sensor networks. *IEEE Communications Mag.*, 40(8):102–114, Aug. 2002.

[10] P. Barooah and J.P. Hespanha. Distributed estimation from relative measurements in sensor networks. In *3rd International Conference on Intelligent Sensing and Information Processing (ICISIP)*, pages 226–231, Dec. 2005.

[11] C. J. Bovy, H. T. Mertodimedjo, G. Hooghiemstra, H. Uijterwaal, and P. Van Mieghem. Analysis of end-to-end delay measurements in internet. In *Proc. Passive Active Meas. Workshop*, pages 26–33, Fort Collins, CO, Mar. 2002.

[12] S. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge University Press, New York, NY, 2003.

[13] D.G. Chapman and H. Robbins. Minimum variance estimation without regularity assumptions. *Annals of Math. Statistics*, 22(4):581–586, Dec. 1951.

[14] Q.M. Chaudhari, E. Serpedin, and Jang-Sub Kim. Energy-efficient estimation of clock offset for inactive nodes in wireless sensor networks. *IEEE Transactions on Information Theory*, 56(1):582–596, Jan. 2010.

[15] Q.M. Chaudhari, E. Serpedin, and K. Qaraqe. On maximum likelihood estimation of clock offset and skew in networks with exponential delays. *IEEE Transactions on Signal Processing*, 56(4):1685–1697, Apr. 2008.

[16] Q.M. Chaudhari, E. Serpedin, and K. Qaraqe. On minimum variance unbiased estimation of clock offset in a two-way message exchange mechanism. *IEEE Transactions on Information Theory*, 56(6):2893–2904, June 2010.

[17] K-Y. Cheng, K-S. Lui, Y-C. Wu, and V. Tam. A distributed multihop time synchronization protocol for wireless sensor networks using pairwise broadcast synchronization. *IEEE Transactions on Wireless Communications*, 8(4):1764–1772, April 2009.

[18] K. W. Cheung, H. C. So, W.-K. Ma, and Y. T. Chan. Least squares algorithms for time-of-arrival-based mobile location. *IEEE Transactions on Signal Processing*, 52(4):1121–1130, Apr. 2004.

[19] A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society*, 39, Series B:1–38, 1977.

[20] B. Denis, J. Pierrot, and C. Abou-Rjeily. Joint distributed synchronization and positioning in UWB *ad hoc* networks using TOA. *IEEE Transactions on Microwave Theory Tech.*, 54(4):1896 –1911, June 2006.

[21] J. Elson, L. Girod, and D. Estrin. Fine-grained network time synchronization using reference broadcasts. *SIGOPS Oper. Syst. Rev.*, 36:147–163, Dec. 2002.

[22] N.M. Freris, V.S. Borkar, and P.R. Kumar. A model-based approach to clock synchronization. In *Proceedings of the 48th IEEE Conference on Deci-*

sion and Control held jointly with the 2009 28th Chinese Control Conference (CDC/CCC), pages 5744–5749, Dec. 2009.

[23] S. Gezici. A survey on wireless position estimation. *Wireless Personal Communications(Special Issue on Towards Global and Seamless Personal Navigation)*, 44(3):263–282, Feb. 2008.

[24] S. Gezici, Z. Tian, G.B. Giannakis, H. Kobayashi, A.F. Molisch, H.V. Poor, and Z. Sahinoglu. Localization via ultra-wideband radios: A look at positioning aspects for future sensor networks. *IEEE Signal Processing Magazine*, 22(4):70 – 84, Jul. 2005.

[25] M. R. Gholami, S. Gezici, M. Rydström, and E. G. Ström. A distributed positioning algorithm for cooperative active and passive sensors. In *Personal and Indoor Mobile Radio Communications (PIMRC)*, pages 1713–1718, Sep. 2010.

[26] M. R. Gholami, S. Gezici, and E. G. Ström. Improved position estimation using hybrid TW-TOA and TDOA in cooperative networks. *IEEE Transactions on Signal Processing*, 60(7):3770–3785, Jul. 2012.

[27] A. Giridhar and P.R. Kumar. Distributed clock synchronization over wireless networks: Algorithms and analysis. In *45th IEEE Conference on Decision and Control*, pages 4915–4920, Dec. 2006.

[28] Y.-W. Hong and A. Scaliogne. A scalable synchronization protocol for large scale sensor networks and its applications. *IEEE Journal on Selected Areas in Communications*, 23(5):1085–1099, May 2005.

[29] Y. Huang, J. Benesty, G. W. Elko, and R. M. Mersereati. Real-time passive source localization: A practical linear-correction least-squares approach. *IEEE Transactions on Speech and Audio Processing*, 9(8):943–956, Nov. 2001.

[30] D.R. Jeske. On maximum-likelihood estimation of clock offset. *IEEE Transactions on Communications*, 53(1):53–54, Jan. 2005.

[31] S. M. Kay. *Fundamentals of Statistical Signal Processing. Estimation Theory.* Prentice-Hall, Upper Saddle River, NJ, 1993.

[32] H. Kim, X. Ma, and B. E. Hamilton. Tracking low-precision clocks with time-varying drifts using Kalman filtering. *IEEE/ACM Transactions on Networking*, 20(1):257–270, Feb. 2012.

[33] J-S. Kim, J. Lee, E. Serpedin, and K. Qaraqe. A robust estimation scheme for clock phase offsets in wireless sensor networks in the presence of non-gaussian random delays. *Signal Processing*, 89(6):1155–1161, 2009.

[34] J-S. Kim, J. Lee, E. Serpedin, and K. Qaraqe. Robust clock synchronization in wireless sensor networks through noise density estimation. *IEEE Transactions on Signal Processing*, 59(7):3035–3047, July 2011.

[35] F.R. Kschischang, B.J. Frey, and H.-A. Loeliger. Factor graphs and the sum-product algorithm. *IEEE Transactions on Information Theory*, 47(2):498–519, Feb. 2001.

