

Sensor Network Localization, Euclidean Distance Matrix Completions, and Graph Realization*

Yichuan Ding Nathan Krislock Jiawei Qian Henry Wolkowicz

November 17, 2008

University of Waterloo
Department of Combinatorics and Optimization
Waterloo, Ontario N2L 3G1, Canada
Research Report CORR 2006-23

Key Words: Sensor Network Localization, Anchors, Graph Realization, Euclidean Distance Matrix Completions, Semidefinite Programming.

AMS Subject Classification: 90C22, 90C51, 51K05, 15A48

Abstract

We study Semidefinite Programming, *SDP*, relaxations for Sensor Network Localization, *SNL*, with anchors and with noisy distance information. The main point of the paper is to view *SNL* as a (nearest) Euclidean Distance Matrix, *EDM*, completion problem that does not distinguish between the anchors and the sensors. We show that there are advantages for using the well studied *EDM* model. In fact, the set of anchors simply corresponds to a given fixed clique for the graph of the *EDM* problem.

We next propose a method of projection when large cliques or dense subgraphs are identified. This projection reduces the size, and improves the stability, of the relaxation. In addition, by viewing the problem as an *EDM* completion problem, we are able to derive a new approximation scheme for the sensors from the *SDP* approximation. This yields, on average, better low rank approximations for the low dimensional realizations. This further emphasizes the theme that *SNL* is in fact just an *EDM* problem.

We solve the *SDP* relaxations using a primal-dual interior/exterior-point algorithm based on the Gauss-Newton search direction. By not restricting iterations to the interior, we usually get lower rank optimal solutions and thus, better approximations for the *SNL* problem. We discuss the relative stability and strength of two formulations and the corresponding algorithms that are used. In particular, we show that the quadratic formulation arising from the *SDP* relaxation is better conditioned than the linearized form that is used in the literature.

*Research supported by Natural Sciences Engineering Research Council Canada.

1 Introduction

We study ad hoc wireless sensor networks and the sensor network localization, ***SNL***, problem with anchors. The anchors have fixed known locations and the sensor-sensor and sensor-anchor distances are known (approximately) if they are within a given (radio) range. The problem is to approximate the positions of all the sensors, given that we have only this partial information on the distances. We use semidefinite programming, ***SDP***, relaxations to find approximate solutions to this problem.

In this paper we emphasize that the existence of anchors is not special. The ***SNL*** problem with anchors can be modelled as the well studied (nearest) Euclidean Distance Matrix, ***EDM***, completion problem. The only property that distinguishes the anchors is that the corresponding set of nodes yields a clique in the graph. This results in the failure of the Slater constraint qualification for the ***SDP*** relaxation. However, we can take advantage of this liability/instability. We can find the smallest face of the ***SDP*** cone that contains the feasible set and project the problem onto this face to obtain a smaller/stable problem. In addition, by viewing the problem as an ***EDM*** completion problem, we are able to derive a new approximation scheme for the sensors from the ***SDP*** approximation. This yields, on average, better low rank approximations for the low dimensional realizations. Moreover, by treating the anchors this way, we show that other cliques of sensors or dense parts of the graph can similarly result in a reduction in the size of the problem.

We solve the ***SDP*** relaxations using a primal-dual interior/exterior-point algorithm based on the Gauss-Newton search direction, [20]. This approach takes a full step of length one once the iterates get *close enough* to the optimal solution. By not restricting the iterations to the interior, they usually do not converge to the analytic center of the optimal set. This often results in lower rank optimal solutions and thus, yields better approximations for the ***SNL*** problem. We also discuss and compare the relative robustness and stability of two primal-dual algorithms for ***SNL***. Our tests show that the approach based on a quadratic formulation is better conditioned and more efficient than the approach based on a linearization. These tests confirm the theoretical results on the conditioning of different barriers, [15, 9].

1.1 Related Work and Applications

The geometry of ***EDM*** has been extensively studied in the literature; see e.g. [14, 10] and more recently [2, 1] and the references therein. The latter two references studied algorithms based on ***SDP*** formulations of the ***EDM*** completion problem. Several recent papers have developed algorithms for the ***SDP*** relaxation designed specifically for ***SNL*** with anchors, e.g. [6, 17, 7, 4, 25, 5, 31, 19]. Relaxations using second order cones are studied in e.g., [28, 29].

The ***SDP*** relaxations solve a closest ***SDP*** matrix problem and generally use the ℓ_1 norm. The ℓ_2 norm is used in [19], where the noise in the radio signal is assumed to come from a multivariate normal distribution with mean 0 and covariance matrix $\sigma^2 I$, so that the least squares estimates are the maximum likelihood estimates. We use the ℓ_2 norm as well in this paper. Our approach follows that in [2] for ***EDM*** completion *without* anchors.

Various applications for **SNL** are discussed in the references mentioned above. These applications include e.g. natural habitat monitoring, earthquake detection, and weather/current monitoring.

1.2 Outline

The formulation of the **SNL** problem as a least squares approximation problem is presented in Section 2. This section also includes background, notation, and information on the linear transformations and adjoints used in the model. In particular, since this paper emphasizes using **EDM**, we provide details on *distance geometry*, and on the linear mappings between **EDM** and **SDP** matrices.

The **SDP** relaxations are presented in Section 3. This section contains the details for the four main contributions of the paper:

- (i) the connection of **SNL** with **EDM**; (ii) the projection technique for cliques and dense sets of sensors; (iii) the improved approximation scheme for locating the sensors from the **SDP** relaxation; and (iv) a derivation and discussion of the better conditioning of the quadratic formulation of the **SDP** relaxation relative to the linear formulation used in the literature.

We then continue with the numerical tests in Section 4. Section 4.1 provides the comparisons for two approximation schemes for locating the sensors from the **SDP** relaxation; Section 4.2 provides the comparisons between the linear and quadratic **SDP** models. Concluding remarks are given in Section 5.¹

2 Background and Notation

Let the n unknown (sensor) points be $p^1, p^2, \dots, p^n \in \mathbb{R}^r$, r the embedding dimension, and let the m known (anchor) points be $a^1, a^2, \dots, a^m \in \mathbb{R}^r$. Let $X^T = [p^1, p^2, \dots, p^n]$, and $A^T = [a^1, a^2, \dots, a^m]$. We identify a^i with p^{n+i} , for $i = 1, \dots, m$, and sometimes treat these as unknowns; we define

$$P^T := (p^1, \dots, p^n, a^1, \dots, a^m) = (p^1, \dots, p^{n+m}) = (X^T \ A^T). \quad (2.1)$$

Since we can always translate all the sensors and anchors, and to avoid some special trivial cases, we assume the following.

Assumption 2.1 *The embedding dimension, and the number of sensors and anchors, satisfy $n \gg m > r$, $A^T e = 0$, and A is full column rank.*

¹A previous version of this paper includes an appendix with: notation needed for the implementation of the algorithms; the optimality and duality theory for the **SDP** relaxations; and a derivation of our primal-dual interior/exterior-point algorithm. URL: orion.math.uwaterloo.ca/~hwoikowi/henry/reports/oct08.pdf

Now define $(\mathcal{N}_e, \mathcal{N}_u, \mathcal{N}_l)$ to be the index sets of specified distance values, upper bounds, lower bounds, respectively, of the distances d_{ij} between pairs of nodes from $\{p^i\}_1^n$ (sensors); let $(\mathcal{M}_e, \mathcal{M}_u, \mathcal{M}_l)$, denote the same for distances between a node from $\{p^i\}_1^n$ (sensor) and a node from $\{a^k\}_1^m$ (anchor). Define (the partial Euclidean Distance Matrix) E with elements

$$E_{ij} = \begin{cases} d_{ij}^2, & \text{if } ij \in \mathcal{N}_e \cup \mathcal{M}_e, \\ \|p^i - p^j\|^2 = \|a^{i-n} - a^{j-n}\|^2, & \text{if } i, j > n, \\ 0, & \text{otherwise.} \end{cases}$$

The underlying graph is $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ with node set $\mathcal{V} = \{1, \dots, m+n\}$ and edge set $\mathcal{E} = \mathcal{N}_e \cup \mathcal{M}_e \cup \{ij : i, j > n\}$. The subgraph induced by the anchors is complete; i.e., the set of anchors forms a clique in the graph. Similarly, we define the matrix of (squared) distance upper bounds U^b and the matrix of (squared) distance lower bounds L^b for $ij \in \mathcal{N}_u \cup \mathcal{M}_u$ and $\mathcal{N}_l \cup \mathcal{M}_l$, respectively. Corresponding to the above, we let W_p, W_{pa}, W_a be weight matrices for the sensor-sensor, sensor-anchor, anchor-anchor, distances respectively. For example, they simply could be 0–1 matrices to indicate when a distance is unknown–known. Alternatively, a weight could be used to verify the confidence in the value of the distance. The weights in W_a corresponding to anchor-anchor distances can be made *large*, since these distances are known.

