Sensor Network Localization via Schatten Quasi-Norm Minimization: An Interior-Point Approach

SZE, Kam Fung

A Thesis Submitted in Partial Fulfilment
of the Requirements for the Degree of
Master of Philosophy
in
Systems Engineering and Engineering Management

The Chinese University of Hong Kong
September 2013

Abstract

With dramatic increase on mobile devices, efficient and inexpensive positioning is crucial in various applications. In wireless sensor network, localization problem is essential as information without location cannot be well utilized. Sensor Network Localization(SNL) is simply determining node positions by using pairwise information among nodes. However, this problem has been proven NP-Hard for a fixed dimension.

Several convex-relaxation based algorithms have been proposed yet are not able to localize all nodes in the required dimension tractably. Semidefinite Programming(SDP) is one of these common relaxation tool and its modeling power to SNL has been demonstrated under various setting. The constraint relaxed by SDP to SNL is the rank constraint of decision matrix, i.e. rank(X) = d, where d is usually two or three.However, standard SDP solver would return the maximum rank solution, thus relaxation cannot be exact.

Based on the work of Biswas and Ye, we proposed a non-convex framework, namely Schatten p regularization, by introducing a regularizer term to penalize the high-rank solutions. A polynomial-time potential reduction algorithm will also be presented to find a global minimizer or an ϵ-first order critical point. Our numerical results show that the proposed algorithm could perform better and act as a refinement step for rank reduction. Besides, some practical issues and heuristic methods, which are combined with some existing methods, would be described.

摘要

隨著手提裝置的數量激增，很多應用都需要高效率和成本低的定位系統。如果沒有感測器的確實位置，無線感測網路收集得來的資料便不能有效地使用。簡單來說，無線感測網路的定位問題就是利用感測器之間的數據來找出每個感測器的位置。然而，這個問題已被証明是非決定性多項式時間困難(NP-hard)。

雖然現在已經有數個以凸鬆弛作為基礎的算法發怖，但是卻不是每一個問題都能找出一個在指定空間下的定位。半正定規劃就是其中一個主要鬆弛的工具，在很多的設定下，它亦能表達出鬆弛化的問題。可惜，一般半正定規劃的算法都會傳回最大秩的答案，相反我們一般都是在二維和三維空間定位，秩需要是二或三才能符合原本的問題。

我們在半正定規劃鬆弛模型的基礎上提出一個加入了對秩懲罰的非凸性的模型，希望藉此對可行域中較高秩的點作出處罰。同時，我們亦找到一個可以在多項式時間內解出一階極值點或極值點的算法。在不同的測試中，這個算法可以更準確地找出更好的答案。

Thesis/Assessment Committee

Professor Ken Wing Kin MA (Chair)

Professor Anthony Man Cho SO (Thesis Supervisor)

Professor Hong CHENG (Committee Member)

Professor Hing Cheung SO (External Examiner)

ACKNOWLEDGMENTS

I would like to express my deepest gratitude to my advisor Professor Anthony So for his generous support of my study and research, warm encouragement, innovative inspiration, immense patience and precious chance to work with the industry. His expertise in operations research deeply strengthen my skill. Most importantly, his personality and dedication to teach and research has had lifelong influences on my personality and attitude. I would say that without his patient guildence, it is impossible for me to finish this thesis.

I want to give gratitude to Professors Hing-Cheung So, Ken Ma and Hong Cheng for serving on my thesis committee and their valuable comments on my thesis. I would like to thanks Professors Janny Leung and Chun-Hung Cheng for the teaching and encouragement to me throughout my undergraduate and post-graduate study.

My keen appreciation goes to the member of Anthony's research group, including Kuncheng Wang, Ke Hou, Senshan Ji, Yang Liu, Zirui Zhou, Chi Zhang, Frank Wong and Man-Chung Yue, who shares their insightful and gives many fruitful discussions in various topics. I wish to extend my thanks to Man-Chung Yue and Frank Wong for thier supports and sharings throughout my post-graduate study.

Finally, I am grateful to my family, which has supported me over the years. They provided me an peaceful and carefree environment for me to do what I want and pursue my studies. Even though they hardly understand what optimization is, my parents are still willing to support me and never complain about my decision.

Contents

Chapter 1

Introduction

1.1 Background and motivation

A wireless sensor network consists of a large group of distributed sensors, which can monitor different conditions(temperature, sound levels and light etc.) from environment. Sensors are equipped with communication power with others, via short-range Bluetooth and even long-range radio signal. These networks can suit for many monitoring and controlling applications in various industries, such as target tracking [33] and mobile advertising [23]. To facilitate the use of information, locations of each sensors are essential. However, exact locations of these sensors usually are unknown due to the distribution method or random nature of sensors. Although one can install the global positioning system(GPS) on each sensors, it is surely a costly solution no matter in terms of hardware or power consumption [13]. Fortunately, sensors can still obtain extra pairwise information(distance or angle) by communication with their neighbors. Sensor Network Localization(SNL) Problem is to localize all nodes in the network using these pairwise information between nodes. It has been shown to be NP-hard if the localized dimension is fixed [2] [22]. In our setting, we will consider the pairwise distances only and assume that we would have some anchors, that are the sensors

with known locations (which may be achieved by GPS).

There are several approaches to SNL, and they can be classified into the centralized and the distributed [6] [15]. Centralized algorithms, which includes Atomic Multilateration[21], Multidimensional scaling and Optimization-based Methods, gather all information and compute the solution in a central computer. In contrast, distributed algorithms, which includes Beacon-based Localization[17] and Coordinate System Stitching[31], compute the positions of a subset of nodes by partial information. Generally speaking, centralized algorithm can give a optimal solution with higher cost while distributed algorithm can only generate sub-optimal solution but with less power consumption and hardware requirement.

In particular, optimization-based algorithms have been getting attention for its precious solution. The main technique that it relies on is the convex relaxation, which has been extensively studied[3][4][5][7][12][14][19][20][25][27][28][34]. Convex relaxation aims to relax the original problem into a convex programming, which is polynomial–time solvable. And the practical performance for these relaxations is usually accurate. Still, these relaxations are not always exact as fixed–dimensional localization problem is NP-hard[2] [22].

We will focus the studies on the Semidefnite Programming Relaxation(SDR) proposed by Biswas et al.[5]. It first reconstructs the SNL into a rank-constrained Semidefinite Programming feasibility problem, then further relaxes its rank constraint. In other words, if solution of the relaxation is of rank d, which is the localization dimension needed, the relaxation is exact, i.e. the corresponding SNL is solved. Moreover, So et al.[27] have developed a notion of localizable graphs and builded the relationship between the SDP solution and these graphs. However, in many cases, the solution solved by Semidefinite Programming solver is of rank greater than d due to the max-rank property of interior–point algorithms[10]. To achieve a lower rank solution, one can use the regularizer to penalize the high rank solution. A good rank regularizer should maintain tractability of the prob-

lem(e.g. convexity) and can assimilate the rank function. Biswas et al. first introduced the trace regularization to SNL[4]. As well known, trace norm is the convex envelop of the rank function[8], so it is surely a good candidate for SNL. However, its performance is not really satisfied especially when the problem is sparse, i.e. only few pairwise distances are known. It is natural for one to ask if there are other more effective regularizers.

In fact, for positive semidefinite matrix, rank function is the cardinality function of the non-zero eigenvalues while trace norm is actually the sum of eigenvalues. Thus, Schatten p–quasi–norm, which is the sum of eigenvalues power to p for $p \in (0, 1]$,would be a possible candidate as the rank regularizer. Moreover, when $p \to 0$, Schatten p–quasi–norm would tend to the rank function. Besides, in the context of low rank matrices recovery, Schatten p–quasi–norm regularizer can take advantages with some theoretical properties. Upon these reasons, we see that Schatten p–quasi–norm would be a potential regularizer for SNL. However, the Schatten p–quasi–norm regularizered Semidefinite Programming is NP–hard i.e. there is no polynomial time algorithm can get global optimum for all instances. Thus, we develope a tractable potential reduction algorithm that can converge to a 'first-order point' instead of the global optimizer. This thesis aims to introduce this algorithm and analyze its performance on SNL.

In Chapter 2, we begin with mathematical models of SNL in various settings. The first is the SDR proposed by Biswas et al.[5] for the accurate, i.e. exact pairwise distance measurements, and anchor-presence setting. Then, we would move to SDR for the accurate and anchor-absence setting[4]. Third, we would mention the SDR with Gaussian noise measurements and anchor-presence setting[3]. Afterall, we introduce the general framework for Schatten p–quasi–norm regularization of Semidefinite Programming and show that it can be applied for the aforementioned models. In Chapter 3, we would present the algorithm and define the notion of *first–order critical point*. We would further show the correctness

proof and complexity bound on it. Furthermore, we show some properties maintained under Schatten p–quasi–norm regularization. In Chapter 4, we would present the extensive numerical results on them. Moreover, we suggest some heuristic methods on SNL with some numerical testing.

1.2 Preliminaries

1.2.1 Basic Notation

We first denote some notations, in particular, for matrices. Let $X \in \mathbb{R}^{n\times n}$ represent a real number matrix with dimension $n \times n$ while X_{ij} represent the ij entry. Trace function of X is the sum of its diagonal values i.e. $\sum_{i=1}^{n} X_{ii}$, denoted by $tr(X)$. We denote $\sigma_i(X)$ be the largest singular value of X. The Frobenius norm of X is defined as $\|X\|_F = \sqrt{\sum_{i,j} X_{ij}^2}$.The sets of $n \times n$ symmetric, symmetric positive semidefinite and symmetric positive definite matrices are denoted by $\mathcal{S}^n$, $\mathcal{S}^n_+$, $\mathcal{S}^n_{++}$ respectively. Moreover, $Y \succeq X$ represents $Y - X$ is a symmetric positive semidefinite matrix. For a vector $x \in \mathbb{R}^n$, we denote the 2-norm by $\|x\| := \sqrt{\sum_{i=1}^{n} x_i^2}$, where x_i is the i-entry of x. For X belongs to $\mathcal{S}^n_{++}$, the nuclear norm of X is defined as $\|X\|_* = tr(X)$, which is the sum of its all absoluted eigenvalues.

1.2.2 Semidefinite Programming

As Semidefinite Programming is the basic tool in our dissertation, we review its standard form and some theory of it. Here is the basic standard form. Here is the standard form in trace form.

$$\begin{aligned} v_p = \ \text{minimize} \quad & \text{tr}(CX) \\ \text{subject to} \quad & \text{tr}(A_i X) = b_i \ \text{ for } i = 1, \ldots, m, \\ & X \in \mathcal{S}^n_+ \end{aligned} \tag{1.1}$$

where $C, \{A_i\} \in \mathcal{S}^n$ and $\{b_i\}$ are real numbers.

The dual problem of (1.1) is as follows:

$$\begin{aligned} v_d = \ & \text{maximize} && b^T y \\ & \text{subject to} && \sum_{i=1}^m y_i A_i + S = C \\ & && S \in \mathcal{S}_+^n \end{aligned} \tag{1.2}$$

where $b = (b_1, \ldots, b_m)^T$.

The set of equality constraints in (1.1) can be expressed as $\mathcal{A}(X) = b$ with the linear mapping $\mathcal{A}(\cdot) : \mathcal{S}^n \to \mathbb{R}^m$ where $\mathcal{A}(X) := \Big(tr(A_1 X), \ldots, tr(A_m X)\Big)^T$. Moreover, the linear mapping can be written as multiplication $\mathcal{A}(X) = Avec(X)$, where $vec(\cdot)$ is the Vectorization and $A := \Big(vec(A_1), \ldots, vec(A_m)\Big)^T \in \mathbb{R}^{m \times n^2}$. On the other hand, the equality constraint in (1.2) can be expressed as $\mathcal{A}^*(y) + S = C$ with the adjoint operator $A^* : \mathbb{R}^m \to \mathcal{S}^n$ of $\mathcal{A}$. Furthermore, $\mathcal{A}^*(y) := mat(A^T y)$, where $mat(\cdot)$ is the Matricization.

Similar to Linear Programming, there is a duality theorem for SDP. However, this is slightly "weaker" than that of Linear Programming. First, we denote the feasible regions of (1.1) and (1.2) as $\mathcal{F}_p$ and $\mathcal{F}_d$ respectively. Weak duality still holds, that is, objective value of (1.1) is greater than (1.2).

Lemma 1.1. *(weak duality[29][11]) For any $X \in \mathcal{F}_p$ and $(y, S) \in \mathcal{F}_d$,*

$$tr(CX) - b^T y = tr(XS) \geq 0.$$

However, strong duality is not true in all cases.

Lemma 1.2. *Let $X \in \mathcal{F}_p$, $(y, S) \in \mathcal{F}_d$ and $tr(XS) = 0$. Then (X, y, S) is optimal for problems (1.1) and (1.2).*

In general, extra conditions are sufficient to make strong duality to hold.

Lemma 1.3. *(Strong duality[29][11]) Let $\mathcal{F}_p$ and $\mathcal{F}_d$ be nonempty and have an interior feasible solution. Then X is optimal for (1.1) if and only if $X \in \mathcal{F}_p$ and there exists $(y, S) \in \mathcal{F}_d$ such that $tr(CX) - b^T y = tr(XS) = 0$.*

To conclude, the duality theorem are as follows:

Theorem 1.1. *(SDP duality therorem[29][11])*

1. *If (1.1) (resp. (1.2)) is strictly feasible, then we have $v_p^* = v_d^*$. If (1.1) (resp. (1.2)) is bounded below(resp. above), then the common optimal value is attained by some $(y^*, S^*) \in \{(y, S) \in \mathbb{R}^m \times \mathcal{S}_+^n : \mathcal{A}^*(y) + S = C\}$ (resp. $X^* \in \{X \in \mathcal{S}_+^n : \mathcal{A}(X) = b\}$).*

2. *Suppose that both (1.1) or (1.2) are bounded and strictly feasible. Then, we have $v_p^* = v_d^*$. The common optimal value is attained by some primal feasible X^* and dual feasible (y^*, S^*). Moreover, each of the following conditions is necesary and sufficient for optimality of a primal-dual feasible pair.*

   (a) (Zero Duality Gap) $tr(CX) = b^T y$

   (b) (Complementary Slackness) $tr(XS) = 0$

3. *If (1.1) (resp. (1.2)) is unbounded, then (1.2) (resp. (1.1)) has no feasible solution.*

1.2.3 Schatten p–quasi–norm

Given a matrix $X \in \mathbb{R}^{n \times n}$, for any given $p \in (0, 1]$, the Schatten p–quasi–norm is defined as $\|X\|_p^p := \left(\sum_{i \geq 1} |\sigma_i(Z)|^p \right)$. In this dissertation, the Schatten p–quasi–norm is applied on the Positive definite matrices. Given a $Z \in \mathcal{S}_{++}^n$, we can get $Z = U \Lambda U^T$ by spectral decomposition of Z. As a result, Schatten p–quasi–norm of Z can be expressed as $\|Z\|_p^p = tr(U \Lambda^p U^T)$, where Λ^p is entrywise power to p.