[36] R. Kumar, S. Ganeriwal, and M. B. Srivastava. Timing-sync protocol for sensor networks. In *ACM Conf. Embedded Networked Sensor Systems*, 2003.

[37] M. Leng and Y-C. Wu. On clock synchronization algorithms for wireless sensor networks under unknown delay. *IEEE Transactions on Vehicular Technology*, 59(1):182–190, Jan. 2010.

[38] M. Leng and Y-C. Wu. Distributed clock synchronization for wireless sensor networks using belief propagation. *IEEE Transactions on Signal Processing*, 59(11):5404–5414, Nov. 2011.

[39] M. Leng and Y-C. Wu. Low complexity maximum likelihood estimators for clock synchronization of wireless sensor nodes under exponential delays. *IEEE Transactions on Signal Processing*, 59(10):4860–4870, Oct. 2011.

[40] 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*, pages 39–49, 2004.

[41] G.J. McLachlan and T. Krishnan. *The EM Algorithm and Extensions*. John Wiley & Sons, New York, NY, 1997.

[42] H. Messer. The Hybrid Cramér-Rao bound - From practice to theory. In *IEEE workshop on Sensors, Array and Multichannel Signal Process.*, 2006.

[43] S.B. Moon, P. Skelly, and D. Towsley. Estimation and removal of clock skew from network delay measurements, vol.1. In *INFOCOM '99. Eighteenth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE*, volume 1, pages 227–234, Mar. 1999.

[44] M. Nicoli, S. Gezici, Z. Sahinoglu, and W. Wemeersch. Localization in mobile wireless and sensor networks. *EURASIP Journal on Wireless Communications and Networking*, Dec. 2011.

[45] K-L. Noh, Q.M. Chaudhari, E. Serpedin, and B.W. Suter. Novel clock phase offset and skew estimation using two-way timing message exchanges for wireless sensor networks. *IEEE Transactions on Communications*, 55(4):766 –777, Apr. 2007.

[46] R. Pagliari and A. Scaliogne. Scalable network synchronization with pulse-coupled oscillators. *IEEE Transactions on Mobile Computing*, 10(3):392–405, Mar. 2005.

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

[48] V. Paxson. On calibrating measurements of packet transit times. In *7th ACM Sigmetrics Conf.*, volume 26, pages 11–21, June 1998.

[49] R. Poovendran, C. Wang, and S. Roy. *Secure Localization and Time Synchronization for Wireless Sensor and ad hoc networks*. New York; Springer, 2006.

[50] G. J. Pottie, W. J. Kaiser, M. Giona, and S. Barbarossa. Wireless integrated network sensors. *Commun., ACM*, 43(5):51–58, May 2000.

[51] H. E. Rauch, F. Tung, and C. T. Striebel. Maximum likelihood estimates of linear dynamic systems. *AIAA Journal*, 3(8):1445–1450, Aug. 1965.

[52] Y. Rockah and P. M. Schultheiss. Array shape calibration with sources in unknown locations, part I: Far field sources. *IEEE Transactions on Acoustics, Speech and Signal Processing*, 35(3):286–299, Mar. 1987.

[53] B.M. Sadler and A. Swami. Synchronization in sensor networks: an overview. In *IEEE Military Communications Conference (MILCOM)*, pages 1–6, Oct. 2006.

[54] A. H. Sayed, A. Tarighat, and N. Khajehnouri. Network-based wireless location: Challenges faced in developing techniques for accurate wireless location information. *IEEE Signal Processing Mag.*, 22(4):24–40, Jul. 2005.

[55] O. Simeone and U. Spagnolini. Distributed time synchronization in wireless sensor networks with coupled discrete-time oscillators. *EURASIP Journal on Wireless Communications and Networking*, 2007.

[56] I. Skog and P. Handel. Synchronization by two-way message exchanges: Cramér-Rao bounds, approximate maximum likelihood, and offshore submarine positioning. *IEEE Transactions on Signal Processing*, 58(4):2351–2362, Apr. 2010.

[57] R. Solis, V.S. Borkar, and P.R. Kumar. A new distributed time synchronization protocol for multihop wireless networks. In *45th IEEE Conference on Decision and Control*, pages 2734–2739, Dec. 2006.

[58] B. Sundararaman, U. Buy, and A. D. Kshemkalyani. Clock synchronization for wireless sensor networks: a survey. *Ad Hoc Networks*, 3(3):281–323, 2005.

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

[60] H.L. Van Trees and K.L. Bell. *Bayesian Bounds for Parameter Estimation and Nonlinear Filtering/Tracking*. John Wiley & Sons, New York, NY, 2007.

[61] M.J. Wainwright, T.S. Jaakkola, and A.S. Willsky. A new class of upper bounds on the log partition function. *IEEE Transactions on Information Theory*, 51(7):2313–2335, Jul. 2005.

[62] Y. Wang, X. Ma, and G. Leus. Robust time-based localization for asynchronous networks. *IEEE Transactions on Signal Processing*, 59(9):4397–4410, Sep. 2011.

[63] G. Xiong and S. Kishore. Analysis of distributed consensus time synchronization with gaussian delay over wireless sensor networks. *EURASIP Journal on Wireless Communication and Networking*, 2009.

[64] D. Zennaro, A. Ahmad, E. Serpedin, L. Vangelista, and H. Nounou. Network-wide clock synchronization via message passing with exponentially distributed link delays. *submitted to IEEE Transactions on Communications*.

[65] D. Zennaro, E. Dall'Anese, T. Erseghe, and L. Vangelista. Fast clock synchronization in wireless sensor networks via admm-based consensus. In *2011 International Symposium on Modeling and Optimization in Mobile, Ad Hoc and Wireless Networks (WiOpt)*, pages 148–153, May 2011.

[66] D. Zennaro, B. Tomasi, L. Vangelista, and M. Zorzi. Light-sync: a low overhead synchronization algorithm for underwater acoustic networks. In *accepted at IEEE/MTS OCEANS*, May 2012.