Since there can be noise in the data, we minimize the weighted least squares error.

$$\begin{aligned} \min f_1(P) := & \frac{1}{2} \sum_{(i,j) \in \mathcal{N}_e} (W_p)_{ij} (\|p^i - p^j\|^2 - E_{ij})^2 \\ & + \frac{1}{2} \sum_{(i,k) \in \mathcal{M}_e} (W_{pa})_{ik} (\|p^i - a^k\|^2 - E_{ik})^2 \\ & \left(+ \frac{1}{2} \sum_{i,j > n} (W_a)_{ij} (\|p^i - p^j\|^2 - E_{ij})^2 \right) \\ (\mathbf{SNL}_{LS}) \quad \text{subject to} \quad & \|p^i - p^j\|^2 \leq U_{ij}^b \quad \forall (i, j) \in \mathcal{N}_u \quad \left(n_u = \frac{|\mathcal{N}_u|}{2} \right) \\ & \|p^i - a^k\|^2 \leq U_{ik}^b \quad \forall (i, k) \in \mathcal{M}_u \quad \left(m_u = \frac{|\mathcal{M}_u|}{2} \right) \\ & \|p^i - p^j\|^2 \geq L_{ij}^b \quad \forall (i, j) \in \mathcal{N}_l \quad \left(n_l = \frac{|\mathcal{N}_l|}{2} \right) \\ & \|p^i - a^k\|^2 \geq L_{ik}^b \quad \forall (i, k) \in \mathcal{M}_l \quad \left(m_l = \frac{|\mathcal{M}_l|}{2} \right) \\ & (\|p^i - p^j\|^2 = E_{ij} \quad \forall i, j > n). \end{aligned} \tag{2.2}$$

This is a *hard* problem to solve due to the nonconvex objective and constraints. We included the anchor-anchor distances within brackets *both* in the objective and constraints. This is to emphasize that we could treat them with large weights in the objective or as holding exactly without error in the constraints.

2.1 Distance Geometry

The geometry for *EDM* has been studied in e.g., [23, 13, 16, 27], and more recently in e.g., [2],[1]. Further theoretical properties can be found in e.g., [3, 12, 18, 21]. Since we emphasize that the *EDM* theory can be used to solve the *SNL*, we now include an overview

of the tools we need for **EDM**. In particular, we show the relationships between **EDM** and **SDP**.

2.1.1 Linear Transformations and Adjoint Related to EDM

We use the notation from [19]. We work in vector spaces of real matrices, $\mathcal{M}^{s \times t}$, equipped with the trace inner-product $\langle A, B \rangle = \text{trace } A^T B$ and induced Frobenius norm $\|A\|_F^2 = \text{trace } A^T A$. We let \mathcal{S}^n denote the vector space of real symmetric $n \times n$ matrices; \mathcal{S}_+^n and \mathcal{S}_{++}^n denote the cones of positive semidefinite matrices and positive definite matrices, respectively. We identify **SDP** = \mathcal{S}_+^n ; and, we use the Löwner partial order notation $A \succeq 0$ and $A \succ 0$ for $A \in \mathcal{S}_+^n$ and $A \in \mathcal{S}_{++}^n$, respectively. For a given $B \in \mathcal{S}^n$, the linear transformation $\text{diag}(B) \in \mathbb{R}^n$ denotes the diagonal of B ; for $v \in \mathbb{R}^n$, the adjoint linear transformation is the diagonal matrix $\text{diag}^*(v) = \text{Diag}(v) \in \mathcal{S}^n$. We now define several linear operators on \mathcal{S}^n .

$$\mathcal{D}_e(B) := \text{diag}(B) e^T + e \text{diag}(B)^T, \quad \mathcal{K}(B) := \mathcal{D}_e(B) - 2B, \quad (2.3)$$

where e is the vector of ones. The adjoint linear operators are

$$\mathcal{D}_e^*(D) = 2\text{Diag}(De), \quad \mathcal{K}^*(D) = 2(\text{Diag}(De) - D). \quad (2.4)$$

By abuse of notation, we allow \mathcal{D}_e to act on \mathbb{R}^n : $\mathcal{D}_e(v) = ve^T + ev^T$, $v \in \mathbb{R}^n$. The linear operator \mathcal{K} maps the cone of positive semidefinite matrices onto the cone of Euclidean distance matrices (denoted **EDM**), i.e., $\mathcal{K}(\mathbf{SDP}) = \mathbf{EDM}$.

Lemma 2.1 ([1]) *Define the linear operator on \mathcal{S}^n , $\text{offDiag}(S) := S - \text{Diag}(\text{diag}(S))$. Then, the following holds:*

- the range and nullspace satisfy $\mathcal{N}(\mathcal{K}) = \mathcal{R}(\mathcal{D}_e)$;
- $\mathcal{R}(\mathcal{K}) = S_H$, where the hollow subspace $S_H := \{D \in \mathcal{S}^n : \text{diag}(D) = 0\}$;
- $\mathcal{R}(\mathcal{K}^*) = S_c$, where the centered subspace $S_c := \{B \in \mathcal{S}^n : Be = 0\}$;
- the Moore-Penrose generalized inverse $\mathcal{K}^\dagger(D) = -\frac{1}{2}J(\text{offDiag}(D))J$, where $J := I - \frac{1}{n}ee^T$.

■

Proposition 2.1 ([1, 19])

1. Let S_D denote the subspace of diagonal matrices in \mathcal{S}^n . Then

$$\begin{aligned} S_c = \mathcal{N}(\mathcal{D}_e^*) = \mathcal{R}(\mathcal{K}^*) = \mathcal{R}(\mathcal{K}^\dagger) &\perp \mathcal{N}(\mathcal{K}) = \mathcal{R}(\mathcal{D}_e) \\ S_H = \mathcal{R}(\mathcal{K}) = \mathcal{N}(\mathcal{D}_e) &\perp S_D = \mathcal{N}(\mathcal{K}^*) = \mathcal{R}(\mathcal{D}_e^*). \end{aligned}$$

2. Let $\left[V \quad \frac{1}{\sqrt{n}}e \right]$ be an $n \times n$ orthogonal matrix. Then

$$Y \succeq 0 \iff Y = V\hat{Y}V^T + \mathcal{D}_e(v) \succeq 0, \text{ for some } \hat{Y} \in \mathcal{S}^{n-1}, v \in \mathbb{R}^n.$$

3. Suppose that $0 \preceq B \in \mathcal{S}^n$. Then $D = \mathcal{K}(B)$ is EDM. ■

3 SDP Relaxations Based on EDM Model

Current **SDP** relaxations for the feasibility problem for **SNL** relax $Y = XX^T$ to

$$Y \succeq XX^T, \text{ or equivalently, } Z_s = \begin{bmatrix} I_r & X^T \\ X & Y \end{bmatrix} \succeq 0. \quad (3.5)$$

This is in combination with the constraints

$$\begin{aligned} \text{trace} \begin{pmatrix} 0 \\ e_i - e_j \end{pmatrix} \begin{pmatrix} 0 \\ e_i - e_j \end{pmatrix}^T Z_s &= E_{ij}, \quad \forall ij \in \mathcal{N}_e, \\ \text{trace} \begin{pmatrix} -a_k \\ e_i \end{pmatrix} \begin{pmatrix} -a_k \\ e_i \end{pmatrix}^T Z_s &= E_{ij}, \quad \forall ij \in \mathcal{M}_e, \quad i < j = n + k; \end{aligned} \quad (3.6)$$

see [6, 4, 25, 5, 17].

We use relaxations of $\bar{Y} := PP^T = \begin{bmatrix} XX^T & XA^T \\ AX^T & AA^T \end{bmatrix}$. The dimensions are: $X \in \mathcal{M}^{n,r}$; $A \in \mathcal{M}^{m,r}$; $P \in \mathcal{M}^{m+n,r}$; $\bar{Y} \in \mathcal{S}^{m+n}$. We first reformulate **SNL** using matrix notation to get the equivalent **EDM** problem

$$\begin{aligned} \min \quad & f_2(\bar{Y}) := \frac{1}{2} \|W \circ (\mathcal{K}(\bar{Y}) - E)\|_F^2 \\ \text{subject to} \quad & g_u(\bar{Y}) := H_u \circ (\mathcal{K}(\bar{Y}) - \bar{U}^b) \leq 0 \\ (\mathbf{SNL}_M) \quad & g_l(\bar{Y}) := H_l \circ (\mathcal{K}(\bar{Y}) - \bar{L}^b) \geq 0 \\ & \bar{Y} - PP^T = 0 \\ & (\mathcal{K}(\bar{Y}))_{22} = \mathcal{K}(AA^T), \end{aligned} \quad (3.7)$$

where $\mathcal{K}(\bar{Y})_{22}$ denotes the 2, 2 block, and $W \in \mathcal{S}^{n+m}$ is the weight matrix having a positive ij-element if $(i, j) \in \mathcal{N}_e \cup \mathcal{M}_e \cup \{(ij) : i, j > n\}$, 0 otherwise. H_u, H_l are 0–1 matrices where the ij-th element equals 1 if an upper (resp. lower) bound exists; and it is 0 otherwise. By abuse of notation, we consider the functions g_u, g_l as implicitly acting on only the nonzero components in the upper triangular parts of the matrices that result from the Hadamard products with H_u, H_l , respectively. We include in brackets the constraints on the clique corresponding to the anchor-anchor distances.

Remark 3.1 *The function $f_2(\bar{Y}) = f_2(PP^T)$, and it is clear that $f_2(PP^T) = f_1(P)$ in (2.2). Note that the functions f_2, g_u, g_l act only on \bar{Y} ; and, the locations of the anchors and sensors are completely hidden in the hard, nonconvex quadratic constraint $\bar{Y} = PP^T = \begin{bmatrix} XX^T & XA^T \\ AX^T & AA^T \end{bmatrix}$. The problem **SNL_M** is a linear least squares problem with nonlinear constraints. The objective function is generally underdetermined. This can result in ill-conditioning problems, e.g., [11]. Therefore, reducing the number of variables helps with stability.*

We now relax the *hard* quadratic constraint in (3.7)

$$\bar{Y} = \begin{bmatrix} \bar{Y}_{11} & \bar{Y}_{21}^T \\ \bar{Y}_{21} & AA^T \end{bmatrix} = PP^T = \begin{bmatrix} XX^T & XA^T \\ AX^T & AA^T \end{bmatrix}, \quad (3.8)$$

with (3.5), or equivalently with $\bar{Y} \succeq PP^T$. We show that this is equivalent to the simpler $\bar{Y} \succeq 0$. We include details on problems and weaknesses with the relaxation. We first present several lemmas.

