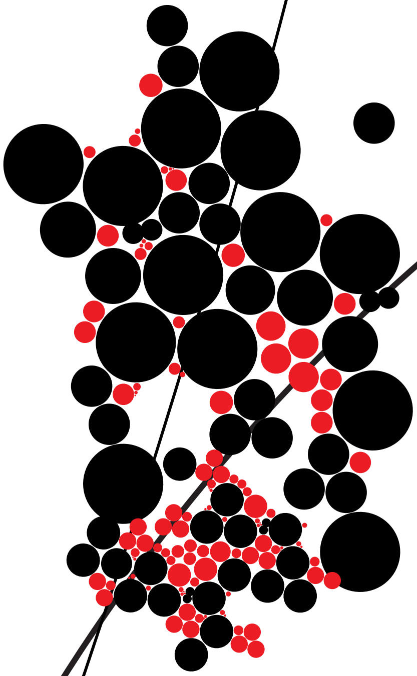


On Analysis and Design of Algorithms for Robust Estimation from Relative Measurements

Nelson P.K. Chan

30 March 2016
MSc. Systems & Control
Specialization: Control Theory
Chair: Hybrid Systems (AM)

Graduation Committee:
prof.dr. H.J. Zwart (UT)
dr. P. Frasca (UT)
dr.ir. J. Goseling (UT)

University of Twente.

Abstract

The problem of estimating the states of a group of agents from noisy pairwise difference measurements between agents' states has been studied extensively in the past. Often, the noise is modeled as a Gaussian distribution with constant variance; in other words, the measurements all have the same quality. However, in reality this is not the case and measurements can be of different quality which, for a sensible estimation, needs to be taken into account. In the current work, we assume the noise to be a mixture of Gaussian distributions.

Our contribution is two fold. First, we look at the problem of estimation. Several Maximum-likelihood type estimators are considered, based on the availability of information regarding the noise distributions We show that for networks represented by a tree, the quality of the measurements is not of importance for the estimation. Also, the WLSP yields the best performance among the approaches considered. Furthermore, the benefit of the approaches as presented in this report as opposed to the least squares approach is apparent when a graph is more connected.

Second, we consider also the problem of adding new edges with possibly unknown quality to the network with the aim to decrease the uncertainty in the estimation. It is observed that the first few edges will more likely add edges which are not close to each other for the cycle graph, or edges which have initially a low value for the degree, that is they have a few neighbors.

Keywords: Weighted Graph, Expectation Maximization, State Estimation, Link Addition.

Preface

If you take a journey because you love to reach a destination, you may not arrive. But if you love the journey, you can reach any destination. ~ *Alexander den Heijer*

The work presented in this report is the result of a journey that lasted for approximately 9 months as the final hurdle towards reaching the destination (referring to the MSc. degree). In hindsight, it has been quite a pleasant time spent on learning how to do research. Along the course of the journey, I had managed to have some *Aha!* moments but most of the time I was faced with a brick wall and the time was spent on how to overcome these brick walls.

In the following, I will spend a few paragraphs (which will cumulate to some pages) to extend my sincere gratitude to a number of people (this is rather a long list) whom have helped me throughout this journey and also the MSc. journey in its entirety.

As the report (which my hope is that you would have a look through it after reading these pages) is regarding networks, I will consider the persons I mention hereafter to be nodes (or a cluster of nodes) which are linked to me.

First, I would like to say a huge *grazie mille* to dr. Paolo Frasca, my supervisor for both the internship and the graduation project. Paolo, I remember our first connection was that of a course instructor - student connection; you being the course instructor for "Hybrid Dynamical Systems" and I the student who took it as part of my curriculum. Until now, I smile when recalling the joke you made in one of the lectures in which you told you were not supposed to talk when you were facing with your back towards the students. Our connection took a turn when I approached you in the summer of 2014 asking for recommended reading related to a MSc. topic, which happens to be estimation from relative measurements (The topic of this report). This event had led me to spending a period abroad in Padova, Italy for my internship and also doing the graduation project under your supervision. Looking back, I am grateful to you for enabling these events to occur. To me, it has been a joyful period working under (or with) you. I sometimes ask myself how it is possible that we could have meetings in which we (or I) lost track of time; needless to say, during those fruitful discussions, you have always managed to steer me to the right direction. Furthermore, I need to also thank you for your efforts in my search for a PhD position. Moreover, you have granted me the opportunity to collaborate on a paper with you, which I am also grateful for.

Next, I would like to thank prof.dr. H.J. Zwart and dr.ir. J. Goseling for putting time aside to read my report and to serve as part of the assessment committee. Thank you.

A *'Dankjewel'* can also be attributed to Mrs. Marja Langkamp, the secretary of Hybrid Systems, and Mrs. Mirande van der Kooij, from BOZ for taking care of the administration regarding the graduation, especially those long e-mails.

Mrs. Belinda Jaarsma-Knol, also a *'Dankjewel'* to you, this time for the administration that was needed for the internship, but in particular for allowing me to pass by your office and just have a talk with you regarding traveling. I will remember one of the encounters in which you told me to smile when my face 'told' you I was quite stressed.

Mrs. Lilian Spijker, I cannot forget you. You have been with me since I was, back then, a MSc. student in Applied Mathematics. You have always been listening to me, when I had problems, when I was facing difficulties, but also when I made some progress with the courses. Through your patience and your openness, I have found myself from reluctantly meeting you to actually looking forward to meeting you. You have been a source of encouragement for the past two and a half years and I am in hindsight happy you sent that e-mail to me back in October 2013, which started the link between us.

Next on the (long) list is Wilbert. Wilbert, *grazie* for the pencil sharpener (*un piccolo regalo*). It has been very useful the past couple of months and will be in the future; also *grazie* for reaching out to me and for the 'light' discussions whenever we do not feel like working (Shh).

Xinwei, *xie xie* for always listening and trying your best to answer the math-related questions that I have. Through our discussions, I have learned to appreciate the theorems and it is also thanks to your observation that the moments of a pdf is simply an integral that has led me to solve the problem related to calculating the moments of the two-sided normal tail distribution and as a consequence proposition 2.9 and proposition 2.10 can be shown, one of the *Aha!* moments!

Connie Wong & Angela Cheung, thank you for setting aside time to talk to me. Connie *jie jie*, I am indebted to you for the life experiences you have shared with me. In talking to you, I have learned to view a problem from the perspective of others and in doing so understand the motivation for their behavior. Angela, thank you for answering my calls when I felt bored and needed company. Also, thank you making the get-aways to Rotterdam enjoyable.

The following few lines are meant for the Surinamese friends here in Enschede. Ignaas, Cyrano, Annemieke, Jina, Chefiek, Roswita, Eline, Dinah, and all the other members of the Surinamese student community in Enschede, thank you for the support the past years. The year-end gathering, the meet-up in the city center, etc. and for organizing twice a birthday party for me (yup); well what can I say other than thank you.

Now I come to "The Fellowship of the Ring", the close group of friends consisting of Mohamed, Hengameh, Gisela, Armando, Abhishek, Charalambos, Carlos, Giuseppe and others. Thank you for letting me in be part of the "Fellowship". We have spent quite some time in the library finishing the courses (Yes, I know you do not want to be remembered about that period!). Also thank you guys for the cinema breaks! Of course I cannot forget the Eqyptian habibi's Shamel, Adel, and Shamer and the Nigerian friends Bobo, and Victor. Not to mention, William Lee. Thank you all!

A word of thanks also to Femi for providing me shelter for the past couple of months and more for keeping up with me. Also the life experiences shared in the living room. Zan-Bo for free coffee (at least once a day) and for discussions considering graphs.

ICF-Enschede has been a "Home away from Home". Here, I have found motivation when I was facing difficulties and I also got the opportunity to meet people from different countries. I am also grateful for being given the opportunity to be a member of the choir and also be involved in the student leadership.

I am indebted to AdeKUS, my home university in Suriname for the financial support and in particular, for not giving up on me even though the journey was far from smooth during the first six months in the Netherlands.

Mom and Dad and Sis, thank you for your unconditional love, your support and for the opportunity to let me go after my dreams. Ironically, our physical distance (you in Suriname and I in the Netherlands) has actually drawn us closer to each other.

I would also like to say thank you to all the others with whom I have made a connection in the past two and a half years. There are so many of you but due to space I have to end here. I thank G.P. van der Beek for allowing me to use his template for the coverpage.

The last person I want to thank is God without whom all these people and the experiences I have mentioned above would not have been possible. He has been faithful to me, putting the right people at the right time in my life guiding me according to his desired plan.

I started this part of the report with a quote so I like to end it also with one, this time taken from the bible;

> *For I know the plans I have for you, declares the Lord, plans to prosper you and not to harm you, plans to give you hope and a future.* (Jeremiah 29:11 NIV)

Enschede, March 2016
Nelson P.K. Chan

Contents

1. Introduction

1.1 Motivation

Wireless sensor network (WSN) is a relatively new field that has gained world-wide attention in recent years due to advances in technology and the availability of small, inexpensive, and smart sensors which leads to cost effective and easily deployable WSNs [1–3]. In Ref. [1], WSNs is considered as one of the most researched areas in the last decade. This can be justified by a quick online search in which several survey papers are being published addressing the developments within the field and also the challenges that researchers are facing, see for example the references mentioned in Ref. [1]. The building blocks of a WSN are the sensor nodes; tiny devices, which are equipped with one or more sensors, a processor, memory, a power supply, a radio, and possibly also an actuator [2]. These devices are usually spatially distributed and work collectively, hence forming a network. Due to the numerous advantages, such as, lower costs, scalability, reliability, accuracy, flexibility, and ease of deployment, that WSN offers, it is being employed in a wide range of areas. These include among others, fields as military, environmental, health care, industrial, household and in marine applications [1, 2]. As mentioned earlier, sensor nodes are being produced which are small and inexpensive. This unavoidable put resource constraints on the nodes, including a limited amount of energy, short communication range, low bandwidth, and limited processing and storage [2]. Due to the short communication range, each node can communicate only with neighbouring nodes which are within a distance apart from it. As WSN covers spatially a large area, this subset of neighbouring nodes is usually small [4]. Furthermore, a node may usually lack knowledge of certain attributes such as its position in a global reference frame or the global time. These can be attributed as a consequence of the resource constraints. They however, are allowed to obtain a relative value for the quantity of interest between themselves and the neighboring nodes which are as mentioned within a certain distance. Hence, it is desired to obtain global estimates using the set of relative measurements.

1.2 Related Work

The problem of estimation from relative measurements has been studied in for example in the papers by Barooah & Hespanha, [4–6]. Applications of this problem can be found in localization, and time syncronization. Another interesting application of the problem is that of statistical ranking, studied in Ref. [7]. In localization, using the set of relative positions between the nodes, it is desired to obtain the absolute position of them in a global frame work. In order to limit the energy usage, usually nodes are put to sleep mode when they are not use for communication. Each node possesses a local clock and by exchanging time-stamped messages with their neighbors, they are able to obtain a measurement of the clock offset between them. The quantity of interest in this is now to obtain the clock offset with respect to a global time. In the ranking problem, the relative measurements are viewed as the difference in rating given to two movies by the same user and the goal is to obtain the rating of the movies in the movie database, see Ref. [7]. Usually, the noise modelled in the measurement model is assumed to be Gaussian additive noise with a constant variance value, meaning that the measurements are all of the same quality. This is considered in Refs. [4–6, 8–11] in which the focus was on distributed algorithms for solving the estimation problem. As mentioned in Ref. [12], this assumption of constant variance, also known as homoscedasticity, is rarely observed in reality. And ignoring the heteroscedasticity, i.e., the fact that the measurement noise can have a non-constant variance (or different quality), the result can be suboptimal and inconsistent. which are unsatisfactory [13]. Also, in the case of heteroscedastic noise models, the noise distribution is a priori not given as an input for estimation and hence the algorithms described in the Refs. may obtain results which are unsatisfactory when applied to real case scenarios.

A follow-up problem related to the estimation problem regards improving the estimation by means of reducing the uncertainty in the estimation. This may be done by optimally choosing a small set of edges to add to the existing graph. The addition of edges to a base graph with the aim of maximizing the algebraic connectivity of the laplacian has been studied previously in [7, 14]. In Ref. [14], heuristic approaches are considered for the edge addition problem applied to an unweighted graph. A greedy approach is described and compared with the convex relaxation approach. Osting et al. [7] has used this approach and applied it to ranking problems in movie rating and sport scheduling. Herein, the graph considered is weighted.

In the optimal design community, maximizing the algebraic connectivity is considered as the E-optimal condition. Other criteria that may be considered are the A-optimal problem in which the negative sum of the inverse of the eigenvalues are considered, i.e., $-\sum_{i=2}^{N} \lambda_i^{-1}$ and the D-optimal which is the product of the eigenvalues, $\prod_{i=2}^{N} \lambda_i$. Note, the sum is taken starting from $i = 2$, as we are aware that the first eigenvalue of the laplacian equals zero and the graph considered is assumed to be connected. The A- and D-optimal criteria are less studied while having interesting interpretation. The D-optimal criterion can be interpreted as the number of spanning trees in a graph while the A-optimal criterion is proportional to the total effective resistance of an electric network in which the edge weights are considered to be the resistance between the vertices [7, 15].

In a recent paper [16] by Summers et al., the addition of edges is considered to optimize the network coherence, which is proportional to the A-optimality criterion. A greedy approach is also applied here as a heuristic for adding edges.

1.3 Approach

In the current study, we relax the assumption of homoscedasticity for the noise and assume it to be a binary mixture of Gaussian distributions; as a consequence we can make a distinction between measurements that are considered to be accurate or ‘good’ in quality and measurements that are ’bad’ in quality. For the estimation, these measurements are then weighted differently, putting more emphasis on the ‘good’ measurements in order to still obtain a sensible estimate. Depending on the availability of information regarding the noise distributions, several estimators based on the Maximum Likelihood Principle are derived and its performance are analysed. Apart from the estimation problem, we also look at how to optimally add new edges to the available edge set with the aim to reduce the uncertainty in the state estimates. A comparison is made between the combinatorial approach in which a set of edges is added all at once and the submodular approach in which the edges are added one at a time. In this problem we include also cases for adding nodes with unknown quality.

1.4 Contribution

The main contribution of this work are summarized; First, for the estimation problem we have derived maximum likelihood estimators and for the WLS and the WLSP estimators, we also have analytical results for obtaining their performance. We have shown that for tree type graph, the approaches considered in this work all yield the same result and the covariance matrix is hence also the same as for the WLS approach. We have performed parametric study on the circle, random graph and complete graph configuration and have shown that as the graph is more connected, the benefit of the approaches considered are apparent. For the edge addtion problem, we have shown that the initial edges are added that links nodes which are far from each other and also have a low-degree, i.e., having few neighbors. Edges may significantly decrease the uncertainty in the estimation when added optimally.

1.5 Outline

The remainder of this report is as follows: we first look at the estimation problem in part I. Therein, first the measurement model is defined in section 2.2. Hereafter, estimators based on the maximum likelihood principle are obtained in section 2.3 and 2.4. Performance analysis are carried out on the estimators when it is possible. In chapter 3, numerical results are presented showing the performance of the estimators and some conclusions are drawn. In part II, we look at the edge addition problem and derive algorithms for the combinatorial and the submodular approaches based on the available information. chapter 6 presents the numerical results obtained from the approaches and as a usecase, the estimation and edge addition problem are combined. In appendix A background material on graph theory are given. In addition, results for the Moore-Penrose pseudoinverse are considered in appendix B, and normal probability distributions in appendix C.

Part I

On Algorithms for Robust State Estimation from Noisy Relative Measurements

2. Robust State Estimation from Noisy Relative Measurements

2.1 Overview

In this part of the report, our attention will be drawn to the state estimation problem.

In the current chapter, we start by defining the measurement model used in the current work, see section 2.2. The novelty herein consists the noise which is modeled as a Gaussian mixture instead of being a constant variance for each measurement. Hereafter, the following estimators, based on the Maximum Likelihood Principle, are derived depending on the availability of information regarding the noise distribution.

In the Weighted Least Squares (WLS) approach, presented in section 2.3, we assume the quality of the measurements to be known, and hence direct estimation can be immediately carried out. Performance analysis are also presented for this estimator.

Next, we consider the WLSP approach, given in section 2.4.1. In the WLSP approach, with the P stands for "Plus", we assume to not have the quality of the measurements but are given the original state vector $\bar{\mathbf{x}}$ and when this information is combined with the available measurements $\mathbf{b}$, the actual noise realization vector η can be obtained. We derive a classification rule based on the Maximum A Posteriori (MAP) estimation. In particular, a threshold value is obtained which decides whether the noise realization is 'good' or 'bad'. After the classification step, we then proceed with estimation using WLS approach. Performance analysis regarding the estimator are also presented.

In section 2.4.2 and 2.4.3, we assume to be given only the noise distribution parameters in addition to the set of measurements and the graph topology. The task is then to first classify the measurements and afterwards perform estimation based on the measurements. Herein, two approaches are considered; that of a naive brute force approach in which for all the possible combinations regarding the quality of the measurements, the WLS estimate is obtained. In the second step, we then choose the WLS estimate $\hat{\mathbf{x}}$ which yields the highest value for the log likelihood function. Hence, in the current approach we have first performed estimation, with classification following after it.

The second approach we consider is based on the Expectation Maximization algorithm. Herein, hidden random variables are introduced in order to complete the measurement data and we alternate between the expectation step, in which a soft classification,i.e., $\hat{z} \in [0, 1]$ with 0 referring to 'good' and 1 to 'bad' measurement, is done and a maximization step, which again boils down to carrying out the WLS approach. In particular, the WLSP approach can be regarded as one iteration of the EM approach in which here hard classification is performed, i.e., $\hat{z} \in \{0, 1\}$.

2.2 Measurement Model

As mentioned already in the Introduction, we are interested in the problem of estimating the state vector from noisy relative measurements (also known as pairwise differences [17]). Problems of this type include localization [11], time synchronization [4, 17], and statistical ranking [7].

The measurement model is described as follows: we consider a network of N agents. Each of the agents in the network possesses a quantity $\bar{x}_i$, which is not known to the agents themselves. In the current work, we assume the quantity to be a scalar value, i.e., $\bar{x}_i \in \mathbb{R}$. The scalar value may for example be, the position in localization problems, the clock offset in time-synchronization problems, or the popularity in the ranking problem. The agents are allowed to take relative noisy measurements with their neighbors, a small subset of the network. The goal is to use the set of noisy relative measurements to construct an estimate of the original state vector $\bar{\mathbf{x}} \in \mathbb{R}^N$. It should be noted that by solely using noisy relative measurements, the original vector can be constructed up to an additive constant [4, 11, 15]. This can be easily seen as adding a constant value to each agent's value will not change the pairwise difference; $\left((\bar{x}_i + c) - (\bar{x}_j + c)\right) = (\bar{x}_i - \bar{x}_j)$. Hence the problem is sometimes referred to as finding an estimate of the pairwise diffferences from noisy measurements [17].

Using terminology from graph theory, the above description can be represented graphically. The agents are regarded as vertices of a graph and the relative measurements as the edges connecting the vertices; see fig. 2.1 for an example graph consisting of 5 vertices and 5 edges connecting the vertices in a pairwise manner.

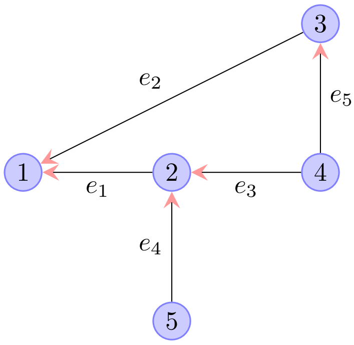


Figure 2.1: An example graph $\mathcal{G}$ with vertex set $\mathcal{V}(\mathcal{G}) = \{1, 2, 3, 4, 5\}$ and edge set $\mathcal{E}(\mathcal{G}) = \{e_1 = (2, 1), e_2 = (3, 1), e_3 = (4, 2), e_4 = (5, 2), e_5 = (4, 3)\}$.

The set of relative measurements can be encoded using the edge-incidence matrix $A \in \{0, \pm 1\}^{M \times N}$ with entries defined as

$$A_{ei} = \begin{cases} +1 & \text{if } e = (j, i) \\ -1 & \text{if } e = (i, j) \\ 0 & \text{otherwise} \end{cases} \quad \text{for } e \in \mathcal{E}(\mathcal{G}). \tag{2.1}$$

Note that each row of the A matrix corresponds to one measurement and we assume to have M measurements. Also, in the above definition, we assume a pair (j, i) to be an element of the edge set $\mathcal{E}(\mathcal{G})$ if and only if $i < j$; in fig. 2.1, this is graphically represented by the arrows in which the orientation is always towards the smallest labelled number of each pair. The A matrix for the graph in fig. 2.1 is

$$A = \begin{bmatrix} 1 & -1 & 0 & 0 & 0 \\ 1 & 0 & -1 & 0 & 0 \\ 0 & 1 & 0 & -1 & 0 \\ 0 & 1 & 0 & 0 & -1 \\ 0 & 0 & 1 & -1 & 0 \end{bmatrix}.$$

By letting $\mathbf{b} \in \mathbb{R}^M$ to denote the set of measurements, we have

$$\mathbf{b} = A\bar{\mathbf{x}} + \eta, \tag{2.2}$$

with $A\bar{\mathbf{x}}$ the uncorrupted relative difference between pairs of vertices within the network and η the vector of Gaussian additive noise corrupting the relative difference. We assume the noise to have the following spefications: each noise term have mean zero, i.e., $\mathbb{E}[\eta_e] = 0$ and the variance equals $\mathbb{E}\left[\eta_e^2\right] = \sigma_e^2 \; \forall e \in \mathcal{E}(\mathcal{G})$, with $\sigma_e^2 = (1 - \bar{z}_e)\alpha^2 + \bar{z}_e\beta^2$, $0 < \alpha < \beta$ and $\bar{z}_e \sim \text{Ber}(p)$, i.e., $\bar{z}_e$ is Bernoulli distributed and the probability that $\bar{z}_e = 1$, which means $\mathcal{N}(0, \beta^2)$ is chosen and hence the measurement is considered 'bad', is p. Furthermore, we assume each noise term η_e to be mutually independent random variables. Graphically, the noise variances can be regarded as the weights on the edges with $w_e = (1 - \bar{z}_e)\alpha^{-2} + \bar{z}_e\beta^{-2} \; \forall e \in \mathcal{E}(\mathcal{G})$.

2.3 The Case of Knowing the Quality of the Measurement Noise; WLS Approach

In the current case, besides the information regarding the graph topology given by A, the set of measurements $\mathbf{b}$ and the noise distribution parameters, we assume to also know the value of $\bar{z}_e$ for each measurement, in other words the quality of each measurement. We have

$$\bar{z}_e = \begin{cases} 1 & \text{if } \eta_e \in \mathcal{N}(0, \beta^2) \\ 0 & \text{if } \eta_e \in \mathcal{N}(0, \alpha^2) \end{cases} \quad \forall e \in \mathcal{E}(\mathcal{G}). \tag{2.3}$$

By knowing this additional information, we know $\mathbb{P}(\eta_e)$ $\forall e \in \mathcal{E}(\mathcal{G})$ exactly and because $b_e = (A\bar{\mathbf{x}})_e + \eta_e$, also $\mathbb{P}\left(b_e \middle| (A\bar{\mathbf{x}})_e\right)$ by shifting the distribution of $\mathbb{P}(\eta_e)$ by the value $(A\bar{x})_e$, see fig. 2.2.

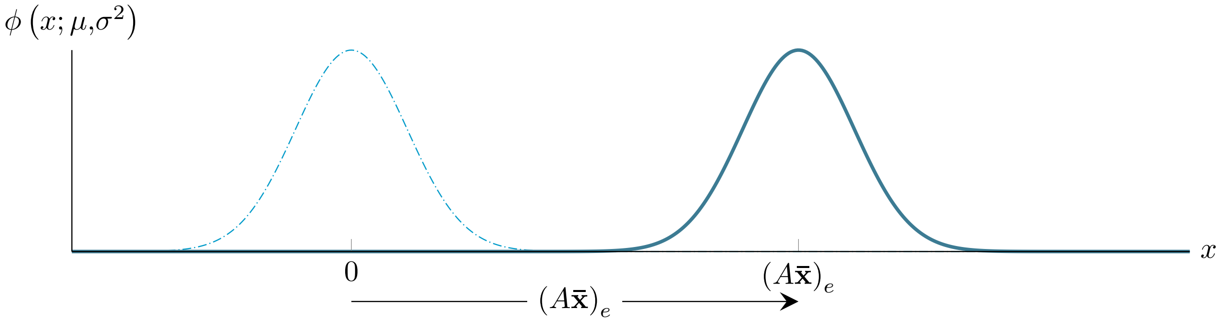


Figure 2.2: The equivalence between $\mathbb{P}(\eta_e)$ and $\mathbb{P}\left(b_e \middle| (A\bar{\mathbf{x}})_e\right)$.

By the iid (independent and identically distributed) assumption, we have for the joint probability function $\mathbb{P}\left(\mathbf{b} \middle| \bar{\mathbf{x}}\right)$

$$\mathbb{P}\left(\mathbf{b} \middle| \bar{\mathbf{x}}\right) = \prod_{e \in \mathcal{E}} \mathbb{P}\left(b_e \middle| (A\bar{\mathbf{x}})_e\right), \tag{2.4}$$

with

$$\mathbb{P}\left(b_e \middle| (A\bar{\mathbf{x}})_e\right) = \frac{1}{\sqrt{2\pi\sigma_e^2}} \exp\left(-\frac{(b_e - (\bar{x}_i - \bar{x}_j))^2}{2\sigma_e^2}\right). \tag{2.5}$$

Subscript e is the pair (j, i) and $\sigma_e^2 = (1 - \bar{z}_e)\alpha^2 + \bar{z}_e\beta^2$. The maximum likelihood estimation (MLE) approach is used in order to obtain an estimate of $\bar{\mathbf{x}}$ when the measurements $\mathbf{b}$ are given. This is based on the following principle:

Definition 2.1 (Maximum Likelihood Principle [18]). *Given a dataset, choose the parameter(s) of interest in such a way that the data are most likely.*

In our case, the data are the measurement vector $\mathbf{b}$ and the parameter of interest the state vector $\bar{\mathbf{x}}$. The MLE approach is the most popular technique for deriving estimators [19] and has among others the invariance property, in which if $\hat{\theta}$ is the maximum likelihood estimate of a parameter θ, then $\tau(\hat{\theta})$ is the MLE for $\tau(\theta)$. Other useful properties of the MLE are the asymptotic unbiasedness and the asymptotic minimum variance property.

The likelihood function is defined as

$$L\left(\mathbf{x} \middle| \mathbf{b}\right) = L\left(x_1, x_2, \ldots, x_N \middle| b_e \forall e \in \mathcal{E}\right) = \mathbb{P}\left(\mathbf{b} \middle| \mathbf{x}\right), \tag{2.6}$$

and according to definition 2.1, we seek to find the parameter values $x_1, x_2, \dots, x_N$ that most likely have produced the observations $b_e \, \forall e \in \mathcal{E}(\mathcal{G})$ (Note: the bar above $\mathbf{x}$ is removed to indicate that $\mathbf{x}$ is now the variable.). This is formulated in the following optimization problem:

$$\hat{\mathbf{x}} = \underset{\mathbf{x} \in \mathbb{R}^N}{\arg\max} \, L\Big(\mathbf{x} \Big| \mathbf{b}\Big). \tag{2.7}$$

In the following we will work with the log likelihood function $\log L\Big(\mathbf{x}\Big|\mathbf{b}\Big)$ by taking the natural logarithm of the likelihood function $L\Big(\mathbf{x}\Big|\mathbf{b}\Big)$. This because in order to solve the maximization problem, we need to differentiate the function and it is easier to work with the log likelihood function as then a product of terms is changed into a sum of the logaritms, Ref. [18]. Also, because the log likelihood function is strictly increasing, the extreme points of $L\Big(\mathbf{x}\Big|\mathbf{b}\Big)$ and $\log L\Big(\mathbf{x}\Big|\mathbf{b}\Big)$ coincide [18, 19]. Hence, we obtain

$$\log L\Big(\mathbf{x}\Big|\mathbf{b}\Big) = \log \mathbb{P}\Big(\mathbf{b}\Big|\mathbf{x}\Big) = \log \prod_{e \in \mathcal{E}(\mathcal{G})} \mathbb{P}\Big(b_e\Big|(A\mathbf{x})_e\Big) = \sum_{e \in \mathcal{E}(\mathcal{G})} \log \mathbb{P}\Big(b_e\Big|(A\mathbf{x})_e\Big). \tag{2.8}$$

The second equality is due to the iid assumption, eq. (2.4) and the third equality by using $\log \prod a_i = \sum \log a_i$.

Proposition 2.1 (WLS Estimator). *Let $A \in \{0, \pm 1\}^{M \times N}$ be the edge-incidence matrix describing the graph topology according to eq.* (2.1)*, $\mathbf{b} \in \mathbb{R}^M$ the vector of measurements, $\bar{\mathbf{z}} \in \{0,1\}^M$ the vector indicating the quality of the measurements and α and β the noise parameters with $0 < \alpha < \beta$, then the solution to eq.* (2.7) *is given by*

$$\hat{\mathbf{x}} = \Big(A^T W_{\bar{\mathbf{z}}} A\Big)^{\dagger} A^T W_{\bar{\mathbf{z}}} \mathbf{b}, \tag{2.9}$$

with $W_{\bar{\mathbf{z}}} = diag\big((1 - \bar{z}_e)\alpha^{-2} + \bar{z}_e \beta^{-2}\big)$.

Proof. In order to solve eq. (2.7), we need to take the derivative of eq. (2.8) with respect to the variables x_k, $k = 1, 2, \dots, N$ and set the system of equations equal to zero, i.e.,

$$\frac{\partial}{\partial x_k} \log L\Big(\mathbf{x}\Big|\mathbf{b}\Big) = 0, \quad \text{for } k = 1,\, 2,\, \dots,\, N. \tag{2.10}$$

As there are N parameters, we have N equations and using the sum rule in differentation, each equation will have M terms corresponding to the number of measurements. For illustration purposes, the derivation for one such term is given below,

$$\log \mathbb{P}\Big(b_e\Big|(A\mathbf{x})_e\Big) = \log\left(\frac{1}{\sqrt{2\pi\sigma_e^2}} \exp\left(-\frac{(b_e - (x_i - x_j))^2}{2\sigma_e^2}\right)\right).$$

Taking the derivative with respect to x_k yields

$$\begin{aligned}
\frac{\partial}{\partial x_k} \log \mathbb{P}\Big(b_e\Big|(A\mathbf{x})_e\Big) &= \frac{\partial}{\partial x_k} \log\left(\frac{1}{\sqrt{2\pi\sigma_e^2}} \exp\left(-\frac{(b_e - (x_i - x_j))^2}{2\sigma_e^2}\right)\right) \\
&= \frac{\partial}{\partial x_k} \log \frac{1}{\sqrt{2\pi\sigma_e^2}} + \frac{\partial}{\partial x_k}\left(-\frac{(b_e - (x_i - x_j))^2}{2\sigma_e^2}\right) \\
&= -\frac{(b_e - (x_i - x_j))}{\sigma_e^2} \frac{\partial(b_e - (x_i - x_j))}{\partial x_k},
\end{aligned}$$

with

$$\frac{\partial(b_e - (x_i - x_j))}{\partial x_k} = \begin{cases} 0 & \text{if edge } e \text{ is not connected to vertex } k \\ -1 & \text{if edge } e \text{ is connected to vertex } k \text{ and } k \text{ is the “to” vertex} \\ 1 & \text{if edge } e \text{ is connected to vertex } k \text{ and } k \text{ is the “from” vertex} \end{cases} \tag{2.11}$$

This corresponds to the $-A_{ek}$ entry in the edge-incidence matrix, hence

$$\frac{\partial}{\partial x_k} \log \mathbb{P}(b_e | (A\mathbf{x})_e) = \frac{A_{ek}\big(b_e - (x_i - x_j)\big)}{\sigma_e^2}. \tag{2.12}$$

As mentioned earlier, there are M terms in each of the N equations and each of the M terms will have the same structure as eq. (2.12). Writing the system of equations in matrix form, we eventually obtain

$$A^T W_{\bar{\mathbf{z}}}(\mathbf{b} - A\mathbf{x}) = 0 \Leftrightarrow A^T W_{\bar{\mathbf{z}}} A\mathbf{x} = A^T W_{\bar{\mathbf{z}}}\mathbf{b} \Leftrightarrow \hat{\mathbf{x}} = \left(A^T W_{\bar{\mathbf{z}}} A\right)^{\dagger} A^T W_{\bar{\mathbf{z}}}\mathbf{b}.$$

with W in the above being a diagonal matrix having entries $w_e = (1 - \bar{z}_e)\alpha^{-2} + \bar{z}_e\beta^{-2}$. The subscript $\bar{\mathbf{z}}$ explicitly states the dependence of W on the indicator vector $\bar{\mathbf{z}}$. □

$()^{\dagger}$ in the above denotes the Moore-Penrose pseudoinverse. We take the Moore-Penrose pseudoinverse for the product $\left(A^T W_{\bar{\mathbf{z}}} A\right)$ is not invertible. Taking a closer look, we can observed that this product is the laplacian of the weighted graph and as the laplacian always has an eigenvalue of zero, it is not invertible, see section A.3 for more information related to the laplacian matrix. $\hat{\mathbf{x}}$ obtained above is the minimum 2-norm solution [20].The estimate $\hat{\mathbf{x}}$ obtained in proposition 2.1 can also be obtained using the weighted least squares (WLS) approach in which we minimize the weighted 2-norm of the difference $(\mathbf{b} - A\mathbf{x})$, i.e.,

$$\hat{\mathbf{x}} = \underset{\mathbf{x}\in\mathbb{R}^N}{\arg\min} \frac{1}{2} ||\mathbf{b} - A\mathbf{x}||^2_{W_{\bar{\mathbf{z}}}}. \tag{2.13}$$

The derivation can be found in Ref. [21]. In the unweighted case, it is known that minimization of the sum of square errors $||\mathbf{b} - A\mathbf{x}||^2$ is equivalent to maximization of the log-likelihood function when the observations are independent of each other and the noise is gaussian [22]; the current result can thus be seen as the weighted analogue to that. As already mentioned earlier, we can estimate $\bar{\mathbf{x}}$ up to an additive constant, hence

$$\hat{\mathbf{x}}_{WLS} = \hat{\mathbf{x}} + c\mathbf{1}, \tag{2.14}$$

with the constant c still undetermined.