[67] J. Zheng and Y-C. Wu. Joint time synchronization and localization of an unknown node in wireless sensor networks. *IEEE Transactions on Signal Processing*, 58(3):1309–1320, Mar. 2010.

[68] S. Zhu and Z. Ding. Joint synchronization and localization using TOAs: A linearization based approach. *IEEE J. Sel. Areas Commun.*, 28(7):1017–1025, Sep. 2010.

APPENDIX A

PROOF OF LEMMA 1

We first show by induction that $A_k < -1/2\sigma^2 \ \forall k$. It is clear from (2.37) that $A_N < -1/2\sigma^2$. Suppose that $A_{k+1} < -1/2\sigma^2$. Using (2.42), we have

$$A_k = -\frac{1}{2\sigma^2} + \gamma_k + B_{k+1} - \frac{C_{k+1}^2}{4A_{k+1}} \ .$$

Using the values of constants B_{k+1} and C_{k+1} and after some algebraic steps, we have

$$\begin{aligned} A_k &= \gamma_k - \frac{1}{\sigma^2}\left(1 + \frac{1}{4\sigma^2 A_{k+1}}\right) \\ &< \gamma_k - \frac{1}{\sigma^2}\left(1 + \frac{1}{4\sigma^2}\cdot\left(-2\sigma^2\right)\right) = \gamma_k - \frac{1}{2\sigma^2} \\ &< -\frac{1}{2\sigma^2} \ . \end{aligned}$$

The proof of the lemma follows by noting that $\frac{-C_k}{2A_k} > 0$ which implies that $g_k(.)$ is a monotonically increasing function.

APPENDIX B

MSE EXPRESSIONS FOR ML ESTIMATORS

Gaussian Distribution

If the likelihood for ξ is Gaussian distributed, the MLE is given by (2.19). Since the variance of the readings U_j is σ_ξ^2 and the MLE (2.19) is unbiased, it can be concluded from (2.16) and Proposition 1 that

$$\text{MSE}\left(\hat{\beta}_{\text{ML}}\right) = \frac{\sigma_\xi^2 + \sigma_\psi^2}{4N} \; . \tag{B.1}$$

Exponential Distribution

If the likelihood for ξ is exponentially distributed, the MLE is given by (2.29). It can be seen that $U_{(1)}$ is exponentially distributed with parameter $\lambda_\xi^{'} = \lambda_\xi N$, so that $\text{Var}\left(\hat{\xi}_{\text{ML}}\right) = 1/\lambda_\xi^2 N^2$. The bias of the estimator $\hat{\xi}_{\text{ML}}$ is $b_{\xi,\text{ML}} \stackrel{\Delta}{=} 1/\lambda_\xi N$. Similarly, $\text{Var}\left(\hat{\psi}_{\text{ML}}\right) = 1/\lambda_\psi^2 N^2$ and $b_{\psi,\text{ML}} \stackrel{\Delta}{=} 1/\lambda_\psi N$. Therefore, given (2.16) and Proposition 1, it can be concluded that

$$\text{MSE}\left(\hat{\beta}_{\text{ML}}\right) = \frac{0.25}{N^2}\left[\left(\frac{1}{\lambda_\xi^2} + \frac{1}{\lambda_\psi^2}\right) + \left(\frac{1}{\lambda_\xi} - \frac{1}{\lambda_\psi}\right)^2\right] \; . \tag{B.2}$$

APPENDIX C

PAIRWISE BROADCAST SYNCHRONIZATION*

In this appendix, we show how the results in Section 2 can be easily extended to the case when an inactive node synchronizes itself by over hearing the message exchanges taking place between two active nodes. In particular, we provide a much simpler analytical proof of the ML estimator proposed in [14]. Our proof utilizes convex optimization techniques and constitutes an alternative to the complex graphical analysis employed in [14] to maximize the likelihood function [2]. In addition, we provide a Bayesian solution to the clock offset estimation problem for inactive node synchronization by utilizing the factor graph approach discussed in Section 2.

C.1 A Simple Alternative Proof of Maximum Likelihood Estimation

The message exchange process for the inactive node synchronization is depicted in Fig. C.1. The mathematical system model describing this message exchange can be expressed as [14]

$$\begin{aligned} U_j &\triangleq T_j^2 - T_j^1 = d + \beta_r + X_j \\ V_j &\triangleq T_j^3 - T_j^1 = d + \beta_o + Y_j \\ W_j &\triangleq T_j^5 - T_j^4 = d + \beta_o - \beta_r + Z_j \ , \end{aligned} \tag{C.1}$$

where β_r and β_o denote the clock offsets of nodes r and o, respectively. The network delays X_j, Y_j and Z_j are considered *i.i.d.* exponentially distributed with common

*Part of this section is reprinted with permission from "Joint clock offset and skew estimation for inactive nodes in wireless sensor networks," Aitzaz Ahmad, Amina Noor, and Erchin Serpedin, 2011, 45th Annual Conference on Information Systems and Sciences (CISS), Copyright 2011 by IEEE.

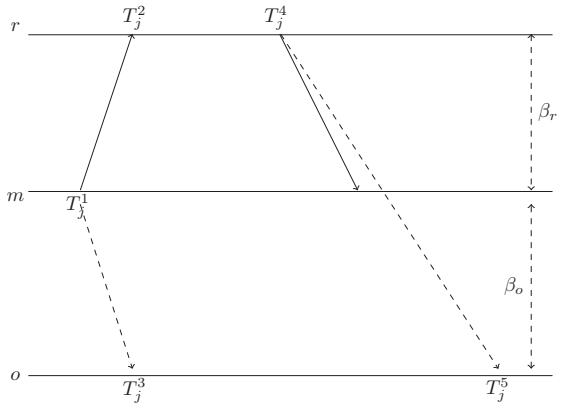

Figure C.1: Message exchange mechanism for inactive node synchronization

mean λ. The likelihood function can be expressed as

$$L(d,\beta_r,\beta_o) = \lambda^{-3N}\exp\left(-\frac{1}{\lambda}\sum_{j=1}^{N} U_j + V_j + W_j - 3d - 2\beta_o\right) \cdot \mathbb{I}\left(U_{(1)} - d - \beta_r\right)\mathbb{I}\left(V_{(1)} - d - \beta_o\right)\mathbb{I}\left(W_{(1)} - d - \beta_o + \beta_r\right) \ . \quad \text{(C.2)}$$