Chapter 2

The Sensor Network Localization Problem

Sensor Network Localization Problem can be deemed as class of Graph Realization Problem. Both of them aim to retrieve geometric information of nodes from some internodes information in an Euclidean space, say $\mathbb{R}^d$. However, as a practical problem, SNL Problem varies from different settings, including the forms of given internodes information and existence of anchor. Before explicitly stating the problem, we assume the graph considered to be simple, undirected and connected. Any two vertices u, v in a connected graph can be connected, which the graph contains a path from u to v. Given a graph $G = (V, E)$ with n vertices and m edges. Vertices Set can be partitioned into two sets, anchor set V_a and sensor set V_s. Anchors are vertices that position are known while sensors are the unknown vertices. An edge $(i, j) \in E$ represents the pairwise Euclidean distance information between vertex i and vertex j. These pairwise Euclidean distance information can be exact, contaminated by noise or intervals. With the presence of anchors, the problem is to assign positions in $\mathbb{R}^d$ to sensors that satisfy these pairwise information, while the problem aims to recover the geometric relationship among vertices if there is no anchors. Let's begin with the simplest

setting. Assume that we have exact pairwise distances and presence of anchor.

2.1 Exact Measurement with Anchors

2.1.1 Problem Statement

Given a graph in above setting. For i-th anchor, there is a corresponding known $a_i \in \mathbb{R}^d$ representing the location in d-Euclidean Space. Moreover, each edge $(i,j) \in E_{ss}$ (resp. $(i,j) \in E_{as}$) is associated with a positive weight d_{ij} (resp. $\bar{d}_{ij}$), which can be regarded as the measured Euclidean distance between sensor i(resp. anchor i) and j. The goal is to assign a vector x_i for i-th sensor such that the following distances constraints are satisfied.

$$\begin{aligned} \|x_i - x_j\|^2 &= d_{ij}^2 \quad \text{for } (i,j) \in E_{ss}, \\ \|a_i - x_j\|^2 &= \bar{d}_{ij}^2 \quad \text{for } (i,j) \in E_{as}. \end{aligned} \tag{2.1}$$

2.1.2 Biswas's Semidefinite Relaxation

The relaxation is based on a traditional trick in SDP, that is mapping the decision variables to a higher dimension. First, we denote $X := [x_1 \dots x_{n_s}] \in \mathbb{R}^{d \times n_s}$. By introducing a $n_s \times n_s$ matrix $Y := X^T X$. The above problem can be reformulated to the following rank-constrained Semidefinite Programming model.

$$\begin{aligned} \text{Find} \quad & Z \\ \text{subject to} \quad & tr((0; e_i - e_j)^T Z (0; e_i - e_j)) = d_{ij}^2, \quad \forall (i,j) \in E_{ss} \\ & tr((a_k; -e_j)^T Z (a_k; -e_j)) = \bar{d}_{ij}^2, \qquad \forall (i,j) \in E_{as} \\ & Z = \begin{pmatrix} I_d & X \\ X^T & Y \end{pmatrix} \\ & Y \succeq X^T X \\ & rank(Z) = d. \end{aligned} \tag{2.2}$$

where e_i is a vector with all zero except 1 in i entry and I_d is d dimensional identity matrix. Thus, after dropping the rank constraint, one can get a SDR

of SNL under the aforementioned setting. Similar to all relaxed problems, the major concern for this relaxation would be the condition for exact relaxation, i.e., the rank of Z is d. However, as well known, the standard interior-point algorithms for solving SDP would return the central solution, that is the solution with maximum rank. In most case, the dimension d we considered is low, says $d = 2$ or 3. In other words, this SDR would be exact if and only if the maximum rank of feasible set is d, which is not frequently happened.

2.1.3 Uniquely Localizable Graph

So et al.[27] showed that if such relaxation is exact, this graph cannot be non-trivially localized in a higher dimension. And they further defined the notion of uniquely localizable graph and showed that the rank-d solution in this SDR corresponds to the d-dimensional localization.

Definition 2.1. *The problem (2.1) is* ***uniquely localizable*** *if there is a unique localization $\bar{X} \in \mathbb{R}^{d\times n_s|}$ and there is no $x_j \in \mathbb{R}^h$, $\forall j \in V_S$, where $h > d$, such that:*

$$\begin{aligned} \|x_i - x_j\|^2 &= d_{ij}^2 \qquad \text{for all } (i,j) \in E_{ss}, \\ \|(a_i;0) - x_j\|^2 &= \bar{d}_{ij}^2 \qquad \text{for all } (i,j) \in E_{as} \\ x_j &\neq (\bar{x}_j;0) \quad \text{for some } j \in \{1,...,n_s\} \end{aligned} \tag{2.3}$$

With unique localizability, model(2.2) without the rank constraint would still have solution in rank d after being solved by SDP solver. On the opposite side, if the solution of relaxation solved by SDP is rank d, the problem (2.1) is uniquely localizable. Here is the theorem.

Theorem 2.1. *[27] Suppose that the given graph is connected. Then the following statements are equivalent:*

1. *Problem (2.1) is uniquely localizable.*

2. *The max-rank solution matrix of relaxation on (2.2) has rank d.*

3. The solution matrix Z of relaxation on (2.2) satisfies $Y = X^T X$.

2.2 Exact Measurement without Anchor

Sometimes, there may not exist any anchors, i.e., sensors get its position. However, one may still want to know more about its geometric structure or unknown distances between some sensors. The localization under this setting could be subject to rotation, reflection and translation.

2.2.1 Problem Statement

Given a graph in aforementioned setting. However, $V_a = \emptyset$, thus $E_{as} = \emptyset$. The problem can be modeled as follows:

$$\begin{aligned} \text{minimize} \quad & \textstyle\sum_{i=1}^{n_s} \|x_i\|^2 \\ \text{subject to} \quad & \|x_i - x_j\|^2 = d_{ij}^2 \quad \text{for } (i,j) \in E_{ss}, \end{aligned} \tag{2.4}$$

Note the objective function is actually the sum of square on norm of each sensor, which can be thought as pulling the center of gravity to the origin, so the translation variant can be removed.

2.2.2 Biswas's Semidefinite Relaxation

Biswas et al.[4] proposed a formulation under such anchor-free setting.

$$\begin{aligned} \text{minimize} \quad & tr(Y) \\ \text{subject to} \quad & tr((e_i - e_j)^T Y (e_i - e_j)) = d_{ij}^2, \quad \forall (i,j) \in E_{ss} \\ & Y \succeq 0 \end{aligned} \tag{2.5}$$

Similar to anchor-presence case, matrix variable Y is introduced and $Y = X^T X$ is relaxed to semidefiniteness constraint.

2.2.3 d-localizable

Similar to anchor-presence case, there is also a localizablity result for the anchor-free case. Here we first state the d-localizable graph.

Definition 2.2. *Problem (2.4) is d-localizable if there is no* $x_j \in \mathbb{R}^h$, $j = 1, \ldots, n_s$, *where* $h \neq d$, *such that:*

$$\|x_i - x_j\|^2 = d_{ij}^2, \quad \forall (i,j) \in E_{ss}$$

For $h > d$, *the condition should exclude the trivial case when we set* $x_j = (\bar{x})_j; 0)$ *for* $j = 1, \ldots, n_s$.

Theorem 2.2. *[4] If Problem (2.4) is d-localizable, then the solution matrix,* $\bar{Y}$, *of (2.5) is unique minimum-norm localization of the graph with* $\sum_{j=1}^{n} \bar{x}_j = 0$ *(subject to only rotation and reflection).*

2.3 Noisy Measurement with Anchors

Practically, the distance measurement is not always accurate. Using the mathematical model in (2.2) may lead to infeasible solution. Thus, the problem would be an estimation problem.

2.3.1 Problem Statement

Assume the measurement on edge$(i,j) \in E_{ss}$ (resp. $(i,j) \in E_{as}$)is contaminated by Gaussian noise with known standard deviation σ_{ij}(resp. $\bar{\sigma}_{ij}$). The maximum likelihood estimation of sensors location would be follow:

$$\min_{x_i \in \mathbb{R}^d, i \in V_s} \left\{ \sum_{(i,j) \in E_{ss}} \frac{1}{\sigma_{ij}^2} (\|x_i - x_j\| - d_{ij})^2 + \sum_{(i,j) \in E_{as}} \frac{1}{\bar{\sigma}_{ij}} (\|a_i - x_j\| - \bar{d}_{ij})^2 \right\}. \quad (2.6)$$

Indeed, Biswas et al.[3] show that such estimation problem can also be relaxed by Semidefinite Programming, which is shown below.

2.3.2 Biswas's Semidefinite Relaxation

$$
\begin{aligned}
\text{minimize} \quad & \sum_{(i,j)\in E_{ss}} \frac{1}{\sigma_{ij}^2}\epsilon_{ij} + \sum_{(i,j)\in E_{as}} \frac{1}{\bar{\sigma}_{ij}^2}\bar{\epsilon}_{ij} \\
\text{subject to} \quad & tr((0; e_i - e_j)^T Z (0; e_i - e_j)) = v_{ij}^2, \quad \forall (i,j) \in E_{ss} \\
& tr((a_k; -e_j)^T Z (a_k; -e_j)) = \bar{v}_{ij}^2, \qquad \forall (i,j) \in E_{as} \\
& tr((-d_{ij}; 1)^T D_{ij} (-d_{ij}; 1)) = \epsilon_{ij} \qquad \forall (i,j) \in E_{ss} \\
& tr((-\bar{d}_{ij}; 1)^T \bar{D}_{ij} (-\bar{d}_{ij}; 1)) = \bar{\epsilon}_{ij} \qquad \forall (i,j) \in E_{as} \\
& Z = \begin{pmatrix} I_d & X \\ X^T & Y \end{pmatrix} \succeq 0 \\
& D_{ij} = \begin{pmatrix} 1 & u_{ij} \\ u_{ij} & v_i j \end{pmatrix} \succeq 0 \qquad \forall (i,j) \in E_{ss} \\
& \bar{D}_{ij} = \begin{pmatrix} 1 & \bar{u}_{ij} \\ \bar{u}_{ij} & \bar{v}_i j \end{pmatrix} \succeq 0 \qquad \forall (i,j) \in E_{as}
\end{aligned}
\tag{2.7}
$$

Same as the model in (2.2), the relaxation drops the rank constraints of Z, D_{ij} *and* $\bar{D}_{ij}$. However, it is not in the standard form of SDP. One can combine the decision matrices into a large matrix diagonally and reformulate it into standard form, which would be shown later. Also, SDP solver(say SDPT3) can exploit the property of block diagonal matrix and the low rank property of the matrix coefficient.

2.4 Schatten p–regularized SDP

2.4.1 Regularizers in SNL

We have demonstrated the modeling power of SDP to SNL in various settings, which is actually only fraction of all cases. However, in above relaxations (2.5), (2.7) and (2.2) without rank constraint, SNL cannot be exactly solved if the relaxation is not exact. A best-rank d approximation may help, but it is more natural to ask whether the high rank solution can be eliminated throughout the

optimization process. Biswas et al. first proposed the regularizer approach in achieving low-rank solution for graph realization problem, under the anchor-free setting. Here, the regularizer they adapted is to maximize the total distances among all sensors, i.e. $\sum_{i\in V_s}\sum_{j\in V_s, j\neq i}\|x_i - x_j\|^2$. This regularizer is also named maximum variance unfolding in non-linear dimensionality reduction in kernel learning.

Besides, regularizer approach has been studied to find the low-rank matrix in SDP, especially in application for the low-rank matrix recovery. Nuclear norm is widely used as regularizer for such setting. In terms of assimilating the rank function and achieving the convexity, nuclear norm is the best choice as it is the convex envelope of rank function. On the other hand, Shamsi et al.[26] proposed some heuristic objective functions to correctly localize for a certain class of graphs.

Recently, there are several algorithms for finding first order point of Schatten p-minimization [18][16][1], yet without the guarantee of polynomial–time complexity.

2.4.2 Formulation

The Schatten p-regularization for standard SDP (1.1) is:

$$\begin{aligned} \Gamma^* = \ \text{minimize} \quad & \pi(Z) = \text{tr}(CZ) + \mu\|Z\|_p^p \\ \text{subject to} \quad & \text{tr}(A_i Z) = b_i \qquad\qquad \text{for } i = 1, \ldots, m, \\ & Z \in \mathcal{S}_+^n \end{aligned} \tag{2.8}$$

where $\mu > 0$ is the regularizer parameter. We denote that $\mathcal{F}$ as the feasible region of above model.

It is not hard to see that relaxation (2.5) and (2.2) without rank constraint are already in standard form. We will show that how (2.7) can also be transformed into standard SDP. Here is the diagonalized version of (2.7) :

$$
\begin{aligned}
\text{minimize} \quad & tr(CW) \\
\text{subject to} \quad & W = \begin{bmatrix} Z & & \\ & Diag(\{\bar{D}_{ij}\}_{(i,j)\in E_{as}}) & \\ & & Diag(\{\bar{D}_{ij}\}_{(i,j)\in E_{as}}) \end{bmatrix}, \\
& D_{ij} = \begin{bmatrix} 1 & u_{ij} \\ u_{ij} & v_{ij} \end{bmatrix} \text{ for } (i,j) \in E_{ss}, \\
& \bar{D}_{ij} = \begin{bmatrix} 1 & \bar{u}_{ij} \\ \bar{u}_{ij} & \bar{v}_{ij} \end{bmatrix} \text{ for } (i,j) \in E_{as}, \\
& tr(K_{ij}Z) = v_{ij} \quad \text{for } (i,j) \in E_{ss}, \\
& tr(\bar{K}_{ij}Z) = \bar{v}_{ij} \quad \text{for } (i,j) \in E_{as}, \\
& Z_{1:d,1:d} = I_d, Z \in \mathcal{S}_+^{d+n_s}, D \in \mathcal{S}_+^{2|E_{ss}|}, \bar{D} \in \mathcal{S}_+^{2|E_{as}|}
\end{aligned} \tag{2.9}
$$

where C takes the form $C = Diag(\mathbf{0}, C_0, C_1)$, C_0 is the $2|E_{ss}| \times 2|E_{ss}|$ block–diagonal matrix whose (i,j)–th diagonal block is the 2×2 matrix $\sigma_{ij}^{-2}(-d_{ij};1)(-d_{ij};1)^T$ (where $(i,j) \in E_{ss}$), and C_1 is the $2|E_{as}| \times 2|E_{as}|$ block–diagonal matrix whose (i,j)–th diagonal block is the 2×2 matrix $\bar{\sigma}_{ij}^{-2}(-\bar{d}_{ij};1)(-\bar{d}_{ij};1)^T$.