In the following, we evaluate the WLS estimator as we are interested in the properties of the estimator. First, we give some definations of properties that are useful for evaluating estimators. We start with the bias of an estimator:

Definition 2.2 (Bias of an estimator [19]). *The bias of a point estimator $\hat{\theta}$ of a parameter θ is the difference between the expected value of $\hat{\theta}$ and θ; that is, Bias$_\theta\hat{\theta} = \mathbb{E}_\theta\left[\hat{\theta}\right] - \theta$. An estimator whose bias is identically equal to* 0 *is called unbiased and satisfies $\mathbb{E}_\theta\left[\hat{\theta}\right] = \theta$ for all θ.*

We continue with a definition for the mean squared error of an estimator:

Definition 2.3 (Mean Squared Error [19]). *The mean squared error (MSE) of an estimator $\hat{\theta}$ of a parameter θ is the function θ defined by $\mathbb{E}_\theta\left[(\hat{\theta} - \theta)^2\right]$.*

The MSE measures the average squared difference between the estimator $\hat{\theta}$ and the parameter θ and can be rewritten as the following

$$\begin{aligned}
\mathbb{E}_\theta\left[(\hat{\theta} - \theta)^2\right] &= \mathbb{E}_\theta\left[(\hat{\theta} - \mathbb{E}_\theta\left[\hat{\theta}\right] + \mathbb{E}_\theta\left[\hat{\theta}\right] - \theta)^2\right] \\
&= \mathbb{E}_\theta\left[(\hat{\theta} - \mathbb{E}_\theta\left[\hat{\theta}\right])^2\right] + 2\underbrace{\mathbb{E}_\theta\left[\left(\hat{\theta} - \mathbb{E}_\theta\left[\hat{\theta}\right]\right)\left(\mathbb{E}_\theta\left[\hat{\theta}\right] - \theta\right)\right]}_{0} + \mathbb{E}_\theta\left[(\mathbb{E}_\theta\left[\hat{\theta}\right] - \theta)^2\right] \\
&= \mathrm{Var}_\theta\hat{\theta} + \left(\mathbb{E}_\theta\left[\hat{\theta}\right] - \theta\right)^2 \\
&= \mathrm{Var}_\theta\hat{\theta} + \left(\mathrm{Bias}_\theta\hat{\theta}\right)^2.
\end{aligned}$$

The MSE can thus be split in two components, $\mathrm{Var}_\theta\hat{\theta}$ measuring the variability of the estimator (precision) and $\left(\mathrm{Bias}_\theta\hat{\theta}\right)^2$ measuring its bias (accuracy). If the estimator is unbiased, then the MSE equals the variance of the estimator, i.e.,

$$\mathbb{E}_\theta\left[(\hat{\theta}-\theta)^2\right] = \mathrm{Var}_\theta\hat{\theta}.$$

Note that for determining whether the estimator is biased or unbiased, we need to calculate the mean of the estimates which is the first moment of a distribution and the variance the second centralized moment of a distribution.

We consider two cases for the evaluation, first is the case in which we hold the random variable $\bar{\mathbf{Z}}$ fixed, i.e., $\bar{\mathbf{Z}} = \bar{\mathbf{z}}$ with $\bar{\mathbf{z}} \in \{0,1\}^M$; second, we consider the case in which $\bar{\mathbf{Z}}$ is random. This is seen as a generalization of the former case.

2.3.1 Case: $\bar{\mathbf{Z}}$ is fixed; $\bar{\mathbf{Z}} = \bar{\mathbf{z}}$

The following proposition sums up the main result of this subsection:

Proposition 2.2 (Moments of the WLS estimator conditioned on $\bar{\mathbf{Z}}$). *Let the WLS estimate of $\bar{\mathbf{x}}$ be given by eq. (2.14) with $\hat{\mathbf{x}}$ obtained from proposition 2.1 and assume $\bar{\mathbf{Z}} = \bar{\mathbf{z}}$ to be fixed. If the additive constant is chosen to be the centroid of the nodes, i.e., $c = mean(\bar{\mathbf{x}})$, and expectation is taken on the noise term, given by the random variable $\mathbf{H}$, we can obtain the following:*

$$\mathbb{E}_{\mathbf{H}}\left[\hat{\mathbf{x}}_{WLS}\middle|\bar{\mathbf{Z}} = \bar{\mathbf{z}}\right] = \bar{\mathbf{x}}, \tag{2.15}$$

and

$$\mathbb{E}_{\mathbf{H}}\left[(\hat{\mathbf{x}}_{WLS}-\bar{\mathbf{x}})(\hat{\mathbf{x}}_{WLS}-\bar{\mathbf{x}})^T\middle|\bar{\mathbf{Z}} = \bar{\mathbf{z}}\right] = \left(A^T W_{\bar{\mathbf{z}}} A\right)^\dagger, \tag{2.16}$$

i.e., the WLS estimator is unbiased and its covariance matrix is given by the laplacian of the weighted graph.

Proof. First, we will prove eq. (2.15).

$$\begin{aligned}
&\mathbb{E}_{\mathbf{H}}\left[\hat{\mathbf{x}}_{WLS}\middle|\bar{\mathbf{Z}} = \bar{\mathbf{z}}\right] \\
&= \mathbb{E}_{\mathbf{H}}\left[\hat{\mathbf{x}} + c\mathbf{1}\middle|\bar{\mathbf{Z}} = \bar{\mathbf{z}}\right] \\
&= \mathbb{E}_{\mathbf{H}}\left[\left(A^T W_{\bar{\mathbf{z}}} A\right)^\dagger A^T W_{\bar{\mathbf{z}}}\mathbf{b} + c\mathbf{1}\middle|\cdots\right] \\
&= \mathbb{E}_{\mathbf{H}}\left[\left(A^T W_{\bar{\mathbf{z}}} A\right)^\dagger A^T W_{\bar{\mathbf{z}}} A\bar{\mathbf{x}}\middle|\cdots\right] + \mathbb{E}_{\mathbf{H}}\left[\left(A^T W_{\bar{\mathbf{z}}} A\right)^\dagger A^T W_{\bar{\mathbf{z}}}\eta_{\bar{\mathbf{z}}}\middle|\cdots\right] + \mathbb{E}\left[c\mathbf{1}\middle|\cdots\right] \\
&= \underbrace{\left(A^T W_{\bar{\mathbf{z}}} A\right)^\dagger\left(A^T W_{\bar{\mathbf{z}}} A\right)}_{\left(I_N - \frac{1}{N}\mathbf{1}\mathbf{1}^T\right)}\bar{\mathbf{x}} + \left(A^T W_{\bar{\mathbf{z}}} A\right)^\dagger A^T W_{\bar{\mathbf{z}}}\underbrace{\mathbb{E}_{\mathbf{H}}\left[\eta_{\bar{\mathbf{z}}}\middle|\bar{\mathbf{Z}} = \bar{\mathbf{z}}\right]}_{\mathbf{0}} + c\mathbf{1} \\
&= \left(I_N - \frac{1}{N}\mathbf{1}\mathbf{1}^T\right)\bar{\mathbf{x}} + c\mathbf{1}.
\end{aligned}$$

With the choice of $c = \frac{1}{N}\mathbf{1}^T\bar{\mathbf{x}} = \left(\frac{1}{N}\sum_{i=1}^N \bar{x}_i\right)$, we obtain eq. (2.15). Plugging this result in eq. (2.14) yields

$$\hat{\mathbf{x}}_{WLS} = \bar{\mathbf{x}} + \left(A^T W_{\bar{\mathbf{z}}} A\right)^\dagger A^T W_{\bar{\mathbf{z}}}\eta_{\bar{\mathbf{z}}}. \tag{2.17}$$

We proceed by showing eq. (2.16).

$$\begin{aligned}
&\mathbb{E}_{\mathbf{H}}\left[(\hat{\mathbf{x}}_{WLS}-\bar{\mathbf{x}})(\hat{\mathbf{x}}_{WLS}-\bar{\mathbf{x}})^T\Big|\bar{\mathbf{Z}}=\bar{\mathbf{z}}\right]\\
&=\mathbb{E}_{\mathbf{H}}\left[\left(A^TW_{\bar{\mathbf{z}}}A\right)^{\dagger}A^TW_{\bar{\mathbf{z}}}\eta_{\bar{\mathbf{z}}}\eta_{\bar{\mathbf{z}}}^TW_{\bar{\mathbf{z}}}A\left(A^TW_{\bar{\mathbf{z}}}A\right)^{\dagger T}\Big|\cdots\right]\\
&=\left(A^TW_{\bar{\mathbf{z}}}A\right)^{\dagger}A^TW_{\bar{\mathbf{z}}}\mathbb{E}_{\mathbf{H}}\left[\eta_{\bar{\mathbf{z}}}\eta_{\bar{\mathbf{z}}}^T\Big|\cdots\right]W_{\bar{\mathbf{z}}}A\left(A^TW_{\bar{\mathbf{z}}}A\right)^{\dagger T}\\
&=\left(A^TW_{\bar{\mathbf{z}}}A\right)^{\dagger}A^T\underbrace{W_{\bar{\mathbf{z}}}Q_{\bar{\mathbf{z}}}}_{I_N}W_{\bar{\mathbf{z}}}A\left(A^TW_{\bar{\mathbf{z}}}A\right)^{\dagger}\\
&=\left(A^TW_{\bar{\mathbf{z}}}A\right)^{\dagger}\left(A^TW_{\bar{\mathbf{z}}}A\right)\left(A^TW_{\bar{\mathbf{z}}}A\right)^{\dagger}\\
&=\left(A^TW_{\bar{\mathbf{z}}}A\right)^{\dagger}.
\end{aligned} \tag{2.18}$$

In the previous derivation, the third equality is obtained using property 2 of the Moore-Penrose pseudoinverse in appendix B and noting that the matrix product $(A^TW_{\bar{\mathbf{z}}}A)$ is the laplacian of the weighted graph, we know it is symmetric, i.e., $(A^TW_{\bar{\mathbf{z}}}A)=(A^TW_{\bar{\mathbf{z}}}A)^T$. The last equality is obtained using the second Penrose equation in theorem B.1. □

2.3.2 Case: $\bar{\mathbf{Z}}$ is random

As mentioned already this is the generatization of results obtained from the previous subsection. Before we state the proposition we give the following theorem which will be used.

Theorem 2.1 ([19]). *If X and Y are any two random variables, then*

$$\mathbb{E}[X]=\mathbb{E}\left[\mathbb{E}\left[X\Big|Y\right]\right]. \tag{2.19}$$

Proposition 2.3 (Moments of the WLS estimator). *Let the WLS estimate of $\bar{\mathbf{x}}$ be given by eq. (2.14) with $\hat{\mathbf{x}}$ obtained from proposition 2.1 and assume $\bar{\mathbf{Z}}$ to be random. If the additive constant is chosen to be the centroid of the nodes, i.e., $c=mean(\bar{\mathbf{x}})$, and expectation is taken on the noise term, given by the random variable $\mathbf{H}$, and on $\bar{\mathbf{Z}}$, then we can obtain the following:*

$$\mathbb{E}_{\bar{\mathbf{Z}},\mathbf{H}}[\hat{\mathbf{x}}_{WLS}]=\bar{\mathbf{x}}, \tag{2.20}$$

and

$$\mathbb{E}_{\bar{\mathbf{Z}},\mathbf{H}}\left[(\hat{\mathbf{x}}_{WLS}-\bar{\mathbf{x}})(\hat{\mathbf{x}}_{WLS}-\bar{\mathbf{x}})^T\right]=\sum_{\mathbf{z}}(1-p)^{\#\alpha}p^{\#\beta}\left(A^TW_{\mathbf{z}}A\right)^{\dagger}. \tag{2.21}$$

with $\#\alpha$ being the number of zeros in $\mathbf{z}$ and $\#\beta$ the number of ones and p the probability of getting a one.

Proof. Again, we will first prove eq. (2.20). Using theorem 2.1 and the definition of expectation, the following can be obtained

$$\mathbb{E}_{\bar{\mathbf{Z}},\mathbf{H}}[\hat{\mathbf{x}}_{WLS}]=\sum_{\mathbf{z}}\mathbb{P}(\bar{\mathbf{Z}}=\mathbf{z})\mathbb{E}_{\mathbf{H}}\left[\hat{\mathbf{x}}_{WLS}\Big|\bar{\mathbf{Z}}=\mathbf{z}\right]. \tag{2.22}$$

In words: for each $\bar{\mathbf{Z}}=\mathbf{z}$, I can obtain a value for $\mathbb{E}_{\mathbf{H}}\left[\hat{\mathbf{x}}_{WLS}\Big|\bar{\mathbf{Z}}=\mathbf{z}\right]$; $\mathbb{E}_{\bar{\mathbf{Z}},\mathbf{H}}[\hat{\mathbf{x}}_{WLS}]$ is seen as taking the weighted sum of $\mathbb{E}_{\mathbf{H}}\left[\hat{\mathbf{x}}_{WLS}\Big|\bar{\mathbf{Z}}=\mathbf{z}\right]$ with weights given by $\mathbb{P}(\bar{\mathbf{Z}}=\mathbf{z})$.
In the above equation, three pieces of information is needed:

- $\sum_{\mathbf{z}}$;
  We need to consider 2^M possible combinations for the random variable $\bar{\mathbf{Z}}$ with M being the number of measurements;

- $\mathbb{P}(\bar{\mathbf{Z}} = \mathbf{z})$;

$$\mathbb{P}(\bar{\mathbf{Z}} = \mathbf{z}) = \mathbb{P}\left(\bigcap_{i=1}^{M} \bar{Z}_i = z_i\right) = \prod_{i=1}^{M} \mathbb{P}(\bar{Z}_i = z_i) = (1-p)^{\#\alpha} p^{\#\beta} \quad \text{with } \#\alpha + \#\beta = M. \tag{2.23}$$

  The second equality is obtained due to the iid assumption and the third equality by the following observation; We know that $z_i \in \{0, 1\}$ with 0 referring to a 'good' measurement and 1 to a 'bad' measurement. The probability of obtaining a 'bad' measurement is p. Grouping the zeros and ones in the vector $\mathbf{z}$ leads to the third equality.
- $\mathbb{E}_{\mathbf{H}}\left[\hat{\mathbf{x}}_{WLS} \middle| \bar{\mathbf{Z}} = \mathbf{z}\right]$; may be obtained using proposition 2.2.

Putting the pieces together yields

$$\mathbb{E}_{\bar{\mathbf{Z}},\mathbf{H}}[\hat{\mathbf{x}}_{WLS}] = \sum_{\mathbf{z}} \mathbb{P}(\bar{\mathbf{Z}} = \mathbf{z}) \mathbb{E}_{\mathbf{H}}\left[\hat{\mathbf{x}}_{WLS} \middle| \bar{\mathbf{Z}} = \mathbf{z}\right] = \sum_{\mathbf{z}} (1-p)^{\#\alpha} p^{\#\beta} \bar{\mathbf{x}} = \bar{\mathbf{x}}.$$

The last equality is obtained because the sum $\sum_{\mathbf{z}} (1-p)^{\#\alpha} p^{\#\beta} = 1$ as we sum over the sample space of $\bar{\mathbf{Z}}$. From the above we can conclude that the WLS estimator is in its general form unbiased. With the same reasoning, we can obtain eq. (2.21). □

2.3.3 Simplification of the Matrix Product $(A^TWA)^\dagger A^TWQWA(A^TWA)^\dagger$

In the following, we are interested in obtaining a simpler form for the matrix product $(A^TWA)^\dagger A^TWQWA(A^TWA)^\dagger$ by using existing results available for the Moore-Penrose pseudoinverse in appendix B. The case for which $W = Q^{-1}$ is already considered in eq. (2.18). There we have found that the simplified form is then $(A^TWA)^\dagger$. We now consider the case when $W \neq Q^{-1}$. This will be useful for the subsequent sections, in particular for the WLSP estimator, 2.4.1.

We first consider rewriting the product $(A^TWA)^\dagger$, by introducing $\tilde{A} = A^TW^{1/2}$.

Proposition 2.4. *Let $A \in \mathbb{R}^{m\times n}$ be a rectangular matrix, $W \in \mathbb{R}^{m\times m}$ a diagonal matrix and consider the matrix product $\tilde{A} = A^TW^{1/2} \in \mathbb{R}^{n\times m}$. Then we have*

$$\left(A^TWA\right)^\dagger = \tilde{A}^{\dagger T}\tilde{A}^\dagger. \tag{2.24}$$

Proof.

$$\left(A^TWA\right)^\dagger = \left(A^TW^{1/2}W^{1/2}A\right)^\dagger = \left(\tilde{A}\tilde{A}^T\right)^\dagger = \tilde{A}^{T\dagger}\tilde{A}^\dagger = \tilde{A}^{\dagger T}\tilde{A}^\dagger. \tag{2.25}$$

The third equality is obtained using special case 2 for the matrix product of theorem B.2. □

Now we can state the main result.

Theorem 2.2. *Let $A \in \mathbb{R}^{m\times n}$ be a rectangular matrix, $W, Q \in \mathbb{R}^{m\times m}$ be diagonal matrices with $W \neq Q^{-1}$ and consider the the matrix product $\tilde{A} = A^TW^{1/2} \in \mathbb{R}^{n\times m}$, then we have*

$$\left(A^TWA\right)^\dagger A^TWQWA\left(A^TWA\right)^\dagger = \tilde{A}^{\dagger T}\underbrace{W^{1/2}QW^{1/2}}_{WQ=QW}\tilde{A}^\dagger.$$

Proof. Using proposition 2.4, we obtain

$$\begin{aligned}\left(A^TWA\right)^\dagger A^TWQWA\left(A^TWA\right)^\dagger &= \tilde{A}^{\dagger T}\tilde{A}^\dagger\tilde{A}W^{1/2}QW^{1/2}\tilde{A}^T\tilde{A}^{\dagger T}\tilde{A}^\dagger \\ &= \left(\tilde{A}^{\dagger T}\tilde{A}^\dagger\tilde{A}\right)W^{1/2}QW^{1/2}\left(\tilde{A}^{\dagger T}\tilde{A}^\dagger\tilde{A}\right)^T.\end{aligned} \tag{2.26}$$

The second equality is due to the transpose property $(AB)^T = B^TA^T$. We consider the product $\tilde{A}^{\dagger T}\tilde{A}^\dagger\tilde{A}$.

$$\tilde{A}^{\dagger T}\tilde{A}^\dagger\tilde{A} = \tilde{A}^{\dagger T}\left(\tilde{A}^\dagger\tilde{A}\right)^T = \left(\tilde{A}^\dagger\tilde{A}\tilde{A}^\dagger\right)^T = \tilde{A}^{\dagger T}.$$

The second equality is a consequence of applying property 4 of theorem B.1; the third is obtained using the transpose property and the fourth equality due to property 2 of again theorem B.1. Plugging the result in eq. (2.26) yields the desired result. We also note that diagonal matrices commute, i.e., $AB = BA$, hence the product $WQ = QW$. □

If in addition, matrix A is full row rank, then further simplification can be obtained. This because we can apply special case of theorem B.2 to $\tilde{A}$ due to the observation that if A if full row rank, then A^T is full column rank. We obtain

$$\tilde{A}^\dagger = \left(A^T W^{1/2}\right)^\dagger = \left(W^{1/2}\right)^\dagger A^{T\dagger} = W^{-1/2} A^{T\dagger}. \tag{2.27}$$

with the third equality as a consequence of applying property 3 for Moore-Penrose pseudoinverse matrices. Application to proposition 2.4 and theorem 2.2, we have

Proposition 2.5. *Assume the conditions in proposition 2.4 holds and in addition, we have that A is full row rank, i.e., rank$(A) = m$, then*

$$\left(A^T W A\right)^\dagger = A^\dagger W^{-1} A^{\dagger T}. \tag{2.28}$$

Proof. By plugging the result of eq. (2.27) in proposition 2.4, we have

$$\tilde{A}^{\dagger T} \tilde{A}^\dagger = \left(W^{-1/2} A^{T\dagger}\right)^T W^{-1/2} A^{T\dagger} = A^\dagger W^{-1} A^{\dagger T}.$$

□

Theorem 2.3. *Assume the conditions in theorem 2.2 holds and in addition, we have that A is full row rank, i.e., rank$(A) = m$, then*

$$\left(A^T W A\right)^\dagger A^T W Q W A \left(A^T W A\right)^\dagger = \left(A^T Q^{-1} A\right)^\dagger. \tag{2.29}$$

Proof. By plugging the result of eq. (2.27) in theorem 2.2, we have

$$\tilde{A}^{\dagger T} W^{1/2} Q W^{1/2} \tilde{A}^\dagger = A^\dagger W^{-1/2} W^{1/2} Q W^{1/2} W^{-1/2} A^{\dagger T} = A^\dagger Q A^{\dagger T} = \left(A^T Q^{-1} A\right)^\dagger. \tag{2.30}$$

An alternative way to show the above is to start from proposition 2.5 and then using the property that A is full row rank and hence A^T is full column rank, see property 6 and 7 for Moore-Penrose pseudoinverse matrices. □

We end this section by providing the WLS algorithm, given in algorithm 1.

Algorithm 1 WLS Approach

```
Require: Data: $(A, \mathbf{b}, \bar{\mathbf{z}}, p, \alpha, \beta)$
Computation of weights:
for all $e \in \mathcal{E}(\mathcal{G})$ do
    $w_e = \frac{1 - \bar{z}_e}{\alpha^2} + \frac{\bar{z}_e}{\beta^2} \quad \forall e \in \mathcal{E}(\mathcal{G})$
end for
Estimation step:
    $\hat{\mathbf{x}} = \left(A^T W_{\bar{\mathbf{z}}} A\right)^\dagger A^T W_{\bar{\mathbf{z}}} \mathbf{b}, \quad$ with $W_{\bar{\mathbf{z}}} = \mathrm{diag}(\mathbf{w})$
Additive constant:
    $\hat{\mathbf{x}}_{WLS} = \hat{\mathbf{x}} + c\mathbf{1}, \quad$ with $c = \mathrm{mean}(\bar{\mathbf{x}})$.
```

2.4 The Case of Not Knowing the Quality of the Measurement Noise

In section 2.3, we consider the case in which we know the quality of the measurement noise; however, in reality this information is a priori not known; hence an estimation of the measurement quality, given by $\bar{\mathbf{z}}$, needs to be found (also known as the classification step) preceding the estimation of the state vector. Three approaches are presented in which classification and estimation are being performed. But first, the log likelihood function will be derived for the current case of unknown measurement quality.

With the quality for each measurement unknown, the probability distribution of the noise term η_e is now a mixture of Gaussian distributions, obtain using the total probability law:

$$\begin{aligned}\mathbb{P}(\eta_e)_{\mathsf{UK}} =&\mathbb{P}\Big(\eta_e\Big|\mathcal{N}(0,\beta^2)\Big)\mathbb{P}(\mathcal{N}(0,\beta^2)) + \mathbb{P}\Big(\eta_e\Big|\mathcal{N}(0,\alpha^2)\Big)\mathbb{P}(\mathcal{N}(0,\alpha^2))\\ =&\frac{1}{\sqrt{2\pi\beta^2}}\exp\left(-\frac{(\eta_e)^2}{2\beta^2}\right)\cdot p + \frac{1}{\sqrt{2\pi\alpha^2}}\exp\left(-\frac{(\eta_e)^2}{2\alpha^2}\right)\cdot(1-p),\end{aligned} \tag{2.31}$$

with $\sum_{\sigma_e^2\in\{\alpha^2,\beta^2\}}\mathbb{P}(\mathcal{N}(0,\sigma_e^2)) = 1$. The subscript UK indicates 'Unknown'. By shifting the probability distribution function by $(A\bar{\mathbf{x}})_e$ the conditional probability of $\mathbb{P}\Big(b_e\Big|(A\bar{\mathbf{x}})_e\Big)_{\mathsf{UK}}$ is found to be

$$\mathbb{P}\Big(b_e\Big|(A\bar{\mathbf{x}})_e\Big)_{\mathsf{UK}} = \frac{p}{\sqrt{2\pi\beta^2}}\exp\left(-\frac{(b_e-(\bar{x}_i-\bar{x}_j))^2}{2\beta^2}\right) + \frac{1-p}{\sqrt{2\pi\alpha^2}}\exp\left(-\frac{(b_e-(\bar{x}_i-\bar{x}_j))^2}{2\alpha^2}\right). \tag{2.32}$$

We again use the MLE approach to obtain an estimate of the state vector given the available measurements. In the current case, the likelihood function is

$$L\Big(\mathbf{x}\Big|\mathbf{b}\Big)_{\mathsf{UK}} = L\Big(x_1, x_2, \ldots, x_N\Big|b_e\,\forall e\in\mathcal{E}\Big) = \mathbb{P}\Big(\mathbf{b}\Big|\mathbf{x}\Big)_{\mathsf{UK}}. \tag{2.33}$$

In order to estimate $\bar{\mathbf{x}}$, we need to maximize the following:

$$\hat{\mathbf{x}} = \underset{\mathbf{x}\in\mathbb{R}^N}{\arg\max}\, L\Big(\mathbf{x}\Big|\mathbf{b}\Big)_{\mathsf{UK}} \tag{2.34}$$

By taking the natural logarithm, we have

$$\log L\Big(\mathbf{x}\Big|\mathbf{b}\Big)_{\mathsf{UK}} = \log\mathbb{P}\Big(\mathbf{b}\Big|\mathbf{x}\Big)_{\mathsf{UK}} = \log\prod_{e\in\mathcal{E}}\mathbb{P}\Big(b_e\Big|(A\mathbf{x})_e\Big)_{\mathsf{UK}} = \sum_{e\in\mathcal{E}}\log\mathbb{P}\Big(b_e\Big|(A\mathbf{x})_e\Big)_{\mathsf{UK}}. \tag{2.35}$$

As pointed out in Ref. [23], maximization of the above equation in a direct manner is not an easy task and no closed form exist for the optimal $\hat{\mathbf{x}}$. Hence, we look for alternatives to solve this maximization problem. In the approaches presented, a 2-step approach is taken; first, the measurements are classified, after which the state vector is then estimated.

2.4.1 WLSP Approach; Having Access to the Noise Realization

In the WLS approach, we assume to know the quality of each measurement ahead of time and hence the noise distribution. In the current approach, we have the inverse and assume to know the noise realization for each measurement. Our goal is then to obtain the noise distribution that most likely have produced the realization and with this information obtain an estimate of the state vector.

The reason for us to consider this approach is because of it serving as an idealized version of the Expectation Maximization approach which will be elaborated on hereafter. We call it the WLSP approach as will be clear, we add a classification step in algorithm 1. Hence the 'P' which stands for 'Plus'. We will elaborate on the classification step as the estimation step is already considered in section 2.3.

In order to obtain the noise realization, we assume to have 'temporary' access to the state vector $\bar{\mathbf{x}}$. The noise realization η can then be obtained using eq. (2.2). We then specifically ask the question: which noise distribution (assuming that we then know the number of distributions that are present, the family of distributions, and the parameters of the distributions) has most likely produced the noise term we observe. This again is an instance of the use of definition 2.1. The question can be formulated as the following optimization problem:

$$\hat{\sigma}_e^2 = \underset{\sigma_e^2 \in \{\alpha^2, \beta^2\}}{\arg\max}\ \mathbb{P}\Big(\mathcal{N}(0, \sigma_e^2)\Big| b_e, (A\bar{\mathbf{x}})_e\Big), \tag{2.36}$$

with $\mathbb{P}\Big(\mathcal{N}(0, \sigma_e^2)\Big| b_e, (A\bar{\mathbf{x}})_e\Big)$ being the posterior distribution of obtaining σ_e^2.

Proposition 2.6 (WLSP Classification with $p < \frac{1}{1+\gamma}$ and $\gamma = \frac{\alpha}{\beta}$). *Let $A \in \{0, \pm 1\}^{M \times N}$ be the edge-incidence matrix describing the graph topology according to eq.* (2.1), *$\mathbf{b} \in \mathbb{R}^M$ the vector of measurements, $\bar{\mathbf{x}} \in \mathbb{R}^N$ the state vector, and α, β, and p the noise parameters with $0 < \alpha < \beta$ and $p < \frac{1}{1+\gamma}$ with $\gamma = \frac{\alpha}{\beta}$, then we have*

$$\hat{z}_e = \begin{cases} 1 & \textit{if} \left|\eta_e\right| > \delta \\ 0 & \textit{otherwise.} \end{cases} \quad \forall e \in \mathcal{E}(\mathcal{G}). \tag{2.37}$$

with η_e being the noise realization and δ the decision boundary given by

$$\delta = \sqrt{2\left(\frac{1}{\alpha^2} - \frac{1}{\beta^2}\right)^{-1} \log\left(\frac{1-p}{p}\frac{\beta}{\alpha}\right)}. \tag{2.38}$$

Proof. We start the proof by rewriting eq. (2.36) for we are dealing with only two alternatives for σ_e^2: The following inequality can be obtained:

$$\mathbb{P}\Big(\mathcal{N}(0, \beta^2)\Big| b_e, (A\bar{\mathbf{x}})_e\Big) > \mathbb{P}\Big(\mathcal{N}(0, \alpha^2)\Big| b_e, (A\bar{\mathbf{x}})_e\Big),$$

known in the statistics literature as the Bayes classifier. This classifier produces the lowest possible test error rate, called the Bayes error rate and serves a a standard against which other methods are compared to [24]. Using the Bayes' theorem, stated as follows

$$\mathbb{P}\Big(D\Big|E, F\Big) = \frac{\mathbb{P}\Big(E\Big|D, F\Big)\mathbb{P}\Big(D\Big|F\Big)}{\mathbb{P}\Big(E\Big|F\Big)}, \tag{2.39}$$

we rewrite the above inequality. In the current case, we have

$$D = \mathcal{N}(0, \beta^2) \text{ or } \mathcal{N}(0, \alpha^2), \quad E = b_e, \quad F = (A\bar{\mathbf{x}})_e.$$

The derivation is only done for $\mathcal{N}(0, \beta^2)$ as the same procedure applies for when $D = \mathcal{N}(0, \alpha^2)$. Substituting yields

$$\begin{aligned} &\mathbb{P}\Big(\mathcal{N}(0, \beta^2)\Big| b_e, (A\bar{\mathbf{x}})_e\Big) \\ &= \frac{\mathbb{P}\Big(b_e\Big|\mathcal{N}(0, \beta^2), (A\bar{\mathbf{x}})_e\Big)\mathbb{P}\Big(\mathcal{N}(0, \beta^2)\Big|(A\bar{\mathbf{x}})_e\Big)}{\mathbb{P}\Big(b_e\Big|(A\bar{\mathbf{x}})_e\Big)} \\ &= \frac{\frac{p}{\sqrt{2\pi\beta^2}}\exp\left(-\frac{(b_e - (\bar{x}_i - \bar{x}_j))^2}{2\beta^2}\right)}{\frac{p}{\sqrt{2\pi\beta^2}}\exp\left(-\frac{(b_e - (\bar{x}_i - \bar{x}_j))^2}{2\beta^2}\right) + \frac{1-p}{\sqrt{2\pi\alpha^2}}\exp\left(-\frac{(b_e - (\bar{x}_i - \bar{x}_j))^2}{2\alpha^2}\right)}. \end{aligned} \tag{2.40}$$

Note: $\mathbb{P}\Big(\mathcal{N}(0,\beta^2)\Big|(A\bar{\mathbf{x}})_e\Big) = \mathbb{P}\big(\mathcal{N}(0,\beta^2)\big)$. As the denominator is the same in the inequality, we only pay attention to the numerator:

$$\begin{aligned}
\mathbb{P}\Big(\mathcal{N}(0,\beta^2)\Big|b_e,(A\bar{\mathbf{x}})_e\Big) &> \mathbb{P}\Big(\mathcal{N}(0,\alpha^2)\Big|b_e,(A\bar{\mathbf{x}})_e\Big) && \Leftrightarrow \\
\frac{p}{\sqrt{2\pi\beta^2}}\exp\left(-\frac{(b_e-(\bar{x}_i-\bar{x}_j))^2}{2\beta^2}\right) &> \frac{1-p}{\sqrt{2\pi\alpha^2}}\exp\left(-\frac{(b_e-(\bar{x}_i-\bar{x}_j))^2}{2\alpha^2}\right) && \Leftrightarrow \\
\exp\left(\frac{(b_e-(\bar{x}_i-\bar{x}_j))^2}{2}\left(-\frac{1}{\beta^2}+\frac{1}{\alpha^2}\right)\right) &> \frac{1-p}{\sqrt{2\pi\alpha^2}}\frac{\sqrt{2\pi\beta^2}}{p} && \Leftrightarrow \\
\left(\frac{(b_e-(\bar{x}_i-\bar{x}_j))^2}{2}\left(-\frac{1}{\beta^2}+\frac{1}{\alpha^2}\right)\right) &> \log\left(\frac{1-p}{p}\frac{\beta}{\alpha}\right) && \Leftrightarrow \\
\Big|\underbrace{b_e-(\bar{x}_i-\bar{x}_j)}_{\eta_e}\Big| &> \underbrace{\sqrt{2\left(\frac{1}{\alpha^2}-\frac{1}{\beta^2}\right)^{-1}\log\left(\frac{1-p}{p}\frac{\beta}{\alpha}\right)}}_{\delta}.
\end{aligned}$$

The fourth inequality is obtained by taking the natural logarithm of both sides. Using the assumption that $0<\alpha<\beta$, we have $\frac{1}{\alpha^2}>\frac{1}{\beta^2}$ and hence no sign change occurs when we multiply both sides with $\left(\frac{1}{\alpha^2}-\frac{1}{\beta^2}\right)^{-1}$. The condition $p<\frac{1}{1+\gamma}$ with $\gamma=\frac{\alpha}{\beta}$ is needed in order for the log term to be greater than 0. This can be easily verified. Hence, the proof is given. □

In fig. 2.3, the normal curve is divided into two regions, with the region indicated in blue, the region in which the noise term η_e is classified as to be from $\mathcal{N}(0,\alpha^2)$ and when η_e is in the red region, then it is assumed to be sampled from $\mathcal{N}(0,\beta^2)$.