After using the maximizing value of λ in (C.2), the reduced likelihood function can be expressed as [14]

$$\begin{aligned} L' &= e^{-3N}\left(\sum_{j=1}^{N} U_j + V_j + W_j - 3d - 2\beta_o\right)^{-3N} \\ &= \cdot\mathbb{I}\left(U_{(1)} - d - \beta_r\right)\mathbb{I}\left(V_{(1)} - d - \beta_o\right)\mathbb{I}\left(W_{(1)} - d - \beta_o + \beta_r\right) \ . \end{aligned} \quad \text{(C.3)}$$

Instead of solving for the ML estimates using a graphical search of the likelihood function as done in [14], we show a simple alternative proof using convex optimization and KKT conditions.

The ML estimation problem can be equivalently represented as an instance of the

convex optimization problem

$$\left(\hat{d}, \hat{\beta}_r, \hat{\beta}_o\right) = \min_{d,\beta_r,\beta_o} \sum_{j=1}^{N} \left(U_j + V_j + W_j - 3d - 2\beta_o\right)$$
$$\text{such that} \quad U_{(1)} - d - \beta_r \geq 0, \quad V_{(1)} - d - \beta_o \geq 0, \quad W_{(1)} - d - \beta_o + \beta_r \geq 0 \, . \tag{C.4}$$

It can be verified that the optimization problem in (C.4) represents a standard linear program (LP). The Lagrangian for (C.4) can be expressed as

$$\mathcal{L} = \sum_{j=1}^{N} \left(U_j + V_j + W_j - 3d - 2\beta_o\right) + \mu_1 \left(d + \beta_r - U_{(1)}\right) + \mu_2 \left(d + \beta_o - V_{(1)}\right) + \mu_3 \left(d + \beta_o - \beta_r - W_{(1)}\right) \, , \tag{C.5}$$

where μ_1, μ_2 and μ_3 are the non-negative Lagrange multipliers. The corresponding KKT conditions can be written as [12]

$$\begin{aligned} -3N + \mu_1^* + \mu_2^* + \mu_3^* &= 0 \\ \mu_1^* - \mu_3^* &= 0 \\ -2N + \mu_2^* + \mu_3^* &= 0 \\ \mu_1^* \left(\hat{d} + \hat{\beta}_r - U_{(1)}\right) = 0, \quad \mu_2^* \left(\hat{d} + \hat{\beta}_o - V_{(1)}\right) &= 0, \quad \mu_3^* \left(\hat{d} + \hat{\beta}_o - \hat{\beta}_r - W_{(1)}\right) = 0 \, . \end{aligned}$$

A solution of the above system of equations yields $\mu_1^* = \mu_2^* = \mu_3^* = N$. This implies that each of the constraints become active. Solving the system of equations presented

by the constraints, the ML estimate of the clock offsets is given by

$$\begin{bmatrix} \hat{d} \\ \hat{\beta}_r \\ \hat{\beta}_o \end{bmatrix} = \begin{bmatrix} U_{(1)} + W_{(1)} - V_{(1)} \\ V_{(1)} - W_{(1)} \\ 2V_{(1)} - U_{(1)} - W_{(1)} \end{bmatrix} \tag{C.6}$$

which coincides with the one reported in [14]. The solution in (C.6), therefore, represents an analytical method to arrive at the ML estimates, bypassing the graphical arguments in [14].

C.2 Bayesian Estimation

The factor graph approach developed for the pairwise synchronization problem in Section 2 can also be extended to the case of inactive node synchronization.

Define

$$\xi = d + \beta_r, \quad \psi = d + \beta_o, \quad \zeta = d + \beta_o - \beta_r \ . \tag{C.7}$$

By considering ξ_j, ψ_j, and ζ_j as Gauss-Markov random processes at the j^{th} message exchange, the system model in (C.1) can be represented as

$$U_j = \xi_j + X_j, \quad V_j = \psi_j + Y_j, \quad W_j = \zeta_j + Z_j \ . \tag{C.8}$$

The posterior pdf can be expressed as

$$\begin{aligned} &f(\boldsymbol{\xi}, \boldsymbol{\psi}, \boldsymbol{\zeta} | \boldsymbol{U}, \boldsymbol{V}, \boldsymbol{W}) \propto f(\boldsymbol{\xi}, \boldsymbol{\psi}) f(\boldsymbol{U}, \boldsymbol{V}, \boldsymbol{W} | \boldsymbol{\xi}, \boldsymbol{\psi}, \boldsymbol{\zeta}) \\ &= f(\xi_0) f(\psi_0) f(\zeta_0) \prod_{j=1}^{N} f(\xi_j | \xi_{j-1}) f(\psi_j | \psi_{j-1}) f(\zeta_j | \zeta_{j-1}) f(U_j | \xi_j) f(V_j | \psi_j) f(W_j | \zeta_j) \ . \end{aligned} \tag{C.9}$$

1. Notice the close resemblance between (2.33) and (C.9). This implies that the

factor graph in this case will have precisely the same structure, just a redefinition of ξ_j, ψ_j and ζ_j.

2. Our message updates based on the max-product algorithm will yield an exact solution for the case of inactive node synchronization as well.