It should be noted that each equality of (2.9) is linear in W. Moreover, as W is block diagonal matrix, we have $W \in \mathcal{S}_+^{d+n+2|E_{ss}|+2|E_{as}|} \Leftrightarrow Z \in \mathcal{S}_+^{d+n_s}$, $D \in \mathcal{S}_+^{2|E_{ss}|}$, $\bar{D} \in \mathcal{S}_+^{2|E_{as}|}$. Therefore, (2.9) can be transformed into Standard SDP.

Note that when the relaxation is exact, rank of each D_{ij} and $\bar{D_{ij}}$ is 1 and rank of Z is d. And extra constraints are needed to add to maintain the block diagonal structure.

Chapter 3

Potential Reduction Algorithm

In this chapter, we would introduce a tractable algorithm namely potential reduction for approximating a first–order critical point of the Schatten p–regularized SDP (2.8). The theoretical results are based on techniques in Ye[32] and Ge et al.[9]. To describe the whole algorithm, we need to define some components. Basically, the algorithm aims to reduce value of a potential function in each iterate. The potential value also plays an essential role in complexity analysis. It had been shown that either ϵ-optimal solution or ϵ-first-order critical point will be achieved in a finite step if the potential function is under certain value.

3.1 ϵ–First–Order critical point

Definition 3.1. *Let $\bar{Z} \in \mathcal{F}$ and $\epsilon > 0$ be given. Suppose that rank$(\bar{Z}) = r$, and let $\bar{Z} = U\Lambda U^T$ be the spectral decomposition of $\bar{Z}$, where $U \in \mathbb{R}^{n\times r}$ has orthonormal columns and $\Lambda \in \mathcal{S}^r_{++}$ is a diagonal matrix. We say that $\bar{Z}$ is an* ϵ–first–order critical point *of Problem (2.8) if there exists a multiplier $\bar{y} \in \mathbb{R}^m$ such that*

$$U^T C U + \mu p \Lambda^{p-1} - \sum_{i=1}^{m} \bar{y}_i \left(U^T A_i U\right) \in \mathcal{S}^r_+ \tag{3.1}$$

and

$$0 \leq \frac{\mathrm{tr}\left(C\bar{Z} + \mu p \bar{Z}^p - \sum_{i=1}^m \bar{y}_i A_i \bar{Z}\right)}{\pi(\bar{Z})} \leq \epsilon. \tag{3.2}$$

For convienice, 0–first–order critical point will be called as first–order critical point.

This definition inherits the dual feasibility and the complementarity condition. Such definition is also used to take care on the marginal region as the objective function π is non-differentiable on $\mathcal{S}^r_+ \setminus \mathcal{S}^r_{++}$. Besides, we define ϵ–optimal solution to (2.8) when $\pi(Z) \leq \Gamma^* + \epsilon$. We will show that potential reduction algorithm can achieve either ϵ–optimal solution or ϵ–first–order critical point in polynomial–time of n, m and ϵ^{-1}.

To approximate the first–order critical point, the above definition onϵ–first–order critical point can take care of the non-differentiablility of objective function π. Moreover, the $\pi(\bar{Z})$ in (3.2) also acts as a normalizer to maintain invariance properties to the middle part of (3.2). Noted that these invarianve properties are essential to an approximation notion.

3.2 Assumption

To show the complexity analysis, we shall make these assumption:

1. The feasible region $\mathcal{F}$ is bounded; i.e., there exists an $R < \infty$ such that $\mathcal{F} \subset \{Z \in \mathcal{S}^n : \|Z\|_F^2 = \mathrm{tr}(Z^2) \leq R^2\}$.

2. A lower bound $\underline{\theta} > -\infty$ on the optimal value of the (unregularized) SDP problem

$$\min\{\mathrm{tr}(CZ) : Z \in \mathcal{F}\} \tag{3.3}$$

   is known or can be efficiently estimated.

3. A strictly feasible solution Z' to Problem (2.8) (i.e., $Z' \in \mathcal{F}$ satisfies $Z' \in \mathcal{S}^n_{++}$) is available or can be efficiently constructed.

For the first assumption, relaxations (2.5) and (2.2) would surely satisfy as long as sensors are connected. In fact, connectivity is a common assumption in SNL. On the other hand, there is no upper bound for model (2.9) even the graph is connected yet one can always put an upper bound on it.

The second assumption is also mild as all objective coefficient matrices C of three models (2.5), (2.2) and (2.9) are positive semi-definite, which implies that there is a trivial lower bound 0. We will discuss the third assumption later.

3.3 Algorithm Architecture

Before go through the analysis, we shall define the potential function, which is actually added a barrier term.

Definition 3.2. *The function $\phi : \mathcal{S}^n_+ \to \mathbb{R}$ is a potential function if*

$$\phi(Z) = \rho \cdot \log(\pi(Z)) - \log\det(Z),$$

where $\rho > 0$ is a parameter.

Using such potential function with the barrier term $-\log\det(Z)$ would ensure that the algorithm iterates within the strictly feasible set of (2.8). We are now ready to describe the algorithm.

Assume now we have the current iterate $\bar{Z}$. The next iterate will then be $\bar{Z}^+ = \bar{Z} + D_{\bar{Z}}$ where $D_{\bar{Z}} \in \mathcal{S}^n$. We aim to choose $D_{\bar{Z}}$ to achieve a maximum reduction on the potential function while satisfying the equality constarint, i.e., $tr(A_i D_{\bar{Z}}) = 0$ for $i = 1, \ldots, m$. We have the following lemma.

Lemma 3.1. *Let $D = \bar{Z}^{-1/2} D_{\bar{Z}} \bar{Z}^{-1/2}$ and $\bar{C} = \bar{Z}^{-1/2} C \bar{Z}^{-1/2}$. If $\|D\|_F^2 \leq \beta < 1$, the following inequality will hold:*

$$\phi(\bar{Z}^+) - \phi(\bar{Z}) \;\leq\; \frac{\rho}{\pi(\bar{Z})} tr\left[(\bar{C} + \mu p \bar{Z}^p) D\right] - tr(D) + \frac{\beta^2}{2(1-\beta)}. \qquad (3.4)$$

We refer the proof to our conference paper[24]. Now, to ensure maximum potential reduction, we can model it as follows:

$$\begin{aligned} \text{minimize} \quad & \operatorname{tr}(QD) := \operatorname{tr}\left[\left(\frac{\rho}{\pi(\bar{Z})}(\bar{C}+\mu p\bar{Z}^p) - I\right) D\right] \\ \text{subject to} \quad & \operatorname{tr}(\bar{Z}^{1/2}A_i\bar{Z}^{1/2}D) = 0 \quad \text{for } i = 1,\ldots,m, \\ & \|D\|_F^2 \le \beta^2. \end{aligned} \tag{3.5}$$

We shall further denote $\mathcal{A}(D) := \left(\operatorname{tr}\left(\bar{Z}^{1/2}A_1\bar{Z}^{1/2}D\right),\ldots,\operatorname{tr}\left(\bar{Z}^{1/2}A_m\bar{Z}^{1/2}D\right)\right)$, and its adjoint operator $\mathcal{A}^T(\nu) := \sum_{i=1}^m \nu_i\bar{Z}^{1/2}A_i\bar{Z}^{1/2}$. The above model is a convex problem and satisfies the Slater condition, so we can draw the KKT conditions, which are both necessary and sufficient for optimality.

$$2\lambda D = \mathcal{A}^T(\nu) - Q, \tag{3.6a}$$

$$\mathcal{A}(D) = \mathbf{0}, \tag{3.6b}$$

$$\|D\|_F^2 = \beta^2, \tag{3.6c}$$

$$\lambda \ge 0. \tag{3.6d}$$

Furthermore, we can express the optimal D in an explicit form from the above conditions if we assume $\mathcal{A}$ has full row rank.

$$D = -\beta\cdot\frac{\left(I-\mathcal{A}^T(\mathcal{A}\mathcal{A}^T)^{-1}\mathcal{A}\right)Q}{\left\|\left(I-\mathcal{A}^T(\mathcal{A}\mathcal{A}^T)^{-1}\mathcal{A}\right)Q\right\|_F}. \tag{3.7}$$

Substituting (3.7) into (3.4), we can get

$$\phi(\bar{Z}^+) - \phi(\bar{Z}) \le -\beta\cdot\left\|\left(I-\mathcal{A}^T(\mathcal{A}\mathcal{A}^T)^{-1}\mathcal{A}\right)Q\right\|_F + \frac{\beta^2}{2(1-\beta)}. \tag{3.8}$$

Here we come to the stopping criteria.

If $\left\|\left(I-\mathcal{A}^T(\mathcal{A}\mathcal{A}^T)^{-1}\mathcal{A}\right)Q\right\|_F \ge 1$, by setting suitable value of β, says $1/3$, one can get $\phi(\bar{Z}^+) - \phi(\bar{Z}) \le -1/4$, i.e. the potential function decreases $1/4$ at each iterate. Here, we shall give a proposition about the ϵ–optimal solution.

Proposition 3.1. *Let $\epsilon > 0$ and $\rho > n/p$ be fixed. Suppose that $\bar{Z}$ is strictly feasible for Problem (2.8) and satisfies*

$$\phi(\bar{Z}) \leq \left(\rho - \frac{n}{p}\right)\log(\epsilon) + \frac{n}{p}\log(\mu n). \tag{3.9}$$

Then, $\bar{Z}$ is an ϵ–optimal solution to Problem (2.8).

Proof. Using the arithmetic–mean geometric–mean inequality, we have

$$\frac{\|Z\|_p^p}{n} = \frac{1}{n}\sum_{i=1}^{n}(\lambda_i(Z))^p \geq \left(\prod_{i=1}^{n}\lambda_i(Z)\right)^{\frac{p}{n}} = (\det(Z))^{p/n}$$

for all $Z \in \mathcal{S}_+^n$. In particular, for any $Z \in \mathcal{F}$,

$$\frac{n}{p}\log(\pi(Z)) - \log\det(Z) \geq \frac{n}{p}\log(\mu n),$$

where we use the assumption that $\mathrm{tr}(CZ) \geq \underline{\theta} = 0$ for all $Z \in \mathcal{F}$. Thus, if condition (3.9) is satisfied, then we have $\pi(\bar{Z}) \leq \epsilon$, which, together with the fact that $\Gamma^* \geq 0$, implies the desired result. □

In case the ϵ–optimal solution is not achieved, next iteration will continue. By the above proposition, the iteration is finite.
Otherwise, if $\left\|\left(I - \mathcal{A}^T(\mathcal{A}\mathcal{A}^T)^{-1}\mathcal{A}\right)Q\right\|_F < 1$, we terminate and the current iterate $\bar{Z}$ is the ϵ–first–order critical point, which will be shown in the complexity analysis. Here is the whole algorithm:

3.4 Correctness and Complexity Analysis

In this section, we will show the correctness of the solution. Here is the theorem.

Theorem 3.1. *Given the current iterate $\bar{Z}$. If $\left\|\left(I - \mathcal{A}^T(\mathcal{A}\mathcal{A}^T)^{-1}\mathcal{A}\right)Q\right\|_F < 1$, $\bar{Z}$ is ϵ–first–order critical point, i.e. (3.1) and (3.2)*

Algorithm 1 Potential Reduction Algorithm

Require: A strictly feasible solution Z_0, C, μ, p, $\{A_i, b_i : \forall i\}$, β, ϵ

Ensure: $\bar{Z}$

1: $\bar{Z} \leftarrow Z_0$

2: $\rho \leftarrow (n + \sqrt{n})/\min\{p, \epsilon\}$

3: $\bar{C} \leftarrow \bar{Z}^{1/2} C \bar{Z}^{1/2}$

4: $Q \leftarrow (\rho/\pi(\bar{Z}))(\bar{C} + \mu p \bar{Z}^p) - I$

5: $\mathcal{A} \leftarrow \left(\mathrm{vec}\left(\bar{Z}^{1/2} A_1 \bar{Z}^{1/2}\right), \ldots, \mathrm{vec}\left(\bar{Z}^{1/2} A_m \bar{Z}^{1/2}\right)\right)^T$

6: **while** $\left\|\left(I - \mathcal{A}^T(\mathcal{A}\mathcal{A}^T)^{-1}\mathcal{A}\right) Q\right\|_F \geq 1$ **do**

7: $\quad D \leftarrow -\beta \cdot \frac{\left(I - \mathcal{A}^T(\mathcal{A}\mathcal{A}^T)^{-1}\mathcal{A}\right)Q}{\|(I - \mathcal{A}^T(\mathcal{A}\mathcal{A}^T)^{-1}\mathcal{A})Q\|_F}$

8: $\quad \bar{Z} \leftarrow \bar{Z} + \bar{Z}^{1/2} D \bar{Z}^{1/2}$

9: $\quad \bar{C} \leftarrow \bar{Z}^{1/2} C \bar{Z}^{1/2}$

10: $\quad Q \leftarrow (\rho/\pi(\bar{Z}))(\bar{C} + \mu p \bar{Z}^p) - I$

11: $\quad \mathcal{A} \leftarrow \left(\mathrm{vec}\left(\bar{Z}^{1/2} A_1 \bar{Z}^{1/2}\right), \ldots, \mathrm{vec}\left(\bar{Z}^{1/2} A_m \bar{Z}^{1/2}\right)\right)^T$