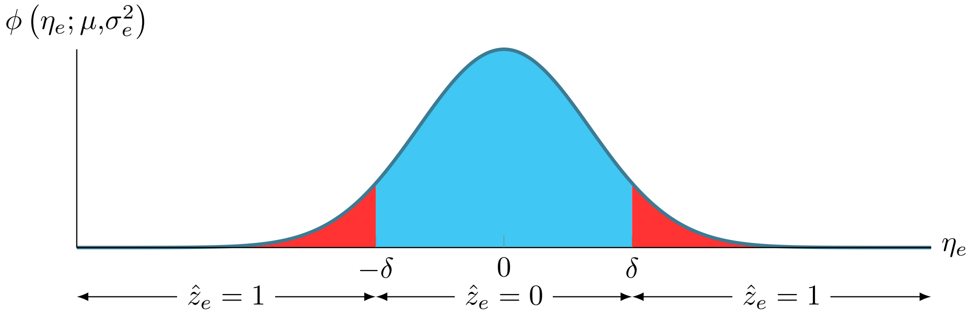


Figure 2.3: Plot of a normal curve and regions showing classification depending on the realization η_e

In case the condition $p<\frac{1}{1+\gamma}$ is not met, we have the following:

Proposition 2.7 (WLSP Classification with $p\geq\frac{1}{1+\gamma}$ and $\gamma=\frac{\alpha}{\beta}$). *Let $A\in\{0,\pm1\}^{M\times N}$ be the edge-incidence matrix describing the graph topology according to eq.* (2.1), *$\mathbf{b}\in\mathbb{R}^M$ the vector of measurements, $\bar{\mathbf{x}}\in\mathbb{R}^N$ the state vector, and α, β, and p the noise parameters with $0<\alpha<\beta$ and $p\geq\frac{1}{1+\gamma}$ with $\gamma=\frac{\alpha}{\beta}$, then we have, regardless of η_e,*

$$\hat{z}_e=1 \quad \forall e\in\mathcal{E}(\mathcal{G}). \tag{2.41}$$

Proof. With $p\geq\frac{1}{1+\gamma}$, we have that the log term in eq. (2.38) is negative and as such, δ is a complex value with real part zero, i.e.,

$$\delta=0+i\sqrt{-2\left(\frac{1}{\alpha^2}-\frac{1}{\beta^2}\right)^{-1}\log\left(\frac{1-p}{p}\frac{\beta}{\alpha}\right)}. \tag{2.42}$$

Hence $\left|\eta_e\right| > \delta$ is always true as $\left|\eta_e\right| > \mathrm{Re}(\delta) = 0$. The noise term is thus regarded as to be sampled from the distribution $\mathcal{N}(0, \beta^2)$ □

In fig. 2.4, we plot γ vs. p indicating the region for which classification is done based on respectively proposition 2.6 and proposition 2.7.

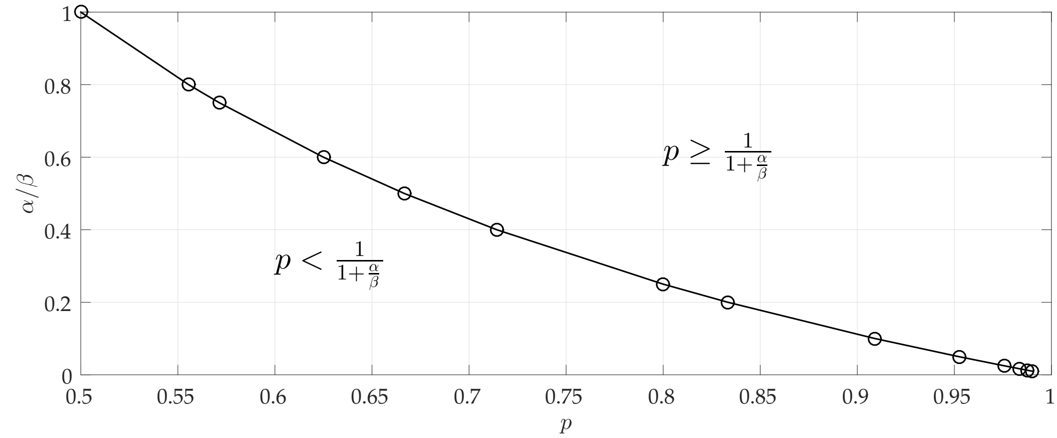


Figure 2.4: Plot of $\gamma = \frac{\alpha}{\beta}$ vs. p indicating the regions for the classification.

After obtaining an estimate of the measurement quality vector, we can use the WLS approach for state estimation, see algorithm 2 for the details.

Algorithm 2 WLSP Approach

```
Require: Data: $(A, \mathbf{b}, \bar{\mathbf{x}}, p, \alpha, \beta)$
Obtain noise vector:
    $$\eta = \mathbf{b} - A\bar{\mathbf{x}}$$
Classification step:
    $$\delta = \sqrt{2\left(\frac{1}{\alpha^2} - \frac{1}{\beta^2}\right)^{-1} \log\left(\frac{1-p}{p}\frac{\beta}{\alpha}\right)}$$
for all $e \in \mathcal{E}(\mathcal{G})$ do
    if $\mathrm{Re}(\delta) > 0$ then
        if $\left|\eta_e\right| > \delta$ then
            $\hat{z}_e = 1$
        else
            $\hat{z}_e = 0$
        end if
    else
        $\hat{z}_e = 1$
    end if
end for
Estimation Step:
Proceed to algorithm 1 with Data: $(A, \mathbf{b}, \hat{\mathbf{z}}, p, \alpha, \beta)$
```

2.4.1.1 Probability of Correct Classification

In this subsection, we are interested in the probability of having correct classification, i.e., $\mathbb{P}\left(\hat{\mathbf{z}}_\delta = \mathbf{z}_g\right)$, with subscript δ denoting $\hat{\mathbf{z}}$ obtained using eq. (2.37), and g the generated vector.

We first define the following four conditional probabilities,

Definition 2.4. *Given $\bar{\mathbf{Z}}$, the random variable for the generated indicator vector of the measurement quality and $\hat{\mathbf{Z}}$ that of the estimated measurement quality, with $\bar{z}_e$ and $\hat{z}_e$ only taking the values $\{0, 1\}$, we can define the following probabilities:*

$$\begin{aligned} TN &= \mathbb{P}\left(\hat{Z}_i = 0 \middle| \bar{Z}_i = 0\right) & FN &= \mathbb{P}\left(\hat{Z}_i = 0 \middle| \bar{Z}_i = 1\right) \\ FP &= \mathbb{P}\left(\hat{Z}_i = 1 \middle| \bar{Z}_i = 0\right) & TP &= \mathbb{P}\left(\hat{Z}_i = 1 \middle| \bar{Z}_i = 1\right) \end{aligned} \tag{2.43}$$

with TN = True Negative, FN = False Negative, FP = False Positive, and TP = True Positive.

The above terminology is commonly used in medical testing application. The above probabilities may be redefined in terms of the decision boundary derived in eq. (2.38) as follows:

$$\begin{aligned} \mathrm{TN} &= \mathbb{P}\left(\left|\eta_e\right| \leq \delta \middle| \eta_e \in \mathcal{N}(0, \alpha^2)\right) & \mathrm{FN} &= \mathbb{P}\left(\left|\eta_e\right| \leq \delta \middle| \eta_e \in \mathcal{N}(0, \beta^2)\right) \\ \mathrm{FP} &= \mathbb{P}\left(\left|\eta_e\right| > \delta \middle| \eta_e \in \mathcal{N}(0, \alpha^2)\right) & \mathrm{TP} &= \mathbb{P}\left(\left|\eta_e\right| > \delta \middle| \eta_e \in \mathcal{N}(0, \beta^2)\right). \end{aligned} \tag{2.44}$$

The calculation of the conditional probabilities may be done by a transformation to the standard normal distribution using $\tilde{\eta}_e = \frac{\eta_e - 0}{\sigma_e}$, see also section C.1.

$$\mathbb{P}\left(\left|\eta_e\right| \leq \delta \middle| \eta_e \in \mathcal{N}(0, \sigma_e^2)\right) = \mathbb{P}\left(\left|\tilde{\eta}_e\right| \leq \frac{\delta}{\sigma_e}\right) = \mathbb{P}\left(-\frac{\delta}{\sigma_e} \leq \tilde{\eta}_e \leq \frac{\delta}{\sigma_e}\right) = 1 - 2\mathbb{P}\left(\tilde{\eta}_e \leq -\frac{\delta}{\sigma_e}\right).$$

The last equality is obtained due to symmetry of the normal curve around the zero mean. The probability calculated above is the probability of the blue region in fig. 2.3. The probability of the red region is calculated below.

$$\mathbb{P}\left(\left|\eta_e\right| > \delta \middle| \eta_e \in \mathcal{N}(0, \sigma_e^2)\right) = 1 - \mathbb{P}\left(\left|\eta_e\right| \leq \delta \middle| \eta_e \in \mathcal{N}(0, \sigma_e^2)\right) = 2\mathbb{P}\left(\tilde{\eta}_e \leq -\frac{\delta}{\sigma_e}\right).$$

These values can be obtained using a look-up table, see Ref. [25], or numerically. We can now state the proposition:

Proposition 2.8. *Let $\mathbf{z}_g$ be the indicator vector obtained by $Ber(p)$ and $\hat{\mathbf{z}}_\delta$ obtained using 2.37, then, for any $\mathbf{z}$, we have,*

$$\mathbb{P}\left(\hat{\mathbf{z}}_\delta = \mathbf{z}_g\right) = (TNp + TP(1-p))^M = \sum_{\mathbf{z}} (TNp)^{\#\alpha} (TP(1-p))^{\#\beta} \quad \textit{with } M = \#\alpha + \#\beta. \tag{2.45}$$

Proof. The proof is as follows:
Using the total probability law, we may obtain

$$\mathbb{P}\left(\hat{\mathbf{z}}_\delta = \mathbf{z}_{\mathrm{g}}\right) = \sum_{\mathbf{z}} \mathbb{P}\left(\hat{\mathbf{Z}} = \mathbf{z} \middle| \bar{\mathbf{Z}} = \mathbf{z}\right) \mathbb{P}(\bar{\mathbf{Z}} = \mathbf{z}). \tag{2.46}$$

We first look at the general case for $\mathbb{P}\left(\hat{\mathbf{Z}} = \hat{\mathbf{z}} \middle| \bar{\mathbf{Z}} = \bar{\mathbf{z}}\right)$; the probability of obtaining $\hat{\mathbf{z}}$ given $\bar{\mathbf{z}}$; Due to independence of both $\hat{\mathbf{Z}}$ and $\bar{\mathbf{Z}}$, we eventually obtain

$$\begin{aligned} \mathbb{P}\left(\hat{\mathbf{Z}} = \hat{\mathbf{z}} \middle| \bar{\mathbf{Z}} = \bar{\mathbf{z}}\right) &= \prod_{i=1}^{M} \mathbb{P}\left(\hat{Z}_i = \hat{z}_i \middle| \bar{Z}_i = \bar{z}_i\right) \\ &= \mathrm{TN}^{\#\mathrm{TN}} \mathrm{TP}^{\#\mathrm{TP}} \mathrm{FP}^{\#\mathrm{FP}} \mathrm{FN}^{\#\mathrm{FN}} \end{aligned} \quad \text{with } \#\mathrm{TN} + \#\mathrm{TP} + \#\mathrm{FP} + \#\mathrm{FN} = M. \tag{2.47}$$

as each factor $\mathbb{P}\left(\hat{Z}_i = \hat{z}_i \middle| \bar{Z}_i = \bar{z}_i\right)$ will be one of the cases defined in definition 2.4, and we grouped the factors which are the same together. For the case of correct classification, we have

$$\mathbb{P}\left(\hat{\mathbf{Z}} = \mathbf{z} \middle| \bar{\mathbf{Z}} = \mathbf{z}\right) = \mathsf{TN}^{\#\alpha}\mathsf{TP}^{\#\beta} \tag{2.48}$$

with $\#\alpha$ and $\#\beta$ being respectively the number of zeros and ones in $\mathbf{z}$ and the constraint $\#\alpha + \#\beta = M$. Plugging the parts in, we obtain

$$\mathbb{P}(\hat{\mathbf{z}}_\delta = \mathbf{z}_\mathrm{g}) = \sum_{\mathbf{z}} \mathbb{P}\left(\hat{\mathbf{Z}} = \mathbf{z} \middle| \bar{\mathbf{Z}} = \mathbf{z}\right)\mathbb{P}(\bar{\mathbf{Z}} = \mathbf{z}) = \sum_{\mathbf{z}} \mathsf{TN}^{\#\alpha}\mathsf{TP}^{\#\beta} p^{\#\alpha}(1-p)^{\#\beta} = \sum_{\mathbf{z}} (\mathsf{TN}p)^{\#\alpha}(\mathsf{TP}(1-p))^{\#\beta}.$$

In order to obtain the second form of eq. (2.45), we may start from an alternative method; we first calculate $\mathbb{P}(\hat{z}_e = \bar{z}_e)$:

$$\begin{aligned}\mathbb{P}(\hat{z}_e = \bar{z}_e) &= \mathbb{P}(\hat{z}_e = 0 \cap \bar{z}_e = 0) + \mathbb{P}(\hat{z}_e = 1 \cap \bar{z}_e = 1)\\ &= \mathbb{P}(\hat{z}_e = 0 | \bar{z}_e = 0)\mathbb{P}(\bar{z}_e = 0) + \mathbb{P}(\hat{z}_e = 1 | \bar{z}_e = 1)\mathbb{P}(\bar{z}_e = 1)\\ &= \mathsf{TN}p + \mathsf{TP}(1-p),\end{aligned}$$

The first equality is an expansion and the second equality is due to the use of the multiplication rule for probabilities. $\mathbb{P}(\hat{\mathbf{z}} = \bar{\mathbf{z}})$ is then

$$\mathbb{P}(\hat{\mathbf{z}} = \bar{\mathbf{z}}) = \mathbb{P}\left(\bigcap_{e\in\mathcal{E}} \hat{z}_e = \bar{z}_e\right) = \prod_{e\in\mathcal{E}} \mathbb{P}(\hat{z}_e = \bar{z}_e) = (\mathsf{TN}p + \mathsf{TP}(1-p))^M.$$

The former and the latter equation are linked by the binomial theorem. □

As is the case for the WLS approach, the evaluation of the WLSP estimator is done by observing whether it possesses the unbiasedness property and its MSE. Again, the cases $\bar{\mathbf{Z}} = \bar{\mathbf{z}}$ is fixed and $\bar{\mathbf{Z}}$ is random are considered.

2.4.1.2 Case: $\bar{\mathbf{Z}}$ is fixed; $\bar{\mathbf{Z}} = \bar{\mathbf{z}}$

Before stating the main result of this subsection, we first introduced two distributions that are useful for the subsequent calculations.

Definition 2.5 (Mean and Variance of the Two-Sided Truncated Normal Distribution). *Given a normal distribution with zero mean and variance σ^2, i.e., $\mathcal{N}(0,\sigma^2)$, a symmetric bound $[-b,b]$ around the mean in which the normal distribution is only defined therein, we have the following:*

$$\begin{aligned}\mu_{TR} &= 0\\ \sigma^2_{TR} &= \sigma^2\left(1 - \frac{2\frac{b}{\sigma}\phi\left(\frac{b}{\sigma};0,1\right)}{1 - 2\Phi\left(-\frac{b}{\sigma};0,1\right)}\right)\end{aligned} \tag{2.49}$$

with subscript TR referring to truncated. ϕ is the standard normal distribution and Φ its cumulative density distribution (cdf).

and

Definition 2.6 (Mean and Variance of the Two-Sided Normal Tail Distribution). *Given a normal distribution with zero mean and variance σ^2, i.e., $\mathcal{N}(0,\sigma^2)$, a symmetric bound $[-b,b]$ around the mean and the normal distribution is only defined outside this bound, we have the following:*

$$\begin{aligned}\mu_{Tail} &= 0\\ \sigma^2_{Tail} &= \sigma^2\left(1 + \frac{\frac{b}{\sigma}\phi\left(\frac{b}{\sigma};0,1\right)}{\Phi\left(-\frac{b}{\sigma};0,1\right)}\right)\end{aligned} \tag{2.50}$$

with subscript Tail referring to the tail of the distribution.

The above definitions are obtained from results in C. In fig. 2.3, definition 2.5 is the blue region and definition 2.6 is the red region.

We first give the main result and consequently the derivation.

Proposition 2.9 (Moments of the WLSP Estimator conditioned on $\bar{\mathbf{Z}}$ with Classification According to proposition 2.6). *Let the WLSP estimate be obtained using algorithm 2 with classification obtained according to proposition 2.6 and assume $\bar{\mathbf{Z}} = \bar{\mathbf{z}}$ to be fixed. When the additive constant is chosen to be the centroid of the nodes, i.e., $c = mean(\bar{\mathbf{x}})$, and expectation is taken on the noise term, given by the random variable $\mathbf{H}$, we can obtain the following:*

$$\mathbb{E}_{\mathbf{H}}\left[\hat{\mathbf{x}}_{WLSP}\middle|\bar{\mathbf{Z}} = \bar{\mathbf{z}}\right] = \bar{\mathbf{x}} \tag{2.51}$$

and

$$\begin{aligned}
&\mathbb{E}_{\mathbf{H}}\left[(\hat{\mathbf{x}}_{WLSP} - \bar{\mathbf{x}})(\hat{\mathbf{x}}_{WLSP} - \bar{\mathbf{x}})^T\middle|\bar{\mathbf{Z}} = \bar{\mathbf{z}}\right]\\
&= \sum_{\tilde{\mathbf{z}}}\left(TN^{\#TN}TP^{\#TP}FP^{\#FP}FN^{\#FN}\right)\tilde{A}^{\dagger T}W_{\tilde{\mathbf{z}}}Q_{(\bar{\mathbf{z}},\tilde{\mathbf{z}})}\tilde{A}^{\dagger} \quad \text{with } \tilde{A} = A^T W_{\tilde{\mathbf{z}}}^{1/2}.\\
&= \sum_{\tilde{\mathbf{z}}}\left(TN^{\#TN}TP^{\#TP}FP^{\#FP}FN^{\#FN}\right)\tilde{A}^{\dagger T}Q_{(\bar{\mathbf{z}},\tilde{\mathbf{z}})}W_{\tilde{\mathbf{z}}}\tilde{A}^{\dagger}
\end{aligned} \tag{2.52}$$

Proof. The proof is constructed in the same manner as the one given for the WLS approach. In the current approach, due to the fact that we have an estimation of $\bar{\mathbf{z}}$, $\hat{\mathbf{z}}$ obtained using the classification rule may or may not be equal to $\bar{\mathbf{z}}$. Using theorem 2.1, we have

$$\mathbb{E}_{\mathbf{H}}\left[\hat{\mathbf{x}}_{WLSP}\middle|\bar{\mathbf{Z}} = \bar{\mathbf{z}}\right] = \sum_{\tilde{\mathbf{z}}}\mathbb{P}\left(\hat{\mathbf{Z}} = \tilde{\mathbf{z}}\middle|\bar{\mathbf{Z}} = \bar{\mathbf{z}}\right)\mathbb{E}_{\mathbf{H}}\left[\hat{\mathbf{x}}_{WLSP}\middle|\bar{\mathbf{Z}} = \bar{\mathbf{z}}, \hat{\mathbf{Z}} = \tilde{\mathbf{z}}\right] \tag{2.53}$$

We focus on the last term as the former one is obtained already previously, see eq. (2.47). From algorithm 2, we know $\hat{\mathbf{x}}$ is obtained using the WLS approach with a realization of the random variable $\hat{\mathbf{Z}}$ responsible for the weighted matrix W, hence

$$\hat{\mathbf{x}}_{WLSP} = \left(A^T W_{\hat{\mathbf{z}}} A\right)^{\dagger} A^T W_{\hat{\mathbf{z}}}\mathbf{b} + c\mathbf{1} = \left(A^T W_{\hat{\mathbf{z}}} A\right)^{\dagger} A^T W_{\hat{\mathbf{z}}}(A\bar{\mathbf{x}} + \eta_{\bar{\mathbf{z}}}) + c\mathbf{1},$$

and

$$\begin{aligned}
&\mathbb{E}_{\mathbf{H}}\left[\hat{\mathbf{x}}_{WLSP}\middle|\bar{\mathbf{Z}} = \bar{\mathbf{z}}, \hat{\mathbf{Z}} = \hat{\mathbf{z}}\right]\\
&= \mathbb{E}_{\mathbf{H}}\left[\left(A^T W_{\hat{\mathbf{z}}} A\right)^{\dagger} A^T W_{\hat{\mathbf{z}}}(A\bar{\mathbf{x}} + \eta_{\bar{\mathbf{z}}}) + c\mathbf{1}\middle|\bar{\mathbf{Z}} = \bar{\mathbf{z}}, \hat{\mathbf{Z}} = \hat{\mathbf{z}}\right]\\
&= \mathbb{E}_{\mathbf{H}}\left[\left(A^T W_{\hat{\mathbf{z}}} A\right)^{\dagger}\left(A^T W_{\hat{\mathbf{z}}} A\right)\bar{\mathbf{x}}\middle|\cdots\right] + \mathbb{E}_{\mathbf{H}}\left[\left(A^T W_{\hat{\mathbf{z}}} A\right)^{\dagger} A^T W_{\hat{\mathbf{z}}}\eta_{\bar{\mathbf{z}}}\middle|\cdots\right] + \mathbb{E}_{\mathbf{H}}\left[c\mathbf{1}\middle|\cdots\right]\\
&= \left(I_N - \frac{1}{N}\mathbf{1}\mathbf{1}^T\right)\bar{\mathbf{x}} + \left(A^T W_{\hat{\mathbf{z}}} A\right)^{\dagger} A^T W_{\hat{\mathbf{z}}}\underbrace{\mathbb{E}_{\mathbf{H}}\left[\eta_{\bar{\mathbf{z}}}\middle|\bar{\mathbf{Z}} = \bar{\mathbf{z}}, \hat{\mathbf{Z}} = \hat{\mathbf{z}}\right]}_{\mathbf{0}} + c\mathbf{1}\\
&= \left(I_N - \frac{1}{N}\mathbf{1}\mathbf{1}^T\right)\bar{\mathbf{x}} + c\mathbf{1}.
\end{aligned}$$

$\mathbb{E}_{\mathbf{H}}\left[\eta_{\bar{\mathbf{z}}}\middle|\bar{\mathbf{Z}} = \bar{\mathbf{z}}, \hat{\mathbf{Z}} = \hat{\mathbf{z}}\right] = \mathbf{0}$ because $\mathbb{E}_{\mathbf{H}}\left[\eta_{\bar{\mathbf{z}}}\middle|\bar{\mathbf{Z}} = \bar{\mathbf{z}}, \hat{\mathbf{Z}} = \hat{\mathbf{z}}\right] = \mu_{\bullet}$ (refer to definition 2.5 and definition 2.6) for any combination of $\bar{\mathbf{Z}} = \bar{\mathbf{z}}$ and $\hat{\mathbf{Z}} = \hat{\mathbf{z}}$ and μ equals zero. With the addictive constant c chosen to be the mean of $\bar{\mathbf{x}}$ we obtain $\mathbb{E}_{\mathbf{H}}\left[\hat{\mathbf{x}}_{WLSP}\middle|\bar{\mathbf{Z}} = \bar{\mathbf{z}}, \hat{\mathbf{Z}} = \hat{\mathbf{z}}\right] = \bar{\mathbf{x}}$. Substituting in eq. (2.53) and noting that $\sum_{\tilde{\mathbf{z}}}\mathbb{P}\left(\hat{\mathbf{Z}} = \tilde{\mathbf{z}}\middle|\bar{\mathbf{Z}} = \bar{\mathbf{z}}\right) = 1$ as we sum over the sample space of $\hat{\mathbf{Z}}$ yields the first equation of the proposition. Plugging c in $\hat{\mathbf{x}}_{WLSP}$ yields

$$\hat{\mathbf{x}}_{WLSP} = \bar{\mathbf{x}} + \left(A^T W_{\hat{\mathbf{z}}} A\right)^{\dagger} A^T W_{\hat{\mathbf{z}}}\eta_{\bar{\mathbf{z}}}. \tag{2.54}$$

We proceed by showing the second equation. Again, using theorem 2.1, we have

$$\begin{aligned}
&\mathbb{E}_{\mathbf{H}}\left[(\hat{\mathbf{x}}_{WLSP}-\bar{\mathbf{x}})(\hat{\mathbf{x}}_{WLSP}-\bar{\mathbf{x}})^T\middle|\bar{\mathbf{Z}}=\bar{\mathbf{z}}\right] \\
&=\sum_{\tilde{\mathbf{z}}}\mathbb{P}\left(\hat{\mathbf{Z}}=\tilde{\mathbf{z}}\middle|\bar{\mathbf{Z}}=\bar{\mathbf{z}}\right)\mathbb{E}_{\mathbf{H}}\left[(\hat{\mathbf{x}}_{WLSP}-\bar{\mathbf{x}})(\hat{\mathbf{x}}_{WLSP}-\bar{\mathbf{x}})^T\middle|\bar{\mathbf{Z}}=\bar{\mathbf{z}},\hat{\mathbf{Z}}=\tilde{\mathbf{z}}\right]
\end{aligned} \tag{2.55}$$

We will focus on the term $\mathbb{E}_{\mathbf{H}}\left[(\hat{\mathbf{x}}_{WLSP}-\bar{\mathbf{x}})(\hat{\mathbf{x}}_{WLSP}-\bar{\mathbf{x}})^T\middle|\bar{\mathbf{Z}}=\bar{\mathbf{z}},\hat{\mathbf{Z}}=\hat{\mathbf{z}}\right]$.

$$\begin{aligned}
&\mathbb{E}_{\mathbf{H}}\left[(\hat{\mathbf{x}}_{WLSP}-\bar{\mathbf{x}})(\hat{\mathbf{x}}_{WLSP}-\bar{\mathbf{x}})^T\middle|\bar{\mathbf{Z}}=\bar{\mathbf{z}},\hat{\mathbf{Z}}=\hat{\mathbf{z}}\right] \\
&=\mathbb{E}_{\mathbf{H}}\left[\left(\left(A^TW_{\hat{\mathbf{z}}}A\right)^{\dagger}A^TW_{\hat{\mathbf{z}}}\boldsymbol{\eta}_{\bar{\mathbf{z}}}\right)\left(\left(A^TW_{\hat{\mathbf{z}}}A\right)^{\dagger}A^TW_{\hat{\mathbf{z}}}\boldsymbol{\eta}_{\bar{\mathbf{z}}}\right)^T\middle|\bar{\mathbf{Z}}=\bar{\mathbf{z}},\hat{\mathbf{Z}}=\hat{\mathbf{z}}\right] \\
&=\mathbb{E}_{\mathbf{H}}\left[\left(A^TW_{\hat{\mathbf{z}}}A\right)^{\dagger}A^TW_{\hat{\mathbf{z}}}\boldsymbol{\eta}_{\bar{\mathbf{z}}}\boldsymbol{\eta}_{\bar{\mathbf{z}}}^TW_{\hat{\mathbf{z}}}A\left(A^TW_{\hat{\mathbf{z}}}A\right)^{\dagger}\middle|\bar{\mathbf{Z}}=\bar{\mathbf{z}},\hat{\mathbf{Z}}=\hat{\mathbf{z}}\right] \\
&=\left(A^TW_{\hat{\mathbf{z}}}A\right)^{\dagger}A^TW_{\hat{\mathbf{z}}}\mathbb{E}_{\mathbf{H}}\left[\boldsymbol{\eta}_{\bar{\mathbf{z}}}\boldsymbol{\eta}_{\bar{\mathbf{z}}}^T\middle|\bar{\mathbf{Z}}=\bar{\mathbf{z}},\hat{\mathbf{Z}}=\hat{\mathbf{z}}\right]W_{\hat{\mathbf{z}}}A\left(A^TW_{\hat{\mathbf{z}}}A\right)^{\dagger}.
\end{aligned} \tag{2.56}$$

The term $\mathbb{E}_{\mathbf{H}}\left[\boldsymbol{\eta}_{\bar{\mathbf{z}}}\boldsymbol{\eta}_{\bar{\mathbf{z}}}^T\middle|\bar{\mathbf{Z}}=\bar{\mathbf{z}},\hat{\mathbf{Z}}=\hat{\mathbf{z}}\right]=\sigma_{\bullet}^2$ is dependent on both $\bar{\mathbf{z}}$ and $\hat{\mathbf{z}}$. $\bar{\mathbf{z}}$ supplies information regarding the normal distribution the noise term is sampled from and $\hat{\mathbf{z}}$ the information in which region of fig. 2.3 the realized noise is for each measurement. The distributions defined in definition 2.5 and definition 2.6 comes into play for determining the variance for each measurement. We consider four different cases:

$$\begin{aligned}
&\mathbb{E}_{\mathbf{H}}\left[\eta_e\eta_e^T\middle|\bar{z}_e=0,\hat{z}_e=0\right] \quad \mathbb{E}_{\mathbf{H}}\left[\eta_e\eta_e^T\middle|\bar{z}_e=0,\hat{z}_e=1\right] \\
&\mathbb{E}_{\mathbf{H}}\left[\eta_e\eta_e^T\middle|\bar{z}_e=1,\hat{z}_e=0\right] \quad \mathbb{E}_{\mathbf{H}}\left[\eta_e\eta_e^T\middle|\bar{z}_e=1,\hat{z}_e=1\right]
\end{aligned} \tag{2.57}$$

The variance of the realized noise will be one of the above-mentioned cases. $\mathbb{E}_{\mathbf{H}}\left[\eta_e\eta_e^T\middle|\bar{z}_e=0,\hat{z}_e=0\right]$ will be elaborated on for the moment. When $\bar{z}_e=0$ and $\hat{z}_e=0$, it means that we know the noise is sampled from the normal distribution having α^2 as the variance and the sampled noise η_e is bounded by δ, i.e, it is in the blue region of fig. 2.3. Hence, we have the normal truncated distribution, definition 2.5. The variance is then

$$\sigma_{\text{TR}}^2=\sigma^2\left(1-\frac{2\frac{\delta}{\alpha}\phi(\frac{\delta}{\alpha};0,1)}{1-2\Phi(-\frac{\delta}{\alpha};0,1)}\right) \tag{2.58}$$

Next, we consider the case $\mathbb{E}_{\mathbf{H}}\left[\eta_e\eta_e^T\middle|\bar{z}_e=0,\hat{z}_e=1\right]$. Again η_e is sampled from $\mathcal{N}(0,\alpha^2)$ and the sampled η_e lies in the red region of fig. 2.3. Hence, η_e has a distribution defined by definition 2.6. Its variance is then

$$\sigma_{\text{Tail}}^2=\sigma^2\left(1+\frac{\frac{\delta}{\alpha}\phi(\frac{\delta}{\alpha};0,1)}{\Phi(-\frac{\delta}{\alpha};0,1)}\right) \tag{2.59}$$

For the case $\bar{z}_e=1$, the derivation is the same with the only difference that instead of α, we now have β. The term $\mathbb{E}_{\mathbf{H}}\left[\boldsymbol{\eta}_{\bar{\mathbf{z}}}\boldsymbol{\eta}_{\bar{\mathbf{z}}}^T\middle|\bar{\mathbf{Z}}=\bar{\mathbf{z}},\hat{\mathbf{Z}}=\hat{\mathbf{z}}\right]=Q_{(\bar{\mathbf{z}},\hat{\mathbf{z}})}$ will thus be the noise covariance matrix with each diagonal entry being one of the four cases in eq. (2.57), depending on the value of $\bar{z}_e$ and $\hat{z}_e$. Putting all the pieces together, we have

$$\begin{aligned}
&\mathbb{E}_{\mathbf{H}}\left[(\hat{\mathbf{x}}_{WLSP}-\bar{\mathbf{x}})(\hat{\mathbf{x}}_{WLSP}-\bar{\mathbf{x}})^T\middle|\bar{\mathbf{Z}}=\bar{\mathbf{z}}\right] \\
&=\sum_{\tilde{\mathbf{z}}}\mathbb{P}\left(\hat{\mathbf{Z}}=\tilde{\mathbf{z}}\middle|\bar{\mathbf{Z}}=\bar{\mathbf{z}}\right)\mathbb{E}_{\mathbf{H}}\left[(\hat{\mathbf{x}}_{WLSP}-\bar{\mathbf{x}})(\hat{\mathbf{x}}_{WLSP}-\bar{\mathbf{x}})^T\middle|\bar{\mathbf{Z}}=\bar{\mathbf{z}},\hat{\mathbf{Z}}=\tilde{\mathbf{z}}\right] \\
&=\sum_{\tilde{\mathbf{z}}}\mathsf{TN}^{\#\mathsf{TN}}\mathsf{TP}^{\#\mathsf{TP}}\mathsf{FP}^{\#\mathsf{FP}}\mathsf{FN}^{\#\mathsf{FN}}\left(A^TW_{\tilde{\mathbf{z}}}A\right)^{\dagger}A^TW_{\tilde{\mathbf{z}}}Q_{(\bar{\mathbf{z}},\tilde{\mathbf{z}})}W_{\tilde{\mathbf{z}}}A\left(A^TW_{\tilde{\mathbf{z}}}A\right)^{\dagger}.
\end{aligned} \tag{2.60}$$

and the final equation is obtained by applying theorem 2.2. □

2.4.1.3 Case: $\bar{\mathbf{Z}}$ is random

We state the result without proof as it follows the same manner as for the WLS case.