Therefore, by proceeding exactly as in Section 2.5, the exact solution for the Bayesian estimate of the clock parameters is given by

$$\begin{aligned}
\hat{\xi}_N &= \min\left(U_N, G_N^N\left(U_{N-1}\right), \ldots, G_2^N\left(U_1\right), G_1^N\left(\hat{\xi}_0\right)\right) \\
\hat{\psi}_N &= \min\left(V_N, \tilde{G}_N^N\left(V_{N-1}\right), \ldots, \tilde{G}_2^N\left(V_1\right), \tilde{G}_1^N\left(\hat{\psi}_0\right)\right) \\
\hat{\zeta}_N &= \min\left(W_N, \bar{G}_N^N\left(W_{N-1}\right), \ldots, \bar{G}_2^N\left(W_1\right), \bar{G}_1^N\left(\hat{\zeta}_0\right)\right) \ . \qquad \text{(C.10)}
\end{aligned}$$

The corresponding estimates of $\hat{\beta}_{r_N}$ and $\hat{\beta}_{o_N}$ can determined from (C.10) using the transformation in (C.7).

APPENDIX D

PROOF OF LEMMA 3

To derive the solution of (3.18), assume first that $r_{i,\ell}^{(t)}$ is an even number. Then there exists an entire interval $[\xi_1, \xi_2]$ for β_i in which the sum of the absolute values is constant with respect to β_i and equal to its minimum. This can be verified by noticing that since there is an even number of absolute value terms, the first derivative (wherever it exists) is negative for $\beta_i < \xi_1$, positive for $\beta_i > \xi_2$, and zero in (ξ_1, ξ_2). Now, recalling that the $\{W_{i\ell}^{(t)}(n)\}$ are sorted in an increasing order and assuming $\beta_\ell - S_{i\ell} \leq W_{i\ell}^{(t)}\left(r_{i,\ell}^{(t)}/2\right)$ (see Fig. D.1 (a)), it turns out that $\xi_1 = \max\{\beta_\ell - S_{i\ell}, W_{i\ell}^{(t)}\left(r_{i,\ell}^{(t)}/2 - 1\right)\}$ and $\xi_2 = W_{i\ell}^{(t)}\left(r_{i,\ell}^{(t)}/2\right)$. On the other hand, if $\beta_\ell - S_{i\ell} > W_{i\ell}^{(t)}\left(r_{i,\ell}^{(t)}/2\right)$, then $\xi_1 = W_{i\ell}^{(t)}\left(r_{i,\ell}^{(t)}/2\right)$ and $\xi_2 = \min\{\beta_\ell - S_{i\ell}, W_{i\ell}^{(t)}\left(r_{i,\ell}^{(t)}/2 + 1\right)\}$. In other words, $W_{i\ell}^{(t)}\left(r_{i,\ell}^{(t)}/2\right)$ is always one of the two extremes of the zero-slope interval $[\xi_1, \xi_2]$. Substituting $\beta_i = W_{i\ell}^{(t)}\left(r_{i,\ell}^{(t)}/2\right)$ in (3.18) leads to

$$m_{h_{i\ell}\to\beta_\ell}^{(t)}(\beta_\ell) \propto \exp\left(-2K\lambda\left|\beta_\ell - S_{i\ell} - W_{i\ell}^{(t)}\left(\frac{r_{i,\ell}^{(t)}}{2}\right)\right|\right) .$$

The case in which $r_{i,\ell}^{(t)}$ is odd is more involved. In fact, since there is an odd number of absolute value terms, there is just a point $\bar{\xi}$ in which the function in (3.18) reaches its maximum. To determine such a point, first assume $\beta_\ell - S_{i\ell} < W_{i\ell}^{(t)}\left(\left(r_{i,\ell}^{(t)} - 1\right)/2\right)$, then it is easy to verify that $\bar{\xi} = W_{i\ell}^{(t)}\left(\left(r_{i,\ell}^{(t)} - 1\right)/2\right)$. On the other hand, if $\beta_\ell - S_{i\ell} > W_{i\ell}^{(t)}\left(\left(r_{i,\ell}^{(t)} + 1\right)/2\right)$, then $\bar{\xi} = W_{i\ell}^{(t)}\left(\left(r_{i,\ell}^{(t)} + 1\right)/2\right)$. Finally, if $\beta_\ell - S_{i\ell} \in [W_{i\ell}^{(t)}\left(\left(r_{i,\ell}^{(t)} - 1\right)/2\right), W_{i\ell}^{(t)}\left(\left(r_{i,\ell}^{(t)} + 1\right)/2\right)]$, then $\bar{\xi} = \beta_\ell - S_{i\ell}$ (see Fig. D.1 (b)).

