

Computing the Nearest Euclidean Distance Matrix with Low Embedding Dimensions

Hou-Duo Qi* and Xiaoming Yuan†

September 12, 2012 and Revised May 12, 2013

Abstract

Euclidean distance embedding appears in many high-profile applications including wireless sensor network localization, where not all pairwise distances among sensors are known or accurate. The classical Multi-Dimensional Scaling (cMDS) generally works well when the partial or contaminated Euclidean Distance Matrix (EDM) is close to the true EDM, but otherwise performs poorly. A natural step preceding cMDS would be to calculate the nearest EDM to the known matrix. A crucial condition on the desired nearest EDM is for it to have a low embedding dimension and this makes the problem nonconvex. There exists a large body of publications that deal with this problem. Some try to solve the problem directly and some are the type of convex relaxations of it. In this paper, we propose a numerical method that aims to solve this problem directly. Our method is strongly motivated by the majorized penalty method of Gao and Sun for low-rank positive semi-definite matrix optimization problems. The basic geometric object in our study is the set of EDMs having a low embedding dimension. We establish a *zero* duality gap result between the problem and its Lagrangian dual problem, which also motivates the majorization approach adopted. Numerical results show that the method works well for the Euclidean embedding of Network coordinate systems and for a class of problems in large scale sensor network localization and molecular conformation.

AMS subject classifications. 49M45, 90C25, 90C33

Keywords: Euclidean distance matrix, Lagrangian duality, low-rank approximation, majorization method, semismooth Newton-CG method.

1 Introduction

Euclidean distance embedding appears in many important applications including wireless sensor network localization, where only partial (accurate or contaminated) pairwise distances between n sensors are known. A key question being asked in such context is whether the known distances contain adequate information to recover the location of the n sensors in typically a low (e.g., 2 or 3) dimensional space. The question is often modeled as a nonconvex optimization problem. There exist a large number of publications on this problem. Some try to solve the problem directly and some are the type of convex relaxations of it. The latter type in particular has attracted tremendous interests of many researchers over the past few years, largely due to it being an important application of Semi-Definite Programming (SDP). For excellent references

*School of Mathematics, The University of Southampton, Highfield, Southampton SO17 1BJ, UK. E-mail: hdqi@soton.ac.uk. This research was supported by the EPSRC grant EP/K007645/1.

†Department of Mathematics, Hong Kong Baptist University, Kowloon Tong, Hong Kong. E-mail: xmyuan@hkbu.edu.hk. This author was supported by the General Research Fund from Hong Kong Research Grants Council: 203712.

on this topic, one may refer to the recent surveys by Krislock and Wolkowicz [30] and Liberti et al. [35], and the thesis [11]. In this paper, we propose a numerical method that aims to solve this problem directly. Numerical results show that our method works well for the Euclidean embedding of Network coordinate systems and for a class of problems in large scale sensor network localization and molecular conformation. We now describe the problem, the motivation that leads to our method, and the major contributions in this paper.

(a) The problem. Suppose we have n points $\{x_1, x_2, \dots, x_n\}$ in \mathbb{R}^r . The primary information that is available for those points is the measured Euclidean distances among them

$$d_{ij} \approx \|x_i - x_j\|, \quad \text{for some pairs } (x_i, x_j). \quad (1)$$

As indicated in (1), those measurements may or may be not accurate. Our main problem is to recover the n points in \mathbb{R}^r purely based on those available distances. There is a wide literature on this problem (for a long list of papers on this problem, see [9, 5, 11, 30, 35]). We are content with just mentioning a few fairly recent publications in three categories: Distance Geometry, Sensor Network Localization (SNL) and Multi-Dimensional Scaling (MDS).

The problem has long been studied in distance geometry (see [35] and the references therein) and there are software tools available including DGSOL of Moré and Wu [41] and MD-jeep of Mucherino et al. (see [36, 33, 35]). The problem can also be posted as the SNL problem, which is often solved by the method of SDP relaxations, see the SDP method of Biswas and Ye [3], SNLSDP of Biswas et al. [4] and SFSDP of Kim et al. [28]. Deeply rooted in the classical MDS (cMDS) (see [9, 5]), the Euclidean Distance Matrix (EDM) completion approach initially studied by Alfakih et al. [1] (see also [46, 29]) uses the (convex) quadratic SDP to tackle the problem. Our approach is also heavily influenced by the classical MDS, which is outlined below.