Lemma 3.1 *Suppose that the partitioned symmetric matrix $\begin{bmatrix} Y_{11} & Y_{21}^T \\ Y_{21} & AA^T \end{bmatrix} \succeq 0$. Then $Y_{21}^T = XA^T$, with $X = Y_{21}^T A(A^T A)^{-1}$.*

Proof. Let $A = U\Sigma_r V^T$ be the compact singular value decomposition, SVD, $0 \prec \Sigma_r \in \mathcal{S}^r$. Moreover, suppose that $\begin{bmatrix} U & \bar{U} \end{bmatrix}$ is an orthogonal matrix. Therefore, the range spaces $\mathcal{R}(U) = \mathcal{R}(A)$ and the nullspace $\mathcal{N}(A^T) = \mathcal{R}(\bar{U})$. Consider the nonsingular congruence

$$\begin{aligned} 0 &\preceq \begin{bmatrix} I & 0 \\ 0 & [U \ \bar{U}] \end{bmatrix}^T \begin{bmatrix} Z & Y_{21}^T \\ Y_{21} & AA^T \end{bmatrix} \begin{bmatrix} I & 0 \\ 0 & [U \ \bar{U}] \end{bmatrix} \\ &= \begin{bmatrix} Z & Y_{21}^T [U \ \bar{U}] \\ [U \ \bar{U}]^T Y_{21} & \begin{bmatrix} \Sigma_r^2 & 0 \\ 0 & 0 \end{bmatrix} \end{bmatrix}. \end{aligned}$$

This implies that $Y_{21}^T \bar{U} = 0$. This in turn means that $\mathcal{N}(Y_{21}^T) \supset \mathcal{N}(A^T)$, or equivalently, $\mathcal{R}(A) \supset \mathcal{R}(Y_{21})$. Note that the orthogonal projection onto $\mathcal{R}(A)$ is $A(A^T A)^{-1} A^T$. Therefore, $Y_{21}^T = Y_{21}^T A(A^T A)^{-1} A^T = (Y_{21}^T A(A^T A)^{-1}) A^T$, i.e., we can choose $X = Y_{21}^T A(A^T A)^{-1}$. \blacksquare

In the recent literature, e.g., [7, 6, 17], it is common practice to relax the hard constraint (3.8) to a tractable semidefinite constraint, $\bar{Y} \succeq PP^T$, or equivalently, $\bar{Y}_{11} \succeq XX^T$ with $\bar{Y}_{21} = AX^T$. The following lemma presents several characterizations for the resulting feasible set.

Lemma 3.2 *Let $A = U\Sigma_r V^T$ be the compact SVD of A , and let P, \bar{Y} be partitioned as in (2.1), (3.8),*

$$P = \begin{bmatrix} P_1 \\ P_2 \end{bmatrix}, \quad \bar{Y} = \begin{bmatrix} \bar{Y}_{11} & \bar{Y}_{21}^T \\ \bar{Y}_{21} & \bar{Y}_{22} \end{bmatrix}.$$

Define the semidefinite relaxation of the hard quadratic constraint (3.8) as:

$$G(P, \bar{Y}) := PP^T - \bar{Y} \preceq 0, \quad \bar{Y}_{22} = AA^T, \quad P_2 = A. \quad (3.9)$$

By abuse of notation, we allow G to act on spaces of different dimensions. Then we get the following equivalent representations of the corresponding feasible set \mathcal{F}_G :

$$\mathcal{F}_G = \{(P, \bar{Y}) : G(P, \bar{Y}) \preceq 0, \bar{Y}_{22} = AA^T, P_2 = A\}; \quad (3.9a)$$

$$\mathcal{F}_G = \left\{ (P, \bar{Y}) : G(X, Y) \preceq 0, \bar{Y}_{11} = Y, \bar{Y}_{21} = AX^T, \bar{Y}_{22} = AA^T, P = \begin{bmatrix} X \\ A \end{bmatrix} \right\}; \quad (3.9b)$$

$$\mathcal{F}_G = \left\{ (P, \bar{Y}) : Z = \begin{bmatrix} Z_{11} & Z_{21}^T \\ Z_{21} & Z_{22} \end{bmatrix} \succeq 0, \bar{Y} = \begin{bmatrix} I_n & 0 \\ 0 & A \end{bmatrix} Z \begin{bmatrix} I_n & 0 \\ 0 & A \end{bmatrix}^T, \right. \\ \left. Z_{22} = I_r, X = Z_{21}^T, P = \begin{bmatrix} X \\ A \end{bmatrix} \right\}; \quad (3.9c)$$

$$\mathcal{F}_G = \left\{ (P, \bar{Y}) : \bar{Y} \succeq 0, \bar{Y}_{22} = AA^T, X = \bar{Y}_{21}^T A (A^T A)^{-1}, P = \begin{bmatrix} X \\ A \end{bmatrix} \right\}; \quad (3.9d)$$

$$\mathcal{F}_G = \left\{ (P, \bar{Y}) : Z = \begin{bmatrix} Z_{11} & Z_{21}^T \\ Z_{21} & Z_{22} \end{bmatrix} \succeq 0, \bar{Y} = \begin{bmatrix} I_n & 0 \\ 0 & U \end{bmatrix} Z \begin{bmatrix} I_n & 0 \\ 0 & U \end{bmatrix}^T, \right. \\ \left. Z_{22} = \Sigma_r^2, X = \bar{Y}_{21}^T A (A^T A)^{-1} = Z_{21}^T \Sigma_r^{-1} V^T, P = \begin{bmatrix} X \\ A \end{bmatrix} \right\}. \quad (3.9e)$$

Moreover, the function G is convex in the Löwner (semidefinite) partial order; and the feasible set \mathcal{F}_G is a closed convex set.

Proof. Recall that the cone of positive semidefinite matrices is self-polar. Let $Q \succeq 0$ and $\phi_Q(P) = \text{trace } QPP^T$. Convexity of G follows from positive semidefiniteness of the Hessian $\nabla^2 \phi_Q(P) = I \otimes Q$, where \otimes denotes the Kronecker product.

In addition,

$$0 \succeq G(P, \bar{Y}) = P P^T - \bar{Y} = \begin{bmatrix} X X^T - \bar{Y}_{11} & X A^T - \bar{Y}_{21}^T \\ A X^T - \bar{Y}_{21} & 0 \end{bmatrix}$$

holds if and only if

$$0 \succeq G(X, \bar{Y}_{11}) = X X^T - \bar{Y}_{11}, \text{ and } A X^T - \bar{Y}_{21} = 0.$$

This shows the equivalence with (3.9b). A Schur complement argument, with $\bar{Y}_{11} = Y$, shows the equivalence with $\begin{bmatrix} Y & X \\ X^T & I_r \end{bmatrix} \succeq 0$, i.e., with the set in (3.9c). The equivalence with (3.9d) follows from Lemma 3.1.

To show the equivalence with the final expression (3.9e), we note that $\bar{Y} \succeq 0, \bar{Y}_{22} = AA^T$, implies that there is no strictly feasible $\bar{Y} \succ 0$. Therefore, we project the feasible set onto the *minimal cone* or face (see [8]). This yields the minimal face that contains the feasible set of \bar{Y} , i.e.,

$$\bar{Y} = \begin{bmatrix} I_n & 0 \\ 0 & U \end{bmatrix} Z_w \begin{bmatrix} I_n & 0 \\ 0 & U \end{bmatrix}^T, Z_w \succeq 0, Z_w \in \mathcal{S}^{n+r}. \quad (3.15)$$

The result follows since the constraint $\bar{Y}_{22} = AA^T$ holds if and only if Z_w is blocked as $Z_w := \begin{bmatrix} Y & W^T \\ W & \Sigma_r^2 \end{bmatrix} \succeq 0$. (More simply, one can show the equivalence of (3.9e) with (3.9c) by

using the compact SVD of A . However, the longer proof given above emphasizes that the reduction comes from using a projection to obtain the Slater constraint qualification.) ■

The above Lemma 3.2 shows that we can treat the set of anchors as a set of sensors for which all the distances are known, i.e., the set of corresponding nodes is a clique. The fact that we have a clique and the diagonal $m \times m$ block AA^T in \bar{Y} is rank deficient, $r < m$, means that the Slater constraint qualification, $\bar{Y} \succ 0$, cannot hold. Therefore, we can project onto the minimal cone containing the feasible set and thus reduce the size of the problem; see Lemma 3.2, (3.9e), i.e., the variable $\bar{Y} \in \mathcal{S}^{n+m}$ is reduced in size to $Z \in \mathcal{S}^{n+r}$. The reduction can be done by using any point in the relative interior of the minimal cone, e.g., any feasible point of maximum rank. The equivalent representations in (3.9c) and in (3.9e) illustrate this.

Remark 3.2 *The above reduction to Y in Lemma 3.2, (3.9b), and the application of the Schur complement, allows us to use the smaller dimensional semidefinite constrained variable*

$$Z_s = \begin{bmatrix} I_r & X^T \\ X & Y \end{bmatrix} \succeq 0 \in \mathcal{S}^{n+m}, \quad \bar{Y}_{11} = Y, \bar{Y}_{21} = AX^T. \quad (3.16)$$

*This appears in e.g., [6]. The **SDP** constraint in (3.16) is linear and is equivalent to the quadratic constraint $Y - XX^T \succeq 0$.*

Note that the mapping $Z_s = Z_s(X, Y) : \mathcal{M}^{n,r} \times \mathcal{S}^n \rightarrow \mathcal{S}^{n+r}$ is not onto. This means that the Jacobian of the optimality conditions cannot be full rank, i.e., this formulation introduces instability into the model. A minor modification corrects this, i.e., the I_r constraint is added explicitly.

$$Z = \begin{bmatrix} Z_{11} & Z_{21}^T \\ Z_{21} & Z_{22} \end{bmatrix} \succeq 0, \quad Z_{11} = I_r, \bar{Y}_{11} = Z_{22}, \bar{Y}_{21} = AZ_{21}^T.$$

A numerical comparison of the linear and quadratic constraints is presented in Section 4.2.