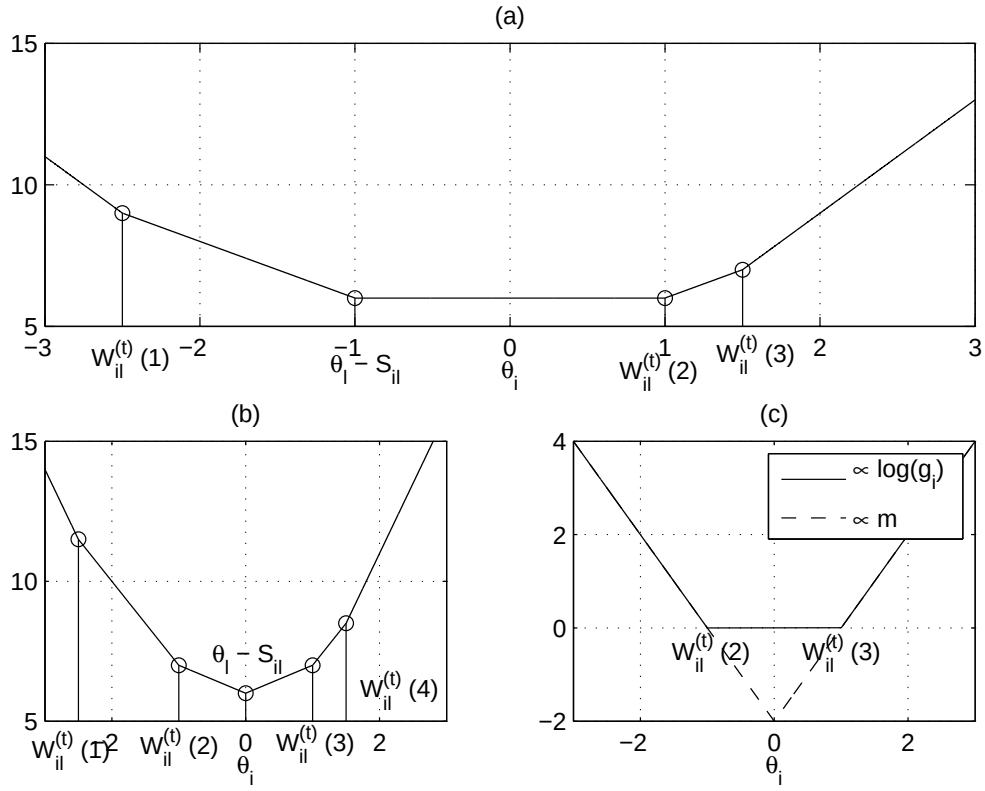

Figure D.1: (a) Function proportional to the exponent of (3.18) with $r_{i,\ell}^{(t)} = 4$ and $\beta_\ell - S_{i\ell} \leq W_{i\ell}^{(t)}(2)$. (b) Function proportional to the exponent of (3.18) with $r_{i,\ell}^{(t)} = 5$ and $\beta_\ell - S_{i\ell} \in [W_{i\ell}^{(t)}(2), W_{i\ell}^{(t)}(3)]$. (c) Approximation and real value of the exponent of $g_i^{(t)}(\beta_\ell)$ with $r_{i,\ell}^{(t)} = 5$.

By summarizing, we have

$$m_{h_{i\ell}\to\beta_\ell}^{(t)}(\beta_\ell) \propto g_i^{(t)}(\beta_\ell) =$$

$$\begin{cases} \exp\left(-2K\lambda\left|\beta_\ell - S_{i\ell} - W_{i\ell}^{(t)}\left(\frac{r_{i,\ell}^{(t)}-1}{2}\right)\right|\right), & \beta_\ell < S_{i\ell} + W_{i\ell}^{(t)}\left(\frac{r_{i,\ell}^{(t)}-1}{2}\right) \\ 1, & W_{i\ell}^{(t)}\left(\frac{r_{i,\ell}^{(t)}-1}{2}\right) + S_{i\ell} \leq \beta_\ell \leq W_{i\ell}^{(t)}\left(\frac{r_{i,\ell}^{(t)}+1}{2}\right) + S_{i\ell} \\ \exp\left(-2K\lambda\left|\beta_\ell - S_{i\ell} - W_{i\ell}^{(t)}\left(\frac{r_{i,\ell}^{(t)}+1}{2}\right)\right|\right), & \beta_\ell > S_{i\ell} + W_{i\ell}^{(t)}\left(\frac{r_{i,\ell}^{(t)}+1}{2}\right) \end{cases}$$

The function $g_\ell^{(t)}(\beta_\ell)$ cannot be written as a Laplace distribution. In order to do so, an approximation is mandatory. The interval in which $g_\ell^{(t)}(\beta_\ell)$ is constant is ignored and the function is extended by continuity in this region (see Fig. D.1 (c)). Mathematically,

$$m_{h_{i\ell}\to\beta_\ell}^{(t)}(\beta_\ell) \propto \exp\left(-2K\lambda \cdot \left| \beta_\ell - S_{i\ell} - \frac{1}{2}\left[W_{i\ell}^{(t)}\left(\frac{r_{i,\ell}^{(t)}-1}{2}\right) + W_{i\ell}^{(t)}\left(\frac{r_{i,\ell}^{(t)}+1}{2}\right)\right]\right|\right).$$

This concludes the proof.

Remark 4. *It is clear that when $r_{i,\ell}^{(t)}$ is even, application of Lemma 3 gives a precise result. The approximation is applied when $r_{i,\ell}^{(t)}$ is odd. In this case, the accuracy of the approximation will improve as the value of $r_{i,\ell}^{(t)}$ increases (cf. Fig. D.1). Hence, the approximation is expected to be better in dense networks.*

APPENDIX E

PROOF OF THEOREM 2

E.1 Computation of $\mathbb{E}_{\boldsymbol{\Theta}|\boldsymbol{x}}\left[\left(\frac{\partial \ln f(\boldsymbol{\Theta}|\boldsymbol{x})}{\partial \boldsymbol{\xi}}\right)\left(\frac{\partial \ln f(\boldsymbol{\Theta}|\boldsymbol{x})}{\partial \boldsymbol{\xi}}\right)^T\right]$.

We first compute the second term in (4.36). Since the parameters $\boldsymbol{\Theta}$ and $\boldsymbol{x}$ are statistically independent, it follows that $f(\boldsymbol{\Theta}|\boldsymbol{x}) = f(\boldsymbol{\Theta})$. By using the fact that $\frac{\partial f(\boldsymbol{\Theta})}{\partial \boldsymbol{x}} = \mathbf{0}$, we can write

$$\mathbb{E}_{\boldsymbol{\Theta}|\boldsymbol{x}}\left[\left(\frac{\partial \ln f(\boldsymbol{\Theta}|\boldsymbol{x})}{\partial \boldsymbol{\xi}}\right)\left(\frac{\partial \ln f(\boldsymbol{\Theta}|\boldsymbol{x})}{\partial \boldsymbol{\xi}}\right)^T\right] = \begin{bmatrix} \mathbb{E}_{\boldsymbol{\Theta}}\left[\left(\frac{\partial \ln f(\boldsymbol{\Theta})}{\partial \boldsymbol{\Theta}}\right)\left(\frac{\partial \ln f(\boldsymbol{\Theta})}{\partial \boldsymbol{\Theta}}\right)^T\right] & \mathbf{0}_{2M\times 2} \\ \mathbf{0}_{2\times 2M} & \mathbf{0}_{2\times 2} \end{bmatrix} \tag{E.1}$$