Proposition 2.10 (Moments of the WLSP Estimator with Classification According to proposition 2.6). *Let the WLSP estimate be obtained using algorithm 2 with classification obtained according to proposition 2.6 and assume $\bar{\mathbf{Z}}$ to be random. When the additive constant is chosen to be the centroid of the nodes, i.e., $c = mean(\bar{\mathbf{x}})$, and expectation is taken on both $\mathbf{H}$ and $\bar{\mathbf{Z}}$, then we can obtain the following:*

$$\mathbb{E}_{\bar{\mathbf{Z}},\mathbf{H}}[\hat{\mathbf{x}}_{WLSP}] = \bar{\mathbf{x}}. \tag{2.61}$$

and

$$\mathbb{E}_{\bar{\mathbf{Z}},\mathbf{H}}\Big[(\hat{\mathbf{x}}_{WLSP} - \bar{\mathbf{x}})(\hat{\mathbf{x}}_{WLSP} - \bar{\mathbf{x}})^T\Big] = \sum_{\mathbf{z}}\sum_{\tilde{\mathbf{z}}}\Big((1-p)^{\#\alpha}p^{\#\beta}\Big)\left(TN^{\#TN}TP^{\#TP}FP^{\#FP}FN^{\#FN}\right)\tilde{A}^{\dagger T}Q_{(\mathbf{z},\tilde{\mathbf{z}})}W_{\tilde{\mathbf{z}}}\tilde{A}^{\dagger}. \tag{2.62}$$

2.4.1.4 Approximation of $\mathbb{E}_{\bar{\mathbf{Z}},\mathbf{H}}\left[(\hat{\mathbf{x}}_{WLSP} - \bar{\mathbf{x}})(\hat{\mathbf{x}}_{WLSP} - \bar{\mathbf{x}})^T\right]$

As observed in proposition 2.10, we see that a double sum is needed for calculating $\mathbb{E}_{\bar{\mathbf{Z}},\mathbf{H}}\left[(\hat{\mathbf{x}}_{WLSP} - \bar{\mathbf{x}})(\hat{\mathbf{x}}_{WLSP} - \bar{\mathbf{x}})^T\right]$, the MSE or in this case also the covariance matrix for the WLSP estimator. As we have determined earlier, the number of combinations for $\mathbf{z}$ is 2^M with M being the measurements. This number thus grows exponentially and as we also have a double sum we have thus $2^M \times 2^M$ terms. This is computationally expensive. Moreover the contribution of some terms is minimal for the resulting outcome, hence an approximation for calculating the covariance matrix $\mathbb{E}_{\bar{\mathbf{Z}},\mathbf{H}}\left[(\hat{\mathbf{x}}_{WLSP} - \bar{\mathbf{x}})(\hat{\mathbf{x}}_{WLSP} - \bar{\mathbf{x}})^T\right]$ may be desired.
The following is proposed as an approximation.
We will approximate $\mathbb{E}_{\mathbf{H}}\Big[(\hat{\mathbf{x}}_{WLSP} - \bar{\mathbf{x}})(\hat{\mathbf{x}}_{WLSP} - \bar{\mathbf{x}})^T\Big|\bar{\mathbf{Z}} = \mathbf{z}\Big]$ in proposition 2.9, by taking only the terms for which $\mathbb{P}\Big(\hat{\mathbf{Z}} = \tilde{\mathbf{z}}\Big|\bar{\mathbf{Z}} = \mathbf{z}\Big) = \left(\mathrm{TN}^{\#\mathrm{TN}}\mathrm{TP}^{\#\mathrm{TP}}\mathrm{FP}^{\#\mathrm{FP}}\mathrm{FN}^{\#\mathrm{FN}}\right) > \epsilon$ in account. By summing the terms satisfying the threshold, we are able to obtain a lower bound for $\mathbb{E}_{\mathbf{H}}\Big[(\hat{\mathbf{x}}_{WLSP} - \bar{\mathbf{x}})(\hat{\mathbf{x}}_{WLSP} - \bar{\mathbf{x}})^T\Big|\bar{\mathbf{Z}} = \mathbf{z}\Big]$ and hence also for $\mathbb{E}_{\bar{\mathbf{Z}},\mathbf{H}}\left[(\hat{\mathbf{x}}_{WLSP} - \bar{\mathbf{x}})(\hat{\mathbf{x}}_{WLSP} - \bar{\mathbf{x}})^T\right]$. The choice of ϵ will be a trade-off between the number of terms considered and a reasonable coverage of $\sum_{\tilde{\mathbf{z}}}\mathbb{P}\Big(\hat{\mathbf{Z}} = \tilde{\mathbf{z}}\Big|\bar{\mathbf{Z}} = \bar{\mathbf{z}}\Big)$.

The previous subsections concern the evaluation of the WLSP estimator when the classification is according to proposition 2.6. We will now evaluate the WLSP estimator when classification is according to proposition 2.7. The unbiasedness of the estimator will still hold (as it holds for all $\mathbf{z}$) but the calculation of MSE will be different.

2.4.1.5 Case: Classification according to proposition 2.7

The main results are summarized in the following propositions, distinguishing between the case $\bar{\mathbf{Z}}$ is fixed and $\bar{\mathbf{Z}}$ is random.

Proposition 2.11 (MSE of the WLSP Estimator conditioned on $\bar{\mathbf{Z}}$ with Classification According to proposition 2.7). *Let the WLSP estimate be obtained using algorithm 2 with classification obtained according to proposition 2.7 and assume $\bar{\mathbf{Z}}$ to be fixed. When the addtitive constant is chosen to be the centroid of the nodes, i.e., $c = mean(\bar{x})$, and expectation is taken on $\mathbf{H}$, then we can obtain the following:*

$$\mathbb{E}_{\mathbf{H}}\Big[(\hat{\mathbf{x}}_{WLSP} - \bar{\mathbf{x}})(\hat{\mathbf{x}}_{WLSP} - \bar{\mathbf{x}})^T\Big|\bar{\mathbf{Z}} = \bar{\mathbf{z}}\Big] = \mathbb{E}_{\mathbf{H}}\Big[(\hat{\mathbf{x}}_{WLSP} - \bar{\mathbf{x}})(\hat{\mathbf{x}}_{WLSP} - \bar{\mathbf{x}})^T\Big|\bar{\mathbf{Z}} = \bar{\mathbf{z}}, \hat{\mathbf{Z}} = \mathbf{1}\Big] = A^{\dagger}Q_{\bar{\mathbf{z}}}A^{\dagger T}. \tag{2.63}$$

Proof. We start with the result from proposition 2.9. Therein, the sum is taken over all the possible combinations for $\tilde{\mathbf{z}}$. As in the current case we will consistently have a vector of ones as the result, this reduced to only the equation given by the second equality and as the W matrix can be rewritten as $\beta^{-2}I_M$ and cancels out, we obtain the third equality. □

Proposition 2.12 (MSE of the WLSP Estimator with Classification According to proposition 2.7). *Let the WLSP estimate be obtained using algorithm 2 with classification obtained according to proposition 2.7 and assume $\bar{\mathbf{Z}}$ to be random. When the additive constant is chosen to be the centroid of the nodes, i.e., $c = mean(\bar{x})$, and expectation is taken on $\mathbf{H}$ and $\bar{\mathbf{Z}}$, then we can obtain the following:*

$$\mathbb{E}_{\bar{\mathbf{Z}},\mathbf{H}}\left[(\hat{\mathbf{x}}_{WLSP} - \bar{\mathbf{x}})(\hat{\mathbf{x}}_{WLSP} - \bar{\mathbf{x}})^T \middle| \hat{\mathbf{Z}} = \mathbf{1}\right] = \sum_{\mathbf{z}} \left((1-p)^{\#\alpha} p^{\#\beta}\right) A^{\dagger} Q_{\mathbf{z}} A^{\dagger T}. \tag{2.64}$$

Proof. The proof follows from applying theorem 2.1 and results obtained previously; hence it is not given. □

2.4.2 Brute Force Approach; Considering all possible $\hat{\mathbf{z}}$ combinations in the sample space $\hat{\mathbf{Z}}$

In the WLSP approach, though we do not know the quality of the measurements, we do have access to the noise realization and as such are able to classify the measurements using a classification rule. This information is, however, not present in real world scenarios. Besides the measurements, the graph topology, and the distribution parameters, no information is known regarding the quality of the measurements and/or the noise terms in the current case. With no information regarding the quality of the measurement, a naive approach would be to examine every possible combination of $\hat{\mathbf{z}}$ and corresponding $\hat{\mathbf{x}}$, and choose as solution the one combination that maximizes the log likelihood $\log L\left(\mathbf{x}\middle|\mathbf{b}\right)_{\text{UK}}$.

We proceed as follows: as we are given a set of M noisy relative measurements, we know that the sample space of $\hat{\mathbf{Z}}$ consists of 2^M elements. This because for each measurement, we have in the current case two possibilities and the measurements are iid. For each element in the sample space $S_{\hat{\mathbf{Z}}}$, we obtain the corresponding estimate of the state vector using the WLS approach. The next step is then to choose as $\hat{\mathbf{x}}$ (and hence implicitly also $\hat{\mathbf{z}}$) the state vector that maximizes $\log L\left(\mathbf{x}\middle|\mathbf{b}\right)_{\text{UK}}$. So, in essense, first, the search space of $\mathbf{x}$ is reduced from $\mathbb{R}^N$ to a set of 2^M vectors, each corresponding to a particular $\hat{\mathbf{z}} \in S_{\hat{\mathbf{Z}}}$ and from this set of 2^M vectors, the one maximizing $\log L\left(\mathbf{x}\middle|\mathbf{b}\right)_{\text{UK}}$ is chosen to be $\hat{\mathbf{x}}$.

$$\hat{\mathbf{x}} = \underset{\mathbf{x}\in\chi}{\arg\max}\ \log L\left(\mathbf{x}\middle|\mathbf{b}\right)_{\text{UK}} \tag{2.65}$$

with χ being the set

$$\chi = \left\{ \tilde{x} \middle| \tilde{x}(\hat{\mathbf{z}}) = \underset{\mathbf{x}\in\mathbb{R}^N}{\arg\max}\ \log L\left(\mathbf{x}\middle|\mathbf{b}\right)_{\hat{\mathbf{z}}} \text{ and } \hat{\mathbf{z}} \in S_{\hat{\mathbf{Z}}} \right\} \tag{2.66}$$

in which the depence on $\hat{\mathbf{z}}$ is explicitly shown. In the exhaustive search approach we have first performed estimation and then classification, as we have first estimated $\hat{\mathbf{x}}$ for each $\hat{\mathbf{z}} \in S_{\hat{\mathbf{Z}}}$ and afterwards chosen the optimal one. Adding the additive constant $c\mathbf{1}$, we obtain

$$\hat{\mathbf{x}}_{BF} = \hat{\mathbf{x}} + c\mathbf{1}. \tag{2.67}$$

As shown in the previous subsections, choosing $c = \text{mean}(\bar{x})$ is a good choice, see algorithm 3.

Algorithm 3 BF Approach

```
Require: Data: $(A, \mathbf{b}, p, \alpha, \beta)$
Initialize:
    $\chi \leftarrow \emptyset$
Obtain all possible $\mathbf{z}$ combinations; Sample space $S_{\hat{\mathbf{Z}}}$
for all $\mathbf{z} \in S_{\hat{\mathbf{Z}}}$ do
    Estimation Step:
    Proceed to algorithm 1 with Data: $(A, \mathbf{b}, \mathbf{z}, p, \alpha, \beta)$
    Store state vector: $\chi \leftarrow \chi \cup \hat{\mathbf{x}}_{\mathbf{z}}$
end for
Classification Step:
    $\hat{\mathbf{x}} = \arg\max_{\mathbf{x} \in \chi} \log L\left(\mathbf{x} \middle| \mathbf{b}\right)_{\mathsf{UK}}$
Additive constant:
    $\hat{\mathbf{x}}_{\mathsf{BF}} = \hat{\mathbf{x}} + c\mathbf{1}, \quad \text{with } c = \text{mean}(\bar{\mathbf{x}}).$
```

2.4.3 EM Approach

As in the brute force approach the sample space $S_{\hat{\mathbf{Z}}}$ consist of 2^M cases to be considered, this will be computationally exhaustive as M grows, hence an alternative approach is needed. The Expectation Maximization (EM) approach is a useful alternative and is considered in this report.

The EM approach can be regarded as a generalization of the MLE approach to the incomplete data case. Incomplete as we lack the necessary information to do estimation directly. It tries to find the solution for $\mathbf{x}$ given the incomplete observed data $\mathbf{b}$. This optimization problem considered for the EM approach is more difficult than the one used for the MLE approach (in the current report, the WLS approach) as it may have multiple local maxima and no closed form solution [26], which is also mentioned previously. In the EM approach the maximization of one difficult likelihood function $L\left(\mathbf{x} \middle| \mathbf{b}\right)_{\mathsf{UK}}$ is replaced by a sequence of simpler maximization [19, 26]; it has an iterative character. In Ref. [27], it is stated that the EM algorithm is only guaranteed to never get worse. This is the monotonicity property of the EM approach and it is in terms of the likelihood. This property alone, however, does not guarantee the convergence of the sequence $\{\mathbf{x}^{(m)}\}$. Under some regularity conditions, it can be proven that the sequence $\{\mathbf{x}^{(m)}\}$ converges to a stationary point (we can think of a local extreme or a saddle point) of the likelihood function. Hence, in the case of convergence, it does not necessary find a global maximum of the likelihood function $L\left(\mathbf{x} \middle| \mathbf{b}\right)$, in particular when the function has multiple peaks.

In the following, we consider the strategy used in EM to replace the difficult likelihood function by a sequence of simpler ones. In order to do that, hidden random variables are introduced to complement the measurement data. In our case, the hidden random variables are the indicator variables. The complete data is then $\mathbf{m} = \left(\mathbf{b}^T, \mathbf{z}^T\right)^T$ with $\mathbf{b}$ being the vector containing the measurements and $\mathbf{z}$ the vector indicating the noise distribution of the measurements. Instead of maximizing $L\left(\mathbf{x} \middle| \mathbf{b}\right)_{\mathsf{UK}}$, we want to find the maximizer of the complete likelihood function, i.e.,

$$\hat{\mathbf{x}} = \arg\max_{\mathbf{x} \in \mathbb{R}^N} L\left(\mathbf{x} \middle| \mathbf{m}\right) = \mathbb{P}\left(\mathbf{m} \middle| \mathbf{x}\right) \tag{2.68}$$

with

$$\mathbb{P}\left(\mathbf{m} \middle| \mathbf{x}\right) = \prod_{e \in \mathcal{E}} \mathbb{P}\left(m_e = (b_e, z_e) \middle| (A\mathbf{x})_e\right) \tag{2.69}$$

and

$$\mathbb{P}\Big(m_e\Big|(A\mathbf{x})_e\Big) = z_e\frac{p}{\sqrt{2\pi\beta^2}}\exp\left(-\frac{\big(b_e-(x_i-x_j)\big)^2}{2\beta^2}\right)+ \\ (1-z_e)\frac{1-p}{\sqrt{2\pi\alpha^2}}\exp\left(-\frac{\big(b_e-(x_i-x_j)\big)^2}{2\alpha^2}\right). \tag{2.70}$$

.

Note, each term is multiplied by an indicator variable. As the indicator vector $\mathbf{z}$ is not known, we do not know the exact expression for $L\Big(\mathbf{x}\Big|\mathbf{m}\Big)$ and hence we replace this by an expected likelihood function given that we know the measurements and an estimate of the state vector. Hereafter, we optimize this expected likelihood function in order to find a next estimate of the state vector. In the following we will again work with the log of the likelihood function. In summary, the following two steps are iterated:

1. Find an estimate of the log-likelihood function

$$\mathbb{E}\left[\log L\Big(\mathbf{x}\Big|\mathbf{m}\Big)\Big|\mathbf{b},\mathbf{x}^{(r)}\right] \tag{2.71}$$

2. Maximize this expectation

$$\mathbf{x}^{(r+1)} = \underset{\mathbf{x}\in\mathbb{R}^N}{\arg\max}\,\mathbb{E}\left[\log L\Big(\mathbf{x}\Big|\mathbf{m}\Big)\Big|\mathbf{m},\mathbf{x}^{(r)}\right] \tag{2.72}$$

The first step can be considered the classification step, which will be clear during the derivation, and the second step the estimation step. In the above, $\mathbf{x}^{(r)}$ is the estimate of the state vector $\bar{\mathbf{x}}$ at iteration r.
From the above, we again have that

$$z_e = \begin{cases} 1 & \text{if } \eta_e \in \mathcal{N}(0,\beta^2) \\ 0 & \text{if } \eta_e \in \mathcal{N}(0,\alpha^2) \end{cases} \tag{2.73}$$

and that if $z_e = 1$, then $(1-z_e) = 0$.

Proposition 2.13 (EM Estimator). *Let $A \in \{0,\pm1\}^{M\times N}$ be the edge-incidence matrix describing the graph topology according to eq.* (2.1), *$\mathbf{b} \in \mathbb{R}^M$ the vector of measurements, $\mathbf{z} \in \{0,1\}^M$ the vector of indicator variables defined as in eq.* (2.73), *and $\alpha,\beta,$ and p be the noise parameters with $0 < \alpha < \beta$, then for each iteration of the optimization problem eq.* (2.68), *we have*

$$\mathbf{x}^{(r+1)} = \Big(A^TW^{(r)}A\Big)^{\dagger}A^TW^{(r)}\mathbf{b}, \tag{2.74}$$

with

$$W^{(r)} = diag\Big(\Big(1-\pi_e^{(r)}\Big)\alpha^{-2}+\pi_e^{(r)}\beta^{-2}\Big) \tag{2.75}$$

and

$$\pi_e^{(r)} = \mathbb{P}\Big(z_e = 1\Big|b_e,\Big(A\mathbf{x}^{(r)}\Big)\Big). \tag{2.76}$$

Proof. By observing that if $z_e = 1$, then $(1-z_e) = 0$, we can obtain the following two separate sums

$$\log L\Big(\mathbf{x}\Big|\mathbf{m}\Big) = \sum_{e\in\mathcal{E}(\mathcal{G})} z_e\log\frac{p}{\sqrt{2\pi\beta^2}}\exp\left(-\frac{\big(b_e-(x_i-x_j)\big)^2}{2\beta^2}\right)+ \\ \sum_{e\in\mathcal{E}(\mathcal{G})}(1-z_e)\log\frac{1-p}{\sqrt{2\pi\alpha^2}}\exp\left(-\frac{\big(b_e-(x_i-x_j)\big)^2}{2\alpha^2}\right).$$

In the estimation step of the EM approach we want to find the expected log likelihood $\mathbb{E}\left[\log L\left(\mathbf{x}\middle|\mathbf{m}\right)\middle|\mathbf{b},\mathbf{x}^{(r)}\right]$. As the measurements $\mathbf{b}$, the distribution parameters α, and β, the parameter p, and the current estimate of the parameter values $\mathbf{x}^{(r)}$ are given, the only unknown in the log likelihood is z_e, hence we need to determine $\mathbb{E}\left[z_e\middle|b_e,\left(A\mathbf{x}^{(r)}\right)_e\right]$. By using the definition of the expectation, we have

$$\begin{aligned}\mathbb{E}\left[z_e\middle|b_e,\left(A\mathbf{x}^{(r)}\right)_e\right] &= 0\cdot\mathbb{P}\left(z_e=0\middle|b_e,\left(A\mathbf{x}^{(r)}\right)_e\right)+1\cdot\mathbb{P}\left(z_e=1\middle|b_e,\left(A\mathbf{x}^{(r)}\right)_e\right)\\ &= \mathbb{P}\left(z_e=1\middle|b_e,\left(A\mathbf{x}^{(r)}\right)_e\right)\\ &= \pi^{(r)}\end{aligned}$$

The above equation is also observed in the classification step of the WLSP approach as $z_e = 1 \Leftrightarrow \mathcal{N}(0,\beta^2)$ and using the Bayes' theorem and with the given information, a value can be easily obtained. Note that in the current case, we have $\pi^{(r)} \in [0,1]$ while in the WLSP approach we π can only take either 0 or 1. The maximization step is similar to the estimation step in the WLS approach and eventually we obtain the following equation:

$$\begin{aligned}\left(\frac{1}{\beta^2}-\frac{1}{\alpha^2}\right)A^T\text{diag}\left(\pi^{(r)}\right)(\mathbf{b}-A\mathbf{x})+\frac{1}{\alpha^2}A^T(\mathbf{b}-A\mathbf{x}) &= 0 \Leftrightarrow\\ A^TW^{(r)}(\mathbf{b}-A\mathbf{x}) &= 0 \Leftrightarrow\\ \mathbf{x}^{(r+1)} &= \left(A^TW^{(r)}A\right)^{\dagger}A^TW^{(r)}\mathbf{b}.\end{aligned}$$

with $W^{(r)} = \text{diag}\left(\left(1-\pi_e^{(r)}\right)\alpha^{-2}+\pi_e^{(r)}\beta^{-2}\right)$. □

The EM approach is iterated until a stopping criterion, for example $\left\|\hat{\mathbf{x}}^{(r+1)}-\hat{\mathbf{x}}^{(r)}\right\| < \text{tol}$ with $\text{tol} > 0$, for convergence is met or until a pre-specified number of iterations is reached. For the final estimate, we have

$$\hat{\mathbf{x}}^{(\infty)} = \left(A^TW^{(\infty)}A\right)^{\dagger}A^TW^{(\infty)}\mathbf{b}, \tag{2.77}$$

with the superscript ∞ indicating the final estimate. Adding the additive constant, we have

$$\hat{\mathbf{x}}_{EM}^{(\infty)} = \hat{\mathbf{x}}^{(\infty)} + c\mathbf{1}. \tag{2.78}$$

2.4.3.1 EM Implementations

In the literature, we learn that the EM algorithm has two main drawbacks, that of slow convergence and also the dependence of the solution on both the initial condition and the stopping criterion used. Hence, this makes it interesting to compare different implementation of the EM algorithms. In Ref. [28, 29], the problem of choosing initial values for the EM algorithm is considered. In the current work, two of the algorithms mentioned in [28] will be considered, that of random initialization, EM-RAND-FI, in which the EM algorithm is run for different randomly chosen initial positions and the solution is chosen to be the one giving the largest value for the log likelihood function and the em-EM (EM-em-FI) approach in which first small runs of the EM approach from different initial positions are performed and then a long run of the EM approach is considered using the initial position which yields the highest log likelihood value for the small runs. For these approaches, the stopping criterion is a fixed number of iterations. Also, a random initialization approach is considered in which as stopping criterion is taken

$$\|\mathbf{x}_{\text{New}}-\mathbf{x}_{\text{Cur}}\| < \text{TOL} \quad \text{with } \text{TOL} > 0. \tag{2.79}$$

This is denoted as EM-RAND-SC approach. Within each approach, comparison will also be made between running the approach once or multiple times, in [28] termed as to be a seach/run/select strategy. In the following, these algorithms are given.

Algorithm 4 EM-Algorithm

```
**procedure** EM($A, \mathbf{b}, p, \alpha, \beta, \mathbf{x}^{(r)}$)
    Expectation step:
    **for all** $e \in \mathcal{E}(\mathcal{G})$ **do**
        $$\pi_e^{(r+1)} = \frac{\frac{p}{\sqrt{2\pi\beta^2}} \exp\left(-\frac{\left(b_e - \left(x_i^{(r)} - x_j^{(r)}\right)\right)^2}{2\beta^2}\right)}{\frac{p}{\sqrt{2\pi\beta^2}} \exp\left(-\frac{\left(b_e - \left(x_i^{(r)} - x_j^{(r)}\right)\right)^2}{2\beta^2}\right) + \frac{1-p}{\sqrt{2\pi\alpha^2}} \exp\left(-\frac{\left(b_e - \left(x_i^{(r)} - x_j^{(r)}\right)\right)^2}{2\alpha^2}\right)}.$$
    **end for**
    Obtain Weights:
    **for all** $e \in \mathcal{E}(\mathcal{G})$ **do**
        $$w_e^{(r+1)} = \frac{\left(1 - \pi_e^{(r+1)}\right)}{\alpha^2} + \frac{\pi_e^{(r+1)}}{\beta^2}$$
    **end for**
    Maximization step:
        $$\mathbf{x}^{(r+1)} = \left(A^T W^{(r+1)} A\right)^{\dagger} A^T W^{(r+1)} \mathbf{b}$$
**end procedure**
```

Algorithm 5 EM-RAND-SC Approach

```
**Require:** Data: $\left(A, \mathbf{b}, p, \alpha, \beta, \mathrm{TOL}, \mathbf{x}^{(0)}\right)$
**Initialize:**
    SC $\leftarrow$1, $t \leftarrow$0
**while** SC $>$ TOL **do**
    Perform algorithm 4 with inputs $\left(A, \mathbf{b}, p, \alpha, \beta, \mathbf{x}^{(t)}\right)$
    SC $\leftarrow \|\mathbf{x}_{t+1} - \mathbf{x}_t\|$, $t \leftarrow t+1$
**end while**
Additive constant
    $$\mathbf{x}_{\text{EM-RAND-SC}}^{(\infty)} = \mathbf{x}^{(\infty)} + c\mathbf{1} \quad \text{with } c = \text{mean}(\bar{x})$$
```

Algorithm 6 EM-RAND-FI Approach

```
Require: Data: $\left(A, \mathbf{b}, p, \alpha, \beta, \mathbf{x}^{(0)}, \text{MaxIter}\right)$
for $t = 1, \ldots, \text{MaxIter}$ do
    Perform algorithm 4 with inputs $\left(A, \mathbf{b}, p, \alpha, \beta, \mathbf{x}^{(t-1)}\right)$
end for
Additive constant
        $\mathbf{x}_{\text{EM-RAND-FI}}^{(\text{MaxIter})} = \mathbf{x}^{(\text{MaxIter})} + c\mathbf{1} \quad \text{with } c = \text{mean}(\bar{x})$
```

Algorithm 7 EM-em-FI Approach

```
Require: Data: $(A, \mathbf{b}, p, \alpha, \beta, \text{MaxIter}, \text{NI}, \text{TOL}_{\text{em}})$
for $n = 1, \ldots, \text{NI}$ do
    Initialize:
    $\text{SC}_{\text{em}} \leftarrow 1$, $t \leftarrow 0$, $\mathbf{x}_n^0 = \text{rand}(N, 1)$
    while $\text{SC}_{\text{em}} > \text{TOL}_{\text{em}}$ do
        Perform algorithm 4 with inputs $\left(A, \mathbf{b}, p, \alpha, \beta, \mathbf{x}_n^{(t)}\right)$
        $\text{SC}_{\text{em}} \leftarrow \frac{\log L\left(\mathbf{x}_n^{(t+1)} \middle| \mathbf{b}\right)_{\text{UK}} - \log L\left(\mathbf{x}_n^{(t)} \middle| \mathbf{b}\right)_{\text{UK}}}{\log L\left(\mathbf{x}_n^{(t+1)} \middle| \mathbf{b}\right)_{\text{UK}} - \log L\left(\mathbf{x}_n^{(0)} \middle| \mathbf{b}\right)_{\text{UK}}}$
        $t \leftarrow t + 1$
    end while
end for
Choose $\mathbf{x}_n$ with highest LLH; use this for the long run.
for $t = 1, \ldots, \text{MaxIter}$ do
    Perform algorithm 4 with inputs $\left(A, \mathbf{b}, p, \alpha, \beta, \mathbf{x}^{(t-1)}\right)$
end for
Additive constant
        $\mathbf{x}_{\text{EM-RAND-FI}}^{(\text{MaxIter})} = \mathbf{x}^{(\text{MaxIter})} + c\mathbf{1} \quad \text{with } c = \text{mean}(\bar{x})$
```

3. Numerical Results

3.1 Overview

In the current chapter, the approaches mentioned in chapter 2 are applied to specific graph topologies. We consider the path, star, cycle, wheel, complete and the random graph model by Erdos and Rényi, see section 3.2.

In section 3.3, the approximate approach for calculating the MSE of the WLSP estimator is considered; also, a choice is made for the EM-implementation to be used in the current work.

Hereafter, we will consider trees (the path and star graphs are examples of trees) in section 3.4 followed by the consideration of cycles in section 3.5. We compare the different topologies for a five node graph in section 3.6. A parameter study is also conducted on a graph consisting of 10 nodes with a cycle, complete and random topology, see section 3.7.

3.2 Graph Configurations Considered

For the numerical simulations, the following graph configurations are considered: path, star, cycle, wheel, and the complete graph, see fig. 3.1. In addition, the Erdos-Rényi random graph model is also considered, see fig. 3.2 for an realization.