3.1 Clique Reductions using Minimal Cone Projection

The equivalent representations of the feasible set given in Lemma 3.2, in particular by (3.9e), show that **SNL** is an **EDM** problem $D = \mathcal{K}(\bar{Y})$, with the additional upper and lower bound constraints as well as the block constraint on the bottom right block of \bar{Y} , see e.g., (3.8).

Remark 3.3 *Suppose that we can increase the size of the clique containing the anchor nodes by adding sensor nodes where the distances are exactly known. Then these sensor nodes can be treated as anchor nodes, though their positions are unknown.*

We can now obtain a relaxation for **SNL** by using the **EDM** problem (3.7) and replacing the hard quadratic constraint with the simpler semidefinite constraint $\bar{Y} \succeq 0$, and $\bar{Y}_{22} = AA^T$. We then observe that the Slater constraint qualification (strict feasibility)

fails. Therefore, we can project onto the minimal cone, i.e., onto the minimal face of the **SDP** cone that contains the feasible set. See [8, 2]. Let

$$U_A = \begin{bmatrix} I_n & 0 \\ 0 & A \end{bmatrix}, \quad U_s = \begin{bmatrix} I_n & 0 \\ 0 & U \end{bmatrix}, \quad \text{where } A = U\Sigma_r V^T. \quad (3.17)$$

We get the following two **SDP** relaxations from the representations in (3.9c) and (3.9e), respectively,

$$\begin{aligned} & \min && f_3(Z) := \frac{1}{2} \|W \circ (\mathcal{K}(U_A Z U_A^T) - E)\|_F^2 \\ & \text{subject to} && H_u \circ (\mathcal{K}(U_A Z U_A^T) - U^b) \leq 0 \\ & (\mathbf{SNL}_{EDMA}) && H_l \circ (\mathcal{K}(U_A Z U_A^T) - L^b) \geq 0 \\ & && Z_{22} = I_r \\ & && Z \succeq 0. \end{aligned} \quad (3.18)$$

and

$$\begin{aligned} & \min && f_3(Z) := \frac{1}{2} \|W \circ (\mathcal{K}(U_s Z U_s^T) - E)\|_F^2 \\ & \text{subject to} && H_u \circ (\mathcal{K}(U_s Z U_s^T) - U^b) \leq 0 \\ & (\mathbf{SNL}_{EDMs}) && H_l \circ (\mathcal{K}(U_s Z U_s^T) - L^b) \geq 0 \\ & && Z_{22} = \Sigma_r^2 \\ & && Z \succeq 0. \end{aligned} \quad (3.19)$$

Remark 3.4 *Note that we do not substitute the constraint on Z_{22} into Z , but leave it explicit. Though this does not change the feasible set, it does change the stability and the dual. This can be compared to the **SDP** relaxation for the Max-Cut problem with constraint that the diagonal of X is all ones, $\text{diag } X = e$, and $X \succeq 0$. However, one does not substitute for the diagonal and rewrite the semidefinite constraint.*

Now suppose that we have another clique of $p > r$ sensors where the exact distances are known and are used as constraints. Then there exists a matrix $\bar{Y} = PP^T$ that has a diagonal rank deficient $p \times p$ block. Since all feasible points are found from elements in the set $\bar{Y} + \mathcal{N}(\mathcal{K})$, we conclude that for p large enough, the diagonal block remains rank deficient for all feasible \bar{Y} , i.e., the Slater constraint qualification fails again, if the corresponding distances are added as constraints.

We now see that we can again take advantage of the loss of the Slater constraint qualification.

Theorem 3.1 *Suppose that the hypotheses and definitions from Lemma 3.2 hold; and suppose that there exists a set of sensors, without loss of generality $\mathcal{T}_c := \{p^{t+1}, \dots, p^n\}$, so that the distances $\|p^i - p^j\|$ are known for all $t+1 \leq i, j \leq n$; i.e., the graph of the partial **EDM** has two cliques, one clique corresponding to the set of known anchors, and the other to the set of sensors \mathcal{T}_c . Let P, \bar{Y} be partitioned as*

$$P = \begin{bmatrix} P_1 \\ P_2 \\ P_3 \end{bmatrix} = \begin{bmatrix} X \\ A \end{bmatrix}, \quad \bar{Y} = \begin{bmatrix} \bar{Y}_{11} & \bar{Y}_{21}^T & \bar{Y}_{31}^T \\ \bar{Y}_{21} & \bar{Y}_{22} & \bar{Y}_{32}^T \\ \bar{Y}_{31} & \bar{Y}_{32} & \bar{Y}_{33} \end{bmatrix} = PP^T,$$

where $P_i = A_i, i = 2, 3$, and $A_3 = A$ corresponds to the known anchors while $P_2 = A_2$ corresponds to the clique of sensors and $X = \begin{bmatrix} P_1 \\ P_2 \end{bmatrix}$ corresponds to all the sensors. Let the **EDM**, $E = \mathcal{K}(\bar{Y})$, be correspondingly blocked

$$E = \begin{bmatrix} E_1 & \cdot & \cdot \\ \cdot & E_2 & \cdot \\ \cdot & \cdot & E_3 \end{bmatrix},$$

so that $E_3 = \mathcal{K}(AA^T)$ are the anchor-anchor squared distances, and $E_2 = \mathcal{K}(P_2P_2^T)$ are the squared distances between the sensors in the set \mathcal{T}_c . Let

$$B = \mathcal{K}^\dagger(E_2).$$

Then the following hold:

1. $Be = 0$ and

$$\bar{Y}_{22} = B + \bar{y}_2 e^T + e \bar{y}_2^T \succeq 0, \text{ for some } \bar{y}_2 \in \mathcal{R}(B) + \alpha e, \alpha \geq 0, \text{ with } \text{rank}(\bar{Y}_{22}) \leq r; \quad (3.20)$$

2. The feasible set \mathcal{F}_G in Lemma 3.2 can be formulated as

$$\mathcal{F}_G := \left\{ (P, \bar{Y}) : Z = \begin{bmatrix} Z_{11} & Z_{21}^T & Z_{31}^T \\ Z_{21} & Z_{22} & Z_{32}^T \\ Z_{31} & Z_{32} & Z_{33} \end{bmatrix} \succeq 0, \bar{Y} = \begin{bmatrix} I_t & 0 & 0 \\ 0 & U_2 & 0 \\ 0 & 0 & U \end{bmatrix} Z \begin{bmatrix} I_t & 0 & 0 \\ 0 & U_2 & 0 \\ 0 & 0 & U \end{bmatrix}^T, \right. \\ \left. Z_{33} = \Sigma_r^2, X = \begin{bmatrix} Z_{31}^T \\ U_2 Z_{32}^T \end{bmatrix} \Sigma_r^{-1} V^T, P = \begin{bmatrix} X \\ A \end{bmatrix} \right\}, \quad (3.21)$$

or equivalently as

$$\mathcal{F}_G = \left\{ (P, \bar{Y}) : Z = \begin{bmatrix} Z_{11} & Z_{21}^T & Z_{31}^T \\ Z_{21} & Z_{22} & Z_{32}^T \\ Z_{31} & Z_{32} & Z_{33} \end{bmatrix} \succeq 0, \bar{Y} = \begin{bmatrix} I_t & 0 & 0 \\ 0 & U_2 & 0 \\ 0 & 0 & A \end{bmatrix} Z \begin{bmatrix} I_t & 0 & 0 \\ 0 & U_2 & 0 \\ 0 & 0 & A \end{bmatrix}^T, \right. \\ \left. Z_{33} = I_r, X = \begin{bmatrix} Z_{31}^T \\ U_2 Z_{32}^T \end{bmatrix}, P = \begin{bmatrix} X \\ A \end{bmatrix} \right\}, \quad (3.22)$$

where $\hat{B} := B + 2ee^T = [U_2 \ \bar{U}_2] \begin{bmatrix} D_2 & 0 \\ 0 & 0 \end{bmatrix} [U_2 \ \bar{U}_2]^T$ is the orthogonal diagonalization of \hat{B} , with $D_2 \in \mathcal{S}_{++}^{r_2}, r_2 \leq r + 1$.

Proof. We proceed just as we did in Lemma 3.2, i.e., we reduce the problem by projecting onto a smaller face in order to obtain the Slater constraint qualification.

The equation for \bar{Y}_{22} for some \bar{y}_2 , given in (3.20), follows from the nullspace characterization in Lemma 2.1. Moreover, $\bar{Y}_{22} = P_2 P_2^T$ implies that $\text{rank}(\bar{Y}_{22}) \leq r$, the embedding

dimension; and, $\bar{Y}_{22} \succeq 0, Be = 0$ implies the inclusion $\bar{y}_2 \in \mathcal{R}(B) + \alpha e, \alpha \geq 0$. We can now shift $\bar{P}_2^T = P_2^T - \frac{1}{n-t}(P_2^T e)e^T$; then, for $B = \bar{P}_2 \bar{P}_2^T$, we get $Be = 0$, i.e., this satisfies $B = \mathcal{K}^\dagger(E_2)$ and $\text{rank}(B) \leq r$. Therefore, for any $Y = B + ye^T + ey^T \succeq 0$, we must have $y = \alpha e, \alpha \geq 0$. This implies that \hat{B} has the maximum rank, at most $r + 1$, among all feasible matrices of the form $0 \preceq Y \in B + \mathcal{N}(\mathcal{K})$. The matrix \hat{B} determines the smallest face containing all such feasible Y .