12: **end while**

Proof. Recall the KKT conditions (3.6a) and (3.6b), we have $v = (\mathcal{A}\mathcal{A}^T)^{-1}\mathcal{A}(Q)$. Thus, we get $\left\|\left(I - \mathcal{A}^T(\mathcal{A}\mathcal{A}^T)^{-1}\mathcal{A}\right) Q\right\|_F = \|Q - \mathcal{A}^T(v)\|_F^2 < 1$, more explicitly,

$$\left\| \frac{\rho}{\pi(\bar{Z})} (\bar{Z}^{1/2} C \bar{Z}^{1/2} + \mu p \bar{Z}^p) - \mathcal{A}^T(v) - I \right\|_F^2 < 1. \tag{3.10}$$

Using the Cauchy-Schwarz inequality, we have

$$\frac{n - \sqrt{n}}{\rho} < \frac{\mathrm{tr}\left[\bar{Z}\left(C + \mu p \bar{Z}^{p-1} - \sum_{i=1}^m \bar{y}_i A_i\right)\right]}{\pi(\bar{Z})} < \frac{n + \sqrt{n}}{\rho},$$

where $\bar{y} = -(\pi(\bar{Z})/\rho)\nu \in \mathbb{R}^m$. By setting $\rho = (n + \sqrt{n})/\min\{p, \epsilon\}$, we have

$$\rho > \frac{n}{p} \quad \text{and} \quad 0 < \frac{\mathrm{tr}\left(\bar{Z}\left(C + \mu p \bar{Z}^{p-1} - \sum_{i=1}^m \bar{y}_i A_i\right)\right)}{\pi(\bar{Z})} < \epsilon.$$

Thus, $\bar{Z}$ satisfies condition (3.2). For the dual feasibility (3.1), we have to show $\bar{S} := C + \mu p \bar{Z}^{p-1} - \sum_{i=1}^m \bar{y}_i A_i \in \mathcal{S}_+^n$. It suffices to show that $\bar{Z}^{1/2} \bar{S} \bar{Z}^{1/2} \in \mathcal{S}_+^n$. Let $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_n$ be the eigenvalues of $\bar{Z}^{1/2} \bar{S} \bar{Z}^{1/2}$. By (3.10), we have

$\sum_{i=1}^{n} \left(\lambda_i - \pi(\bar{Z})/\rho\right)^2 \leq \left(\pi(\bar{Z})/\rho\right)^2$. This implies that $\lambda_n \geq 0$, i.e.

$$\bar{Z}^{1/2}\bar{S}\bar{Z}^{1/2} \in \mathcal{S}_+^n \Rightarrow \bar{S} \in \mathcal{S}_+^n.$$

Thus, $\bar{Z}$ is an ϵ–first–order critical point. □

The remainding part would be the complexity analysis, that is, to show that potential reduction algorithm is polynomial–time implementable.

Theorem 3.2. *The potential reduction algorithm described above would terminate in polynomial time.*

Proof. To show the complexity, it is essential to know more about the potential value of initialized strictly feasible point Z_0. With the third assumption, we have $\|Z_0\|_F \leq R$ and $\lambda_{min}(Z_0) \geq r$. It can be shown that

$$\phi(Z_0) \leq \bar{\phi} = \rho\log(R\|C\|_F + \mu R^p n^{1-p/2}) - n\log\, r.$$

To conclude, the algorithm would terminates in at most

$$\eta = \frac{\bar{\phi} - \underline{\phi}}{1/4} = \mathcal{O}\left(\frac{n}{\Delta}\log\left(\frac{R}{\epsilon r'}\left(\|C\|_F + \mu n^{1-p/2}\right)\right)\right)$$

iterations. As each iteration is polynomial–time implementable, the algorithm would terminate in polynomial time. □

3.5 Initialization and Conversion

Recall the third assumption that we can construct a strictly feasible solution to (2.8). The initialization can be done by solving the unregularized SDP, which is polynomial-time solvable. However, there is no guarantee that this solution is strictly feasible. To tackle this problem, we adapt a conversion technique to tackle this problem.

Suppose we solved the unregularized SDP and obtained a solution $\hat{Z}$ with rank $\hat{r}$. Performing the Schur decomposition, we will get $\hat{Z} = V\Pi_0 V^T$, where $V \in \mathbb{R}^{n\times\hat{r}}$ and $\Pi \in \mathbb{R}^{\hat{r}\times\hat{r}}$. We converse (2.8) into the following problem:

$$\begin{aligned} \Gamma^* = \ \text{minimize} \quad & \operatorname{tr}(V^T C V \Pi) + \mu\|\Pi\|_p^p \\ \text{subject to} \quad & \operatorname{tr}(V^T A_i V \Pi) = b_i \qquad \text{for } i = 1, \ldots, m, \\ & \Pi \in \mathcal{S}_+^{\hat{r}} \end{aligned} \tag{3.11}$$

It is clear that Π_0 is a strictly feasible point to the above problem. Thus, we can apply the potential reduction algorithm to (3.11) and get the solution Π^*. Afterall, we can restore the solution by combining back the decomposition $V\Pi^* V^T$.

Such conversion not only solves the problem of initialization, but also reduces the problem size, which does shorten the solution time a lot. Moreover, the rank of final solution should be less or equal to the rank of unregularized problem, which is very promising in the perspective of rank minimization.

3.6 Localizabiliy Result

Recall that there is a localizability property(Theorem 2.1) of the global solution of traditional SDP approach to SNL problem with the presence of anchor. It can also be preserved in the solution of Schatten p–regularized SDP. It is not a surprising result as the proof of that rely heavily on the feasible set, which is actually an unique point. We refer reader to [27] for more idea.

For the absence of anchor, we show that for the 0–first–order–critical point of regularized Problem (2.5) can extend the result of Theorem 2.2.

Chapter 4

Simulation Results

In this chapter, extensive simulation results are presented under different settings. We will demonstrate the effectiveness of the Schatten p–regularization approach to SNL, especially in the sparse cases, which the information given is not much. All sensors are uniformly random generated in a unit square $[-0.5, 0.5] \times [-0.5, 0.5]$ while the setting of anchors will be described for each case. The nodes are connected under two different schemes, with different testing purpose, will be introduced in the following section. Then, the measure criteria for performance will be introduced. All simulations are implemented in MATLAB(2011b) and run on a 2.66GHz CPU PC with 3 GB memory.

4.1 Graph Generations

4.1.1 Unit-Disk Graphs

In traditional SNL, two nodes can communicate each other if their distances are with certain threshold, namely radio range (denoted by ρ). This type of graph is also called Unit-disk graph as intuitively we place a unit-disk on each vertex and connect it to others that lie on the disk. In fact, the choice of radio range is one major study in SNL as it is useful to know how large is sufficient to

localize all nodes practically. Recently, there are several results on the bound of this radio range. During the simulation, radio range would gradually increase to increase the density. However, unit-disk graphs may have multiple localizations, especially when the radio range is small. In other words, even the problem is exactly solved, the localization may still be different to the ground truth. To further investigate the performance on unique realization case, we would have another graph generation.

4.1.2 Globally Rigid Graphs

Globally rigid graph in $\mathbb{R}^2$ can only have unique localization in $\mathbb{R}^2$. To generate such graphs, one can construct it from the complete graph K_4. There are two operations that preserve the rigidity.

1. **Edge Addition**

   Adding an edge to a pair of non-adjacent vertices.

2. **1-extension**

   For any edge (i, j) and a new vertex k. We form (i, k) and (k, j) to replace the (i, j). And one more edge (k, h) to other connected vertex h should be added to preserve the rigidity.

Although we are generating global rigid graphs, we still aim to preserve the idea of unit-disk, that is, two nodes are connected if their pairwise distance is small. After the random generation of vertices, we would pick 4 vertices that are the closest to the origin and forms a k_4. Then we select a new vertex that is closest to any edge in the connected graph, and perform the **1-extension**. With similar idea, while we are executing the 1-extension, the extra vertex h would be the closest connected vertex besides i and j.

After all the vertices are connected, **Edge Addition** would execute gradually. The adding order of edges is based on their distances. Such generation can greatly

help in investigating the performance in the sparse case with unique realization.

4.2 Measure Criteria

We have three criteria on measuring the performance. The first one is the traditional position error(PE). It is the root mean square of the difference between the computed location and the ground true location of sensors. It is defined as

$$\mathsf{PE} = \sqrt{\frac{1}{n_s}\sum_{i\in V_s}\|x_i - \bar{x}_i\|_2^2},$$

where $x_i, \bar{x}_i \in \mathbb{R}^d$ are the computed location and true location of sensor i respectively. However, as the realization may not be unique in some settings, e.g. unit-disk graph, exact realization that satisifies constraints may still induce high PE. As a result, we would introduce the second criterion, that is the edge error(EE). It measures the discrepancy between the computed distances and the given distance measurments. Similar to PE, it is the root mean square of the difference, i.e. $\mathsf{EE} = \sqrt{\frac{1}{|E|}\left(\left(\sum_{(i,j)\in E_{ss}}\|x_i - x_j\|_2 - d_{ij}\right)^2 + \left(\sum_{(i,j)\in E_{as}}\|a_i - x_j\|_2 - \bar{d}_{ij}\right)^2\right)}$.

The third criterion is the rank of the solution matrix. It represents the dimension of computed realization. If the rank equals to d, then SNL is exactly solved. For the computation of the rank, we would base on the singular value decomposition as it is more reliable yet time consuming. The rank of a matrix would be the cardinality of non-zero singular value. For numerical issue, a tolerance would be used for counting.

4.3 Exact Measurement with Anchors

4.3.1 Unit-disk Graph

We begin with the simplest case. We randomly generate 50 sensors and 3 anchors and employ the unit-disk graph setting with radio range $\rho = 0.18, 0.19, \ldots, 0.3$. The reason that we started from 0.18 is that the randomly generated vertices would be connected with a high probability. When the $\rho = 0.3$, most instances are uniquely localizable(Definition 2.1), i.e., Biswas's relaxation can solve them exactly. Thus, this range of simulation can cover sparse cases to dense cases.

Let's state a unified framework for the models of problem (2.1) we would consider.

$$\begin{array}{ll} \text{Minimize} & \mu\|Z\|_p^p \\ \text{subject to} & tr((0; e_i - e_j)^T Z(0; e_i - e_j)) = d_{ij}^2, \quad \forall (i,j) \in E_{ss} \\ & tr((a_k; -e_j)^T Z(a_k; -e_j)) = \bar{d}_{ij}^2, \qquad \forall (i,j) \in E_{as} \\ & Z = \begin{pmatrix} I_d & X \\ X^T & Y \end{pmatrix} \\ & Y \succeq X^T X. \end{array} \tag{4.1}$$

In this experiment, we will compare the following approaches:

SDR: Biswas's relaxation(i.e. $\mu = 0$) solved by CVX with Solver SeDuMi.

Trace: Trace regularized SDR(i.e. $\mu = 1, p = 1$) solved by CVX with Solver SeDuMi.

Schatten: Schatten 0.5−regularized SDR(i.e. $\mu = 1, p = 0.5$) solved for ϵ–first–order critical point by aforementioned potential reduction algorithm with the conversion, where the tolerance $\epsilon = 10^{-5}$.

Lets have a look on a single instance where the radio range $\rho = 0.23$ in Figure 4.1. When the anchors position are biased or the edge are sparse, the sensors

are very flexible, i.e. they have high chance to be localized in a high dimension. Recall the max-rank property of SDP solver, the SDR would return a high rank solution. The solution is then projected to a lower dimension, resulting high error. With the regularizer, the solution of rank decreases thus it may incur less projection error.

We have run 100 simluations with varying radio range $\rho = 0.18 - 0.3$. Results are shown in Figure 4.2 and Figure 4.3a. It can be seen that EE of Schatten is lower when the radio range is small, i.e., the graph is sparse. Schatten also plays a better role than others in terms of dimension requirement, i.e., the rank of solution. In the terms of PE, Schatten perform worse than SDR. Mentioned before, the unit-disk graph may have multiple realizations. So this may not imply that SDR is not effective.

One may note that the performance of Schatten will converge to SDR when the radio range goes larger. It is based on the fact that rank of SDR decreases with radio range. When the rank is lower, Schatten might be faster to go to the ϵ first–order critical point. In fact, numerical results show that the number of iterations decreases when the radio range increases, see Figure 4.3b. Here is a remark about the computation time. Although the computation time for each iteration increases with the number of constraints, the total solution time decreases with radio range due to the dramatic reduction on iteration number.

4.3.2 Globally Rigid Graphs

To avoid multiple realizations due to graph structure, we start the third simulations under the globally rigid graph setting. We generate 50 sensors and 3 anchors randomly with varying number of extra edges from 0 to 100 (by Edge Addition in graph generation). We adapt $p = 0.5$ for Schatten approach with the conversion. Again, 100 instances had been under investigated and results are in Figure. 4.4. Surprisingly, the position error of Schatten is still higher than SDP