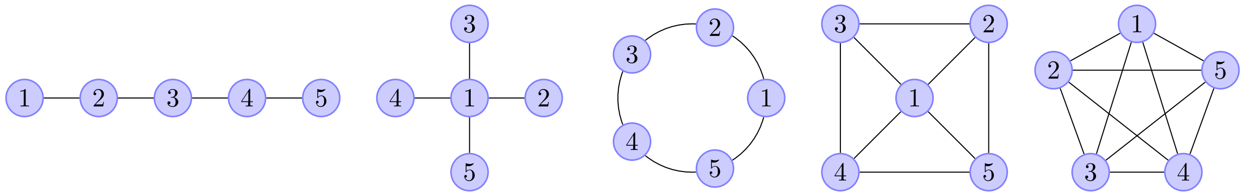


Figure 3.1: 5 nodes graph configurations; from left to right, we have the path, star, cycle, wheel and the complete graph configuration.

Path and Star Graph The path and star graph configuration are examples of trees. A tree is a connected, undirected graph containing no loops. Trees are interesting to study for the following reasons. One, there is exactly one path between any pair of vertices. Two, for it to be connected, a tree of N vertices always has exactly $N-1$ edges. This number is the lower bound for a graph to be connected. As a consequence, a connected graph with the minimum number of edges is always a tree.

An interesting observation for the path and star graph is that the edge-incidence matrix associated with it is of full row rank. This leads to the following proposition

Proposition 3.1. *Let $A \in \{0, \pm 1\}^{N-1\times N}$ be the edge-incidence matrix associated to a tree type graph, then the state estimate can be obtained by the following*

$$\hat{\mathbf{x}} = A^{\dagger}\mathbf{b} + c\mathbf{1}. \tag{3.1}$$

Proof. In each of the algorithms that is considered in this report, for the estimation step we are essentially calculating the following equation

$$\hat{\mathbf{x}} = \left(A^T W_\gamma A\right)^{\dagger} A^T W_\gamma \mathbf{b} + c\mathbf{1}$$

with γ being $\bar{\mathbf{z}}$ for the WLS approach and $\hat{\mathbf{z}}$ for the WLSP, BF, and EM approach. c is taken to be mean$(\bar{\mathbf{x}})$. Using property 7 of the Moore-Penrose pseudoinverse in appendix B and proposition 2.5, we can obtain

$$\hat{\mathbf{x}} = \left(A^T W_\gamma A\right)^{\dagger} A^T W_\gamma \mathbf{b} + c\mathbf{1} = A^{\dagger} W_\gamma^{-1} \left(AA^T\right)^{-1} AA^T W_\gamma \mathbf{b} + c\mathbf{1} = A^{\dagger}\mathbf{b} + c\mathbf{1}.$$

□

From proposition 3.1, we may inferred that all the approaches will yield the same estimate and hence the classification step has no influence on the state estimation. It is though needed for calculating the MSE considered in the WLS and the WLSP approach. But as both approaches yield the same result, we may also infer that the MSE or the covariance matrix for the WLS and the WLSP approach will be the same. Another consequence is that for the EM algorithm, eventhough we have no analytical results regarding its properties, we may state that it is unbiased and yields an MSE that is the same as for the WLS approach for tree types graph.

Cycle Graph A simple cycle graph can be obtained by connecting the first and the last vertex of a path graph. In a cycle graph, the vertices all have the same number of edges, which is two, connected to them. It is an instance of a 2-regular graph. The number of edges for a cycle equals the number of nodes, which is N.

Wheel Graph A wheel graph can be observed as to be a combination of a star graph of N nodes and a cycle graph of $N-1$ nodes. The number of edges for a wheel graph is $2N-1$ with N being the number of nodes.

Complete Graph In a complete graph of N vertices, from each vertex, there are $N-1$ edges, each connected to one of the remaining vertices in the graph. A graph in which all vertices have the same degree is called a regular graph. A k-regular graph is one in which the degree of all the vertices is k. The complete graph is hence a $(N-1)$-regular graph. The number of edges in a complete graph equals $\binom{N}{2} = \frac{1}{2}N(N-1)$.

Random Graph Model The Erdos-Rényi (ER) random graph model considered in the current report is of the type $\text{ER}(N, p_e)$ with N being the number of nodes of the graph and p_e the probability of having an edge between each pair of vertices, independent of the rest. For each realization, a different graph topology is obtained with the fixed number of nodes. It is known that the ER model yields with high probability a connected graph if $p_e = c \log N/N$ with $c > 1$ [30].

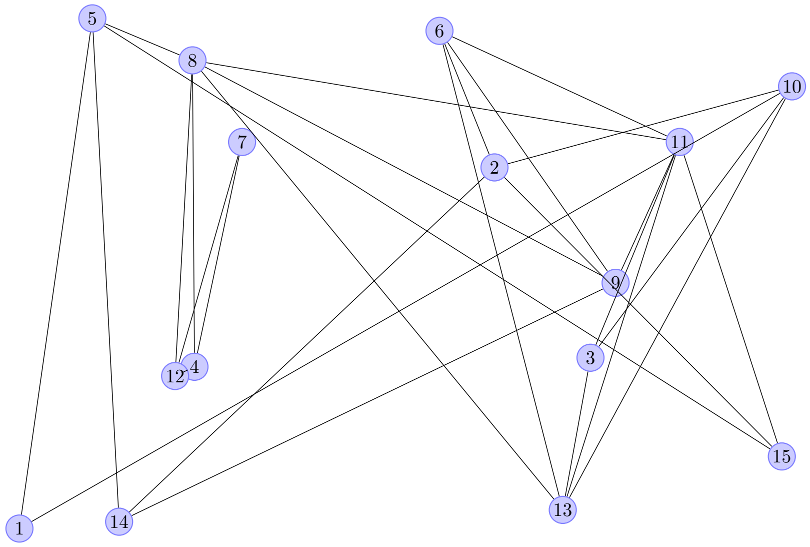


Figure 3.2: A realization of a Erdos-Rényi graph consisting of 15 nodes and $p_e = 0.3$. The number of edges for the current realization is 28 with vertices having a (unweighted) degree in the range $2-6$.

3.3 Implementation

3.3.1 Comparison of the Exact and the Approximate Approach for MSE of WLSP Estimator

In this subsection, we consider the performance of the heuristic approach mentioned in section 2.4.1.4. Simulations are carried out for different configurations of a graph consisting of 5 nodes. In particular, the path (P), star (S), circle (C), wheel (W) and the complete (K) graph configuration are considered. For the simulation, the following is assumed $\alpha = 0.1, \beta/\alpha = 5, p = 0.25$. Threshold for ϵ are taken to be $10^{-2}, 10^{-4}, 10^{-6}$. In fig. 3.3 and fig. 3.4, the mean proportion of $\mathbf{z}$ combinations considered and the coverage using those $\mathbf{z}$ combinations for the MSE of WLSP are presented.

The coverage is defined to be the total sum of the contribution of the $\mathbf{z}$ combinations which are larger than the threshold ϵ to $\mathbb{E}_{\mathbf{H}}\left[(\hat{\mathbf{x}}_{WLSP} - \bar{\mathbf{x}})(\hat{\mathbf{x}}_{WLSP} - \bar{\mathbf{x}})^T \middle| \bar{\mathbf{Z}} = \mathbf{z}\right]$ as we know the latter is a weighted sum. The overall coverage is obtained as to be a weighted sum of the coverage for each given term of $\bar{\mathbf{Z}}$. The mean proportion of $\mathbf{z}$ combinations is the mean of the number of $\mathbf{z}$ combinations considered for each given term of $\bar{\mathbf{Z}}$.

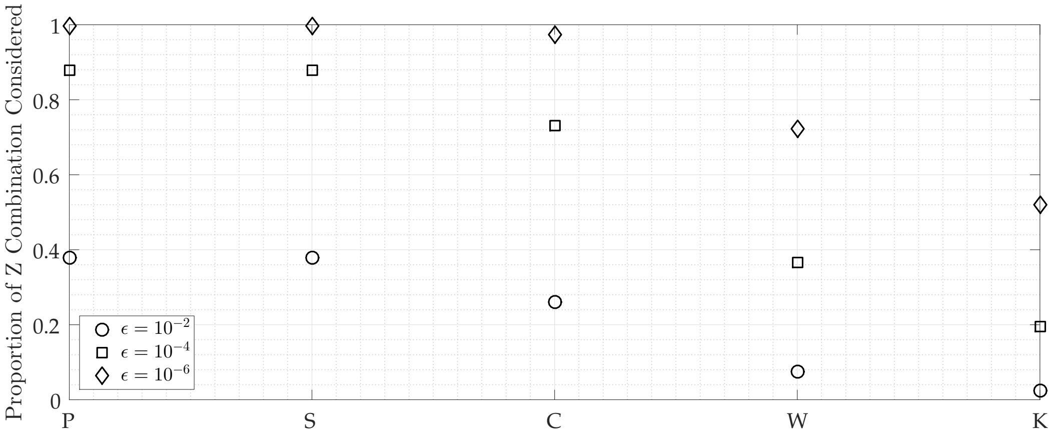


Figure 3.3: Proportion of $\mathbf{z}$ combinations considered for the approximation of the MSE $\mathbb{E}_{\bar{\mathbf{Z}},\mathbf{H}}\left[(\hat{\mathbf{x}}_{WLSP} - \bar{\mathbf{x}})(\hat{\mathbf{x}}_{WLSP} - \bar{\mathbf{x}})^T\right]$ of the WLSP estimator for the different graph configurations with the following simulation parameters: $N = 5, \alpha = 0.1, \beta/\alpha = 5, p = 0.25$.

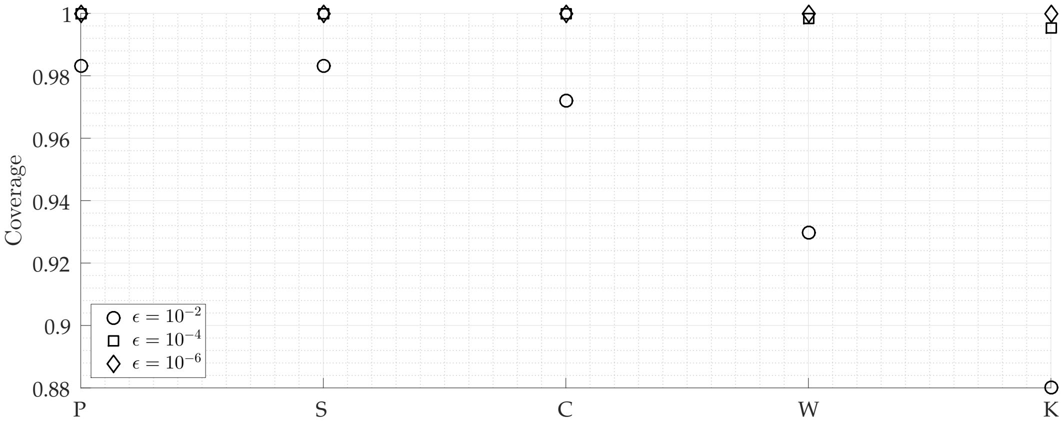


Figure 3.4: Coverage of the MSE $\mathbb{E}_{\bar{\mathbf{Z}},\mathbf{H}}\left[(\hat{\mathbf{x}}_{WLSP} - \bar{\mathbf{x}})(\hat{\mathbf{x}}_{WLSP} - \bar{\mathbf{x}})^T\right]$ of the WLSP estimator for the different graph configurations with the following simulation parameters: $N = 5, \alpha = 0.1, \beta/\alpha = 5, p = 0.25$.

It can be readily observed from the figures that a smaller value for ϵ leads to the inclusion of a larger number of $\mathbf{z}$ combinations for the approximation and as such a better coverage which is according to intuition. In fig. 3.3, it can be observed that for the complete graph configuration, only half the number of combinations are considered for the approximation when ϵ is set to 10^{-6}. Despite this, the coverage can be considered very good as the value is almost one, see fig. 3.4. Choosing $\epsilon = 10^{-4}$, leads to another cut of more than half for the number of combinations considered, albeit with a small decrease in the coverage as the price to paid.

From this small analysis, it can be concluded that the heuristic mentioned in section 2.4.1.4 is applicable for small graphs. However, it should be noted that for larger graphs which consist of a larger number of edges, stricter values for ϵ needs to be considered in order to have a good approximation. An example is $\epsilon = 10^{-2}$ for the five nodes graph. In fig. 3.4, it can be observed that the decrease in coverage is significant when more edges are added to the same set of nodes. This trend is likely to occur for a larger set of nodes with a large number of connections between those nodes.

3.3.2 Comparison of EM Implementations

As mentioned in section 2.4.3.1, in the current work three EM-type implementations are considered; the EM-RAND-FI and the EM-em-FI approach in which the number of iterations is fixed, and the EM-RAND-SC approach in which a stopping criterion is needed. Also, it is mentioned whether we take several runs and choose the best one or we just run the algorithm starting from one initial position. In the current subsection, we choose the EM algorithm that will be used for further simulations.

First, in order to choose the value for tol use for the EM-RAND-SC approach, we run the EM-RAND-FI and the EM-em-FI approach with a fixed number of iterations. We consider for the current case a complete graph of five nodes and also the following parameters, $\alpha = 0.1, \beta/\alpha = \{5, 10, 15, 20\}, p = \{0.25, 0.4\}$. A simulation of 10000 trials is run in which for each trial resampling of the measurement quality is done, i.e., $\bar{\mathbf{Z}}$ is random. For the 10000 trials considered, we determine for a fixed number of iteration, how many of those trials have a value that is less than a specified threshold value for the following norm $\|\mathbf{x}_{\text{New}} - \mathbf{x}_{\text{Cur}}\|$ used in the EM-RAND-SC approach. In the plots given, the iteration is set to 20 and tol $= 10^{-8}$.

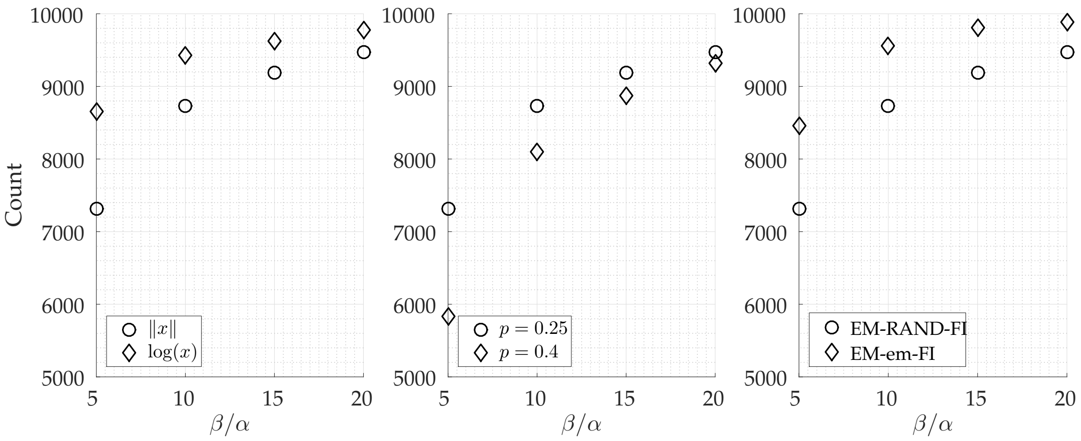


Figure 3.5: Evaluation of EM-RAND-FI and EM-em-FI approach for a five node complete graph. The number of trials that satisfies a threshold value for $\|\mathbf{x}_{\text{New}} - \mathbf{x}_{\text{Cur}}\|$ are counted. The simulation parameters are $\alpha = 0.1, \beta/\alpha = \{5, 10, 15, 20\}, p = \{0.25, 0.4\}, \text{NT} = 10000, \text{tol} = 10^{-8}$; Left: a comparison between the 2-Norm of the difference in $\mathbf{x}$ and the log value for EM-RAND-FI aproach. Center: comparison for a variable p for the EM-RAND-FI approach; Right: comparison between the EM-RAND-FI and the EM-em-FI approach for $p = 0.25$.

We will give the observations starting from the left plot in fig. 3.5. It is observed that the number of trials for which the difference in the log likelihood value meets the threshold value is greater than the difference in the 2-norm. This is not unexpected as from literature it is known that the EM approach have rapid convergence in the likelihood [31]. With an increasing value for β/α both sequences increase. In the center plot, we give the EM-RAND-SC approach for 2 different values of p. It is observed that for an increasing value of p, the value drops. This is reasonable as a larger value for p indicates a larger proportion of 'bad' measurements in the set and as such in order to have the sequence $\mathbf{x}^{(r)}$ converges, a larger number of iterations is needed. Note in the plots, the iteration number is fixed to 20. For the right plot, it is observed that the EM-em-FI approach yields better result than the EM-RAND-FI approach. A possible explanation is due to the wider range for searching an appropriate initial position $\mathbf{x}^{(0)}$ in the EM-em approach. Note the number of runs considered is one for both approaches but for the inner run of EM-em-FI, we consider a number of 10, i.e., we choose 10 initial positions for the short run and choose for the long run the initial position yielding the largest log likelihood value. The result observed here is in agreement with [28].

From the above it can be observed that when the goal is to have the 2-norm of the difference for the $\mathbf{x}^{(r)}$ sequence to be smaller than a threshold value, the number of iterations is sensitive to the distribution parameters, as can be observed in the center plot. As our goal is to have convergence for the $\mathbf{x}^{(r)}$ sequence, these two algorithms are not considered further.

Next, we want to determine whether it is beneficial to consider several runs for the EM-RAND-SC approach or just a single run. In the literature it is mentioned that starting the EM-algorithm from different initial positions is beneficial [27, 28]. We ran the simulation with the noise parameters as mentioned previously. We consider a case of running the EM-RAND-SC approach with 10 initial positions. For the comparisons, we partition the 10 trials considered in two 'classes', one containing one trial and the other class the remaining trials. We determine which class yields the largest log likelihood value for eq. (2.35). In the following, this is plotted, see fig. 3.6.

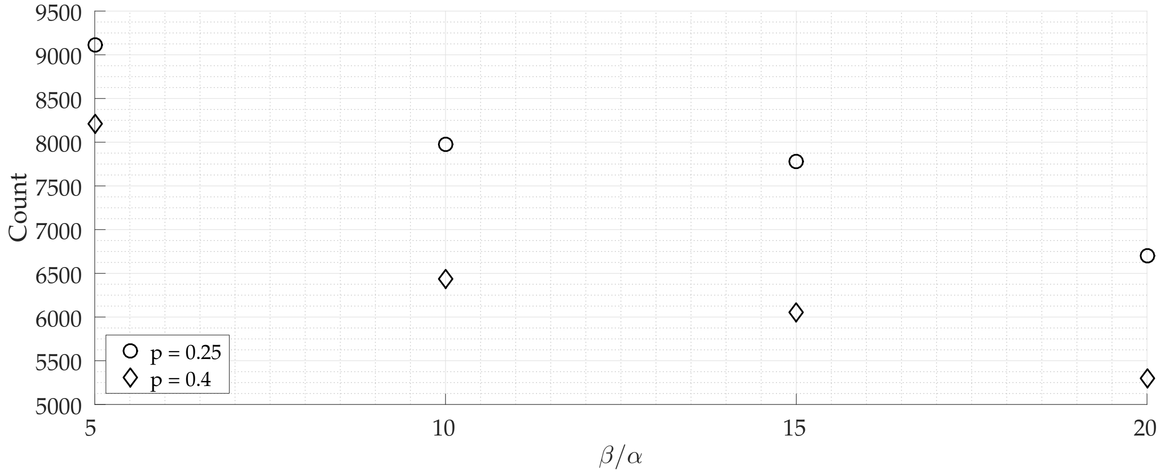


Figure 3.6: Evaluation of EM-RAND-SC considering whether to have only one run or several runs. The simulation is done for a five node complete graph. The simulation parameters are $\alpha = 0.1, \beta/\alpha = \{5, 10, 15, 20\}, p = \{0.25, 0.4\}, \mathrm{NT} = 10000, \mathrm{tol} = 10^{-8}$; In the plot, the number of trials for which the log likelihood value of the one run is greater than or equal to the log likelihood value of the many runs is given.

From the figure we can observed that with an increasing value for β/α and also for p, the number of trials for which the one run yield the same or a higher log likelihood value is decreasing. Hence, from this it is clear that starting from different random initial positions yields a higher value for the log likelihood function.

In the remainder of this section, we will thus use the EM-RAND-SC approach with $\mathrm{tol} = 10^{-8}$ and we run the EM-RAND-SC approach 5 times before choosing the final solution.

3.4 Trees

In the current subsection, we present the observations when the algorithms are applied to tree type graphs which are the path and the star graph considered in this report. As already obtained in proposition 3.1, we will have the same state estimate regardless of the approach taken. This is also observed from the simulation. Also the MSE for the approaches, both analytically and numerically, are the same as the WLS approach. A possible explanation for this behavior may be due to the fact that between each pair of nodes there is only one single path available and hence they are considered equally important for the estimation; we can say that the WLS has turned into a Ordinary Least Squares estimate. As a consequence of this, the log likelihood can be determined using only the noise distribution parameters as the exponential term for each term is ≈ 1. This because the difference $\mathbf{b} - A\hat{\mathbf{x}}$ for all the measurements are observed to be very small (range $< 10^{-13}$). The loglikelihood function thus has the following form

$$\log L\Big(\mathbf{x}\Big|\mathbf{b}\Big) = \sum_{e\in\mathcal{E}(\mathcal{G})} \log \frac{1}{\sqrt{2\pi\sigma_e^2}} = \log\left(\frac{1}{\sqrt{2\pi\alpha^2}}\right)^{\#0} + \log\left(\frac{1}{\sqrt{2\pi\beta^2}}\right)^{\#1} \quad \text{with } \#0 + \#1 = M. \tag{3.2}$$

The second equality is obtained as the exponential term in $\mathbb{P}\Big(b_e\Big|(A\mathbf{x})_e\Big)$ is very close to one and the third by clustering the zeros and ones of $\bar{\mathbf{z}}$.

$$\log L\Big(\mathbf{x}\Big|\mathbf{b}\Big)_{\text{UK}} = \log\left(\frac{1-p}{\sqrt{2\pi\alpha^2}} + \frac{p}{\sqrt{2\pi\beta^2}}\right)^M \tag{3.3}$$

in which we also use the fact the exponential term ≈ 1 and $a \log b = \log b^a$. Note: $\log L\Big(\mathbf{x}\Big|\mathbf{b}\Big)$ depends on the $\bar{\mathbf{z}}$ combination and hence can be different when the case $\bar{\mathbf{Z}}$ is random is simulated; for $\log L\Big(\mathbf{x}\Big|\mathbf{b}\Big)_{\text{UK}}$, this is not the case and hence one fixed value (up to a number of digits) is observed.

3.5 Cycles

For the cycle graph, the following interesting fact is observed by the brute force approach. Therein, we observe that different combination of $\hat{\mathbf{z}}$ and $\hat{\mathbf{x}}$ yields the same log likelihood value. Comparing the log likelihood values for the whole sample space of $\hat{\mathbf{Z}}$, we observe that M distinct values can be obtained, each corresponding to the number of ones that are present in a particular $\mathbf{z}$ combination (Note: the combination all 'good' and all 'bad' yields the same log likelihood value). For this, at the moment, no clear motivation can be given. This behavior regarding the log likelihood values is not observed for the random, wheel and complete graph configuration.

3.6 Five Nodes Graphs

In the current section, we compare the performance of the considered approaches applied to different topologies for a graph consisting of five nodes. The graph topologies considered are the path, star, cycle, wheel, complete graph and two realization of the Erdos Renyi graph; one is a tree and the other one has 7 edges. $p_e = 0.4$ is chosen for the realization. As performance metric, the normalized mean squared error is chosen which is defined as follows

$$\begin{aligned} J &= \frac{1}{N}\mathbb{E}\left[\|\hat{\mathbf{x}}^\bullet - \bar{\mathbf{x}}\|^2\right] \\ &= \frac{1}{N}\text{trace}(P). \end{aligned} \quad \text{with } P \text{ being the covariance matrix (the MSE given in the previous chaper).} \tag{3.4}$$

An approximation of eq. (3.4) is the following:

$$J_{\text{est}} = \frac{1}{\text{NT}} \frac{1}{N} \sum_{i=1}^{\text{NT}} \sum_{j=1}^{N} \left(x_{ij} - \left(\frac{1}{\text{NT}} \sum_{i=1}^{\text{NT}} x_{ij} \right) \right)^2. \tag{3.5}$$

We first observe the performance of eq. (3.5) with that of eq. (3.4) for the WLS, WLSP approach, see fig. 3.8. In the plot, the ordinary least squares approach, LS, in which all measurements are considered to be of equal quality, is also given. The LS approach is considered, as we regard it to be a reference, upon which we want to improve in the current work.

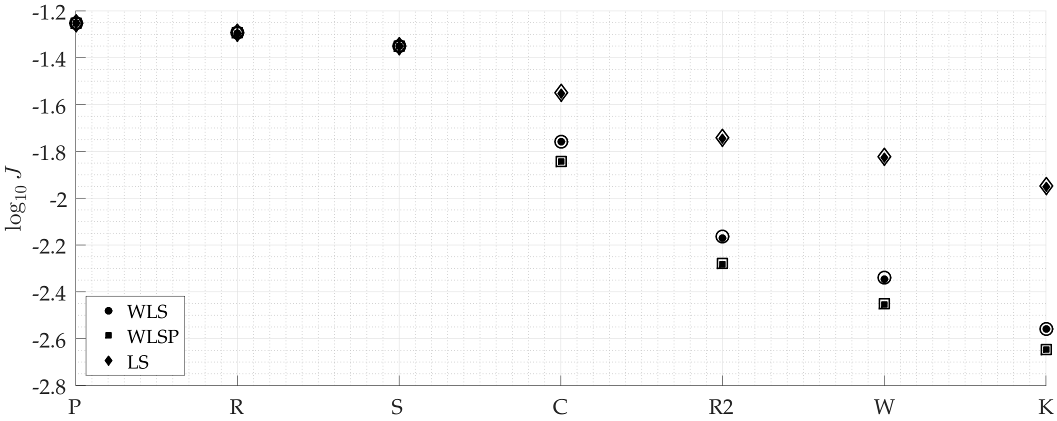


Figure 3.7: Comparison of J and J_{est} for the WLS, WLSP, and the LS approach for the different graph topologies consisting of five nodes. The closed symbols are values obtained using eq. (3.4) and the open symbols are values obtained numerically, using eq. (3.5). The simulation parameters are $\alpha = 0.1, \beta/\alpha = 5, p = 0.25, \text{NT} = 50000$ and $\bar{\mathbf{Z}}$ is random.

From the figure, we can observe that the value obtained using the simulation data (open symbol) and the one calculated analytically (closed symbol) are in agreement with each other. Furthermore, it can be observed that for trees, the value is the same for all three approaches WLS, WLSP, and LS approach by proposition 3.1. Next, we give the plot in which the WLS, WLSP, LS, and the EM-SC approach are compared against each other.

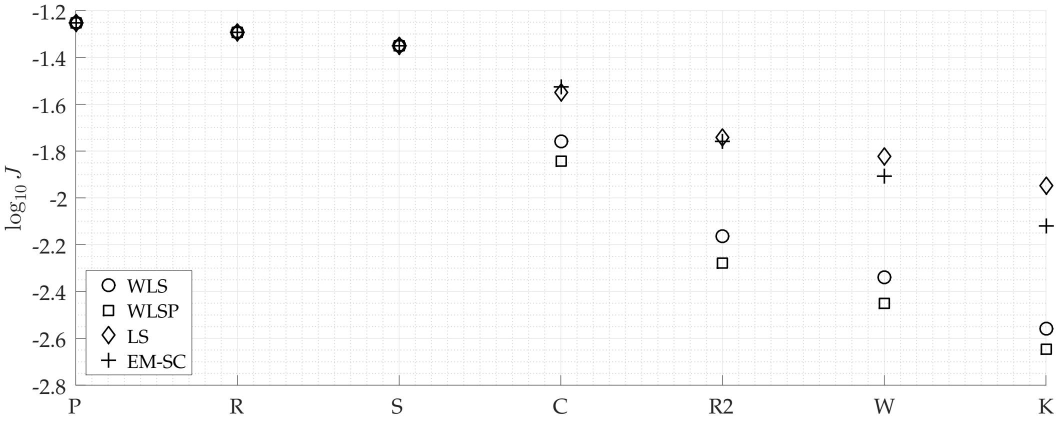
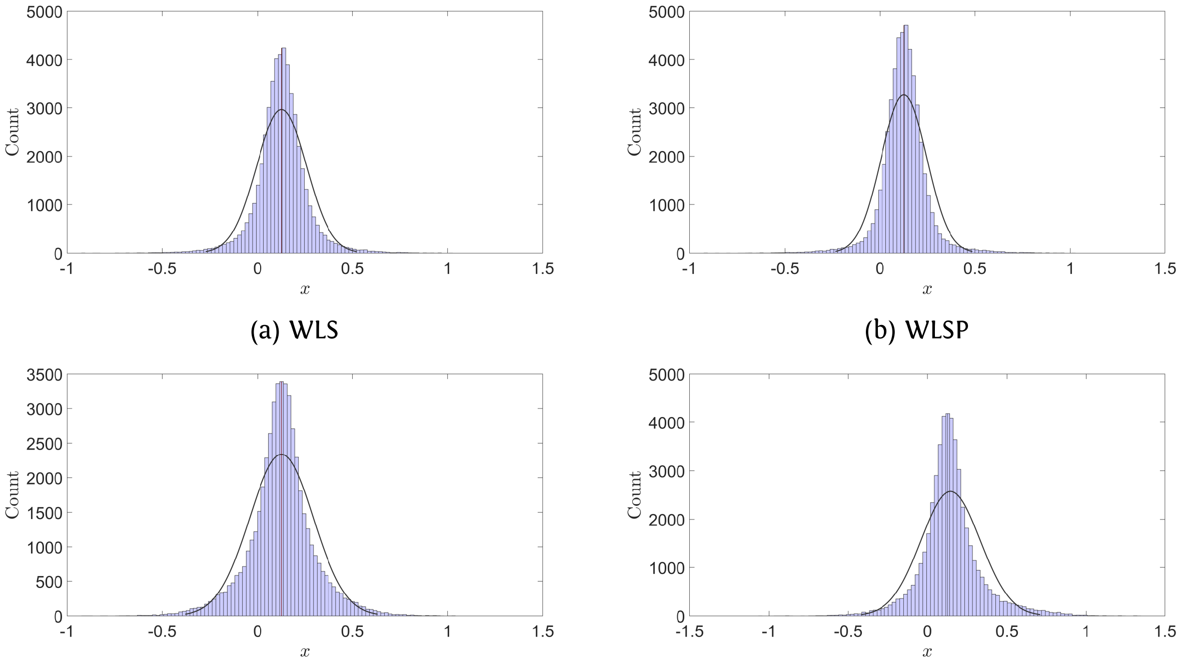


Figure 3.8: Comparison of J_{est} for the WLS, WLSP, EM-RAND-SC, and the LS approach for the different graph topologies consisting of five nodes. The simulation parameters are $\alpha = 0.1, \beta/\alpha = 5, p = 0.25, \text{NT} = 50000$ and $\bar{\mathbf{Z}}$ is random.

From the plot, we can observe that having a more connected graph yields a smaller value for J. Also observe is that the WLSP approach outperforms the other approaches, and that the EM-

RAND-SC and the LS approach are quite similar regarding the performance, with the EM-RAND-SC approach yielding better results as the graph is more connected (and thus the number of edges is increasing). The results obtain herein, is a motivation for considering the problem of adding edges to an existing network, which is the subject of part II.

In addition to considering the performance, we also looked at the distribution of the state variables for the different approaches. In the WLS approach, we can observe that the state estimate is a weighted sum of the measurements with each being a gaussian distribution. From theory, we know that the sum of a Gaussian distribution is also Gaussian, hence, we are interested in whether the state variables are also distributed in a gaussian manner. In the following, the distribution of x_3 is given for all the approaches considered. We have considered the circle graph. For the histogram, the number of bins is chosen to be 100.