Let $\mathcal{L} := \bar{Y}_{22} + \mathcal{R}(\mathcal{D}_e)$ and \mathcal{F}_e denote the smallest face of \mathcal{S}_+^{n-t} that contains $\mathcal{L} \cap \mathcal{S}_+^{n-t}$. Since \hat{B} is a feasible point of maximum rank, we get

$$\hat{B} = B + \mathcal{D}_e(\bar{y}_2) \in (\mathcal{L} \cap \text{relint } \mathcal{F}_e).$$

Thus, the face

$$\mathcal{F}_e = \{U_2 Z U_2^T : Z \in \mathcal{S}_+^{r_2}\} = \{Y \in \mathcal{S}_+^{n-t} : \text{trace } Y(\bar{U}_2 \bar{U}_2^T) = 0\}.$$

Now, we expand

$$\begin{aligned} \begin{bmatrix} \bar{Y}_{11} & \bar{Y}_{21}^T & \bar{Y}_{31}^T \\ \bar{Y}_{21} & \bar{Y}_{22} & \bar{Y}_{32}^T \\ \bar{Y}_{31} & \bar{Y}_{32} & \bar{Y}_{33} \end{bmatrix} &= \begin{bmatrix} I_t & 0 & 0 \\ 0 & U_2 & 0 \\ 0 & 0 & U \end{bmatrix} \begin{bmatrix} Z_{11} & Z_{21}^T & Z_{31}^T \\ Z_{21} & Z_{22} & Z_{32}^T \\ Z_{31} & Z_{32} & \Sigma_r^2 \end{bmatrix} \begin{bmatrix} I_t & 0 & 0 \\ 0 & U_2 & 0 \\ 0 & 0 & U \end{bmatrix}^T \\ &= \begin{bmatrix} Z_{11} & Z_{21}^T U_2^T & Z_{31}^T U^T \\ U_2 Z_{21} & U_2 Z_{22} U_2^T & U_2 Z_{32}^T U^T \\ U Z_{31} & U Z_{32} U_2^T & U \Sigma_r^2 U^T \end{bmatrix}. \end{aligned}$$

Therefore, $\begin{bmatrix} Y_{31}^T \\ Y_{32}^T \end{bmatrix} = \begin{bmatrix} Z_{31}^T U^T \\ U_2 Z_{32}^T U^T \end{bmatrix}$, and the expressions for Z_{33} and X in (3.21) follow from equation (3.9e) in Lemma 3.2. The result in (3.22) can be obtained similarly. \blacksquare

Remark 3.5 *The above Theorem 3.1 can be extended to sets of sensors that are not cliques, but have many known edges. The key idea is to be able to use $(W_i \circ \mathcal{K})^\dagger(W_i \circ E_i)$ and to characterize the nullspace of $W_i \circ \mathcal{K}$.*

We can apply Theorem 3.1 to further reduce the **SDP** relaxation. Suppose there are a group of sensors for which pairwise distances are all known. This should be a common occurrence, since distances between sensors within radio range are all known. Without loss of generality, we assume the set of sensors to be $\{p^{t+1}, \dots, p^n\}$. Let $E_2, B = \mathcal{K}^\dagger(E_2)$, and U_2 , be found using Theorem 3.1 and denote

$$U_{2A} := \begin{bmatrix} I_n & 0 & 0 \\ 0 & U_2 & 0 \\ 0 & 0 & A \end{bmatrix}, \quad U_{2s} := \begin{bmatrix} I_n & 0 & 0 \\ 0 & U_2 & 0 \\ 0 & 0 & U \end{bmatrix}. \quad (3.23)$$

In \mathbf{SNL}_{EDM_s} , we can replace U_s with U_{2s} and reach a reduced \mathbf{SDP} formulation. Similarly, for \mathbf{SNL}_{EDM_A} . Furthermore, we may generalize to the k clique cases for any positive integer k . We similarly define each $U_i, 2 \leq i \leq k$, and define

$$U_{kA} = \begin{bmatrix} I_n & 0 & \dots & 0 & 0 \\ 0 & U_2 & \dots & 0 & 0 \\ \vdots & & \ddots & & \vdots \\ 0 & 0 & 0 & U_k & 0 \\ 0 & 0 & 0 & 0 & A \end{bmatrix}, \quad U_{ks} = \begin{bmatrix} I_n & 0 & \dots & 0 & 0 \\ 0 & U_2 & \dots & 0 & 0 \\ \vdots & & \ddots & & \vdots \\ 0 & 0 & 0 & U_k & 0 \\ 0 & 0 & 0 & 0 & U \end{bmatrix}. \quad (3.24)$$

Then we can formulate a reduced \mathbf{SDP} for k cliques:

$$\begin{aligned} & \min && f_4(Z) := \frac{1}{2} \|W \circ (\mathcal{K}(U_{kA} Z U_{kA}^T) - E)\|_F^2 \\ & \text{subject to} && H_u \circ (\mathcal{K}(U_{kA} Z U_{kA}^T) - U^b) \leq 0 \\ (\mathbf{SNL}_{k\text{-cliques-A}}) &&& H_l \circ (\mathcal{K}(U_{kA} Z U_{kA}^T) - L^b) \geq 0 \\ &&& Z_{kk} = I_r \\ &&& Z \succeq 0, \end{aligned} \quad (3.25)$$

where Z_{kk} is the last r by r diagonal block of Z . Similarly, we get

$$\begin{aligned} & \min && f_4(Z) := \frac{1}{2} \|W \circ (\mathcal{K}(U_{ks} Z U_{ks}^T) - E)\|_F^2 \\ & \text{subject to} && H_u \circ (\mathcal{K}(U_{ks} Z U_{ks}^T) - U^b) \leq 0 \\ (\mathbf{SNL}_{k\text{-cliques-s}}) &&& H_l \circ (\mathcal{K}(U_{ks} Z U_{ks}^T) - L^b) \geq 0 \\ &&& Z_{kk} = \Sigma_r^2 \\ &&& Z \succeq 0. \end{aligned} \quad (3.26)$$

For a clique with r_e sensors, a U_i is constructed with r_e rows and at most $r + 1$ columns. This implies the dimension of Z has been reduced by $r_e - r - 1$. So if $r = 2$, cliques larger than a triangle help reduce the dimension of Z . As mentioned above, the existence of cliques is highly likely, since edges in the graph exist when sensors are within radio range. Moreover, the above technique extends to dense sets, rather than cliques. The key is finding $B = (W \circ \mathcal{K})^\dagger(W \circ E_{ii})$, for an appropriate submatrix E_{ii} , as well as deriving the nullspace of $W \circ \mathcal{K}$.

3.2 Estimating Sensor Positions

After we solve \mathbf{SNL}_{EDM_A} (or equivalently \mathbf{SNL}_{EDM_s}) to get an optimal solution Z , we can express

$$\bar{Y} = U_s Z U_s^T = \begin{bmatrix} \bar{Y}_{11} & \bar{Y}_{21}^T \\ \bar{Y}_{21} & \bar{Y}_{22} \end{bmatrix}, \quad \bar{Y}_{22} = A A^T, \bar{Y}_{21} = A X^T, \text{ for some } X.$$

To complete the \mathbf{SNL} problem, we have to find an approximation to the matrix $P \in \mathcal{M}^{n+m,r}$, i.e., the matrix that has the sensor locations in the first n rows, denoted X , and the anchor locations in the last m rows, denoted A .

Since $\bar{Y} \in S_+^{n+m}$, there exists $\hat{P} = \begin{bmatrix} \hat{P}_{11} & \hat{P}_{21} \\ \hat{P}_{12} & \hat{P}_{22} \end{bmatrix} \in \mathcal{M}^{n+m}$ such that $\hat{P}\hat{P}^T = \bar{Y}$. By Assumption 2.1, the anchors are centered, i.e., $A^T e = 0$. We can translate the locations in \hat{P} , so that the last m locations are centered, i.e., without loss of generality we have

$$\begin{bmatrix} \hat{P}_{12} & \hat{P}_{22} \end{bmatrix}^T e = 0, \quad \hat{P}\hat{P}^T = \bar{Y}. \quad (3.27)$$

Also,

$$\{\bar{P} \in \mathcal{M}^{n+m} : \bar{Y} = \bar{P}\bar{P}^T\} = \{\bar{P} \in \mathcal{M}^{n+m} : \bar{P} = \hat{P}Q, \text{ for some orthogonal } Q \in \mathcal{M}^{n+m}\}.$$

In other words, from the optimal \bar{Y} , all the possible locations can be obtained by a rotation/reflection of \hat{P} . However, these locations in the rows of \hat{P} are in \mathbb{R}^{n+m} , rather than in the desired embedding space \mathbb{R}^r , where the anchors lie.

Remark 3.6 *Since \mathbf{SNL} is underdetermined, in general, the optimum \bar{Y} is not unique. Therefore, finding a lower rank optimum \bar{Y} should result in better approximations for the sensor locations.*

Following are two methods and comparisons for finding an estimate to the sensor locations, X . First, Method 3.1 is the one currently used in the literature. Second, Method 3.2 is a strengthened new method based on the **EDM** interpretation.