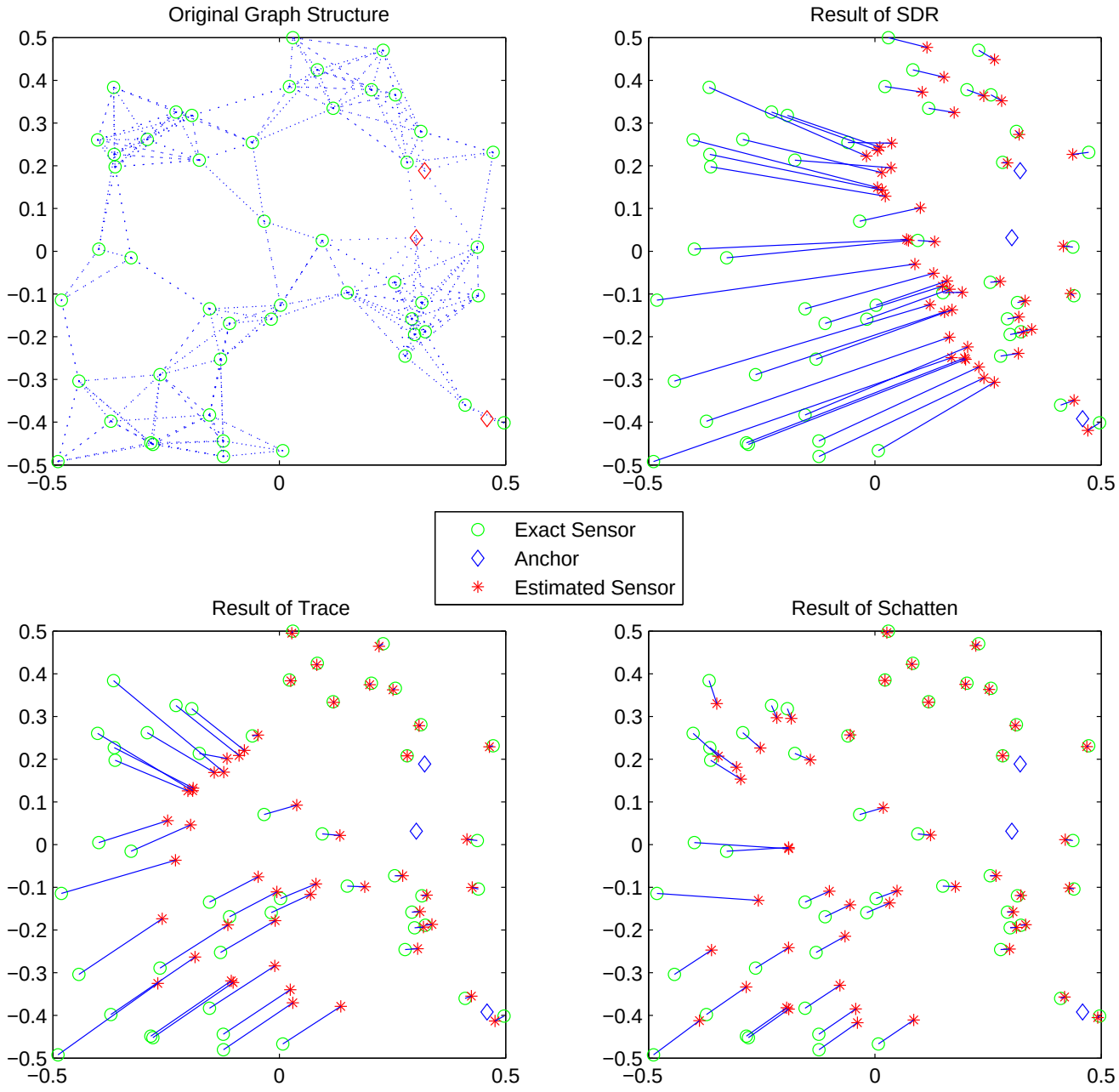


Figure 4.1: The Graph illustration with 50 sensors and 3 anchors, unit–disk graphs with $\rho = 0.23$

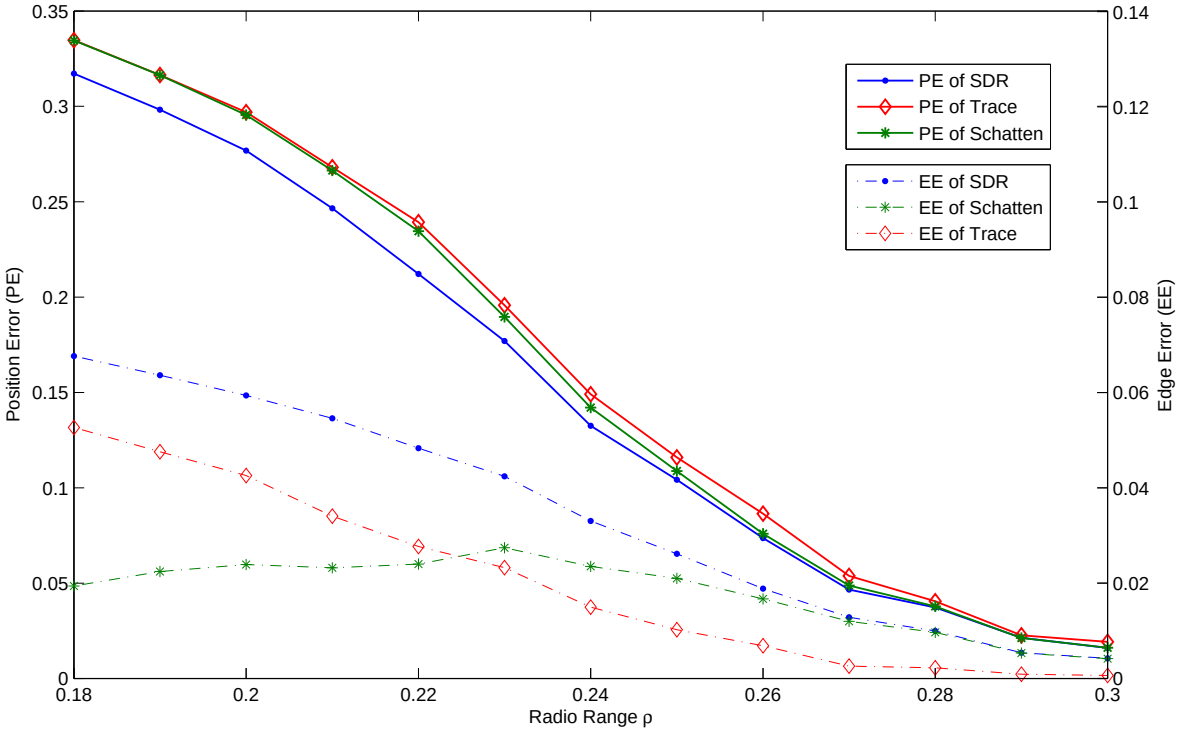
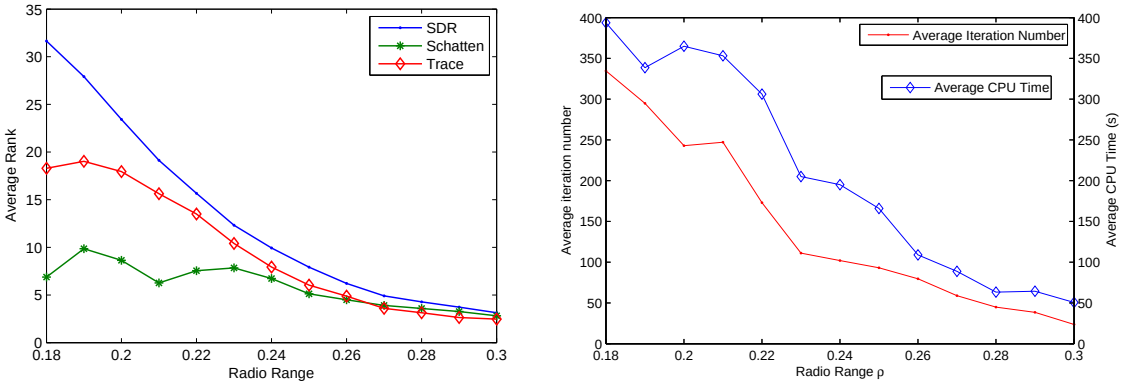


Figure 4.2: The average PE and EE with 50 sensors and 3 anchors, unit–disk graphs with $\rho = 0.18 - 0.3$



(a) Rank (b) Iteration number and time

Figure 4.3: The average Rank, number of iteration and time with 50 sensors and 3 anchors, unit–disk graphs with $\rho = 0.18 - 0.3$

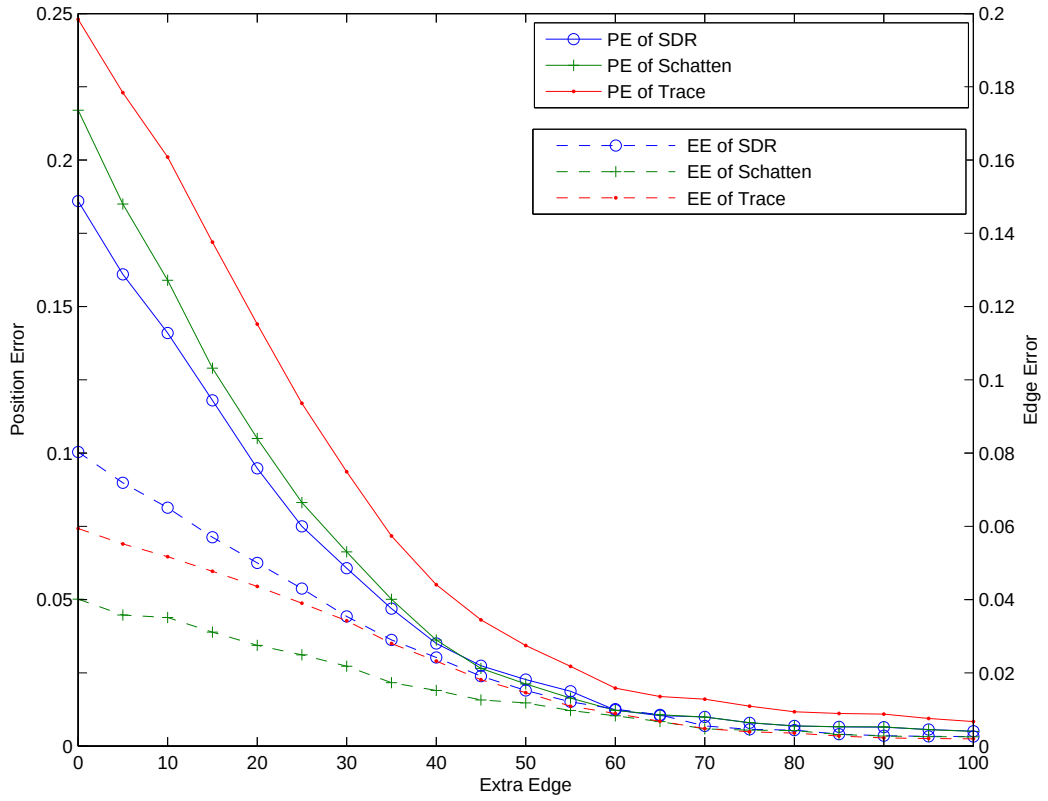


Figure 4.4: The average PE and EE with 50 sensors and 3 anchors,globally rigid graph with extra edges from 0 to 100

in most of time while Schatten shows its strong reduction in edge error and rank. To deeply investigate the effect of rank reduction, one can have a look on the eigenvalues of solutions, See Figure. 4.5. It shows that eigenvalues of solution obtained by potential reduction algorithm is much small.

4.3.3 Choice of Parameter p

In the above simulation, we choose $p = 0.5$ for the Schatten approach for illustration. To investigate the effects of choice of p, we start the second simulation from $p = 0.1$ to 0.9 with 0.1 increment. We generate 50 sensors and 3 anchors

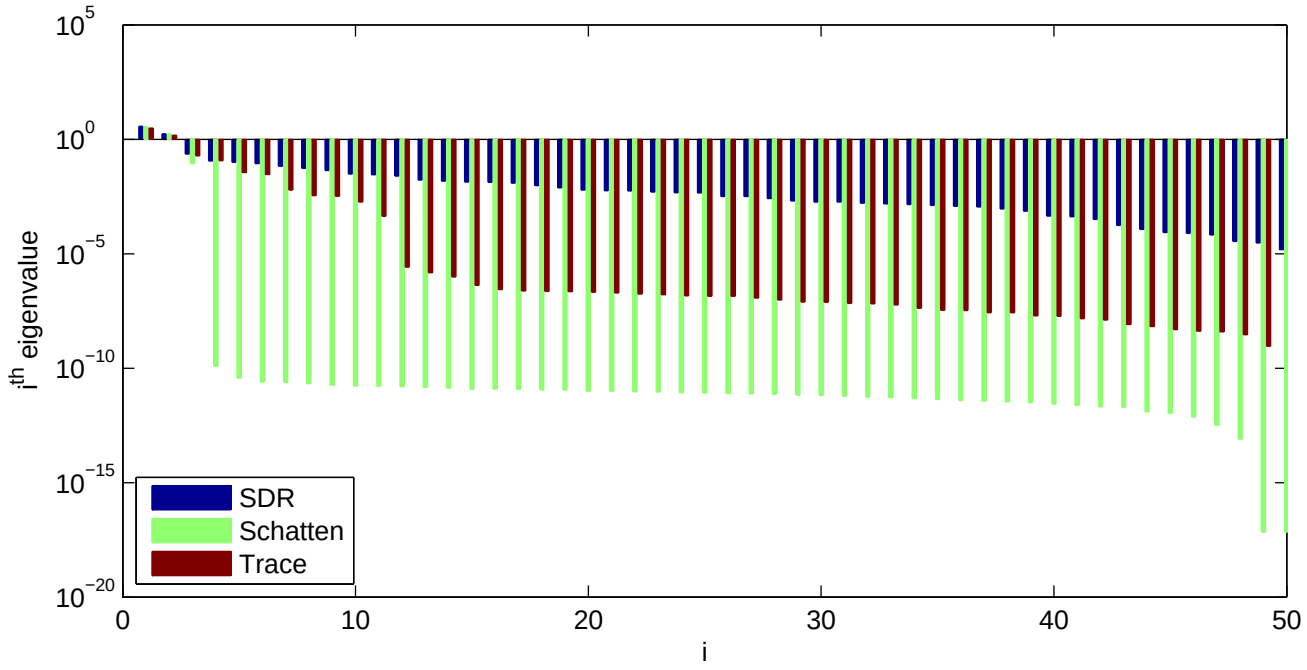
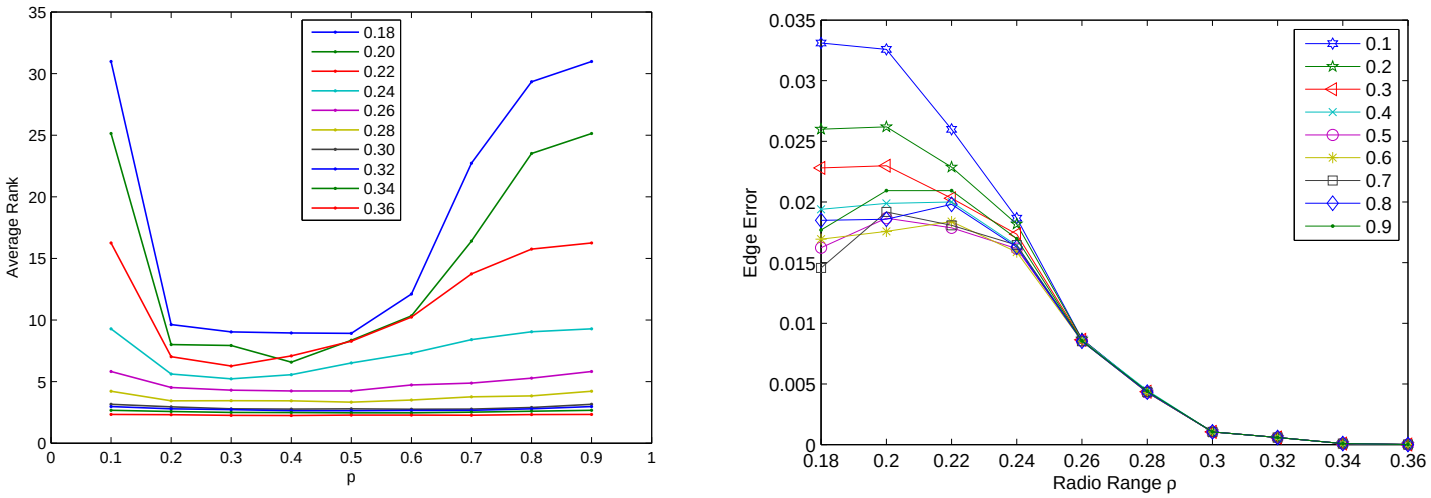


Figure 4.5: The eigenvalues of solutions of a single instance

randomly with varying radio range $\rho = 0.18 - 0.36$. Recall that $\|\cdot\|_p^p \to$ rank$(\cdot)$ when $p \to 0$. However, as our algorithm only guarantees first order point but not optimal solution, lower value of p may not help. Indeed, numerical result shows that a too small p would not be a good choice, in terms of EE and Rank. Again, we have run 100 simulations and results are shown in Figure 4.6. In Figure 4.6a, under the perspective of rank reduction, it is clear that the potential reduction algorithm would perform better when $p \in [0.2, 0.6]$. On the other hand, if one considers the EE, the best range of p would lie on $[0.4, 0.8]$. To conclude, one should select $p \in [0.4, 0.6]$ for the potential algorithm, at least under current setting.