(a) WLS

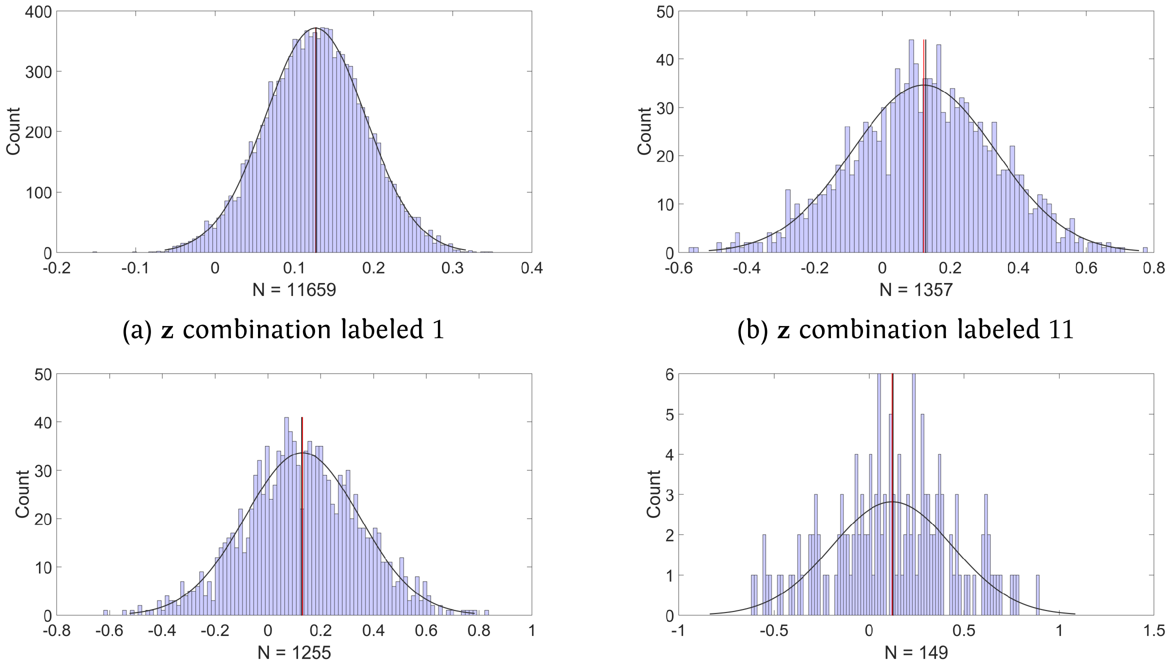
(b) WLSP

(c) LS

(d) EM-SC

Figure 3.9: Distribution of x_3 based on the data for the approaches. The simulation parameters are $\alpha = 0.1, \beta/\alpha = 5, p = 0.25, \mathrm{NT} = 50000$ and $\bar{\mathbf{Z}}$ is random. The red line is mean of the data and the black line the real value $\bar{x}_3$

From the plots, it can be observed that $\hat{x}_3$ is not a normal distribution. From eq. (2.21) and eq. (2.62), we know that the the MSE of the XWLS and also XWLSP is a weighted sum of the $\bar{\mathbf{Z}}$ combinations. Hence, the following is done. We partition the 50000 trials in subsets, each with a common $\bar{\mathbf{z}}$ combination and within these subsets, observe the distribution of $\hat{x}_3$. In the following, some plots are presented in fig. 3.10.



(a) $\mathbf{z}$ combination labeled 1

(b) $\mathbf{z}$ combination labeled 11

(c) $\mathbf{z}$ combination labeled 21

(d) $\mathbf{z}$ combination labeled 31

Figure 3.10: Distribution of x_3 based on the data for the WLS approach. We consider different $\mathbf{z}$ combinations. The simulation parameters are $\alpha = 0.1, \beta/\alpha = 5, p = 0.25, \text{NT} = 50000$ and $\bar{\mathbf{Z}}$ is random. The red line is mean of the data and the black line the real value $\bar{x}_3$

It can be observed that the subsets are normally distributed judging from the plots. Only in the last plot is the normal distribution not obvious, but this is due to the number of trials that are within the subset for determining in a numerical manner the distribution parameters considered. Note the plots are shown only for the WLS approach. No detailed analysis is done for the WLSP approach. This for in the WLSP approach, at the level of one fixed given $\bar{\mathbf{z}}$ combination, we are already taking sums to obtain the MSE, hence we expect those distributions to not be normally distributed.

An observation that has not yet been mentioned is that the mean of the distribution(s) is close and in some plots not distinguishable from the original value x_3, see again the plots in fig. 3.9 and fig. 3.10; this shows that the estimators considered are unbiased, even the EM type estimator for which no analytical evaluation is done.

Another interesting observation considers the diagonal entries of the covariance matrix (the covariance matrix is the MSE calculated) obtained for the WLS and the WLSP estimators. As we know, the diagonal entries of the covariance matrix are the variance of the state estimates. Based on the topologies, we observe that the variance for the case in which $\bar{Z}$ is considered random, has at least a connection with the placement of the vertex and the degree of it. This is motivated by the following; for the cycle and the complete graph which are both graphs for which each vertex has the same number of edges, called regular graphs, the variances for the different variables are the same. It is also observed that for the wheel graph and the star graph in which all except one vertex has the same number for the degree, their variances are also exactly the same and the one vertex with a (in this case) higher degree has a lower variance which is proportional to the variance of the other nodes. For the path graph, however, it is observed that the center node has a smaller variance compared to the direct neighbors which are assumed not to be the end vertices. This is remarkable, as the center node and both its direct neighbors have the same number of edges. Hence, the placement of the vertex also plays a role for determining an expected value for the uncertainty in the state estimates. The values obtained for the five nodes graph are given in table 3.1

	WLS					WLSP				
Path	0.0840	0.0420	0.0280	0.0420	0.0840	0.0840	0.0420	0.0280	0.0420	0.0840
Random Tree	0.0196	0.0616	0.0756	0.0616	0.0336	0.0196	0.0616	0.0756	0.0616	0.0336
Star	0.0112	0.0532	0.0532	0.0532	0.0532	0.0112	0.0532	0.0532	0.0532	0.0532
Cycle	0.0174	0.0174	0.0174	0.0174	0.0174	0.0143	0.0143	0.0143	0.0143	0.0143
Random	0.0056	0.0056	0.0113	0.0056	0.0056	0.0043	0.0043	0.0089	0.0043	0.0043
Wheel	0.0031	0.0049	0.0049	0.0049	0.0049	0.0025	0.0038	0.0038	0.0038	0.0038
Complete	0.0028	0.0028	0.0028	0.0028	0.0028	0.0023	0.0023	0.0023	0.0023	0.0023

Table 3.1: Table with the variances for the state estimates calculated analytically for the WLS and the WLSP approach; Parameters are $\alpha = 0.1, \beta/\alpha = 5, p = 0.25, N = 5$.

3.7 Parameter Study for a Ten Nodes Graphs

We have also carried out a parameter study, this time on a ten nodes graph. The circle, a random generalization with $p_e = 0.4$, and the complete graph are considered. In the first set of plots, the ratio β/α is fixed to be 5 and $p = 0.1 : 0.025 : 0.5$, see fig. 3.11 to 3.13, while in the second set of plots, the roles are turned. Here p is fixed to be 0.25 and $\beta/\alpha = (1 : 1 : 16) \cdot 2.5$, see fig. 3.14 to 3.15. The other parameters are $\alpha = 0.1, \mathsf{NT} = 100000$.

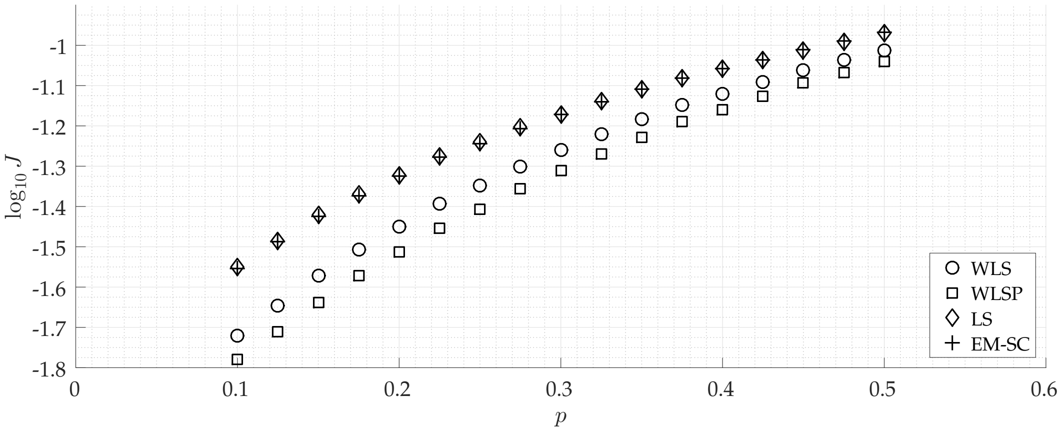


Figure 3.11: $\log_{10} J$ plotted against p for a 10 nodes graph with a circle configuration; Parameters: $\alpha = 0.1, \beta/\alpha = 5, \mathsf{NT} = 100000$. The p-range considered is 0.1 : 0.025 : 0.5.

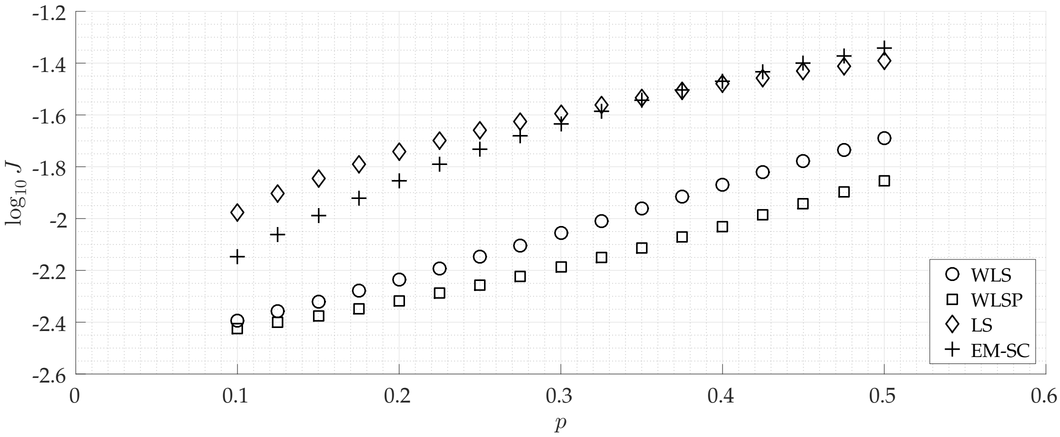


Figure 3.12: $\log_{10} J$ plotted against p for a 10 nodes graph with a random configuration; Parameters: $\alpha = 0.1, \beta/\alpha = 5, \text{NT} = 100000, p_e = 0.4$. The p-range considered is 0.1 : 0.025 : 0.5.

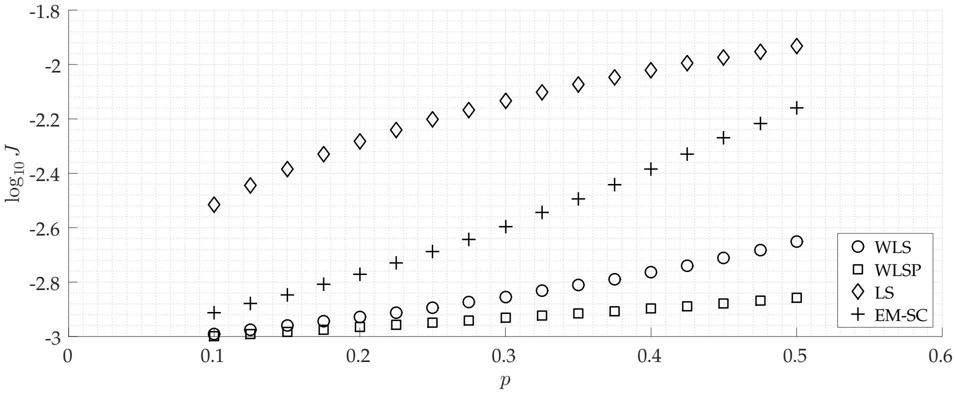


Figure 3.13: $\log_{10} J$ plotted against p for a 10 nodes graph with a complete configuration; Parameters: $\alpha = 0.1, \beta/\alpha = 5, \text{NT} = 100000$. The p-range considered is 0.1 : 0.025 : 0.5.

From the first set of plots, it can be readily observed that with increasing p, J does not decrease, which is expected as with an increasing value for p, we have more measurements of a 'bad' quality in the set. It is observed that for a fixed β/α ratio, the 4 approaches converge for the cycle graph; in the random graph, which consists of 17 edges, we observe that after $p = 0.35$ the EM-RAND-SC approach performs worse than the LS approach and also that a tendency to converge is observed. This is not surprising as increasing p furthermore until the value 1, all the 4 approaches should meet at the same point as we then have only 'one' noise source. For the complete graph, we clearly see that all the approaches performs better than the LS approach. From the above, it can be concluded that when the graph is more connected, then the approaches considered yield better results than the LS approach. Note, we have only consider it for one value of β/α, hence other values needs to be considered to justify this statement. Also observed from the plots is that the growth of the WLSP can be considered the slowest with regard to the WLS and the EM approach; this is obvious from fig. 3.13.

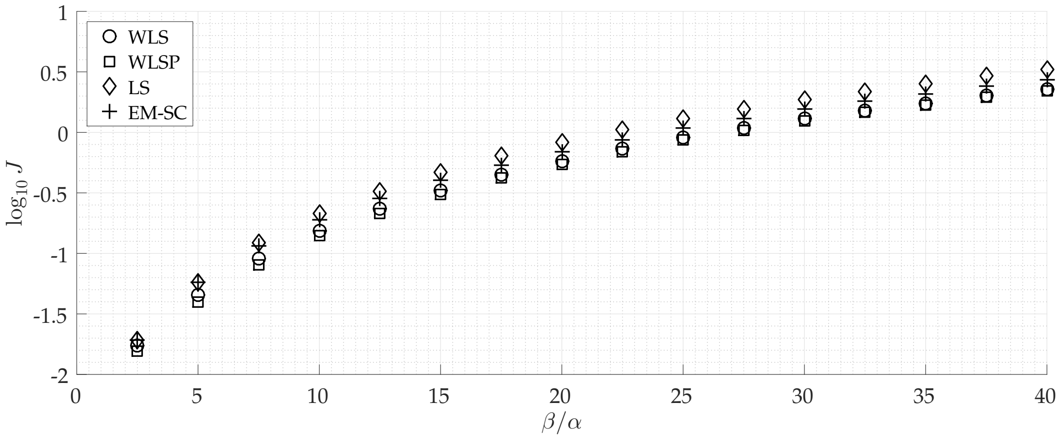


Figure 3.14: $\log_{10} J$ plotted against β/α for a 10 nodes graph with a cycle configuration; Parameters: $\alpha = 0.1, p = 0.25, \mathrm{NT} = 100000$. The β/α-range considered is $(1:1:16) \cdot 2.5$.

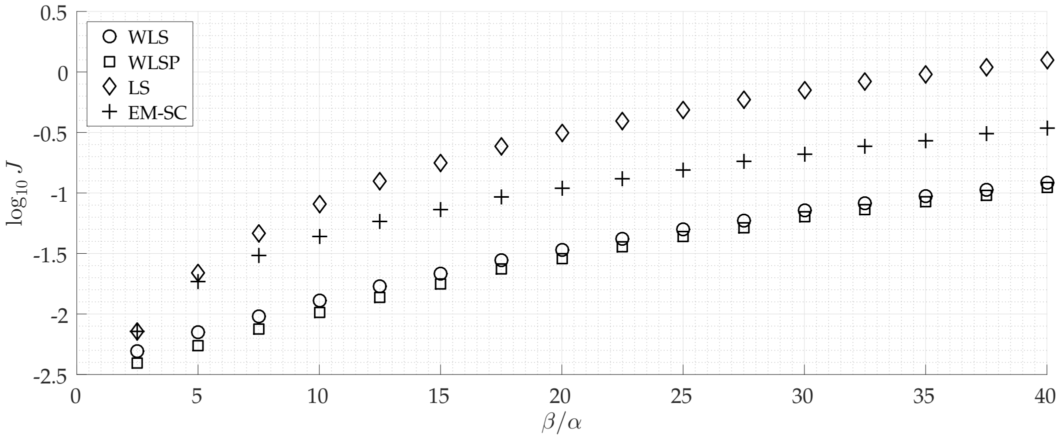


Figure 3.15: $\log_{10} J$ plotted against β/α for a 10 nodes graph with a random configuration; Parameters: $\alpha = 0.1, p = 0.25, \mathrm{NT} = 100000, p_e = 0.4$. The β/α-range considered is $(1:1:16) \cdot 2.5$.

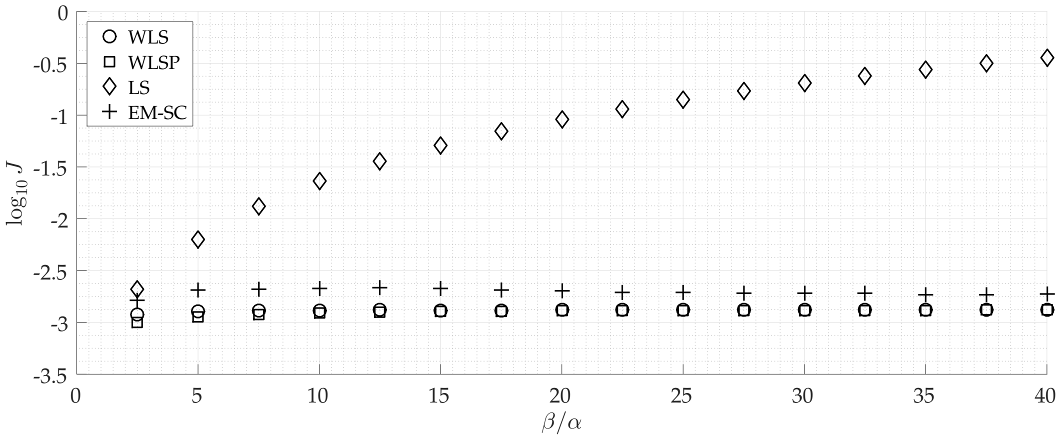


Figure 3.16: $\log_{10} J$ plotted against β/α for a 10 nodes graph with a complete configuration; Parameters: $\alpha = 0.1, p = 0.25, \mathrm{NT} = 100000$. The β/α-range considered is $(1:1:16) \cdot 2.5$.

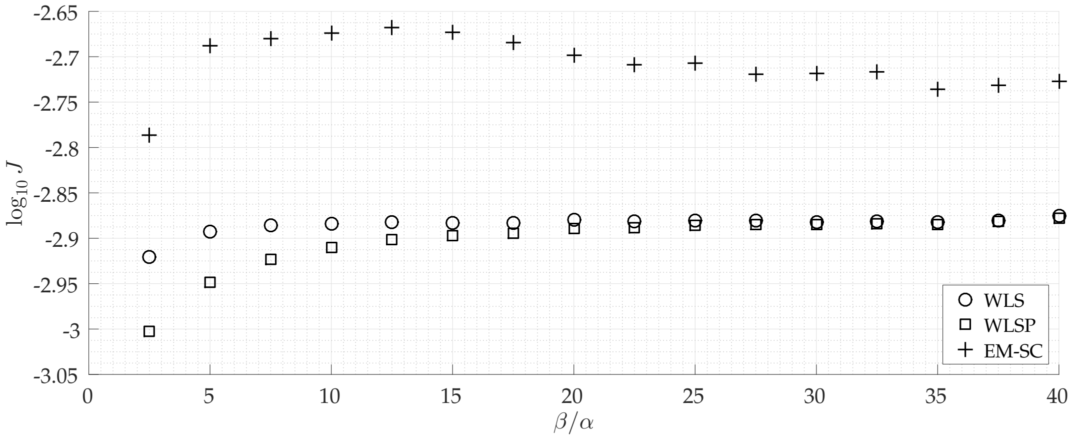


Figure 3.17: $\log_{10} J$ plotted against β/α for a 10 nodes graph with a complete configuration; Parameters: $\alpha = 0.1, p = 0.25, \mathrm{NT} = 100000$. The β/α-range considered is $(1 : 1 : 16) \cdot 2.5$.

Turning our attention to the second set of plots, we observed again overall an increase in J for increasing values of β/α. Only for the complete graph, we observe some interesting behavior for the WLS, WLSP and the EM-RAND-SC approach which can be observed in fig. 3.17. For the circle graph, we observe that the approaches are very tight to each other with the LS approach performing the worst. In the random graph, we see some clear distinction. For the complete graph, as already mentioned, we observe for the WLS, WLSP an oscillating effect, while for the EM-RAND-SC approach, it surprisingly performs better when the value for β/α is increased. This is not anticipated on and a motivation for this behavior is at the moment not available.

4. Conclusion & Outlook

In the current chapter, a summary is given of the work done on the robust estimation problem and also some further work is discussed.

In the current work, we have derived maximum likelihood type algorithms for the problem of estimation from relative measurements in which the noise term is modeled as a binary mixture of Gaussian distributions. It should be mentioned that the current algorithms are anchor-free, hence no nodes need to be considered as known compared to the work by Barooah [4]. The additive constant that needs to be added to the estimate is chosen to be the centroid of the nodes which leads to the estimators having the property of unbiasedness. Also, the centroid solution is unique [32]. For the WLS and the WLSP approach we have obtained analytically its performance for the case when the measurement quality is fixed, i.e., $\bar{\mathbf{Z}} = \bar{\mathbf{z}}$ and for the case when this is random. For the WLSP approach, we have also proposed a heuristic for finding the covariance matrix as it is computationally expensive to compute the exact value. From simulations on a five nodes graph, it is observed that this heuristic works good. It should be noted though that for larger graphs, a smaller value for the threshold needs to be taken in order to have a reasonable approximation. For the current study, we consider the EM-RAND-SC algorithm in which a stopping criterion is specified rather than a fixed number of iterations. This is done because we are interested in the convergence of $\mathbf{x}^{(r)}$. Also, it is observed that the number of iterations is sensitive to the noise distribution parameters, while the stopping criterion is not.

Regarding application of the algorithms to the graph topologies considered, it is observed that for tree type graphs, all the approaches yield the same state estimate. This leads to the conclusion that for tree type graphs, the measurement quality is irrelevant for the state estimation. However, it is needed for determining the uncertainty in the estimation, which is also the same for all the approaches considered. For cycle graphs, we observe from the Brute Force approach that the log likelihood function have M distinct values, each corresponding to the number of ones in the quality vector and as such not dependent on their location in the vector. Hence different combination of $\mathbf{z}$ and corresponding $\mathbf{x}$ may be obtained as the result for the estimation problem. This ambiguity is not observed in the wheel, random and complete graph. Comparing the different graph topologies, we observe as the graph is more connected, the performance measure J, which is the normalized mean square error, decreases. This serves as motivation for the part II. Also observed is that the WLSP approach outperforms the other approaches. In the comparison, we have also considered the ordinary least squares (LS) approach in which all the measurements are considered to be 'good' measurements, i.e. of the same quality. In the parametric study, carried out for a graph consisting of 10 elements with graph topology chosen as cycle, random, and the complete graph, it is observed that for the cycle graph, the performance of all the approaches are similar. A clear distinction is observed for the random graph considered. Herein, it is observed also that the EM approach performs worse than the LS approach. For complete graph configuration, the benefits of the approaches considered in the current work are obvious compared to the LS approach. However, for the case in which we vary the β/α ratio, we observe the EM approach to perform better with an increasing value of β/α ratio which is not expected, while both the WLS and the WLSP approach appears to 'oscillate' for the complete graph configuration.

4.1 Outlook

Having given a summary of the contribution and the conclusion, we have some open problems which need attention for the near future. First, the Gaussian mixture model may be extended. In the current report, we have assumed it to contain two components. Expanding it to the case of more than two components can be done easily. Of course, then the decision boundary as derived for the WLSP approach may not be applicable and in this case we need to determine using the MAP distribution to which distribution the noise is most likely from. This is still a discrete case.

Another extension is to consider the variance to be a function of the state variable, as modeled in [12]. This for we usually do not know the actual value of the variance. Furthermore, in the current approaches we assume to know the noise distributions. A step further would be to let the algorithm also determine these parameters for the discrete case and the number of components, which is a priori unknown.

Second, a better heuristic can perhaps be found for approximating the covariance matrix of the WLSP estimator. Our current approach is based on the fact of discarding the $\mathbf{z}$ combinations which are very small. As the number of edges M increases, we have that 2^M increases exponentially and as such a large portion of the $\mathbf{z}$ combinations are discarded when a not so small value for the threshold is taken. Perhaps a method can be found which is independent on the number of $\mathbf{z}$ combinations.

The next thing that needs attention is the EM algorithm. Instead of looking at how to analytically calculate the performance of the approach as done in the WLS and the WLSP approach, we may be interested in deriving bounds on the performance. Also regarding the implementation, maybe the EM-em approach can be combined with the EM-RAND-SC approach. As observed, for the same number of iterations, a larger number of trials which satisfy a given threshold are obtained. This hints that the EM-em-FI approach have a faster convergence compared to the EM-RAND-FI approach. By combining both, we can use a smaller number of iterations for meeting the same condition. It should be mentioned though that for the EM-em approach also for the short runs, we need some iterations.

It is of interest to study the behavior as observed in fig. 3.17 where the EM-RAND-SC performs 'better'. This is counter intuitive and hence needs attention. Also, running the algorithms on different set of parameters may also of interest. This gives a better picture of the performance of the algorithms.

The algorithms presented in this report are centralized algorithms. This means that all the information must be sent to a central location where information processing is done and an estimate is obtained. This is then sent back to the vertices. As described in [4], this approach puts burdens on the nodes that are close to the central location. Moreover, whenever a link failure occurs in the routing tree to the central node, a possible loss of information may occur. With the above in mind, it is desired to construct the optimal estimate in a distributed manner, in which each node locally computes an estimate of its own variable based on information received from its local neighbors; an application of this may be found in autonomous robots. In the references mentioned in the Introduction, distributed algorithms are already considered; however with the assumption of equal quality for measurements. This need to be expanded upon for which the work presented in the current report can serve as a first stepping stone. Due to the assumption of a constant variance, the distributed algorithms mentioned in [4, 5, 8, 9, 11] focused on the estimation step. The classification step can be easily included as in the current work, the MAP estimation is already done in a distributed manner for the classification of each measurement is based on the noise parameters and the state estimate of the vertices involved in the measurement.

It should be mentioned that we are already working on a distributed version of the algorithms considered in the current work. Recently, in collaboration with C. Ravazzi from the Politecnico di Torino, Italy, a conference paper is submitted to the Conference on Decision and Control (CDC) 2016, titled *"Distributed estimation from relative measurements of mixed quality"*.

Part II

On Design of the Network Topology for Improved Estimation

5. Optimal Extension of the Graph

5.1 Overview

Up until now, we have paid attention to the estimation of the state vector from noisy relative measurements. The covariance matrix for the estimates, which provides information regarding the uncertainty in the estimation, is obtained analytically for the WLS approach (proposition 2.2) and the WLSP approach (proposition 2.9). In this part of the report, we are interested in the problem of optimally adding edges to the existing edge set in order to reduce the uncertainty in the estimation. This problem is motivated by fig. 3.1 in which it can be observed that the more connected the graph is, i.e., the more edges exist between the vertices, the smaller the value is for J which is proportional to the trace of the covariance matrix. This problem is also of interest from the perspective of network robustness, which in Ref. [33] is defined as to be "the ability of a network to continue performing well when it is subject to failures or attacks". As the graphs considered in this report are already connected, i.e. each vertex can be reached starting from any other vertex by following a chosen path, adding edges to the existing graph results in creating alternative paths between the vertices and as such will result in the network being more robust against link failures.

The structure of the current chapter is as follows; we first begin by defining the problem in section 5.2. Hereafter, attention will be paid on approaches for solving the problem. The approaches considered in the current work are the combinatorial approach in which we consider all the possible alternatives for adding edges to the existing graph and then choose the one that yields the best result. This is detailed in section 5.3. An alternative to the combinatorial approach, which is in practice infeasible due to the number of alternatives that are available, is the submodular approach in section 5.4. This as a consequence of the observation that choosing the edges with as goal reducing the uncertainty in the estimation can be regarded as a submodular function. Hence, a greedy algorithm may be used. In the greedy algorithm, at each iteration, the edge is chosen that yields the largest decrease in the uncertainty. So instead of adding a specified number of edges all at once, which is the case in the combinatorial approach, we will add the edges one at a time, with for each iteration the augmented graph from the previous run as the base graph. From the literature, we know that this heuristic yields near optimal solution, [7, 14, 16, 34].

The novelty in this report will be the addition of edges with a priori unknown quality. By this, we mean that we only know the distribution parameters of each edge and not the exact quality of it. The edge addition is done based on the consideration that all edges are equally weighted with the value being the expected quality based on knowledge of the distribution parameters. After the addition, we then are provided the missing information regarding the quality of the measurement and as such can observe how well we have done.

5.2 Problem Statement for Edge Addition

In this section, the problem statement will be given. Starting from a base graph $\mathcal{G}_{\text{base}}$, given a set of possible candidates $\mathcal{E}_{\text{cand}}$ with cardinality $\left|\mathcal{E}_{\text{cand}}\right| = M_{\text{cand}}$ and the possibility to add k edges with $0 \leq k \leq M_{\text{cand}}$, we are interested in how to add these k edges from $\mathcal{E}_{\text{cand}}$ to $\mathcal{G}_{\text{base}}$ such that we maximize the A-criterion, i.e.,

$$\begin{aligned} \arg\max_{\mathcal{E}} \quad & -\operatorname{trace}\left(L^{\dagger}_{(\mathcal{E}_{\text{base}}\cup\mathcal{E})}\right) \\ \text{subject to} \quad & \left|\mathcal{E}\right| = k \\ & \mathcal{E} \subseteq \mathcal{E}_{\text{cand}}. \end{aligned} \tag{5.1}$$

Note, the A-optimal criterion is defined as $-\text{trace}(C) = -\sum_i \lambda_i(C^{\dagger})$ with λ_i being the non-zero eigenvalues of the Moore-Penrose pseudoinverse of matrix C. Here, the variable is the set $\mathcal{E}$. We assume that $\mathcal{E}_{\text{cand}} \cap \mathcal{E}_{\text{base}} = \emptyset$. In addition, the edges in $\mathcal{E}_{\text{cand}}$ have weights given by the following, $w_e = (1 - z_e)\alpha^{-2} + z_e\beta^{-2}\, \forall e \in \mathcal{E}_{\text{cand}}$ with z_e indicating the quality of the edge as was done in **??**.

5.3 Combinatorial Approach; Adding Edges All at Once

The above problem can be reformulated as follows. Consider each subset $\mathcal{E} \subseteq \mathcal{E}_{\text{cand}}$ to be encoded by a boolean vector $\mathbf{y} \in \{0,1\}^{M_{\text{cand}}}$, in which a 1 for y_e means that the edge is added to $\mathcal{G}_{\text{base}}$ and 0 otherwise, the problem can be rewritten as

$$\begin{aligned} \underset{\mathbf{y}}{\arg\max} \quad & -\text{trace}\Big(L(\mathbf{y})^{\dagger}\Big) \\ \text{subject to} \quad & \mathbf{1}^T\mathbf{y} = k \\ & \mathbf{y} \in \{0,1\}^{M_{\text{cand}}}. \end{aligned} \tag{5.2}$$

with $L(\mathbf{y}) = L_{\text{base}} + \sum_{e=1}^{M_{\text{cand}}} y_e w_e a_e^T a_e$. L_{base} is the laplacian of $\mathcal{G}_{\text{base}}$ while w_e denotes the weight of edge e. The problem can be solved exactly by using exhaustive search, in which $\binom{M_{\text{cand}}}{k}$ combinations for $\mathbf{y}$ are considered. In the current work, we consider two cases regarding the weight of the edges to be added. First, we assume to have full knowledge regarding the weigth of the edge, i.e., we know w_e is either α^{-2} or $\beta^{-2}\, \forall e \in \mathcal{E}_{\text{cand}}$. The procedure for solving the optimization problem would then be to first obtain the $\binom{M_{\text{cand}}}{k}$ combinations, then for each combination, add the corresponding edges to $\mathcal{G}_{\text{base}}$, and obtain the A-optimum criterion which is stored in a vector. Afterwards, run through the whole vector and choose the edge set which gives the highest value for the A-optimum criterion, see algorithm 8.

Algorithm 8 BF Approach for Adding Edges with Known Quality

```
Require: Data: $(L_{\text{base}}, \mathcal{E}_{\text{cand}}, k, W_{\text{cand}})$
Initialize:
    $R \leftarrow \emptyset$
Compute $R_{\text{total}}$ for all $\binom{M_{\text{cand}}}{k}$ combinations
for all $\binom{M_{\text{cand}}}{k}$ combinations do
    Calculate $L_{\text{New}}$

        $L_{\text{New}} = L_{\text{base}} + L_{\text{add}}$

    Obtain the A-optimum criterion value and store it in $R$
    $R \leftarrow R \cup -\text{trace}(L_{\text{New}}^{\dagger})$
end for
Find $\mathcal{E}^*$

        $\mathcal{E}^* = \underset{\mathcal{E} \in \binom{M_{\text{cand}}}{k}}{\arg\max} R$
```

In the second case, which we regard as to be an extension of the first case, we assume to not know the exact value of w_e. We are only given the set of values that w_e can take and their probability. The following approach is then considered. First, all the edges in $\mathcal{E}_{\text{cand}}$ are considered to have weights with values $\mathbb{E}[w_e]$. In the current work $\mathbb{E}[w_e] = (1-p)\alpha^{-2} + p\beta^{-2}$. algorithm 8 will be carried out to find the edge set that is the solution of the optimization problem 5.2 with consideration of equal weights for all the edges. After addition of the edges to the existing graph, the actual weights of the added edges become known and we can calculate the correct value for the A-optimality criterion and as such know how well we have done by adding the chosen edges.