Method 3.1 *Estimate the location of the sensors using X in the optimal Z_s or, equivalently, solve for X using the equation $AX^T = \bar{Y}_{21}$, where \bar{Y}_{21} is from the optimal \bar{Y} .*

In the recent papers on **SNL**, e.g., [7, 6, 17], X is taken directly from the optimal $Z_s = \begin{bmatrix} I_r & X^T \\ X & Y \end{bmatrix}$, see e.g., (3.16). Equivalently, since A is full column rank r , and $AX^T = \bar{Y}_{21}$ is consistent, we can solve for X uniquely from the equation $AX^T = \bar{Y}_{21}$. To provide a comparison with the second Method 3.2, we now describe the underlying geometry of using this X .

Recall that $A = U\Sigma_r V^T$ and $\begin{bmatrix} \hat{P}_{12} & \hat{P}_{22} \end{bmatrix} \begin{bmatrix} \hat{P}_{12} & \hat{P}_{22} \end{bmatrix}^T = AA^T = \begin{bmatrix} A & 0 \end{bmatrix} \begin{bmatrix} A & 0 \end{bmatrix}^T$. Therefore, these three matrices all have the same spectral decomposition and all can be diagonalized using U . This implies that the three matrices $\begin{bmatrix} \hat{P}_{12} & \hat{P}_{22} \end{bmatrix}$, A , $\begin{bmatrix} A & 0 \end{bmatrix}$ can all use the same set of left singular vectors in a compact SVD. Therefore, $\begin{bmatrix} \hat{P}_{12} & \hat{P}_{22} \end{bmatrix} Q = \begin{bmatrix} A & 0 \end{bmatrix}$, for some orthogonal Q , which implies that

$$\exists \hat{Q}, \hat{Q}^T \hat{Q} = I, \text{ with } \bar{P} = \hat{P}\hat{Q} = \begin{bmatrix} \bar{P}_{11} & \bar{P}_{12} \\ A & 0 \end{bmatrix}. \quad (3.28)$$

This yields

$$\bar{P} = \begin{bmatrix} \bar{P}_{11} & \bar{P}_{12} \\ A & 0 \end{bmatrix}, \quad \bar{Y} = \bar{P}\bar{P}^T = \begin{bmatrix} \bar{Y}_{11} & \bar{P}_{11}A^T \\ A\bar{P}_{11}^T & AA^T \end{bmatrix}. \quad (3.29)$$

Since $A\bar{P}_{11}^T = \bar{Y}_{21} = AX^T$, we get $\bar{P}_{11} = X$. Thus, after the rotation/reflection with \hat{Q} to make the bottom m rows equal to A , we see that the first n rows of \bar{P} project exactly onto the rows of X . If we denote the orthogonal projection onto the first r coordinates by P_r , then the resulting operation on the locations in the rows of \hat{P} can be summarized by

$$P_r\bar{P}^T = \left(P_r\hat{Q}^T\right)\hat{P}^T \in \mathbb{R}^r \otimes \{0\}^{n+m-r}, \bar{Y} \approx \bar{Y}_p := \bar{P}P_r\bar{P}^T.$$

We get two sources of error in the approximation for X . First, the product $P_r\hat{Q}^T$ is not necessarily idempotent or symmetric, i.e., not necessarily a projection. Second, the term that is deleted, \bar{P}_{12} , can be arbitrary large, while the rank of \bar{Y} can be as small as $r + 1$. However, we can expect a larger error when the rank of \bar{Y} is larger, since the deleted term \bar{P}_{12} can have more nonzero columns. The relaxation from $\bar{Y}_{11} = XX^T$ to $\bar{Y}_{11} = [\bar{P}_{11} \ \bar{P}_{12}] [\bar{P}_{11} \ \bar{P}_{12}]^T = \bar{P}_{11}\bar{P}_{11}^T + \bar{P}_{12}\bar{P}_{12}^T \succeq XX^T$, shows that using $X = \bar{P}_{11}$ has an error of the order of $\|\bar{Y}_{11} - XX^T\|^2 = \|\bar{P}_{12}\|^2$. \blacksquare

Method 3.2 Suppose that $\hat{P}_r = U_r\Sigma_r^{1/2}$, as found in (3.30) with $\hat{P}_r = \begin{bmatrix} \hat{P}_1 \\ \hat{P}_2 \end{bmatrix}$. We find \hat{Q} as a minimum for $\min_{Q^T Q=I} \|\hat{P}_2 Q - A\|_F^2$. The solution is given analytically by $\hat{Q} = V_Q U_Q^T$, where $U_Q \Sigma_Q V_Q = A^T \hat{P}_2$ is the SVD for $A^T \hat{P}_2$. Then the rows of $\hat{P}_1 \hat{Q}$ are used to estimate the locations of the sensors.

In Method 1, the matrix $\bar{P}P_r\bar{P}^T$ provides a rank r approximation to \bar{Y} . However, if $\|\bar{P}_{12}\|$ in (3.29) is large, then it appears that we have lost information. It is desirable to keep as much of the information from the high dimensional locations in \hat{P} as we can, i.e., the information that is contained in \bar{Y}_{11} . If we do not consider the anchors distinct from the sensors, then we would like to rotate and then project all the rows of \hat{P} onto a subspace of dimension r , i.e., we consider the problem to be an **EDM** completion problem and would like to extract a good approximation of the positions of all the nodes. Since the last m rows corresponding to the anchors originated from a clique, the corresponding graph is rigid and the corresponding projected points will be close to the original anchor positions. We realize this using the spectral decomposition. (See e.g., [2], where error estimates are included.) Let

$$\bar{Y} = \begin{bmatrix} U_r & \bar{U}_r \end{bmatrix} \begin{bmatrix} \Sigma_r & 0 \\ 0 & \Sigma_{n+m-r} \end{bmatrix} \begin{bmatrix} U_r & \bar{U}_r \end{bmatrix}^T.$$

Then, considering the problem as an **EDM** completion problem, we first find a best rank r approximation to \bar{Y} , denoted $\bar{Y}_r := U_r \Sigma_r U_r^T$. Only then do we find a particular full rank factorization

$$\hat{P}_r \in \mathcal{M}^{n+m,r} \text{ with } \bar{Y}_r = \hat{P}_r \hat{P}_r^T, \quad (3.30)$$

i.e., $\hat{P}_r = U_r \Sigma_r^{1/2}$. It remains to find an orthogonal Q in order to find $P = \hat{P}_r Q$. Fortunately, we can use the information from the anchors to find the orthogonal Q . ■

Numerical tests for the two methods are given in Section 4.1. Method 3.2 proved to be, almost always, more accurate. However, Method 3.1 locates all sets of sensors that are uniquely localizable in R^r , see [25].

Remark 3.7 *As above, suppose that \bar{Y} is an optimum for the **SDP** relaxation. The problem of finding a best P to estimate the sensor locations is equivalent to finding*

$$P^* \in \operatorname{argmin}_P \left\{ \|W \circ (\mathcal{K}(PP^T - \bar{Y}))\|_F : P = \begin{bmatrix} X \\ A \end{bmatrix} \right\}.$$

Equivalently, we want to find

$$Y^* \in S_+^{n+m}, \operatorname{rank}(Y^*) = r, Y_{22}^* = AA^T, Y^* = \bar{Y} + \mathcal{N}(W \circ \mathcal{K}).$$

However, finding such a Y^ is equivalent to finding the minimal rank matrix in the intersection of the semidefinite cone and an affine space. This is still an open/hard problem. Recently, [24, 22] proposed randomization methods for **SDP** rank reduction. These methods can generate a low rank positive semidefinite matrix in an approximate affine space.*

4 Numerical Tests

We now discuss results on randomly generated, small **SNL** problems, with connected underlying graphs. The tests were done using MATLAB 7.4. The method for generating the tests problems follows the approach used in [17, 19].

A typical example is presented in Figure 4.1. The first plot is the original graph. The starred positions correspond to the anchors. Then, we get the solution labelled *Linear SDP*, which corresponds to solving the **SDP** relaxation using the ℓ_1 norm. The visible straight lines joining two points correspond to significant errors in the sensor positions found using the relaxation. Next, we added lower bounds to the **SDP** relaxation and used the ℓ_2 norm and a second order cone model. This yields the improved picture labelled *SOC SDP*. The solver in this case was SeDuMi (URL: sedumi.mcmaster.ca). We still see several straight lines indicating a significant difference between the positions of the sensors found and the original sensor positions. The final picture labelled *GN* was found using our own code with a Gauss-Newton direction and a crossover technique. The obvious improvement was a result of the lower rank in the optimal solution of the **SDP** relaxation. This lower rank is a result of our crossover technique, i.e., by not following the central path once we get close enough to the optimal solution, we do not converge to the analytic center of the optimal set of semidefinite matrices.