It should be noticed that the extreme values are not the good choice. Also when the radio range goes larger, their performances converge as they all converge to SDR's performance. So far, there is no any theoretical justification on the choice of p.



(a) Rank against p
(Legend: radio range)

(b) EE against radio range
(Legend: p)

Figure 4.6: Illustration of effect of p with 50 sensors and 3 anchors, unit–disk graphs with $\rho = 0.18 - 0.36$

4.3.4 Choice of Initial Point

In section 3.5, we mentioned the initialization technique and the conversion of problem before applying potential reduction. We select the solution of unregularized SDP as the initial feasible piont. However, such starting point is with the highest rank to other feasible point(if any). We want to investigate the performance for different initialization and thus performed the following experiment. We would consider the following approaches:

SDR and Schatten-SDR Biswas's relaxation solved by CVX and Schatten-0.5-regularized approach with initialization with SDR.

Trace and Schatten-Trace Trace regularized SDR solved by CVX and Schatten-0.5-regularized approach with initialization with Trace.

MVU and Schatten-MVU Maximum variance unfolding regularized SDR solved by CVX and Schatten-0.5-regularized approach with initialization with MVU.

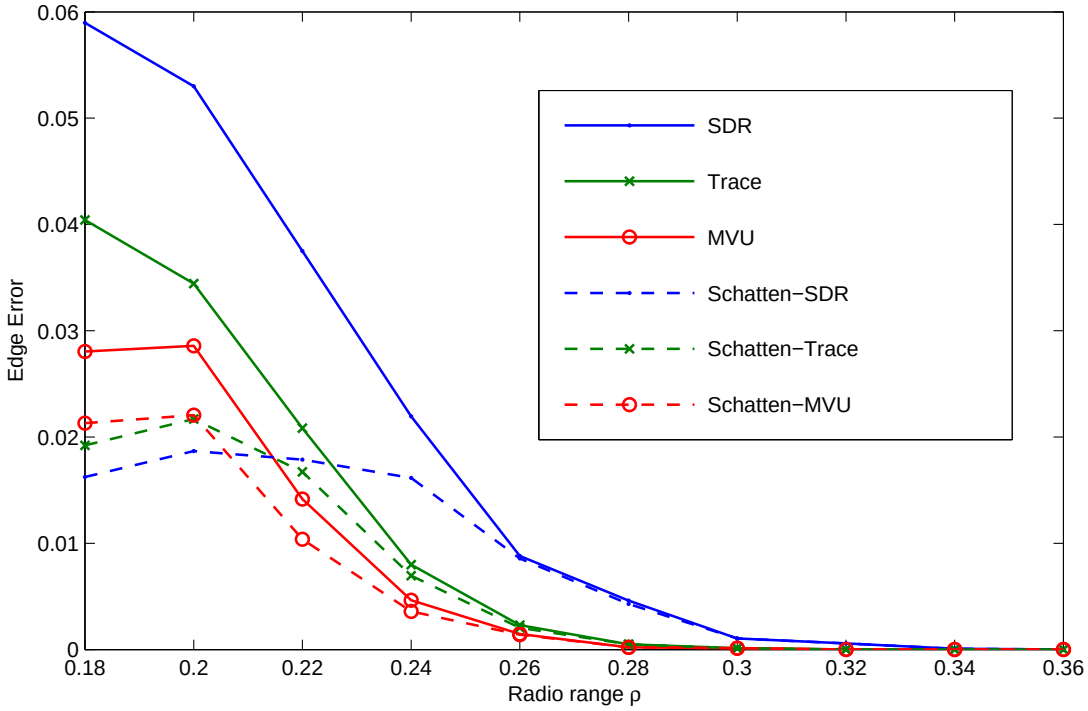


Figure 4.7: The Average edge error of different initialization

It is also the first time that we compare with MVU-regularization. Indeed, MVU demonstrated its strong regularization power in terms of EE and rank. Again we run 100 simulations with radio range $\rho = 0.18 - 0.36$ for 50 random sensors and 3 random anchors, and the result is in Figure 4.7.

For each approach in SDR, Trace and MVU, its Schatten version would give better average edge error and their performances converge to their initialization. When the graph is very sparse, i.e. $\rho = 0.18$ and 0.2, Schatten-SDR performs better. MVU has beaten others and its Schatten version further refines the solution and produces even more better result.

This numerical experiment shows that there is a strong relation between first-order point of Schatten p-regularization and the lower rank and better EE solution. And this potential algorithm can be deemed as a refinement step to lower the rank of solution.

4.4 Exact Measurement without Anchors

It is interesting whether Schatten-p regularization would have similar performance when anchors are absent. In this section, we would demonstrate the effectiveness of Schatten-p regularization to such anchor-free case. As the realization is subject to rotation, reflection and transformation, it is hard to measure its position error. Thus we would use the edge error and the rank as the metric only. Furthermore, we use the globally rigid graph so that if the rank of solution is equal to d, the exact geometric relation is reconstructed. In this experiment we compare two approaches. To begin, let's introduce the unified framework we would consider.

$$\begin{aligned} \text{minimize} \quad & \mu\|Y\|_p^p \\ \text{subject to} \quad & tr((e_i - e_j)^T Y (e_i - e_j)) = d_{ij}^2, \quad \forall (i,j) \in E_{ss} \\ & Y \succeq 0 \end{aligned} \tag{4.2}$$

And here are the approaches,

Trace: Trace regularized SDR(i.e., $\mu = 1, p = 1$) solved by CVX with Solver SeDuMi.

Schatten: Schatten 0.5−regularized SDR(i.e. $\mu = 1, p = 0.5$) solved for ϵ–first–order critical point by aforementioned potential reduction algorithm with the conversion, yet choosing initial point from Trace.

We run 100 simulations with varying adding edges from 0 to 150. And we count number of instances that can be fully recovered, i.e. the rank is d. When the instance is fully recovered, the global optimizer is achieved. One can find out that Schatten can fully recover more cases than the Trace in Figure 4.8. We should remark that more information may not always help under the framework of rank minimization as there are some recovered instances cannot be exactly recovered after the edge adding.

Moreover, similar to previous experiment, the rank of Schatten is much lower to the Trace approach, see Figure 4.9.

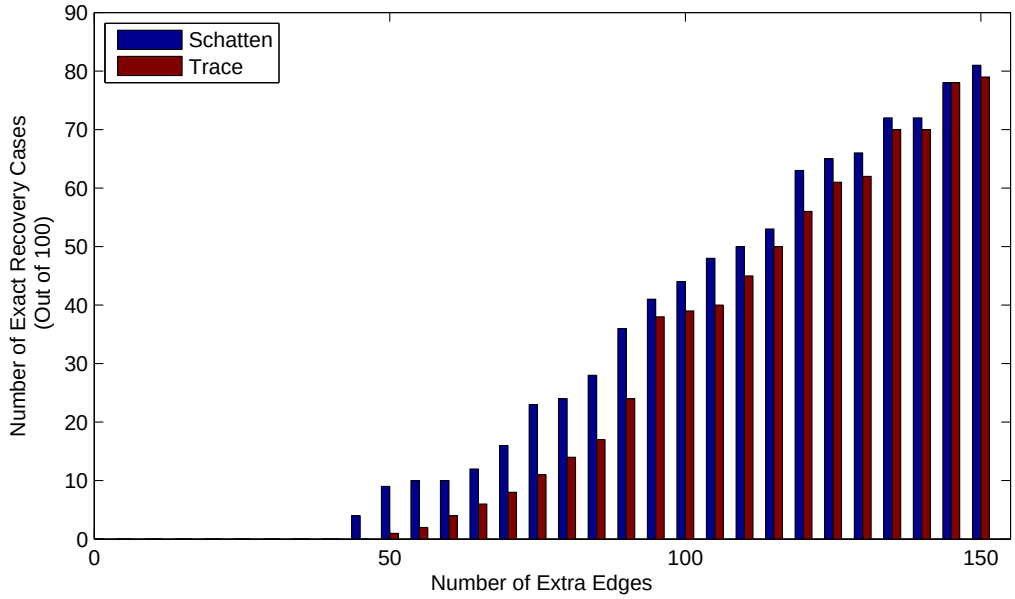


Figure 4.8: The number of exact recovery cases

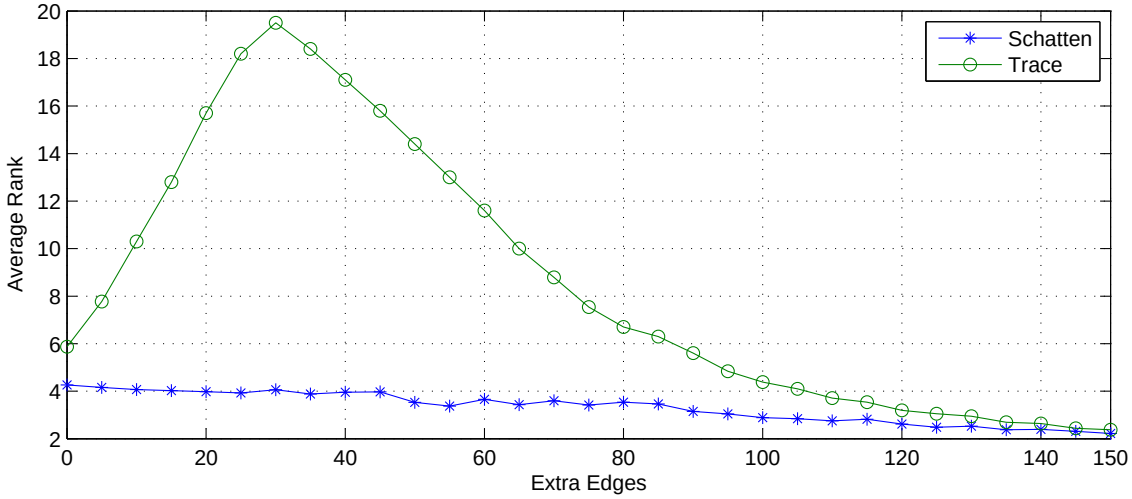


Figure 4.9: The average rank, globally rigid graph with extra edges 0 to 150

4.5 Noisy Measurement with Anchors

In this section, we would demonstrate the effectiveness of Schatten p-regularization when the measurement is not accurate. We assume the measurements are perturbed by the Gaussian noise, i.e.

$$\begin{aligned} d_{ij} &= \hat{d}_{ij}|1+\epsilon_{ij}|, \quad \text{for } (i,j) \in E_{ss} \quad \text{and } \epsilon_{ij} \sim \mathcal{N}(0,\sigma_{ij}^2) \\ \bar{d}_{ij} &= \hat{\bar{d}}_{ij}|1+\bar{\epsilon}_{ij}|, \quad \text{for } (i,j) \in E_{as} \quad \text{and } \bar{\epsilon_{ij}} \sim \mathcal{N}(0,\bar{\sigma}_{ij}^2) \end{aligned} \tag{4.3}$$

where $\hat{d}_{ij}$ and $\hat{\bar{d}}_{ij}$ are the true distances.

Lets recall the model (2.9) and we would add the regularizer to it. Let's restate the regularized model.

$$\begin{aligned} \text{minimize} \quad & \pi(W) = \text{tr}(CW) + \mu\|W\|_p^p \\ \text{subject to} \quad & \text{tr}(A_i W) = b_i \qquad\qquad \text{for } i = 1,\ldots,m, \\ & W = \begin{bmatrix} X_1 & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \ddots & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & X_{n_b} \end{bmatrix}, \\ & W \in \mathcal{S}_+^n \end{aligned} \tag{4.4}$$

where n_b is the number of block diagonal matrices. The number of constraints is numerous, and the size of decision matrix is large too. If there are 50 sensors with 3 anchors, the size of matrix and number of constraints will be at least 300, not even counting the constraints for non-block diagonal entry.

4.5.1 Block Diagonal Structure Preserving Property

For each entry that does not belong to the block diagonal matrices, we need an equality to force it to be zero, otherwise the whole matrix is no longer positive semi-definite. Of course, such constraints can be easily handled in some standard SDP algorithms, such as SDPT3[30], which can exploit the property of sparsity, block-diagonal decision matrices and low-rank coefficient. However, the potential

reduction algorithm for Schatten p regularization has not yet be adjusted to do so. Still, we can show that for some given instances and conditions, block diagonal structure can be maintained.

Proposition 4.1. *For model (4.4), define* $\mathcal{B} := \{(j,k)|W_{jk} \neq 0\}$. *A matrix* X *has same block-diagonal structure with* W *if* $X_{jk} = 0,\ \forall (j,k) \notin \mathcal{B}$. *If the following statements are true, then the solution returned by aforementioned potential reduction algorithm preserves the same block-diagonal structure.*

1. $(A_i)_{jk} = 0,\ \forall (j,k) \notin \mathcal{B}$ *for* $i = 1$ *to* m,
2. $C_{jk} = 0,\ \forall (j,k) \notin \mathcal{B}$, *and*
3. *An initial point with same block-diagonal structure is given.*

Proof. We need two important properties of block diagonal matrix to complete the proof. Given two block diagonal matrices $A, B \in \mathcal{S}_n^+$ with same block structure.