Algorithm 9 BF Approach for Adding Edges with a Priori Unknown Quality

```
Require: Data: $(L_{\text{base}}, \mathcal{E}_{\text{cand}}, k, W_{\text{cand}}, \alpha, \beta, p)$
1: Calculate $\mathbb{E}[w_e]$
       $$\mathbb{E}[w_e] = \frac{p}{\beta^2} + \frac{1-p}{\alpha^2} \quad \forall e \in \mathcal{E}_{\text{cand}}$$
2: Run algorithm 8 in order to obtain $\mathcal{E}^*$ based on edge weights $\mathbb{E}[w_e]$.
3: Find correct edge weigths corresponding to $\mathcal{E}^*$.
4: Calculate the A-criterion value with the proper weights for the added edges.
```

As it is known, exhaustive search will be impractical when M_{cand} and k are large hence the following heuristic is considered.

5.4 Submodular Approach; Adding Edges One at a Time

In this section, the greedy algorithm in Ref. [16] will be used for selecting edges to add to the existing graph. We will also give extension of the algorithm to cases in which we do not know the exact quality of each edge in the candidate set. The reason for considering this approach is based on the observation that the A-optimum criterion is submodular. In the following, we briefly give the definition of a submodular function and the main theorem found in [16].

Definition 5.1 (Submodular Function [16, 34]). *Let V be a given finite set $V = 1, \ldots, M$, and let $f : 2^V \to \mathbb{R}$ be a function that assigns a real number to each subset of V. Then f is submodular if, for any sets A and B satisfying $A \subseteq B \subseteq V$ and any element $s \notin B$, it holds that*

$$f(A \cup \{s\}) - f(A) \geq f(B \cup \{s\}) - f(B) \tag{5.3}$$

or equivalently, if for all subsets $A, B \subseteq V$, it holds that

$$f(A) + f(B) \geq f(A \cup B) + f(A \cap B). \tag{5.4}$$

We also know that if a function f is submodular, then $-f$ is supermodular [34]. Intuitively, from the above, we can derive that adding an element to a smaller set yields a greater gain than adding the same element to a larger set. Thm 3 of Ref. [16], which is an important result, is given below

Theorem 5.1 ([16]). *Let $G = (V, E, w_E)$ be a given connected weighted graph, let $\mathcal{E} \subseteq (V \times V) \setminus E$ with weights $w_{\mathcal{E}}$, and let $L_{\mathcal{E}}$ be the weighted graph Laplacian matrix associated with the edge set $E \cup \mathcal{E}$. Then the set function $f : (V \times V) \setminus E \to \mathbb{R}$ defined by $f(\mathcal{E}) = -\text{trace}(L_{\mathcal{E}}^{\dagger})$ is submodular.*

As a consequence of this result and the fact that the function $f(\mathcal{E}) = -\text{trace}(L_{\mathcal{E}}^{\dagger})$ is monotone increasing, the greedy heuristic can be applied which produces a near optimal set selection. In Thm. 2 of the same paper, it is stated that when f is submodular and monotone increasing, we have the following upper bound regarding the performance of the greedy algorithm,

$$\frac{f^* - f(S_{\text{greedy}})}{f^* - f(\emptyset)} \leq \left(\frac{k-1}{k}\right)^k \leq \frac{1}{e} \approx 0.37 \tag{5.5}$$

Here, f^* is the optimal value obtained using the combinatorial approach, S_{greedy} the set of edges obtain using the greedy approach and $f(S_{\text{greedy}})$ its function value and k the number of edges to add. It is known that the greedy algorithm performs often better than the bound. Pseudocode for the greedy algorithm is given below.

?? in algorithm 10 is used for speeding up the greedy algorithm and its derivation can be found in Ref. [16].

Algorithm 10 Greedy Approach for Adding Edges with Known Quality

```
Require: Data: $(L_{\text{base}}, \mathcal{E}_{\text{cand}}, k, W_{\text{cand}})$
Initialize:
    $S \leftarrow \emptyset$, $\mathcal{E}_{\text{cur}} \leftarrow \mathcal{E}_{\text{cand}}$, $L_{\text{cur}} \leftarrow L_{\text{base}}$, $W_{\text{cur}} \leftarrow W_{\text{cand}}$
for $t = 1, \ldots, k$ do
    Obtain the edge to add
        $\hat{e} = \arg\max_{e \in \mathcal{E}_{\text{cur}}} \; -\operatorname{trace}\left(L^{\dagger}_{(\text{cur}+e)}\right)$
    with $\operatorname{trace}\left(L^{\dagger}_{(\text{cur}+e)}\right)$ calculated using
        $\operatorname{trace}\left(L^{\dagger}_{(\text{cur}+e)}\right) = \operatorname{trace}\left(L^{\dagger}_{\text{cur}}\right) - \frac{w_e}{1 + w_e \mathbf{a}_e L^{\dagger}_{\text{cur}} \mathbf{a}_e^T} \left\| L^{\dagger}_{\text{cur}} \mathbf{a}_e^T \right\|^2$    (5.6)
    $\mathbf{a}_e$ is a row vector in which the entries corresponding to the edge $e$ are nonzero.
    Update the sets
    $S \leftarrow S \cup \hat{e}$, $\mathcal{E}_{\text{cur}} \leftarrow \mathcal{E}_{\text{cur}} \backslash \hat{e}$, $L_{\text{cur}} \leftarrow L_{\text{cur}} + w_{\hat{e}} \mathbf{a}_{\hat{e}} \mathbf{a}_{\hat{e}}^T$, $W_{\text{cur}} \leftarrow W_{\text{cur}} \backslash w_{\hat{e}}$
end for
```

Having presented algorithm 10, we consider two variations to it; again due to not knowing the measurement quality ahead of time. In the first variation we assume to know the actual weight of the added edge immediately after adding it to the graph. In the second variation, we know the actual weight of the edge after adding all the edges to the graph. This is the same assumption as considered in algorithm 9. Pseudocode for both the variations are given in algorithm 11 and algorithm 12.

Algorithm 11 Greedy Approach for Adding Edges with Quality of the Edges Immediately Known After Addition

```
Require: Data: $(L_{\text{base}}, \mathcal{E}_{\text{cand}}, k, W_{\text{cand}}, \alpha, \beta, p)$
Initialize:
    $S \leftarrow \emptyset$, $\mathcal{E}_{\text{cur}} \leftarrow \mathcal{E}_{\text{cand}}$, $L_{\text{cur}} \leftarrow L_{\text{base}}$, $W_{\text{cur}} \leftarrow W_{\text{cand}}$
Calculate $\mathbb{E}[w_e]$
    $\mathbb{E}[w_e] = \frac{p}{\beta^2} + \frac{1-p}{\alpha^2}$
for $t = 1, \ldots, k$ do
    Obtain the edge to add
        $\hat{e} = \arg\max_{e \in \mathcal{E}_{\text{cur}}} \; -\operatorname{trace}\left(L^{\dagger}_{(\text{cur}+e)}\right)$
    with $\operatorname{trace}\left(L^{\dagger}_{(\text{cur}+e)}\right)$ calculated using
        $\operatorname{trace}\left(L^{\dagger}_{(\text{cur}+e)}\right) = \operatorname{trace}\left(L^{\dagger}_{\text{cur}}\right) - \frac{\mathbb{E}[w_e]}{1 + \mathbb{E}[w_e] \mathbf{a}_e L^{\dagger}_{\text{cur}} \mathbf{a}_e^T} \left\| L^{\dagger}_{\text{cur}} \mathbf{a}_e^T \right\|^2$
    $\mathbf{a}_e$ is a row vector in which the entries corresponding to the edge $e$ are nonzero.
    Obtain the actual weight of the added edge
    Update the sets
    $S \leftarrow S \cup \hat{e}$, $\mathcal{E}_{\text{cur}} \leftarrow \mathcal{E}_{\text{cur}} \backslash \hat{e}$, $L_{\text{cur}} \leftarrow L_{\text{cur}} + w_{\hat{e}} \mathbf{a}_{\hat{e}} \mathbf{a}_{\hat{e}}^T$, $W_{\text{cur}} \leftarrow W_{\text{cur}} \backslash w_{\hat{e}}$
end for
```

Algorithm 12 Greedy Approach for Adding Edges with Quality of the Edges Known After Addition of all k edges

```
Require: Data: $(L_{\text{base}}, \mathcal{E}_{\text{cand}}, k, W_{\text{cand}}, \alpha, \beta, p)$
Initialize:
    $S \leftarrow \emptyset$, $\mathcal{E}_{\text{cur}} \leftarrow \mathcal{E}_{\text{cand}}$, $L_{\text{cur}} \leftarrow L_{\text{base}}$
Calculate $\mathbb{E}[w_e]$
    $$\mathbb{E}[w_e] = \frac{p}{\beta^2} + \frac{1-p}{\alpha^2}$$
for $t = 1, \dots, k$ do
    Obtain the edge to add
        $$\hat{e} = \arg\max_{e \in \mathcal{E}_{\text{cur}}} -\text{trace}\left(L^{\dagger}_{(\text{cur}+e)}\right)$$
    with $\text{trace}\left(L^{\dagger}_{(\text{cur}+e)}\right)$ calculated using
        $$\text{trace}\left(L^{\dagger}_{(\text{cur}+e)}\right) = \text{trace}\left(L^{\dagger}_{\text{cur}}\right) - \frac{\mathbb{E}[w_e]}{1 + \mathbb{E}[w_e]\mathbf{a}_e L^{\dagger}_{\text{cur}} \mathbf{a}_e^T} \left\| L^{\dagger}_{\text{cur}} \mathbf{a}_e^T \right\|^2$$
    $\mathbf{a}_e$ is a row vector in which the entries corresponding to the edge $e$ are nonzero.
    Update the sets
    $S \leftarrow S \cup \hat{e}$, $\mathcal{E}_{\text{cur}} \leftarrow \mathcal{E}_{\text{cur}} \backslash \hat{e}$, $L_{\text{cur}} \leftarrow L_{\text{cur}} + \mathbb{E}[w_e]\mathbf{a}_e\mathbf{a}_e^T$
end for
Obtain weights of the edges in the set $S$.
Calculate the A-criterion value with the proper weights for the added edges in the set $S$.
```

6. Numerical Results

6.1 Overview

In the current chapter, we present the application of the algorithms considered in the previous chapter to some graph configurations. First, we apply the algorithms to the cycle graph configuration consisting of 10 nodes in section 6.2. Afterwards, in the same section, we consider as base graph a realization of a ER random graph consisting of 20 nodes and with $p_e = 0.25$. Hereafter, in section 6.3, we consider the estimation problem for the cycle graph after edge addition.

6.2 Evaluation of the Combinatorial and the Submodular Approaches

In the following, we evaluate the edge addition strategy using the combinatorial and the submodular approach; we also consider the cases in which we do not know the measurement quality prior to the addition process.

6.2.1 Ten Nodes Cycle Graph as Base Graph

We first consider a cycle consisting of 10 vertices generated with the following noise distribution parameters, $\alpha = 0.1, \beta/\alpha = 5, p = 0.25$. The five algorithms (2 for the combinatorial strategy and 3 for the greedy strategy) are applied to the cycle graph. We consider the addition of 5 additional edges from the possible set of edges to add, each with its quality. In fig. 6.1, the edges added using each algorithm are in red.

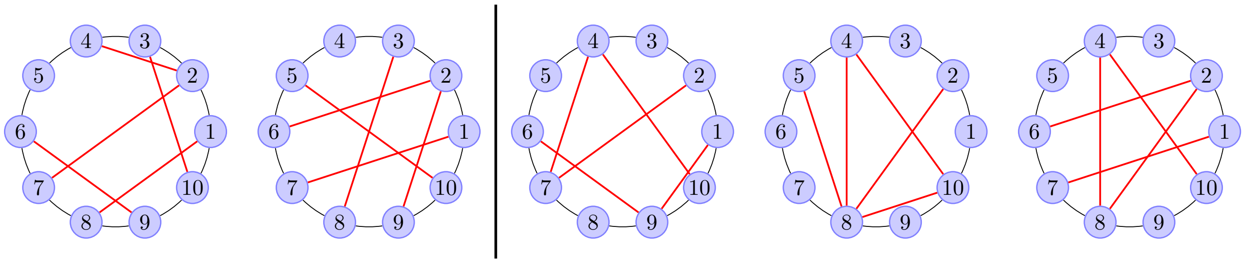


Figure 6.1: Edge addition on a cycle graph consisting of 10 Nodes; from left to right: algorithm 8 to algorithm 12 are applied; the bar in the middle separate the combinatorial approach from the greedy approach. The simulation parameters are $\alpha = 0.1, \beta/\alpha = 5, p = 0.25$ and we consider addition of 5 edges from the possible 35 candidates.

From the plots, it can be observed that the edges added to the existing graph are between nodes which are not close to each other. We will illustrate this using the edge between vertex 2 and 7 for the left most plot. In the base graph, there are two ways to go from node 2 to node 7; by choosing the sequence $2-3-4-5-6-7$ or by choosing the other way $2-1-10-9-8-7$. In both scenarios, 5 hops are needed to reach vertex 7 from vertex 2. By the addition of the link $(7,2)$ we have now a direct link between 2 and 7. This is also useful when we want to get to node 6 starting from 2. It takes us only 2 hops now $(2-7-6)$ to reach vertex 6 while in the base graph it will be either 4 or 6. This is also observed in [16] and this behaviour is as described therein, characteristic for the Watts-Strogatz small world graph model.

When we consider the quality of the edges added, we observe that for the case in which we know this quality prior to addition, all the added edges are ‘good’. This is according to expectation. For the other approaches in which the quality is not known in advance, the actual quality of the added edge is based on a coin flip using a biased coin represented by the parameter p in the current study as we assume that the quality of the edges are independent. It should be noted that also in these cases nodes which are further from each other are connected.

In fig. 6.2 to 6.4 , we present plots for J as a function of the edge added to the graph. This will be done only for the greedy approach as in this approach, we can obtain the decrease in J after addition of each edge to the current graph. For the combinatorial approach, as the edges are added all at once, we cannot determine the sequence in which they are added, only the final result after adding them to the base graph.

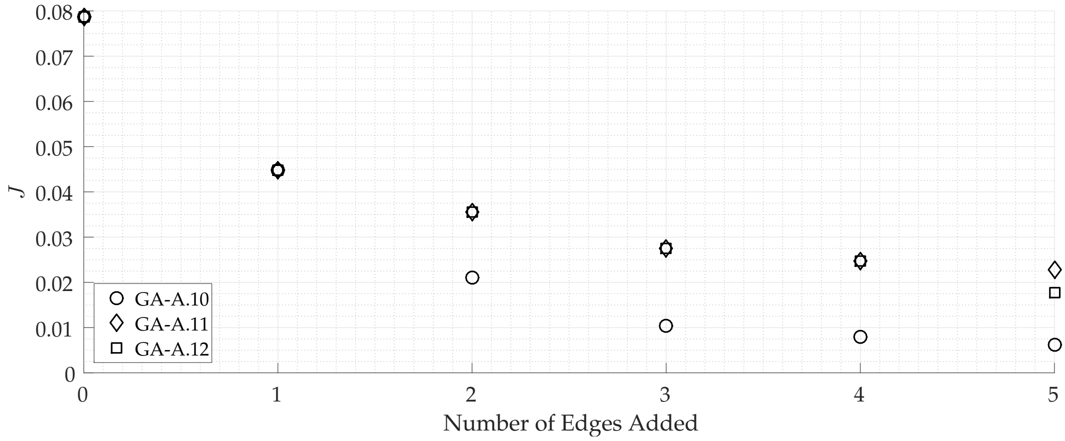


Figure 6.2: J as a function of the number of edges added for greedy approaches considered

It can be observed from the plot that algorithm 10 performs the best which is to be expected. We can observe that algorithm 11 and algorithm 12 both have similar performance except for the last edge added. Both the approaches perform the same as the first three edges added to the base graph are the same. The fact that algorithm 12 performs better at the end is due to the fact that the edge added is a 'good' edge while for algorithm 11 it is a 'bad' edge.

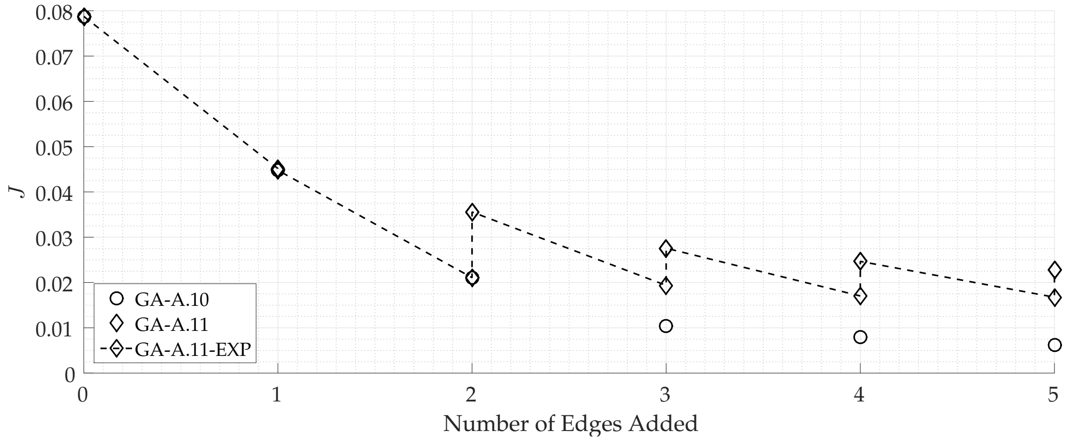


Figure 6.3: J as a function of the number of edges added for algorithm 10 and algorithm 11

From the plot, a zig zag course is observed for algorithm 11; this is to be expected as we first add the edge with weight value equal to $\mathbb{E}[w_e]$ after which we are then given the information whether it is 'good' or 'bad'. Vertical upwards mean the edge is in reality 'bad' while downward indicates that the edge added is 'good'. The first edge added to the base graph is a good one hence, we do not see the course of the plot going upwards. As the difference between for J is small, this vertical downwards action is not observed in the plot.

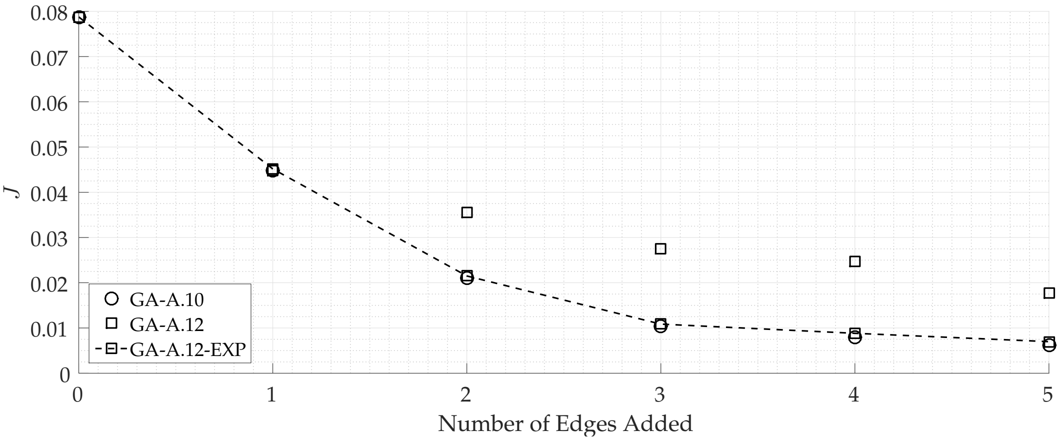


Figure 6.4: J as a function of the number of edges added for algorithm 10 and algorithm 12

The main observation from this plot is that the edge addition considering the candidate set having all the same quality (again $\mathbb{E}[w_e]$) yield a J value that is for each step very close to the case when we know the edge quality ahead of time. It should be noted though that $\mathbb{E}[w_e] = 76$ and $w_\alpha = 100$ so they are also close to each other.

It should be mentioned that from the above we cannot conclude whether algorithm 11 or algorithm 12 performs better when the edge quality is a priori unknown. This as the plots given are for one realization of the base graph and depending on the quality of the edges added. It happened so that for the current case, the edges added using algorithm 11 happens to be, except for the first one, all 'bad' while for algorithm 12, the first and the last edge added are 'good'. As mentioned already, the actual quality is based on a coin flip with parameter p.

For the case, the edges are known ahead of time, we observe for the combinatorial approach J after addition is 0.0582 while for the submodular approach 0.0612 is obtained. The starting value is 0.07876. Plugging the values in eq. (5.5), yields a value of $0.0042 < 0.37$, justifying that the greedy approach performs better than suggested.

6.2.2 Random Graph as Base Graph

Next, we consider the addition of edges to a randomly generated graph consisting of 20 vertices and the probability of linking 2 vertices to be 0.25 which is $p_c = \frac{\log 20}{20} \approx 0.15$. The realized graph has 51 edges. The number of edges added to the random graph is 15 out of the 139 candidates.

In the matrix of eq. (6.1), the degree of each node is given; hereby we do not take the weights into account yet. The first row shows the degree for each node of the base graph, the second row when applying algorithm 10; third row is for algorithm 11 and fourth row for algorithm 12.

$$\begin{bmatrix} 7 & 3 & 3 & 5 & 8 & 2 & 4 & 2 & 7 & 5 & 4 & 8 & 8 & 3 & 6 & 7 & 5 & 5 & 5 & 5 \\ 7 & 5 & 5 & 6 & 8 & 5 & 5 & 6 & 7 & 7 & 7 & 9 & 8 & 5 & 6 & 7 & 7 & 6 & 8 & 8 \\ 7 & 5 & 6 & 6 & 8 & 4 & 4 & 6 & 7 & 6 & 9 & 8 & 8 & 4 & 6 & 7 & 7 & 5 & 11 & 8 \\ 7 & 4 & 6 & 7 & 8 & 5 & 5 & 6 & 7 & 7 & 8 & 8 & 8 & 5 & 6 & 7 & 6 & 5 & 8 & 9 \end{bmatrix} \tag{6.1}$$

It is observed from the matrix that edges are added mostly between vertices which have a small degree. The only exception are the last two nodes which has an increase in the number of edges of three and more for the approaches considered. The reason for this is due to the fact that their initital variance is quite large as is shown below (6.2).

$$\begin{bmatrix} 24 & 40 & 94 & 38 & 25 & 60 & 32 & 133 & 27 & 39 & 153 & 27 & 21 & 56 & 32 & 24 & 50 & 29 & 97 & 95 \\ 21 & 21 & 27 & 26 & 21 & 21 & 21 & 21 & 21 & 21 & 26 & 20 & 17 & 26 & 26 & 21 & 21 & 21 & 25 & 25 \\ 21 & 22 & 27 & 27 & 22 & 28 & 28 & 28 & 23 & 27 & 26 & 26 & 18 & 36 & 27 & 21 & 27 & 28 & 32 & 34 \\ 21 & 27 & 35 & 26 & 22 & 29 & 22 & 22 & 22 & 27 & 27 & 26 & 18 & 35 & 27 & 21 & 35 & 27 & 49 & 33 \end{bmatrix} \times 10^{-4}$$

(6.2)

In the next matrix, the variance of the states are given when adding $15, 30, 45, 60, 75, 93$ and all the edges to the graph. This is done only for algorithm 10. The number 93 is also considered for among the possible candidates (139), 93 of them are of 'good' quality and the remaining ones are 'bad'.

$$\begin{bmatrix} 24 & 40 & 94 & 38 & 25 & 60 & 32 & 133 & 27 & 39 & 153 & 27 & 21 & 56 & 32 & 24 & 50 & 29 & 97 & 95 \\ 21 & 21 & 27 & 26 & 21 & 21 & 21 & 21 & 21 & 21 & 26 & 20 & 17 & 26 & 26 & 21 & 21 & 21 & 25 & 25 \\ 16 & 14 & 17 & 16 & 16 & 17 & 17 & 14 & 14 & 17 & 16 & 16 & 16 & 17 & 14 & 14 & 16 & 17 & 16 & 19 \\ 12 & 12 & 14 & 14 & 12 & 12 & 12 & 12 & 12 & 14 & 14 & 12 & 12 & 12 & 12 & 12 & 14 & 12 & 14 & 12 \\ 9 & 9 & 11 & 10 & 10 & 9 & 9 & 9 & 9 & 10 & 13 & 10 & 9 & 11 & 10 & 9 & 10 & 10 & 12 & 12 \\ 8 & 8 & 9 & 8 & 8 & 9 & 9 & 8 & 8 & 9 & 13 & 9 & 8 & 8 & 8 & 8 & 8 & 9 & 12 & 12 \\ 7 & 5 & 8 & 7 & 7 & 7 & 9 & 5 & 6 & 7 & 13 & 9 & 7 & 8 & 6 & 6 & 8 & 7 & 12 & 12 \\ 7 & 5 & 8 & 7 & 7 & 7 & 8 & 5 & 6 & 7 & 13 & 9 & 7 & 8 & 6 & 6 & 8 & 7 & 11 & 11 \end{bmatrix} \times 10^{-}4 \tag{6.3}$$

We also give a plot of the variance of $\hat{x}_8$ against the number of edges added using algorithm 10.

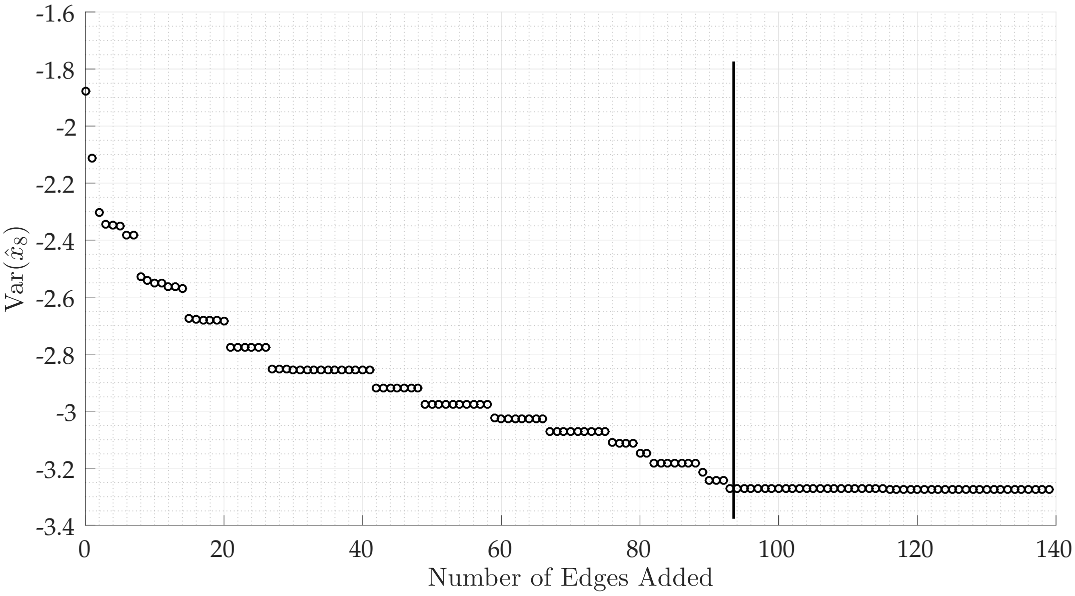


Figure 6.5: VAR$(\hat{x}_8)$ as a function of the number of edges added using algorithm 10. The black bar seperates the addition of good edges from bad edges.

The jumps in the graphs are mostly due to the fact that an edge is added which links vertex 8 with another vertex. Only in the beginning are also jumps observed when edges are added that indirectly has a significant decrease in the variance of vertex 8. It can be mentioned that from the set of edges, first all the 'good' edges are added to the base graph. The black bar indicates the boundary between 'good' and 'bad' edges. From the first 20 edges added apart from the jumps, it can be also observed that the value decreases. Another observation, which can also be observed from the matrix is that the addition of the first few edges leads to a greater decrease than addition of edges when the graph is already highly connected.

6.3 Performance of the Cycle Graph After Edge Addition

In the following, we briefly consider the estimation problem for the cycle graph before and after edge addition. In 6.6 , J is plotted for the different MLE approaches considered and also the different addition algorithms considered. It is expected that the values obtained from addition coincides with that of the estimation which is also the case (the open and close circles). It is observed that the WLSP performs better when bad measurements are in the graph. Also, it is observed that more bad measurements in the graph yields poor results for the EM algorithm as its value is above that of the LS approach for the approaches in which the noise quality is not known ahead of time.

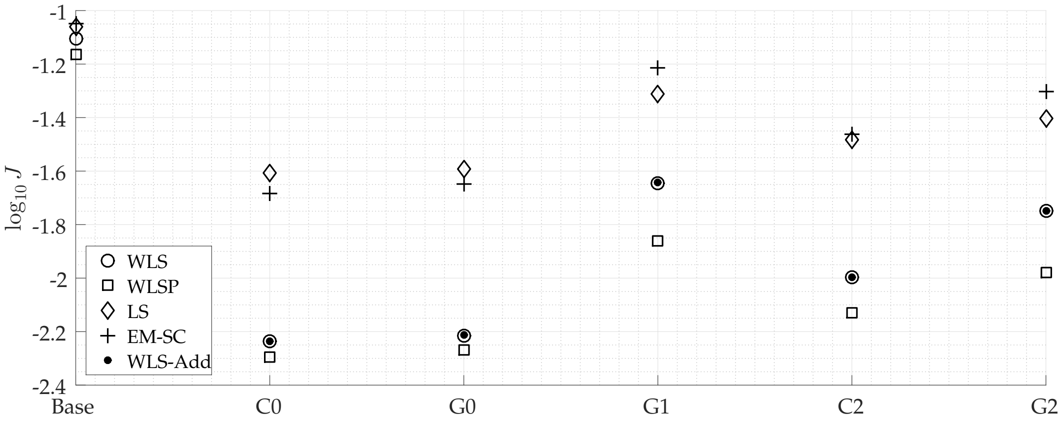


Figure 6.6: J values plotted for the estimation algorithms applied to graph topologies obtained from the edge addition algorithms.

7. Conclusion & Outlook

In the current chapter, we summarize the main observations for the graph addition problem.

The work done here can be regarded as an extension to the work in Ref. [16] in which we consider cases for which the quality of the noise is not known ahead of time. As seen, the algorithm can be easily extended to include this. We have then applied the algorithms to first a cycle consisting of 10 nodes and afterwards a random graph consisting of 20 nodes. Application to the cycle graph has shown that nodes which are not close to each other are connected in the first stage, this is also the case for the extension as shown in fig. 6.1. This is in agreement with the result obtained in Ref. [16]. In the case we know the measurement quality, then all the 'good' edges are first added and then the 'bad' edges follows as can be seen in fig. 6.5. Furthermore, we have shown that how well we do when we do not know the measurement quality ahead of time is based on the quality of the measurements which are either revealed to us after each iteration or when all the edges are added. Hence no conclusion can be drawn regarding which algorithm performs the best. We have also considered the application of the algorithms to a random graph. Here we observe again that vertices which initially have a low degree are more likely to be chosen to be linked. From fig. 6.5, it can be observed that jumps occur when an direct edges is added consisting of the node considered, while the variance stays 'flat' when no direct connection is made to the considered node. Note that this is the case for algorithm 10.

As a last step, we combine both the estimation and the edge addition in one for the cycle graph. We see that for the WLS approach, the values from the edge addition and the estimation case coincides, which is not surprising. We also observe that the EM approach performs worse when the added edges to the graph are 'bad' edges.

7.1 Outlook

In the current work, we have extended the greedy algorithm to the case for which information regarding the edge quality is not known ahead of time. We have observed that eventually the outcome is dependent on the actual quality of the edges and hence we cannot conclude which algorithm performs best. It is of interest to explore more in depth a way to predict the performance.

In the literature, another relaxation used when dealing with combinatorial problems is the convex relaxation. In Ref. [14], for the maximization of the algebraic connectivity, the convex relaxation and a greedy heuristic were compared to each other. We may consider a similar comparison for the A-criterion; this as it is known, the combinatorial approach is not feasible for larger graphs.

Another interesting direction can be to look into minimizing the MSE for the WLSP approach and observe the effect on the other approaches.

Part III

Appendices

A. Graph Theory

In this appendix, we provide some background material in graph theory. The purpose is to give the interested reader, who previously was not exposed to this area, a basic knowledge in order to follow the content presented in this report. The main references consulted are [14, 33, 35]. Basic knowledge in linear algebra and calculus is assumed.

Graph theory is the branch of mathematics that deals with networks [35]. A graph is simply stated a collection of vertices joined together in pairs by edges. In the following, a formal definition, taken from Ref. [36], is given.