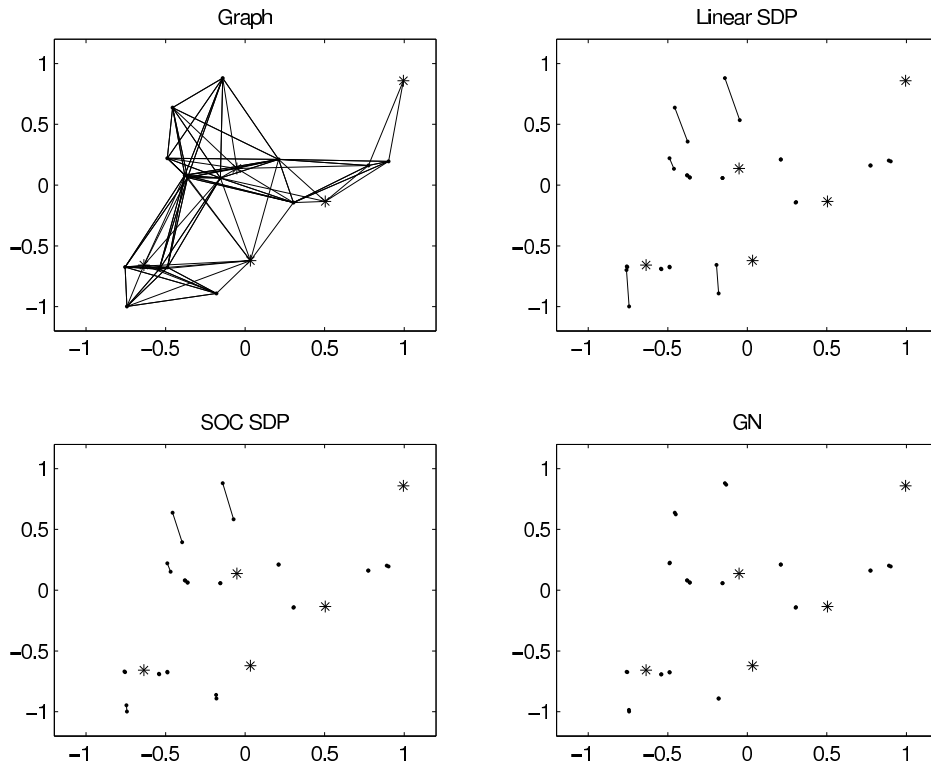


Figure 4.1: Better Solutions from Lower Rank

4.1 Two Methods for Estimating Sensor Locations

Two methods for estimating the sensor locations from a given optimum of the *SDP* relaxation were presented in Section 3.2. Method 3.1 used X directly from the optimal Z obtained from the *SDP* relaxation. This is the method currently employed in the literature. Method 3.2 treated the *SNL* problem as an *EDM* and found a best rank- r approximation. Then the problem was rotated to get the anchors back into their original positions. See Tables 4.1–4.4 for results of 100 random tests, with noise factor nf , for the three solvers labelled above: i.e., *Linear SDP*; *SOC SDP*; *GN*, respectively.

The results in Tables 4.1 and 4.2 are for noise factors $nf = 0$ and $nf = 0.05$, respectively. The tables used the true objective function f_1 in (2.2) as a measure. The three rows correspond to: the percentage of times that Method 3.2 performed better (reduced objective function f_1) over Method 3.1; the average relative reduction of Method 3.2 over Method 3.1; the minimum (and maximum) relative reduction of Method 3.2 over Method 3.1.

Tables 4.3 and 4.4 show the results of comparing the rank of the optimal Y returned by *GN* over the rank of the optimal Y returned by *Linear SDP* or *SOC SDP*. In our numerical experiments, we found that the rank of the optimal solution returned by *GN* was always less or equal to the rank of the optimal solutions returned by *Linear SDP* or *SOC SDP*. The first row gives a percentage of the random instances for which the optimal solution returned by *GN* had a strictly smaller rank than the optimal solution returned by

Method 3.2 over Method 3.1	Linear <i>SDP</i>	SOC <i>SDP</i>	GN
% successful reduction	75%	82%	99%
average % reduction (in total / in successful)	13.5% / 19.5%	16.8% / 21.3%	24.0% / 24.3%
min/max % reduction	-25.7% / 58.9%	-17.9% / 61.4%	-1.3% / 62.5%

Table 4.1: % reduction of Method 3.2 over Method 3.1 for true objective function f_1 in (2.2) (100 problems, $n = 15, m = 5, densityL = .1, densityW = .6$, noise $nf = 0$)

Method 3.2 over Method 3.1	Linear <i>SDP</i>	SOC <i>SDP</i>	GN
% successful reduction	86%	92%	95%
average % reduction (in total / in successful)	9.5% / 11.7%	11.4% / 12.8%	16.6% / 18.2%
min/max % reduction	-13.7% / 38.0%	-13.6% / 38.3%	-33.2% / 48.4%

Table 4.2: % reduction of Method 3.2 over Method 3.1 for true objective function f_1 in (2.2), (100 problems, $n = 15, m = 5, densityL = .1, densityW = .6$, noise $nf = 0.05$)

Linear *SDP* or SOC *SDP*. The second row gives the average percentage reduction in the rank over those instances for which GN returned an optimal solution of strictly smaller rank. The third and fourth rows show the average percentage reduction in the true objective function f_1 in (2.2) over those instances for which GN returned an optimal solution of strictly smaller rank. The third and fourth rows differ in the method used for extracting the sensor positions from the optimal solution. Table 4.3 considers random instances with a zero noise factor; Table 4.4 considers random instances with a 5% noise factor.

GN over (Linear <i>SDP</i> / SOC <i>SDP</i>)	Linear <i>SDP</i>	SOC <i>SDP</i>
% strictly smaller rank	53%	54%
average % rank reduction	44.6%	44.2%
average % obj. reduction (Method 3.1)	89.9%	90.0%
average % obj. reduction (Method 3.2)	90.4%	90.1%

Table 4.3: % reduction of rank and true objective function, f_1 in (2.2), of GN over Linear *SDP* and SOC *SDP* (100 problems, $n = 15, m = 5, densityL = .1, densityW = .6$, noise $nf = 0$)

4.2 Two Methods for Solving SNL

In Figures 4.2 and 4.3, we present results for using the quadratic constraint $Y - XX^T \succeq 0$ compared to the linearized version $\begin{bmatrix} I & X^T \\ X & Y \end{bmatrix} \succeq 0$. We solved many randomly generated problems with various values for the parameters. We present typical results in the figures.

GN over (Linear <i>SDP</i> / SOC <i>SDP</i>)	Linear <i>SDP</i>	SOC <i>SDP</i>
% strictly smaller rank	99%	96%
average % rank reduction	40.5%	37.7%
average % obj. reduction (Method 3.1)	26.4%	21.9%
average % obj. reduction (Method 3.2)	31.8%	24.5%

Table 4.4: % reduction of rank and true objective function, f_1 in (2.2), of GN over Linear *SDP* and SOC *SDP* (100 problems, $n = 15, m = 5, densityL = .1, densityW = .6$, noise $nf = 0.05$)

Figure 4.2 shows the $(-log)$ of the optimal value at each iteration. Figure 4.3 shows the $(-log)$ of the relative gap. Both figures illustrate the surprising result that the quadratic formulation is more efficient, i.e., it obtains higher accuracy with fewer iterations. This is surprising, since we are using a Newton based method that should be faster on functions that are less nonlinear. Therefore, from a numerical analysis viewpoint, since the constraint is not *onto*, it appears that the linear version is more ill-conditioned; see Remark 3.2. In addition, the figures show the high accuracy that can be obtained using the Gauss-Newton direction with a crossover technique, even though these problems are highly ill-conditioned.

These tests provide empirical evidence for the theoretical comparison results on different barriers given in [15, 9]. The results in these references show that the central path is distorted due to the I in the linear formulation constraint. Moreover, the distortion increases with increasing dimension of the I . This agrees with our interpretation that the linear constraint is not *onto*, and the Jacobian is singular.

5 Concluding Remarks

In this paper, we have analyzed the well known *SNL* problem from a new perspective. By considering the set of anchors as a clique in the underlying graph, the *SNL* problem can be studied using traditional *EDM* theory. Our contributions follow mainly due to this *EDM* approach:

1. The Slater constraint qualification can fail for cliques and/or dense subgraphs in the underlying graph. If this happens, then we can project the feasible set of the *SDP* relaxation onto the *minimal cone*. This projection improves the stability and can also significantly reduce the size of the *SDP*. (Algorithms for finding dense subgraphs exist in the literature, e.g., [30, 32, 26].)
2. We provided a geometric interpretation for the current approach of directly using the X from the optimal Z of the *SDP* relaxation, when estimating the sensor positions. We then proposed another method of estimating the sensor positions based on viewing the problem as an *EDM*. This uses a principal component analysis. Our numerical tests showed that the new method, almost always, gave more accurate solutions. This further emphasizes that the *SNL* is best viewed as an *EDM*.

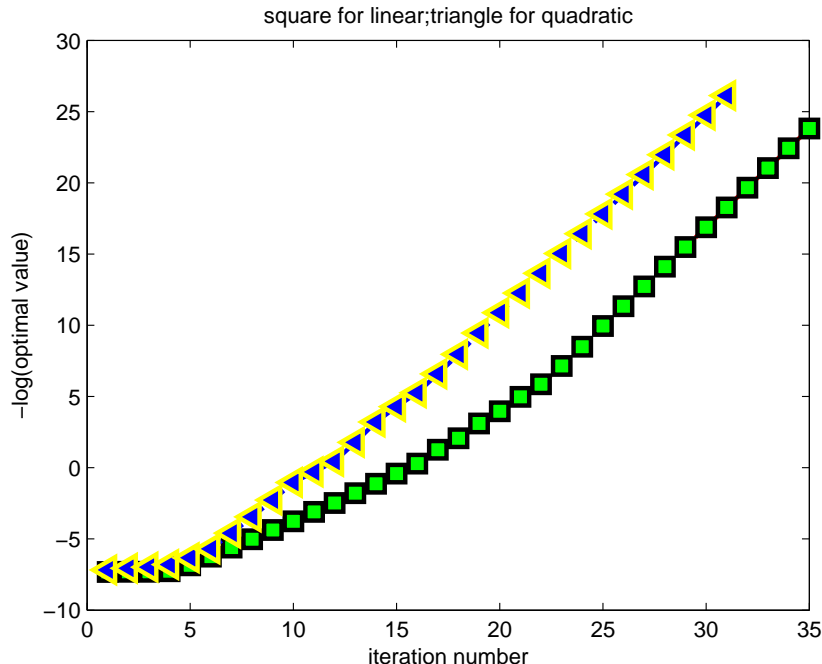


Figure 4.2: Number of iterations vs optimal value; a comparison between two barriers

3. We used the ℓ_2 norm formulation instead of the ℓ_1 norm. This is a better fit for the data that we used. However, the quadratic objective makes the problem more difficult to solve.
4. We solved the ℓ_2 norm formulation of the **SDP** relaxation with a Gauss-Newton primal-dual interior-exterior path following method. This was a robust approach compared with the traditional symmetrization and a Newton method. We compared using the quadratic constraint with the linearized version used in the literature. The numerical results showed that the quadratic constraint is more stable. This agrees with theoretical results in the literature on the deformation of the central path based on the size of the I in the linearized version.
5. The Gauss-Newton approach with a crossover technique often converged to an optimal solution of strictly lower rank which resulted in a better **SDP** approximation for the underlying **SNL**. The results in Tables 4.3 and 4.4 demonstrate how converging to optimal solutions of lower rank allow us to compute approximate solutions whose true objective function is smaller. Although the problems we solved were of a small scale, the significance of these numerical results warrants the application of our approach to large scale problems.