1. AB has the same block structure.
2. A+B has the same block structure.
3. Let $U\Pi U^T$ be the singular value decomposition of A. $A^p := U\Pi^p U^T$ has the same block diagonal structure.

Now, we consider the current iterate $\bar{W}$ with the block diagonal structure mentioned in model (4.4). Recall we would update the iterate by $\bar{W}^+ = \bar{W} + W^{1/2}DW^{1/2}$, where $D = \beta \cdot \frac{\left(I - \mathcal{A}^T(\mathcal{A}\mathcal{A}^T)^{-1}\mathcal{A}\right)Q}{\|(I - \mathcal{A}^T(\mathcal{A}\mathcal{A}^T)^{-1}\mathcal{A})Q\|_F}$. By the aforementioned properties, to show $\bar{W}^+$ has same block diagonal structure, it is sufficient to show D has same block diagonal structure.

For D, we neglect the scalar term $\frac{\beta}{\|(I - \mathcal{A}^T(\mathcal{A}\mathcal{A}^T)^{-1}\mathcal{A})Q\|_F}$ as it would not affect the block diagonal structure and only consider the matrix part $Q - \mathcal{A}^T(\mathcal{A}\mathcal{A}^T)^{-1}\mathcal{A}Q$. As $Q = (\rho/\pi(\bar{W}))(W^{1/2}CW^{1/2} + \mu p \bar{W}^p) - I$, Q has same block diagonal structure.

The remained component would be $\mathcal{A}^T\left((\mathcal{A}\mathcal{A}^T)^{-1}\mathcal{A}Q\right)$. As the adjoint operator $\mathcal{A}^T(\nu) := \sum_{i=1}^m \nu_i \bar{W}^{1/2} A_i \bar{W}^{1/2}$, we can show that $\mathcal{A}^T\left((\mathcal{A}\mathcal{A}^T)^{-1}\mathcal{A}Q\right)$ has same block diagonal structure, thus D has same block diagonal structure. □

4.5.2 Unit-disk Graph

Due to the complexity problem, we would only test for the cases with 30 sensors and 3 anchors randomly with unit-disk graph setting. We will compare the following approaches:

SDR Model (4.4) with $\mu = 0$ solved by SDPT3

Schatten Model (4.4) with $p = 0.5$ and varying μ solved by potential reduction algorithm

For the regularizer parameter μ, we will select from 1e-2 to 1e2. And we would use three levels of noisy factor(the standard derviation σ), that is 0.05, 0.15 and 0.25. The result is in the Figure. 4.10. The result shows that Schatten is an effective estimator only when the noise is low. Moreover, this experiment shows the effect of μ is not obvious, whatever in terms of rank, EE and PE. However, one can still find the performance difference in a finer scale. Indeed, the optimal choice of μ in terms of quality is still unknown as it varies a lot from case to case. But it would form a nearly U-shape, which can be further investigation later. In the view of computation complexity, it is easy to observe that the algorithm would terminate faster when the μ is smaller.

4.5.3 Gradient Descent Method

During the relaxation, the solution would be projected into the required dimension d. After the projection, a refinement by gradient descent method would be applied on the solution[3]. Recall the original objective function (2.6), which

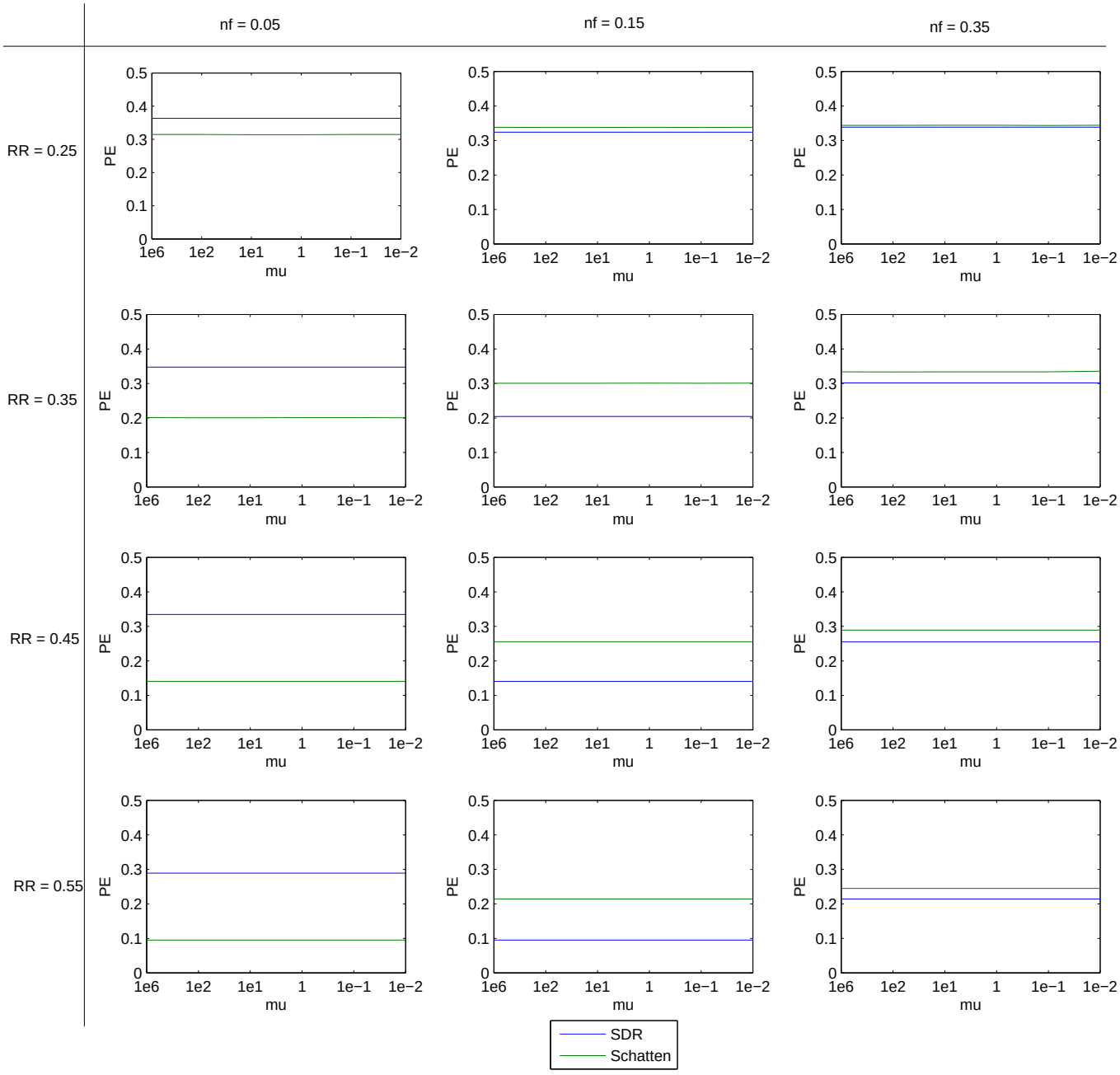


Figure 4.10: The average PE, unit-disk graph and inexact measurement(RR: radio range, nf = noisy factor)

we denote it by $f(X)$. We can would get the following gradient with respect to sensor x_j:

$$\bigtriangledown_j f(X) = \left\{ \sum_{(i,j)\in E_{ss}} \frac{1}{\sigma_{ij}^2}(1 - \frac{d_{ij}}{\|x_i - x_j\|})(x_j - x_i) + \sum_{(i,j)\in E_{as}} \frac{1}{\bar{\sigma}_{ij}^2}(1 - \frac{\bar{d}_{ij}}{\|a_i - x_j\|})(x_j - a_i) \right\}.$$

In each iteration, we update $X^+ = X - [\alpha \bigtriangledown_1 f(X), \ldots, \alpha \bigtriangledown_n f(X)]$, where the step size $\alpha \in (0, 1]$ is chosen by back-tracking procedure. We applied the gradient descent method for 50 iterations to solution of above simulation. And we get the result in Figure. 4.11 and 4.12.

After the extra refinement by gradient descent method, the PE of Schatten seems slightly being better than that of SDR. However, the performances are not stable at all. It should be noted that the gradient descent method is also a local minimization method. Beside the PE, another essential metric is the objective value, in the perspective of optimization. In Figure 4.12, Schatten returned better objective values as Schatten could help in lowering dimension and thus producing a better initial point for gradient descent method. However, the lower objective value do not guarentee that the PE would be better, which is related to the estimation issue. For example, for the case that radio range is 0.55 and noisy factor is 0.05, although there is improvement in the objective value, the PE is even worse.

4.6 Heuristic Methods

In this section, we will introduce some heuristic methods, which combine the existing rank regularizers. For simplicity, we would consider for the simple case, i.e., model (4.1). Here is the heuristic method we would consider:

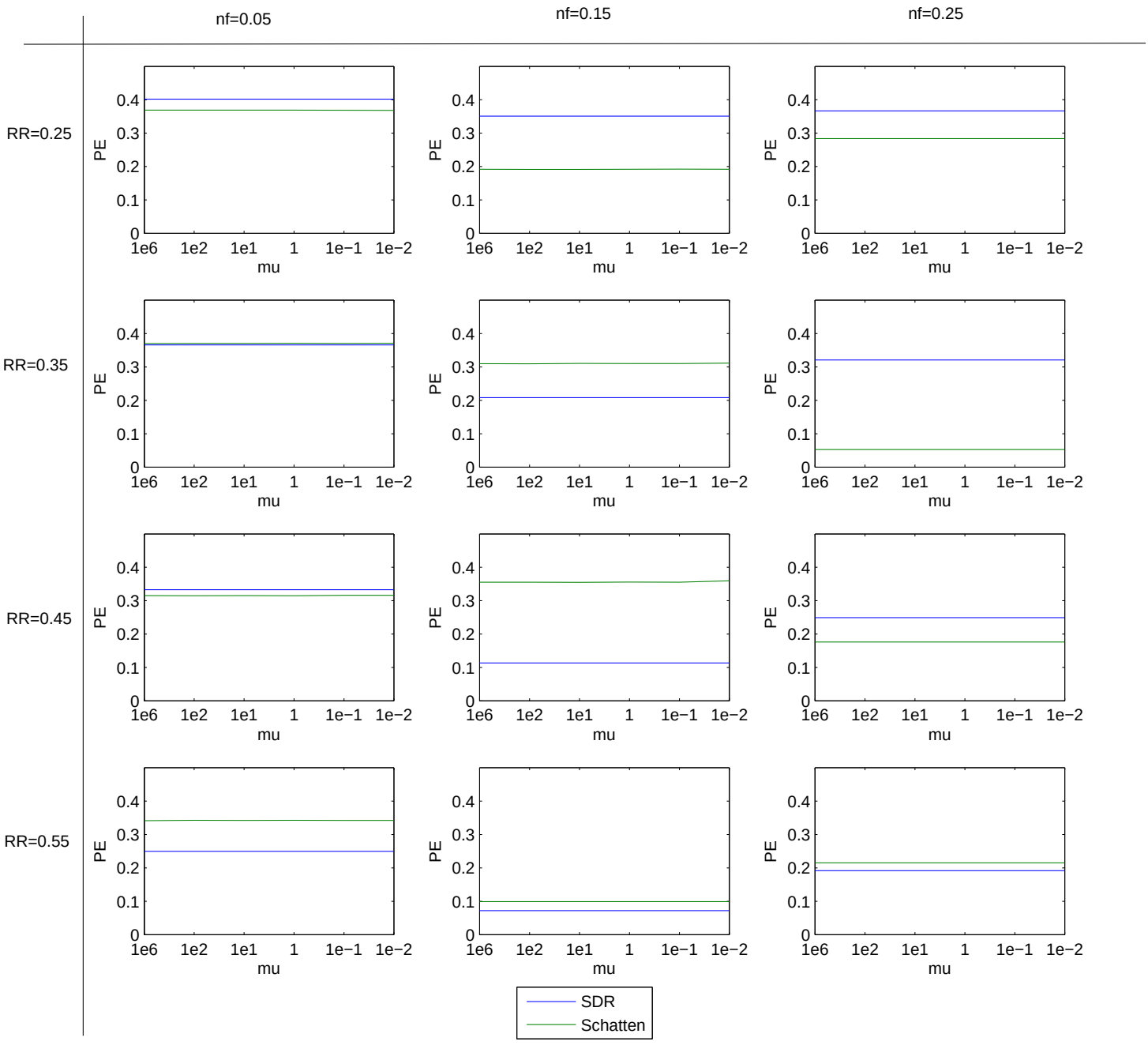


Figure 4.11: The average PE after gradient descent method, unit-disk graph and inexact measurement(RR: radio range, nf = noisy factor)