Definition A.1 (Graph, [36]). *A graph $\mathcal{G}$ is a triple consisting of a vertex set $\mathcal{V}(\mathcal{G})$, an edge set $\mathcal{E}(\mathcal{G})$, and a relation that associates with each edge two vertices (not necessarily distinct) called its endpoints.*

The cardinality of the vertex set $\mathcal{V}(\mathcal{G})$ is $\left|\mathcal{V}(\mathcal{G})\right| = N$ and of the edge set $\mathcal{E}(\mathcal{G})$ is $\left|\mathcal{E}(\mathcal{G})\right| = M$. In the current report, we consider graphs having the following properties :

- simple; A graph is simple when it has no self-edges or multi-edges. Self-edges are edges which connect a vertex to itself. Multi-edges are the collection of edges (> 1) present between a pair of vertices.
- connected; A graph is connected when starting from an arbitrary vertex, by following the edges, every other vertex can be reached. If this is not the case, then it is disconnected.
- undirected; A graph is undirected when no specific orientation is assigned to each edge.
- weighted; A graph is weighted when a non-negative real number is assigned to each edge indicating its strength. In case the same value is assigned to all the edges, then it can be considered as a constant times a graph with weights all equal to one (we call this graph the unweighted graph).

In the following, when we refer to a graph, we assume it to be simple, connected and undirected. Distinction will be made between the unweighted and the weighted graph. In fig. A.1, an example of an unweighted graph is given. The graph consist of 8 vertices and 12 edges.

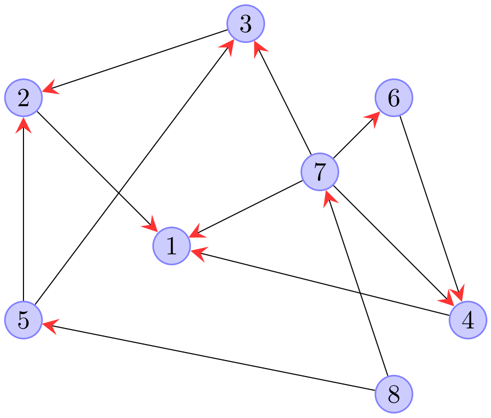


Figure A.1: A small unweighted graph consisting of 8 vertices and 12 edges. The vertex set is $\mathcal{V}(\mathcal{G}) = \{1,2,3,4,5,6,7,8\}$ and the edge set is $\mathcal{E}(\mathcal{G}) = \{(2,1),(4,1),(7,1),(3,2),(5,2),(5,3),(7,3),(6,4),(7,4),(8,5),(7,6),(8,7)\}$. The arrows in red indicate an orientation and is chosen arbitrary. It will be useful for setting up the edge-vertex incidence matrix described in A.3.

A.1 The Adjacency Matrix

One of the ways to mathematically represent a graph is by means of the adjacency matrix $B \in \mathbb{R}^{N \times N}$ with elements B_{ij} having values

$$B_{ij} = \begin{cases} 1 & \text{if there is an edge between vertices } i \text{ and } j \\ 0 & \text{otherwise.} \end{cases} \tag{A.1}$$

This representation holds for an unweighted graph. For a weighted graph, the weights (indicating the strength) for the corresponding edges are substituted for the value 1, i.e.,

$$B^W_{ij} = \begin{cases} w_{ij} & \text{if there is an edge between vertices } i \text{ and } j \\ 0 & \text{otherwise.} \end{cases} \tag{A.2}$$

with the superscript W indicating a weighted graph. The adjacency matrix of a graph has the following properties. First, the diagonal elements B^W_{ii} are zeros (because we assume the graph to be simple and hence it has no self-edges) and second, the adjacency matrix is symmetric, i.e., $B^W = \left(B^W\right)^T$ with the superscript T denoting the transpose of a matrix. (Note, we will use the weighted version when the property holds for both the unweighted and the weighted case. The unweighted case can be easily obtained by setting without loss of generality all the weights w_{ij} to 1). This can be derived from the fact that if there is an edge between the vertex i and j, then there is also an edge between j and i due to assumption of undirected graphs.

A.2 The Degree Matrix

The degree of a vertex in an unweighted graph is the number of edges connected to it. It can be written in terms of the adjacency matrix B as

$$d_i = \sum_{j=1}^{N} B_{ij} \tag{A.3}$$

with d denoting the degree and subscript i the ith vertex. The sum of the degrees of all the vertices equals twice the number of edges,

$$\sum_{i}^{N} k_i = 2M. \tag{A.4}$$

Note that this holds only for unweighted graphs. In the weighted case, the sum of the weights connected to a vertex can be regarded as the strength of the vertex, see Ref. [33]:

$$s_i = \sum_{j=1}^{N} B^W_{ij} \tag{A.5}$$

The degree matrix is a diagonal matrix with on the diagonal entries, $D_{ii} = d_i$ for the unweighted case and $D^W_{ii} = s_i$ for the weighted case, and all the other entries are zeros.

A.3 The Graph Laplacian Matrix

Apart from the adjacency matrix B, another matrix that fully characterize the graph [33] is the graph laplacian matrix L. It is related to the adjacency matrix B and degree matrix D by the following equation

$$L^{(W)} = D^{(W)} - B^{(W)}. \tag{A.6}$$

The elements of the unweighted graph laplacian are thus

$$L_{ij} = \begin{cases} k_i & \text{if } i = j \\ -1 & \text{if } i \neq j \text{ and there is an edge } (i,j) \\ 0 & \text{otherwise.} \end{cases} \tag{A.7}$$

In the weighted setting, we have

$$L_{ij}^W = \begin{cases} s_i & \text{if } i = j \\ -w_{ij} & \text{if } i \neq j \text{ and there is an edge } (i,j) \\ 0 & \text{otherwise.} \end{cases} \tag{A.8}$$

For ease of notation, when we write laplacian, we refer to the graph laplacian matrix. Another way to construct the laplacian is by means of the incidence matrix A. In this report, we adopt the convention $A \in \mathbb{R}^{M \times N}$, i.e., each row encodes an edge. This is called an edge-vertex incidence matrix. In the literature, the transpose, vertex-edge incidence matrix $A \in \mathbb{R}^{N \times M}$, is also used, see for example Ref. [4, 15, 34]. In order to encode the edges, an orientation needs to be chosen. We assume that $(j,i) \in \mathcal{E}(\mathcal{G})$ only if $i < j$, a convention adopted from Ref. [11]. The edge thus points *from* vertex j *to* vertex i. In fig. A.1, the red arrows indicate the orientation. It should be noted that this orientation is arbitrary chosen. The elements of the matrix A are

$$A_{ei} = \begin{cases} +1 & \text{if } e = (j,i) \\ -1 & \text{if } e = (i,j) \\ 0 & \text{otherwise.} \end{cases} \tag{A.9}$$

The unweighted laplacian is related to the edge-vertex incidence matrix by

$$L = A^T A = \sum_e a_e^T a_e. \tag{A.10}$$

in which a_e is the eth row of the incidence matrix corresponding to the edge $e = (i,j)$. In the weighted case, it is

$$L^W = A^T W A = \sum_e w_{ij} a_e^T a_e. \tag{A.11}$$

with $W \in \mathbb{R}^{M \times M}$ being the diagonal matrix of weights w_{ij} corresponding to edge e.

The adjacency, degree, edge-vertex incidence and the graph laplacian for the graph in fig. A.1

is

$$B = \begin{bmatrix} 0 & 1 & 0 & 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 1 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 \end{bmatrix} \quad L = \begin{bmatrix} 3 & -1 & 0 & -1 & 0 & 0 & -1 & 0 \\ -1 & 3 & -1 & 0 & -1 & 0 & 0 & 0 \\ 0 & -1 & 3 & 0 & -1 & 0 & -1 & 0 \\ -1 & 0 & 0 & 3 & 0 & -1 & -1 & 0 \\ 0 & -1 & -1 & 0 & 3 & 0 & 0 & -1 \\ 0 & 0 & 0 & -1 & 0 & 2 & -1 & 0 \\ -1 & 0 & -1 & -1 & 0 & -1 & 5 & -1 \\ 0 & 0 & 0 & 0 & -1 & 0 & -1 & 2 \end{bmatrix}$$

$$D = \begin{bmatrix} 3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 3 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 3 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 3 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 3 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 5 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 2 \end{bmatrix} \quad A = \begin{bmatrix} 1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & -1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & -1 & 0 \\ 0 & 1 & -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & -1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & -1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 & 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & -1 \\ 0 & 0 & 0 & 0 & 0 & 1 & -1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 \end{bmatrix}$$

The laplacian is a symmetric positive semidefinite matrix , i.e., $L^W = \left(L^W\right)^T$ (symmetric) and $L^W \succeq 0$ (positive semidefinite, PSD). From this, we can infer that the eigenvalues of the laplacian are all real and non-negative. We can observed that $L^W\mathbf{1} = 0$, so zero is an eigenvalue of L^W and the corresponding eigenvector is a vector of ones, $\mathbf{1} = \begin{bmatrix} 1 & 1 & \dots & 1 \end{bmatrix}^T$. Let $\lambda_1 \leq \lambda_2 \leq \cdots \leq \lambda_N$ be the eigenvalues of the laplacian, then $\lambda_1 = 0$ as we know that the eigenvalues are all non-negative. In fact, the laplacian has always at least one zero eigenvalue. In Ref. [33], it is shown that the multiplicity of the eigenvalue zeros equals the number of components.

Definition A.2 (Component [35]). *A component of a graph is a subset of the vertex set $\mathcal{V}(\mathcal{G})$ in which there exist at least one path for each member within the subset to another member in the subset and such that no other vertex in $\mathcal{V}(\mathcal{G})$ can be added to the subset while preserving the property.*

As a consequence, for a graph to be connected (consisting of only one component), the second smallest eigenvalue, λ_2 needs to be positive [33, 35]. This second smallest eigenvalue of the laplacian is called the algebraic connectivity of the graph. As seen earlier, it has a direct relation to the number of connected components [14, 33]. Furthermore, λ_2 is considered to be a measure of the connectivity of a graph as it is monotone increasing in the edge set [14]. In fact, from Ref. [33], it is known that adding an edge or increasing the edge weight does not decrease the laplacian eigenvalues. This result is in particular used in part II of this report. λ_2 has also connection to among others the sparsity of cuts in the graph, linear embeddings of graphs, etc., see Ref. [14].

The sum of the inverse of the nonzero laplacian eigenvalues, $\sum_{i=2}^{n} \lambda_i^{-1}$, is another quantity that has interesting interpretation. In Ref. [15, 33], this quantity is known as the mean effective resistance of a resistive network where the edge weights are considered to be resistances between vertices. The total effective resistance is then $R_{\text{tot}} = n \sum_{i=2}^{n} \lambda_i^{-1}$. The total effective resistance is also considered as a measure for quantifying how connected a network is, see Ref. [33]. Other applications of R_{tot} can be found in for example Ref. [15].

B. The Moore-Penrose Pseudoinverse

The Moore-Penrose pseudoinverse, denoted by $(\,)^\dagger$, is a generalization of the inverse of a matrix and is defined for any (possibly rectangular) matrix. A purely algebraic characterization of this pseudoinverse is given in the next theorem by Penrose [37].

Theorem B.1 (Moore-Penrose Pseudoinverse [37]). *Let $A \in \mathbb{R}^{m\times n}$. Then $G = A^\dagger$ if and only if*

1. *$AGA = A$;*
2. *$GAG = G$;*
3. *$(AG)^* = AG$;*
4. *$(GA)^* = GA$.*

Furthermore, $A^\dagger$ always exists and is unique.

In the following, some useful properties of the Moore-Penrose pseudoinverse are presented. These are obtained through consulting, among others, Ref. [37–39].

1. $A^{\dagger\dagger} = A$;
2. $A^{*\dagger} = A^{\dagger *}$;
3. if A is non-singular, then $A^\dagger = A^{-1}$;
4. $(\lambda A)^\dagger = \lambda^\dagger A^\dagger$;
5. $A^\dagger = (A^*A)^\dagger A^* = A^*(AA^*)^\dagger$;
6. If $A \in \mathbb{R}_r^{r\times m}$, in which case A is onto, then $A^\dagger = A^T\big(AA^T\big)^{-1}$;
7. if $A \in \mathbb{R}_r^{n\times r}$, in which case A is 1-1, then $A^\dagger = \big(A^TA\big)^{-1}A^T$;
8. $A^\dagger AA^* = A^*AA^\dagger = A^*$;
9. $AA^*A^{*\dagger} = A^{*\dagger}A^*A = A$.

with $(\,)^*$ denoting the conjugate transpose of a complex matrix. In case the elements of the matrix are all real numbers, then the conjugate transpose reduces to a transpose operation $(\,)^T$.

In Ref. [40], a representation for the generalized inverse of the product of matrices is presented. The following theorem is the main result;

Theorem B.2 (Generalized Inverse of Matrix Product [40]). *$(AB)^\dagger = B_1^\dagger A_1^\dagger$, where $B_1 = A^\dagger AB$ and $A_1 = AB_1B_1{}^\dagger$.*

In the following special cases [39], $(AB)^\dagger = B^\dagger A^\dagger$ holds, i.e., $B_1 = B$ and $A_1 = A$:

1. if $A \in \mathbb{R}_r^{n\times r}, B \in \mathbb{R}_r^{r\times m}$, i.e., A is full column rank and B is full row rank.;
2. if $B = A^*$;
3. if $B = A^\dagger$.

C. Normal Probability Distribution

C.1 General Normal Distribution

Note: The current section and the next one are taken from Ref. [41].
The general normal distribution is characterized by two parameters, μ and σ^2, which are in fact the mean and variance of the distribution. The probability density function (pdf) $\phi(x;\mu,\sigma^2)$ or $\mathcal{N}(\mu,\sigma^2)$ is given by

$$\phi(x;\mu,\sigma^2) = \frac{1}{\sigma\sqrt{2\pi}} \exp^{-\frac{(x-\mu)^2}{2\sigma^2}}, \tag{C.1}$$

and the cumulative distribution function (cdf) $\Phi(x;\mu,\sigma^2)$ by

$$\Phi(x;\mu,\sigma^2) = \int_{-\infty}^{x} \frac{1}{\sigma\sqrt{2\pi}} \exp^{-\frac{(t-\mu)^2}{2\sigma^2}} \, \mathrm{d}t. \tag{C.2}$$

Note that for the general normal distribution, we have $-\infty \leq x \leq \infty$.

The normal pdf with $\mu = 0$ and $\sigma^2 = 1$ is called the standard normal distribution $\phi(x;0,1)$. Using the following equation, every general normal distribution can be mapped to the standard normal distribution:

$$\xi = \frac{x-\mu}{\sigma}. \tag{C.3}$$

Look-up tables exist for the cumulative standard normal distribution, see for example Ref. [25]. Next, The kth moment of the general normal pdf is given by the following equation:

$$\mu_k^{'}(\phi(*;\mu,\sigma^2)) = \frac{1}{\sigma\sqrt{2\pi}} \int_{-\infty}^{\infty} x^k \exp^{-\frac{(x-\mu)^2}{2\sigma^2}} \, \mathrm{d}t \tag{C.4}$$

which can be rewritten as

$$\mu_k^{'} = \sum_{j=0}^{\lfloor k/2 \rfloor} \binom{k}{2j} (2j-1)!!\sigma^{2j}\mu^{k-2j} \tag{C.5}$$

in which $\lfloor k/2 \rfloor$ is the floor function, rounding the value down to an integer and $(2j-1)!!$ is the double factorial function which is here the product of odd numbers starting from 1 to $2j-1$ (Note: $(-1)!! = 1$ and $0!! = 1$). When $k = 0$, we are calculating the area of the pdf which is equal to 1 and for $k = 1$, we have $\mu_1^{'} = \mu$.

C.2 Two-sided Truncated Normal Distribution

The truncated normal distribution can be obtained by putting bounds on the general normal pdf. The two-sided truncated normal distribution $\phi_{\mathrm{TR}}(x;\mu,\sigma^2)$ can be given by the following formula,

$$\phi_{\mathrm{TR}}(x;\mu,\sigma^2) = \begin{cases} \dfrac{\phi(x;\mu,\sigma^2)}{\Phi(b;\mu,\sigma^2) - \Phi(a;\mu,\sigma^2)} & \text{if } a \leq x \leq b \\ 0 & \text{if } x < a \text{ or } x > b \end{cases} \tag{C.6}$$

Here, the parameters μ and σ^2 are parameters from the 'parent' general normal pdf and a, b are the left and right boundary of the range we are interested in. The denominator $\Phi(b;\mu,\sigma^2) -$

$\Phi(a;\mu,\sigma^2)$ is a scaling constant in order to have the total integral in the range $[a,b]$ to be equal to 1. The mean and the variance of the truncated normal distribution are given by

$$\begin{aligned}\mu_{\mathrm{TR}} &= \mu - \sigma\frac{\phi(\tilde{b};0,1)-\phi(\tilde{a};0,1)}{\Phi(\tilde{b};0,1)-\Phi(\tilde{a};0,1)}\\ \sigma^2_{\mathrm{TR}} &= \sigma^2\left(1-\frac{\tilde{b}\phi(\tilde{b};0,1)-\tilde{a}\phi(\tilde{a};0,1)}{\Phi(\tilde{b};0,1)-\Phi(\tilde{a};0,1)}\right)-\left(\frac{\phi(\tilde{b};0,1)-\phi(\tilde{a};0,1)}{\Phi(\tilde{b};0,1)-\Phi(\tilde{a};0,1)}\right)^2\end{aligned} \tag{C.7}$$

with $\tilde{a}$ and $\tilde{b}$ obtained using eq. (C.3)

$$\tilde{a}=\frac{a-\mu}{\sigma},\quad \tilde{b}=\frac{b-\mu}{\sigma}. \tag{C.8}$$

The kth moment of the truncated normal distribution is the quantity

$$\mu'_{\mathrm{TR}_k}=\int_a^b x^k\phi_{\mathrm{TR}}(x;\mu,\sigma^2)\,\mathrm{d}x \tag{C.9}$$

which can be reformulated as

$$\mu'_{\mathrm{TR}_k}=\sum_{i=0}^{k}\binom{k}{i}\sigma^i\mu^{k-i}L_i \tag{C.10}$$

with L_i satisfying

$$\begin{aligned}L_0 &= 1\\ L_1 &= \frac{\phi(\tilde{b};0,1)-\phi(\tilde{a};0,1)}{\Phi(\tilde{b};0,1)-\Phi(\tilde{a};0,1)}\\ L_i &= -\frac{\tilde{b}^{i-1}\phi(\tilde{b};0,1)-\tilde{a}^{i-1}\phi(\tilde{a};0,1)}{\Phi(\tilde{b};0,1)-\Phi(\tilde{a};0,1)}+(i-1)L_{i-2}\quad \text{for } i\geq 2.\end{aligned} \tag{C.11}$$

The mean μ_{TR} and the variance σ^2_{TR} can be rewritten as

$$\begin{aligned}\mu_{\mathrm{TR}} &= \mu-\sigma L_1\\ \sigma^2_{\mathrm{TR}} &= \sigma^2 L_2 - L_1^2.\end{aligned} \tag{C.12}$$

C.3 Two-sided Normal Tail Distribution

Note: The current subsection is inspired by Ref. [41].
The two-sided normal tail distribution $\phi_{\mathrm{Tail}}(x;\mu,\sigma^2)$ is defined as to be the complement of the two-sided truncated normal distribution, i.e.,

$$\phi_{\mathrm{Tail}}(x;\mu,\sigma^2)=\begin{cases}\dfrac{\phi(x;\mu,\sigma^2)}{1-(\Phi(b;\mu,\sigma^2)-\Phi(a;\mu,\sigma^2))} & \text{if } x<a \text{ or } x>b\\ 0 & \text{if } a\leq x\leq b.\end{cases} \tag{C.13}$$

It can be readily seen that $\phi_{\mathrm{TR}}(x;\mu,\sigma^2)+\phi_{\mathrm{Tail}}(x;\mu,\sigma^2)=\phi(x;\mu,\sigma^2)$. The mean and variance of the two-sided normal tail distribution are obtained using the general normal pdf and the truncated normal pdf.

Theorem C.1 (Mean and Variance of the Two-sided Normal Tail Distribution). *Let the two-sided normal tail distribution be defined as in eq.* (C.13) *with μ and σ^2 being parameters of the general normal pdf, then*

$$\begin{aligned}\mu_{Tail} &= \frac{\mu-\mathbb{X}(\mu-\sigma L_1)}{(1-\mathbb{X})}\\ \sigma^2_{Tail} &= \frac{-4(1-\mathbb{X})\mathbb{X}\mu\sigma L_1+\left(1-(1+L2)\mathbb{X}+\left(L_2-L_1^2\right)\mathbb{X}^2\right)\sigma^2}{(1-\mathbb{X})^2}.\end{aligned} \tag{C.14}$$

with $\mathbb{X}=\left(\Phi(b;\mu,\sigma^2)-\Phi(a;\mu,\sigma^2)\right)$ and the L_is defined in eq. (C.11).

Proof. First, we turn our attention to the mean. By definition, the mean is the first moment of the probability distribution. For the general normal pdf, it is

$$\begin{aligned}\mu &= \int_{-\infty}^{\infty} x\phi(x;\mu,\sigma^2)\,\mathrm{d}x \\ &= \int_{-\infty}^{a} x\phi(x;\mu,\sigma^2)\,\mathrm{d}x + \int_{a}^{b} x\phi(x;\mu,\sigma^2)\,\mathrm{d}x + \int_{b}^{\infty} x\phi(x;\mu,\sigma^2)\,\mathrm{d}x \\ &= \underbrace{(1-\mathbb{X})\left(\int_{-\infty}^{a} x\phi_{\text{Tail}}(x;\mu,\sigma^2) + \int_{b}^{\infty} x\phi_{\text{Tail}}(x;\mu,\sigma^2)\right)}_{\mu_{\text{Tail}}} + \mathbb{X}\underbrace{\int_{a}^{b} x\phi_{\text{TR}}(x;\mu,\sigma^2)}_{\mu_{\text{TR}}}\end{aligned}$$

with $\mathbb{X} = \left(\Phi(b;\mu,\sigma^2) - \Phi(a;\mu,\sigma^2)\right)$. Plugging the quantity for μ_{TR} and rearranging, we have

$$\begin{aligned}\mu_{\text{Tail}} &= \frac{\mu - \mathbb{X}\mu_{\text{TR}}}{(1-\mathbb{X})} \\ &= \frac{\mu - \mathbb{X}(\mu - \sigma L_1)}{(1-\mathbb{X})}.\end{aligned}$$

The variance of ϕ_{Tail} is calculated using the moments equation which in turn is related to ϕ and ϕ_{TR}. We know that for univariate distribution, the variance of a random variable X can be given by the alternative equation

$$\mathbb{E}\left[(X-\mu_X)^2\right] = \mathbb{E}\left[X^2\right] - (\mu_X)^2. \tag{C.15}$$

The term on the left hand side is the second central moment of the pdf and equals σ_X^2 while the first term on the right hand side is the second moment of the pdf. In a similar fashion as the first moment (which is the mean), the second moment of ϕ_{Tail} can be expressed in terms of the second moments of ϕ and ϕ_{TR}.

The second moment for the general normal pdf ϕ can be expressed as

$$\begin{aligned}\mu_2^{'} &= \binom{2}{0}(0-1)!!\sigma^0\mu^{2-0} + \binom{2}{2}(2-1)!!\sigma^2\mu^{2-2} \\ &= \mu^2 + \sigma^2.\end{aligned}$$

while for ϕ_{TR}, it is

$$\begin{aligned}\mu_{\text{TR}_2}^{'} &= \binom{2}{0}\sigma^0\mu^{2-0}L_0 + \binom{2}{1}\sigma^1\mu^{2-1}L_1 + \binom{2}{2}\sigma^2\mu^{2-2}L_2 \\ &= \mu^2 + 2\sigma\mu L_1 + \sigma^2 L_2\end{aligned}$$

Hence, in a similar fashion as the mean, we have

$$\begin{aligned}\mu_{\text{Tail}_2}^{'} &= \frac{\mu_2^{'} - \mathbb{X}\mu_{\text{TR}_2}^{'}}{(1-\mathbb{X})} \\ &= \frac{(\mu^2+\sigma^2) - \mathbb{X}(\mu^2 + 2\sigma\mu L_1 + \sigma^2 L_2)}{(1-\mathbb{X})}.\end{aligned}$$

and gathering the results yields

$$\begin{aligned}\sigma_{\text{Tail}}^2 &= \frac{(\mu^2+\sigma^2) - \mathbb{X}(\mu^2 + 2\sigma\mu L_1 + \sigma^2 L_2)}{(1-\mathbb{X})} - \left(\frac{\mu - \mathbb{X}(\mu - \sigma L_1)}{(1-\mathbb{X})}\right)^2 \\ &= \frac{-4(1-\mathbb{X})\mathbb{X}\mu\sigma L_1 + \left(1 - (1+L2)\mathbb{X} + \left(L_2 - L_1^2\right)\mathbb{X}^2\right)\sigma^2}{(1-\mathbb{X})^2}.\end{aligned}$$

which is the desired result. □

Bibliography

[1] P. Rawat, K. D. Singh, H. Chaouchi, and J. M. Bonnin, "Wireless sensor networks: a survey on recent developments and potential synergies," *The Journal of supercomputing*, vol. 68, no. 1, pp. 1–48, 2014.

[2] J. Yick, B. Mukherjee, and D. Ghosal, "Wireless sensor network survey," *Computer Networks*, vol. 52, no. 12, pp. 2292 – 2330, 2008.

[3] S. Shue and J. M. Conrad, "A survey of robotic applications in wireless sensor networks," in *Southeastcon, 2013 Proceedings of IEEE*, pp. 1–5, April 2013.

[4] P. Barooah and J. Hespanha, "Estimation on Graphs from Relative Measurements," *Control Systems, IEEE*, vol. 27, pp. 57–74, Aug 2007.

[5] P. Barooah, N. da Silva, and J. Hespanha, "Distributed Optimal Estimation from Relative Measurements for Localization and Time Synchronization," in *Distributed Computing in Sensor Systems* (P. Gibbons, T. Abdelzaher, J. Aspnes, and R. Rao, eds.), vol. 4026 of *Lecture Notes in Computer Science*, pp. 266–281, Springer Berlin Heidelberg, 2006.

[6] P. Barooah and J. Hespanha, "Estimation From Relative Measurements: Electrical Analogy and Large Graphs," *Signal Processing, IEEE Transactions on*, vol. 56, pp. 2181–2193, June 2008.

[7] B. Osting, C. Brune, and S. J. Osher, "Optimal Data Collection For Informative Rankings Expose Well-Connected Graphs," *Journal of Machine Learning Research*, vol. 15, pp. 2981–3012, 2014.

[8] P. Frasca, C. Ravazzi, R. Tempo, and H. Ishii, "Gossips and prejudices: ergodic randomized dynamics in social networks," in *Estimation and Control of Networked Systems - Proceedings of the 4th IFAC Workshop on Distributed Estimation and Control in Networked Systems , Koblenz, Germany*, vol. 4, pp. 212–219, International Federation of Automatic Control, September 2013.

[9] C. Ravazzi, P. Frasca, H. Ishii, and R. Tempo, "A distributed randomized algorithm for relative localization in sensor networks," in *Control Conference (ECC), 2013 European*, pp. 1776–1781, July 2013.

[10] C. Ravazzi, P. Frasca, R. Tempo, and H. Ishii, "Almost sure convergence of a randomized algorithm for relative localization in sensor networks," in *Decision and Control (CDC), 2013 IEEE 52nd Annual Conference on*, pp. 4778–4783, Dec 2013.

[11] W. Rossi, P. Frasca, and F. Fagnani, "Transient and limit performance of distributed relative localization," in *Decision and Control (CDC), 2012 IEEE 51st Annual Conference on*, pp. 2744–2748, Dec 2012.

[12] H. Hu and G. Kantor, "Parametric covariance prediction for heteroscedastic noise," in *Intelligent Robots and Systems (IROS), 2015 IEEE/RSJ International Conference on*, pp. 3052–3057, Sept 2015.

[13] E. D. Niri and T. Singh, "Unscented Transformation based estimation of parameters of nonlinear models using heteroscedastic data," *Pattern Recognition*, pp. –, 2016.

[14] A. Ghosh and S. Boyd, "Growing Well-connected Graphs," in *Decision and Control, 2006 45th IEEE Conference on*, pp. 6605–6611, Dec 2006.

[15] A. S. Arpita Ghosh, Stephen Boyd, "Minimizing Effective Resistance of a Graph," *SIAM Review*, vol. 50, no. 1, pp. 37–66, 2008.

[16] T. Summers, I. Shames, J. Lygeros, and F. Dorfler, "Topology design for optimal network coherence," in *Control Conference (ECC), 2015 European*, pp. 575–580, July 2015.

[17] P. Molavi and A. Jadbabaie, "A topological view of estimation from noisy relative measurements," in *American Control Conference (ACC), 2011*, pp. 3615–3620, June 2011.

[18] F. M. Dekking, *A Modern Introduction to Probability and Statistics: Understanding why and how*. Springer Science & Business Media, 2005.

[19] G. Casella and R. L. Berger, *Statistical Inference*, vol. 2. Duxbury Pacific Grove, CA, 2002.

[20] C. D. Meyer, *Matrix analysis and applied linear algebra*, vol. 2. SIAM, 2000.

[21] J. Humpherys and J. West, "Kalman Filtering with Newton's Method [Lecture Notes]," *Control Systems, IEEE*, vol. 30, pp. 101–106, Dec 2010.

[22] I. J. Myung, "Tutorial on maximum likelihood estimation," *Journal of Mathematical Psychology*, vol. 47, no. 1, pp. 90 – 100, 2003.

[23] M. Blume, "Expectation Maximization: A Gentle Introduction." `http://campar.in.tum.de/twiki/pub/Main/MoritzBlume/EMGaussianMix.pdf`, 2008.

[24] G. James, D. Witten, T. Hastie, and R. Tibshirani, *An Introduction to Statistical Learning*, vol. 112. Springer, 2013.

[25] D. C. Montgomery and G. C. Runger, *Applied Statistics and Probability for Engineers*. John Wiley & Sons, 2010.

[26] C. B. Do and S. Batzoglou, "What is the expectation maximization algorithm?," *Nature biotechnology*, vol. 26, no. 8, pp. 897–900, 2008.

[27] M. R. Gupta and Y. Chen, *Theory and use of the EM algorithm*. Now Publishers Inc, 2011.

[28] C. Biernacki, G. Celeux, and G. Govaert, "Choosing starting values for the EM algorithm for getting the highest likelihood in multivariate Gaussian mixture models," *Computational Statistics & Data Analysis*, vol. 41, no. 3-4, pp. 561 – 575, 2003. Recent Developments in Mixture Model.

[29] D. Karlis and E. Xekalaki, "Choosing initial values for the EM algorithm for finite mixtures," *Computational Statistics & Data Analysis*, vol. 41, no. 3-4, pp. 577 – 590, 2003. Recent Developments in Mixture Model.

[30] N. Bof, G. Baggio, and S. Zampieri, "On the Role of Network Centrality in the Controllability of Complex Networks," *arXiv preprint arXiv:1509.04154*, 2015.

[31] L. Xu and M. I. Jordan, "On Convergence Properties of the EM Algorithm for Gaussian Mixtures," *Neural computation*, vol. 8, no. 1, pp. 129–151, 1996.

[32] R. Aragues, L. Carlone, C. Sagues, and G. Calafiore, "Distributed centroid estimation from noisy relative measurements," *Systems & Control Letters*, vol. 61, no. 7, pp. 773 – 779, 2012.

[33] W. Ellens, "Effective Resistance," Master's thesis, University of Leiden, 2011.

[34] A. Clark and R. Poovendran, "A submodular optimization framework for leader selection in linear multi-agent systems," in *Decision and Control and European Control Conference (CDC-ECC), 2011 50th IEEE Conference on*, pp. 3614–3621, Dec 2011.

[35] M. Newman, *Networks: An Introduction*. Oxford University Press, 2010.

[36] D. B. West, *Introduction to Graph Theory*. Pearson, 2 ed., 2001.

[37] R. Penrose, "A Generalized Inverse for Matrices," in *Mathematical Proceedings of the Cambridge Philosophical Society*, vol. 51, pp. 406–413, Cambridge Univ Press, 1955.

[38] A. J. Laub, *Matrix Analysis for Scientists and Engineers.* Siam, 2005.

[39] T. N. E. Greville, "Note on the Generalized Inverse of a Matrix Product," *SIAM Review*, vol. 8, no. 4, pp. 518–521, 1966.

[40] R. E. Cline, "Note on the Generalized Inverse of the Product of Matrices," *SIAM Review*, vol. 6, no. 1, p. 57, 1964.

[41] J. Burkardt, "The Truncated Normal Distribution." `http://people.sc.fsu.edu/~jburkardt/presentations/truncated_normal.pdf`, October 2014.