Using (4.8) and (4.9),

$$\ln f(\boldsymbol{\Theta}) = \ln f(\boldsymbol{\theta}_0) + \sum_{k=1}^{K} \ln f(\boldsymbol{\theta}_k|\boldsymbol{\theta}_{k-1})$$

and it follows that

$$\begin{aligned} \frac{\partial \ln f(\boldsymbol{\Theta})}{\partial \boldsymbol{\theta}_k} &= \frac{\boldsymbol{n}_{k+1} - \boldsymbol{n}_k}{\sigma_n^2}\ ,\ k = 1, \ldots, K-1 \\ \frac{\partial \ln f(\boldsymbol{\Theta})}{\partial \boldsymbol{\theta}_K} &= \frac{-\boldsymbol{n}_K}{\sigma_n^2}\ . \end{aligned}$$

After some algebraic calculations, we have

$$\mathbb{E}_{\boldsymbol{\Theta}}\left[\left(\frac{\partial \ln f(\boldsymbol{\Theta})}{\partial \boldsymbol{\Theta}}\right)\left(\frac{\partial \ln f(\boldsymbol{\Theta})}{\partial \boldsymbol{\Theta}}\right)^T\right] = \boldsymbol{\Upsilon}\ , \tag{E.2}$$

where Υ is defined in (4.40).

E.2 Computation of $\mathbb{E}_{\boldsymbol{\Theta}}\left[\boldsymbol{F}(\boldsymbol{\Theta}, \boldsymbol{x})\right]$

The FIM can be partitioned as

$$\boldsymbol{F}(\boldsymbol{\Theta}, \boldsymbol{x}) = \begin{bmatrix} \boldsymbol{F}_{11} & \boldsymbol{F}_{12} \\ \boldsymbol{F}_{21} & \boldsymbol{F}_{22} \end{bmatrix} ,$$

where

$$\begin{aligned}
\boldsymbol{F}_{11} &= \mathbb{E}_{\boldsymbol{y}|\boldsymbol{\Theta},\boldsymbol{x}} \left[\left(\frac{\partial \ln f\left(\boldsymbol{y}|\boldsymbol{\Theta}, \boldsymbol{x}\right)}{\partial \boldsymbol{\Theta}} \right) \left(\frac{\partial \ln f\left(\boldsymbol{y}|\boldsymbol{\Theta}, \boldsymbol{x}\right)}{\partial \boldsymbol{\Theta}} \right)^T \right] \\
\boldsymbol{F}_{12} &= \mathbb{E}_{\boldsymbol{y}|\boldsymbol{\Theta},\boldsymbol{x}} \left[\left(\frac{\partial \ln f\left(\boldsymbol{y}|\boldsymbol{\Theta}, \boldsymbol{x}\right)}{\partial \boldsymbol{\Theta}} \right) \left(\frac{\partial \ln f\left(\boldsymbol{y}|\boldsymbol{\Theta}, \boldsymbol{x}\right)}{\partial \boldsymbol{x}} \right)^T \right] \\
\boldsymbol{F}_{21} &= \mathbb{E}_{\boldsymbol{y}|\boldsymbol{\Theta},\boldsymbol{x}} \left[\left(\frac{\partial \ln f\left(\boldsymbol{y}|\boldsymbol{\Theta}, \boldsymbol{x}\right)}{\partial \boldsymbol{x}} \right) \left(\frac{\partial \ln f\left(\boldsymbol{y}|\boldsymbol{\Theta}, \boldsymbol{x}\right)}{\partial \boldsymbol{\Theta}} \right)^T \right] \\
\boldsymbol{F}_{22} &= \mathbb{E}_{\boldsymbol{y}|\boldsymbol{\Theta},\boldsymbol{x}} \left[\left(\frac{\partial \ln f\left(\boldsymbol{y}|\boldsymbol{\Theta}, \boldsymbol{x}\right)}{\partial \boldsymbol{x}} \right) \left(\frac{\partial \ln f\left(\boldsymbol{y}|\boldsymbol{\Theta}, \boldsymbol{x}\right)}{\partial \boldsymbol{x}} \right)^T \right] .
\end{aligned} \tag{E.3}$$

E.2.0.1 Sub-matrix $\mathbb{E}_{\boldsymbol{\Theta}}\left(\boldsymbol{F}_{11}\right)$

The sub-matrix $\boldsymbol{F}_{11}$ can be computed more readily by using the alternative expression

$$\boldsymbol{F}_{11} = \mathbb{E}_{\boldsymbol{y}|\boldsymbol{\Theta},\boldsymbol{x}} \left[-\frac{\partial^2 \ln f\left(\boldsymbol{y}|\boldsymbol{\Theta}, \boldsymbol{x}\right)}{\partial \boldsymbol{\Theta} \partial \boldsymbol{\Theta}^T} \right] . \tag{E.4}$$

It follows from the *i.i.d* nature of the network delays that

$$\frac{\partial^2 \ln f\left(\boldsymbol{y}|\boldsymbol{\Theta}, \boldsymbol{x}\right)}{\partial \boldsymbol{\Theta} \partial \boldsymbol{\Theta}^T} = \sum_{k=1}^{K} \frac{\partial^2 \ln f\left(\boldsymbol{y_k}|\boldsymbol{\theta_k}, \boldsymbol{x}\right)}{\partial \boldsymbol{\Theta} \partial \boldsymbol{\Theta}^T} , \tag{E.5}$$

where using (4.10)

$$\ln f\left(\boldsymbol{y_k}|\boldsymbol{\theta_k},\boldsymbol{x}\right)\propto -\frac{\left(\boldsymbol{y}_k-\boldsymbol{d}\left(\boldsymbol{x}\right)-\boldsymbol{H}_k\boldsymbol{\theta}_k\right)^T\left(\boldsymbol{y}_k-\boldsymbol{d}\left(\boldsymbol{x}\right)-\boldsymbol{H}_k\boldsymbol{\theta}_k\right)}{2\sigma_w^2}$$

and

$$\boldsymbol{f}_k' \triangleq \frac{\ln f\left(\boldsymbol{y_k}|\boldsymbol{\theta_k},\boldsymbol{x}\right)}{\partial\boldsymbol{\theta}_k} = \frac{\boldsymbol{H}_k^T\left(\boldsymbol{y}_k-\boldsymbol{d}\left(x\right)\right)-\boldsymbol{H}_k^T\boldsymbol{H}\boldsymbol{\theta}_k}{\sigma_w^2}\ . \tag{E.6}$$