Future work involves making the algorithm more efficient. In particular, this requires finding appropriate preconditioners.

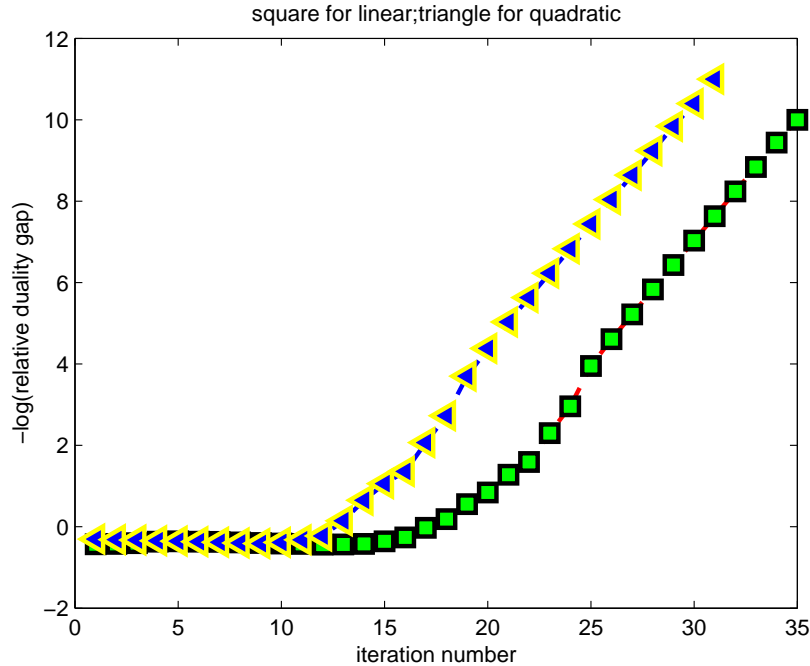


Figure 4.3: Number of iterations vs duality gap; a comparison between two barriers

References

- [1] S. AL-HOMIDAN and H. WOLKOWICZ. Approximate and exact completion problems for Euclidean distance matrices using semidefinite programming. *Linear Algebra Appl.*, 406:109–141, 2005.
- [2] A. ALFAKIH, A. KHANDANI, and H. WOLKOWICZ. Solving Euclidean distance matrix completion problems via semidefinite programming. *Comput. Optim. Appl.*, 12(1-3):13–30, 1999. Computational optimization—a tribute to Olvi Mangasarian, Part I.
- [3] M. BAKONYI and C. R. JOHNSON. The Euclidean distance matrix completion problem. *SIAM J. Matrix Anal. Appl.*, 16(2):646–654, 1995.
- [4] P. BISWAS, T.C. LIANG, K.C. TOH, T.C. WANG, and Y. YE. Semidefinite programming approaches for sensor network localization with noisy distance measurements. *IEEE Transactions on Automation Science and Engineering*, 2006. To appear.
- [5] P. BISWAS, T.C. LIANG, K.C. TOH, and Y. YE. An SDP based approach for anchor-free 3D graph realization. Technical report, Operation Research, Stanford University, Stanford, CA, 2005.
- [6] P. BISWAS and Y. YE. Semidefinite programming for ad hoc wireless sensor network localization. In *Information Processing In Sensor Networks, Proceedings of the third*

- international symposium on Information processing in sensor networks*, pages 46–54, Berkeley, Calif., 2004.
- [7] P. BISWAS and Y. YE. A distributed method for solving semidefinite programs arising from ad hoc wireless sensor network localization. In *Multiscale optimization methods and applications*, volume 82 of *Nonconvex Optim. Appl.*, pages 69–84. Springer, New York, 2006.
- [8] J.M. BORWEIN and H. WOLKOWICZ. Facial reduction for a cone-convex programming problem. *J. Austral. Math. Soc. Ser. A*, 30(3):369–380, 1980/81.
- [9] C.B. CHUA and L. TUNÇEL. Invariance and efficiency of convex representations. *Math. Program.*, 111(1-2, Ser. B):113–140, 2008.
- [10] G. M. CRIPPEN and T. F. HAVEL. *Distance geometry and molecular conformation*. Research Studies Press Ltd., Letchworth, 1988.
- [11] J. ERIKSSON and M.E. GULLIKSSON. Local results for the Gauss-Newton method on constrained rank-deficient nonlinear least squares. *Math. Comp.*, 73(248):1865–1883 (electronic), 2004.
- [12] R.W. FAREBROTHER. Three theorems with applications to Euclidean distance matrices. *Linear Algebra Appl.*, 95:11–16, 1987.
- [13] J. C. GOWER. Properties of Euclidean and non-Euclidean distance matrices. *Linear Algebra Appl.*, 67:81–97, 1985.
- [14] J.C. GOWER. Properties of Euclidean and non-Euclidean distance matrices. *Linear Algebra Appl.*, 67:81–97, 1985.
- [15] O. GÜLER and L. TUNÇEL. Characterization of the barrier parameter of homogeneous convex cones. *Math. Programming*, 81(1, Ser. A):55–76, 1998.
- [16] T.L. HAYDEN, J. WELLS, W-M. LIU, and P. TARAZAGA. The cone of distance matrices. *Linear Algebra Appl.*, 144:153–169, 1991.
- [17] H. JIN. *Scalable Sensor Localization Algorithms for Wireless Sensor Networks*. PhD thesis, Toronto University, Toronto, Ontario, Canada, 2005.
- [18] C.R. JOHNSON and P. TARAZAGA. Connections between the real positive semidefinite and distance matrix completion problems. *Linear Algebra Appl.*, 223/224:375–391, 1995. Special issue honoring Miroslav Fiedler and Vlastimil Pták.
- [19] N. KRISLOCK, V. PICCIALLI, and H. WOLKOWICZ. Robust semidefinite programming approaches for sensor network localization with anchors. Technical Report CORR 2006-12, University of Waterloo, Waterloo, Ontario, 2006. URL:orion.uwaterloo.ca/~hwolkowi/henry/reports/ABSTRACTS.html#sensorKPW.

- [20] S. KRUK, M. MURAMATSU, F. RENDL, R.J. VANDERBEI, and H. WOLKOWICZ. The Gauss-Newton direction in semidefinite programming. *Optim. Methods Softw.*, 15(1):1–28, 2001.
- [21] M. LAURENT. A tour d’horizon on positive semidefinite and Euclidean distance matrix completion problems. In *Topics in Semidefinite and Interior-Point Methods*, volume 18 of *The Fields Institute for Research in Mathematical Sciences, Communications Series*, pages 51–76, Providence, Rhode Island, 1998. American Mathematical Society.
- [22] Z-Q. LUO, N.D. SIDIROPOULOS, P. TSENG, and S. ZHANG. Approximation bounds for quadratic optimization with homogeneous quadratic constraints. *SIAM J. Optim.*, 18(1):1–28, 2007.
- [23] I.J. SCHOENBERG. Remarks to Maurice Frechet’s article: Sur la definition axiomatique d’une classe d’espaces vectoriels distances applicables vectoriellement sur l’espace de Hilbert. *Ann. Math.*, 36:724–732, 1935.
- [24] A. SO, Y. YE, and J. ZHANG. A unified theorem on SDP rank reduction. Technical report, Operation Research, Stanford University, Stanford, CA, 2006.
- [25] A.M. SO and Y. YE. Theory of semidefinite programming for sensor network localization. *Math. Program.*, 109(2):367–384, 2007.
- [26] A. SRIVASTAV and K. WOLF. Finding dense subgraphs with semidefinite programming. In *Approximation algorithms for combinatorial optimization (Aalborg, 1998)*, volume 1444 of *Lecture Notes in Comput. Sci.*, pages 181–191. Springer, Berlin, 1998.
- [27] W.S. TORGERSON. Multidimensional scaling. I. Theory and method. *Psychometrika*, 17:401–419, 1952.
- [28] P. TSENG. SOCP relaxation for nonconvex optimization. Technical Report Aug-04, University of Washington, Seattle, WA, 2004. presented at ICCOPT I, RPI, Troy, NY.
- [29] P. TSENG. Second-order cone programming relaxation of sensor network localization. *SIAM J. Optim.*, 18(1):156–185, 2007.
- [30] O.V. VERBITSKY. A note on the approximability of the dense subgraph problem. *Mat. Stud.*, 22(2):198–201, 2004.
- [31] Z. WANG, S. ZHENG, S. BOYD, and Y. YE. Further relaxations of the semidefinite programming approach to sensor network localization. *SIAM J. Optim.*, 19(2):655–673, 2008.
- [32] D. XU, J. HAN, Z. HUANG, and L. ZHANG. Improved approximation algorithms for MAX $\frac{n}{2}$ -DIRECTED-BISECTION and MAX $\frac{n}{2}$ -DENSE-SUBGRAPH. *J. Global Optim.*, 27(4):399–410, 2003.