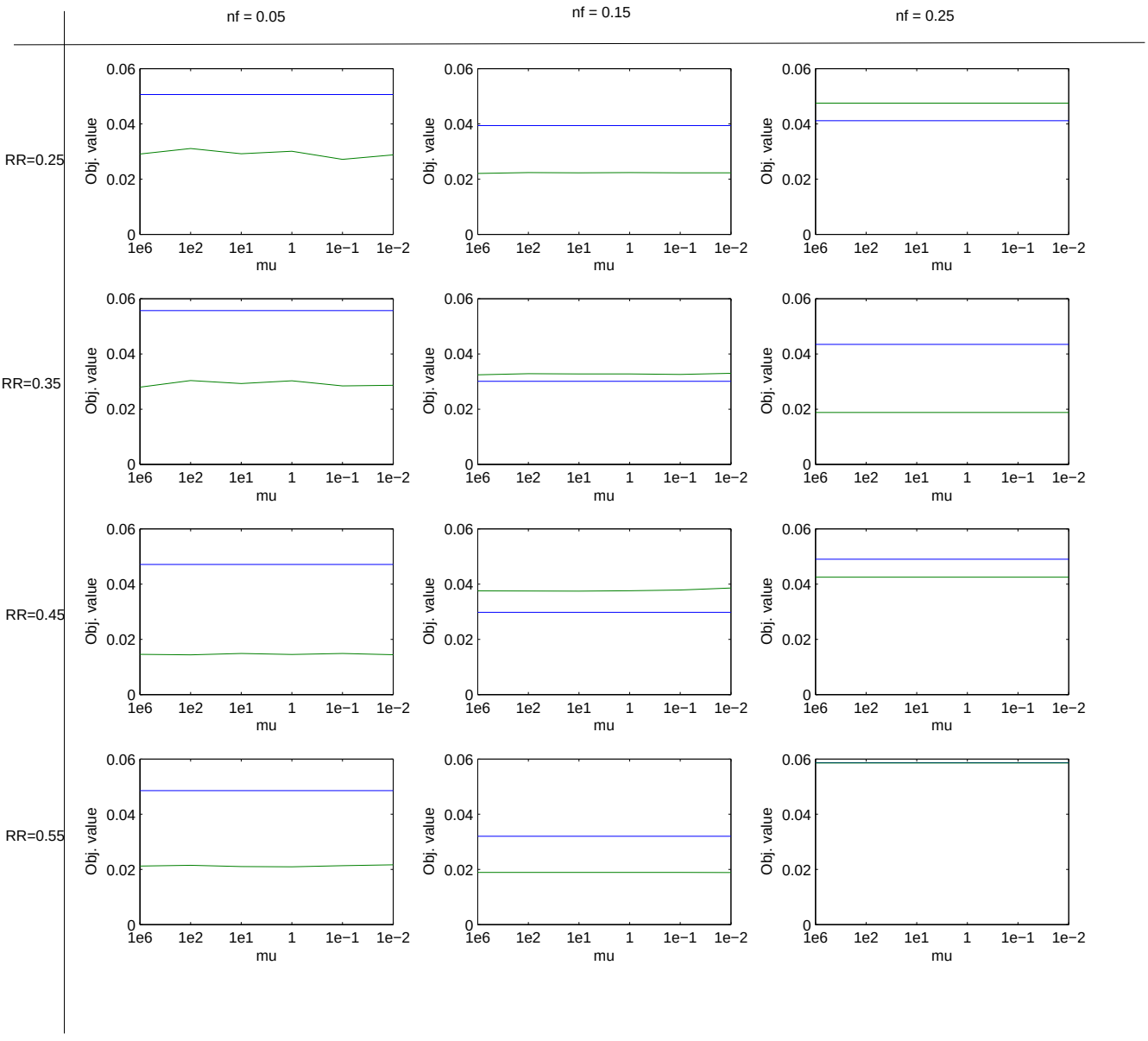


Figure 4.12: The average objective value after gradient descent method, unit-disk graph and inexact measurement(RR: radio range, nf = noisy factor)

$$
\begin{aligned}
\text{Minimize} \quad & \mu\|Z\|_p^p + f_r(Z) \\
\text{subject to} \quad & tr((0; e_i - e_j)^T Z(0; e_i - e_j)) = d_{ij}^2, \quad \forall (i,j) \in E_{ss} \\
& tr((a_k; -e_j)^T Z(a_k; -e_j)) = \bar{d}_{ij}^2, \qquad \forall (i,j) \in E_{as} \\
& Z = \begin{pmatrix} I_d & X \\ X^T & Y \end{pmatrix} \\
& Y \succeq X^T X,
\end{aligned}
\tag{4.5}
$$

where f_r is other regularizer or heuristic objective function. Mentioned before, Shamsi et al. proposed various heuristic objective functions for unit-disk graph setting. However, for simplicity, we would only consider the traditional regularizers:

$$f_{r1}(X) = tr(X) \text{ and } f_{r2}(X) = -tr(X),$$

where the former is the trace-norm regularizer and the latter is the MVU regularizer.

The result is very similar to subsection 4.3.4 where the combination of Schatten and other regularizer can help much in reducing rank and EE, especially when the problem is sparse. Result are shown in Figures. 4.13 and 4.14.

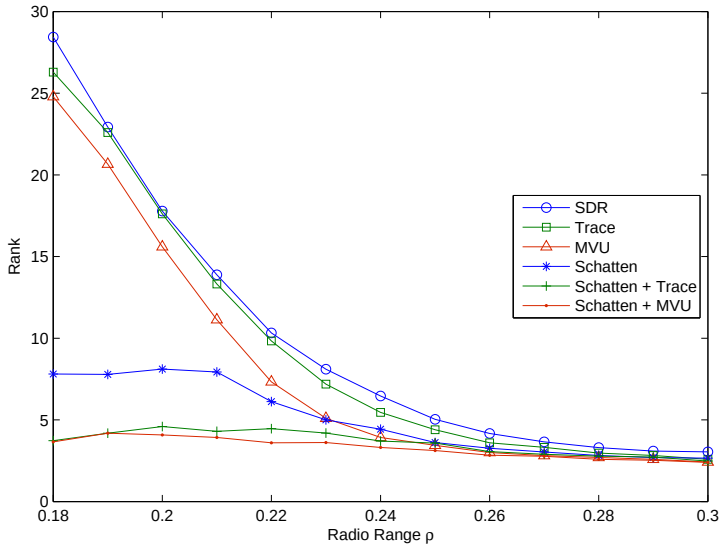


Figure 4.13: The average rank of heuristic methods, unit-disk graph with radio range from 0.18-0.3

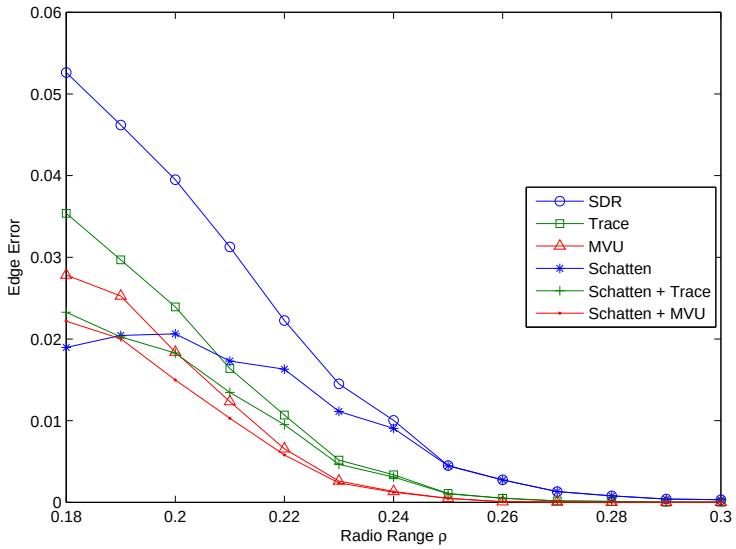


Figure 4.14: The average EE of heuristic methods, unit-disk graph with radio range from 0.18-0.3

Chapter 5

Conclusion and Future Works

In this thesis, we proposed the Schatten Quasi-Norm Minimization to the Sensor Network Localization Problem. We presented an interior-point algorithm to approximate the ϵ–first–order point of the Schatten regularized Semidefinite Programming. Based on the strong modeling power of Semidefinite Programming on Sensor Network Localization Problem, we believed that such algorithm can be applied to most cases, which is not only limited to the three cases we covered.

We performed extensive simulations on demonstrating the effect of the Schatten Quasi-Norm Minimization. There is an effective reduction on the rank of solution. And it also gave improvement in terms of Edge Error, which is a measure on how the information is recovered. Moreover, we also discussed some computation issues, such as choice of parameters and a specified conversion process to tackle the assumption. However, we believed that there might exist some theoretical justification on the parameters as the choice of parameters plays an important role to the algorithm.

At last, we also provided some heuristic methods on combining some existing regularized methods. It would be interesting to investigate the combinations with other existing methods.

Bibliography

[1] Rabab K. Ward Angshul Majumdar. An algorithm for sparse mri reconstruction by schatten p-norm minimization. In *Magnetic resonance imaging*, 2012.

[2] James Aspnes, David Goldenberg, and Yang Richard Yang. On the Computational Complexity of Sensor Network Localization. In *Proceedings of the 1st International Workshop on Algorithmic Aspects of Wireless Sensor Networks (ALGOSENSORS 2004)*, volume 3121 of *Lecture Notes in Computer Science*, pages 32–44. Springer–Verlag, 2004.

[3] Pratik Biswas, Tzu-Chen Lian, Ta-Chung Wang, and Yinyu Ye. Semidefinite Programming Based Algorithms for Sensor Network Localization. *ACM Transactions on Sensor Networks*, 2(2):188–220, 2006.

[4] Pratik Biswas, Kim-Chuan Toh, and Yinyu Ye. A Distributed SDP Approach for Large–Scale Noisy Anchor–Free Graph Realization with Applications to Molecular Conformation. *SIAM Journal on Scientific Computing*, 30(3):1251–1277, 2008.

[5] Pratik Biswas and Yinyu Ye. Semidefinite Programming for Ad Hoc Wireless Sensor Network Localization. In *Proceedings of the 3rd International Symposium on Information Processing in Sensor Networks (IPSN 2004)*, pages 46–54, 2004.

[6] R. Alsindi C. Gentile, N. Alsindi and C. Teolis. *Geolocation Techniques*. Springer New York, 2013.

[7] Lance Doherty, Kristofer S. J. Pister, and Laurent El Ghaoui. Convex Position Estimation in Wireless Sensor Networks. In *Proceedings of the 20th Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM 2001)*, volume 3, pages 1655–1663, 2001.

[8] Maryam Fazel. *Matrix Rank Minimization with Applications*. PhD thesis, Department of Electrical Engineering, Stanford University, Stanford, CA 94305, 2002.

[9] Dongdong Ge, Xiaoye Jiang, and Yinyu Ye. A Note on the Complexity of L_p Minimization. *Mathematical Programming, Series B*, 129(2):285–299, 2011.

[10] Osman Gler and Yinyu Ye. Convergence behavior of interior-point algorithms. *Mathematical Programming*, 60(1-3):215–228, 1993.

[11] Christoph Helmberg. Semidefinite Programming for Combinatorial Optimization. Technical Report ZR–00–34, Konrad–Zuse–Zentrum für Informationstechnik Berlin, Takustraße 7, D–14195, Berlin, Germany, 2000.

[12] Adel Javanmard and Andrea Montanari. Localization from Incomplete Noisy Distance Measurements. In *Proceedings of the 2011 IEEE International Symposium on Information Theory (ISIT 2011)*, pages 1584–1588, 2011.

[13] E. Kaplan. *Understanding GPS: Principles and Applications*. Artech House, 1996.

[14] Sunyoung Kim, Masakazu Kojima, and Hayato Waki. Exploiting Sparsity in SDP Relaxation for Sensor Network Localization. Technical Report B–447, Department of Mathematical and Computing Sciences, Tokyo Institute of Technology, Oh–Okayama, Meguro–ku, Tokyo 152–8552, Japan, 2008.

[15] Yunhao Liu and Zheng Yang. *Location, Localization, and Localizability*. Springer Science+Business Media, LLC, New York, 2011.

[16] Karthik Mohan and Maryam Fazel. Iterative reweighted algorithms for matrix rank minimization. In *Journal of Machine Learning Research*, 2012.

[17] Dragos Niculescu and Badri Nath. Ad hoc positioning system (aps). In *IN GLOBECOM*, pages 2926–2931, 2001.

[18] Feiping Nie, Heng Huang, and Chris Ding. Low–Rank Matrix Recovery via Efficient Schatten p–Norm Minimization. In *Proceedings of the 26th AAAI Conference on Artificial Intelligence (AAAI-12)*, page to appear, 2012.

[19] Jiawang Nie. Sum of Squares Method for Sensor Network Localization. *Computational Optimization and Applications*, 43(2):151–179, 2006.

[20] Ting Kei Pong and Paul Tseng. (Robust) Edge–Based Semidefinite Programming Relaxation of Sensor Network Localization. *Mathematical Programming, Series A*, 130(2):321–358, 2011.

[21] Andreas Savvides, Chih-Chieh Han, and Mani B. Strivastava. Dynamic fine-grained localization in ad-hoc networks of sensors. In *Proceedings of the 7th annual international conference on Mobile computing and networking*, MobiCom '01, pages 166–179, New York, NY, USA, 2001. ACM.

[22] James B. Saxe. Embeddability of Weighted Graphs in k–Space is Strongly NP–Hard. In *Proceedings of the 17th Allerton Conference in Communication, Control, and Computing*, pages 480–489, 1979.

[23] Ali H. Sayed, Alireza Tarighat, and Nima Khajehnouri. Network–Based Wireless Location: Challenges Faced in Developing Techniques for Accurate Wireless Location Information. *IEEE Signal Processing Magazine*, 22(4):24–40, 2005.

[24] Zirui Zhou Anthony Man-Cho So Yinyu Ye Senshan Ji, Kam-Fung Sze. Beyond convex relaxation: A polynomial-time non-convex optimization approach to network localization. In *Proceedings of the 25th Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM 2001)*, page To appear, 2013.

[25] Iman Shames, Brian D. O. Anderson, and Barış Fidan. On the Use of Convex Optimization in Sensor Network Localization and Synchronization. In *Proceedings of the 1st IFAC Workshop on Estimation and Control of Networked Systems*, pages 228–233, 2009.

[26] D. Shamsi, N. Taheri, Z. Zhu, and Y. Ye. On sensor network localization using sdp relaxation. 2010.

[27] Anthony Man-Cho So and Yinyu Ye. Theory of Semidefinite Programming for Sensor Network Localization. *Mathematical Programming, Series B*, 109(2):367–384, 2007.

[28] Ryo Sugihara and Rajesh K. Gupta. Sensor Localization with Deterministic Accuracy Guarantee. In *Proceedings of the 30th IEEE International Conference on Computer Communications (INFOCOM 2011)*, pages 1772–1780, 2011.

[29] M.J. Todd. Semidefinite optimization. *Acta Numerica*, 10:515–560, 2001.

[30] K. C. Toh, M.J. Todd, and R. H. Ttnc. Sdpt3 – a matlab software package for semidefinite programming. *OPTIMIZATION METHODS AND SOFTWARE*, 11:545–581, 1999.

[31] Xiaoping Wang, Jun Luo, Yunhao Liu, Shanshan Li, and Dezun Dong. Component-based localization in sparse wireless networks. *IEEE/ACM Trans. Netw.*, 19(2):540–548, April 2011.

[32] Yinyu Ye. Approximating Quadratic Programming with Bound and Quadratic Constraints. *Mathematical Programming*, 84(2):219–226, 1999.

[33] Ke Zhou and Stergios I. Roumeliotis. Multirobot Active Target Tracking with Combinations of Relative Observations. *IEEE Transactions on Robotics*, 27(4):678–695, 2011.

[34] Zhisu Zhu, Anthony Man-Cho So, and Yinyu Ye. Universal Rigidity and Edge Sparsification for Sensor Network Localization. *SIAM Journal on Optimization*, 20(6):3059–3081, 2010.