This implies that

$$\begin{aligned}\frac{\ln f\left(\boldsymbol{y_k}|\boldsymbol{\theta_k},\boldsymbol{x}\right)}{\partial\boldsymbol{\Theta}} &= [\boldsymbol{0}_{2\times1}^T,\ldots,\boldsymbol{f}_k^{T'},\ldots,\boldsymbol{0}_{2\times1}^T]^T\\ \frac{\partial^2\ln f\left(\boldsymbol{y}_k|\boldsymbol{\theta}_k,\boldsymbol{x}\right)}{\partial\boldsymbol{\Theta}\partial\boldsymbol{\Theta}^T} &= \text{blkdiag}\left(\boldsymbol{0}_{2\times2},\ldots,\frac{\boldsymbol{H}_k^T\boldsymbol{H}_k}{\sigma_w^2},\ldots,\boldsymbol{0}_{2\times2}\right)\ .\end{aligned}$$

Collecting all terms from (E.5) followed by the expectation in (E.4), it follows that

$$\mathbb{E}_{\boldsymbol{\Theta}}\left[\boldsymbol{F}_{11}\right] = \text{blkdiag}\left(\frac{\boldsymbol{H}_1^T\boldsymbol{H}_1}{\sigma_w^2},\ldots,\frac{\boldsymbol{H}_K^T\boldsymbol{H}_K}{\sigma_w^2}\right)\ . \tag{E.7}$$

E.2.0.2 Sub-matrix $\mathbb{E}_{\boldsymbol{\Theta}}\left(\boldsymbol{F}_{22}\right)$

Using the fact that

$$\frac{\partial\left\|\boldsymbol{x}-\boldsymbol{s}_j\right\|}{\partial\boldsymbol{x}} = \frac{\boldsymbol{x}-\boldsymbol{s}_j}{\left\|\boldsymbol{x}-\boldsymbol{s}_j\right\|} \tag{E.8}$$

we can write

$$\boldsymbol{d}'\left(\boldsymbol{x}\right) \triangleq \frac{\partial\boldsymbol{d}\left(\boldsymbol{x}\right)}{\partial\boldsymbol{x}} = \left[\frac{\boldsymbol{x}-\boldsymbol{s}_1}{\left\|\boldsymbol{x}-\boldsymbol{s}_1\right\|},\frac{\boldsymbol{x}-\boldsymbol{s}_1}{\left\|\boldsymbol{x}-\boldsymbol{s}_1\right\|},\ldots,\frac{\boldsymbol{x}-\boldsymbol{s}_N}{\left\|\boldsymbol{x}-\boldsymbol{s}_N\right\|},\frac{\boldsymbol{x}-\boldsymbol{s}_N}{\left\|\boldsymbol{x}-\boldsymbol{s}_N\right\|}\right]^T\ . \tag{E.9}$$

By defining

$$\boldsymbol{d}'_k \triangleq \frac{\ln f\left(\boldsymbol{y_k}|\boldsymbol{\theta_k}, \boldsymbol{x}\right)}{\partial \boldsymbol{x}} = \frac{\boldsymbol{d}'^T\left(\boldsymbol{x}\right)\left(\boldsymbol{y}_k - \boldsymbol{d}\left(\boldsymbol{x}\right) - \boldsymbol{H}_k\boldsymbol{\theta}_k\right)}{\sigma_w^2}\ , \tag{E.10}$$

the matrix $\boldsymbol{F}_{22}$ in (E.3) is given by

$$\boldsymbol{F}_{22} = \sum_{k=1}^{K} \mathbb{E}_{\boldsymbol{y}|\boldsymbol{\Theta},\boldsymbol{x}}\left[\frac{\boldsymbol{d}'_k\boldsymbol{d}'^T_k}{\sigma_w^4}\right] = \frac{2M}{\sigma_w^2}\sum_{j=1}^{N}\boldsymbol{d}'^T\left(\boldsymbol{x}\right)\boldsymbol{d}'\left(\boldsymbol{x}\right) \tag{E.11}$$

which after some simplification yields (4.39).

E.2.0.3 The sub-matrix $\mathbb{E}_{\boldsymbol{\Theta}}\left[\boldsymbol{F}_{12}\right]$

The sub-matrix $\boldsymbol{F}_{12}$ in (E.3) can be computed by using (E.6) and (E.10). After matrix multiplication and taking expectation with respect to $\mathbb{E}_{\boldsymbol{y},\boldsymbol{\Theta}}$, $\mathbb{E}_{\boldsymbol{\Theta}}\left[\boldsymbol{F}_{12}\right]$ can be expressed as

$$\mathbb{E}_{\boldsymbol{\Theta}}\left[\boldsymbol{F}_{12}\right] = \left[\left(\frac{\boldsymbol{H}_1^T\boldsymbol{d}'\left(\boldsymbol{x}\right)}{\sigma_w^2}\right)^T, \ldots, \left(\frac{\boldsymbol{H}_K^T\boldsymbol{d}'\left(\boldsymbol{x}\right)}{\sigma_w^2}\right)^T\right]^T \tag{E.12}$$

$$= \mathbb{E}_{\boldsymbol{\Theta}}\left[\boldsymbol{F}_{21}^T\right]\ . \tag{E.13}$$

The mathematical details in computing (E.12) are omitted for brevity. The proof of the theorem follows from the sub-matrices calculated above.