(b) cMDS and EDM-based formulations. In cMDS, a $n \times n$ matrix D is called a EDM if there exist points p_1, \dots, p_n in \mathbb{R}^r such that $D_{ij} = \|p_i - p_j\|^2$ for $i, j = 1, \dots, n$ (note: the distance is squared). \mathbb{R}^r is often referred to as the embedding space and r is the embedding dimension when it is the smallest such r . The cMDS is based on the following well-known result. A $n \times n$ symmetric matrix D is EDM if and only if

$$\text{diag}(D) = 0, \quad J(-D)J \succeq 0 \quad \text{and} \quad J := I - ee^T/n, \quad (2)$$

where $\text{diag}(\cdot)$ is the diagonal operator that takes the diagonal elements of a matrix, $X \succeq 0$ means that X is positive semidefinite, I (or I_n when the indication of dimension is needed) is the identity matrix, and e is the vector of all ones in \mathbb{R}^n . The origin of this result can be traced back to Schoenberg [44] and an independent work [49] by Young and Householder. See also Gower [21] for a nice derivation of (2). The corresponding embedding dimension is $r = \text{rank}(JDJ)$. In this case, the n embedding points p_1, \dots, p_n in \mathbb{R}^r are given by the columns of P^T , where $P \in \mathbb{R}^{n \times r}$ satisfies

$$-JDJ/2 = PP^T. \quad (3)$$

This embedding result is the foundation of the cMDS, which deals with the case when D is not a true EDM.

Let \mathcal{S}^n denote the space of $n \times n$ symmetric matrices equipped with the standard inner product $\langle A, B \rangle = \text{trace}(AB)$ for $A, B \in \mathcal{S}^n$. Let $\|\cdot\|$ denote the induced Frobenius norm. Let \mathcal{S}_+^n denote the cone of positive semidefinite matrices in \mathcal{S}^n (often abbreviated as $X \succeq 0$ for $X \in \mathcal{S}_+^n$) and let $\mathcal{S}_+^n(r)$ denote the set of positive semidefinite matrices whose ranks are not greater than r . The cMDS actually solves the following optimization problem:

$$Y^* \in \arg \min \|Y - (-JDJ/2)\|^2, \quad \text{s.t. } Y \in \mathcal{S}_+^n(r). \quad (4)$$

The n embedding points are given by the columns of $P^T \in \mathbb{R}^{r \times n}$ obtained through $Y^* = PP^T$. The solution Y^* can be calculated through the spectral decomposition of JDJ (see e.g., [37, Thm. 1]). When D is a true EDM with embedding dimension r , it is obvious $Y^* = -JDJ/2$ and the embedding formula (3) is recovered. However, when D is not a true EDM, the cMDS is often not robust [6] as the nearest distance from D is measured through the transformation JDJ rather than on D itself.

A more natural “nearness” measurement is the following:

$$Y^* \in \arg \min \|Y - D\|^2/2, \quad \text{s.t. } Y \in \mathcal{E}^n(r), \quad (5)$$

where $\mathcal{E}^n(r)$ is the set of EDMs with embedding dimensions not greater than r , and Y^* is the nearest EDM to D from this set. A more general model is

$$Y^* \in \arg \min \|H \circ (Y - D)\|^2/2, \quad \text{s.t. } Y \in \mathcal{E}^n(r), \quad (6)$$

where $H \in \mathcal{S}^n$ is the weight matrix ($H_{ij} \geq 0$) and “ \circ ” is the Hadamard product between matrices. If $H = E$ (the matrix of all ones), (6) reduces to (5). The n embedding points from (5) or (6) are given by the columns of $P^T \in \mathbb{R}^{r \times n}$ satisfying $-JY^*J/2 = PP^T$. The cost of having such natural models is that both problems have no closed-form solutions and have to rely on iterative algorithms for their optimal solutions. This is challenging as both problems are nonconvex.

(c) Methods of convex relaxations. Therefore, convex relaxation methods form an important approach to those problems. We use (5) below to demonstrate this approach. By the Schoenberg characterization (2), we have

$$Y \in \mathcal{E}^n(r) \iff Y \in \mathcal{S}_h^n, \quad -JYJ \succeq 0, \quad \text{and} \quad \text{rank}(JYJ) \leq r,$$

where \mathcal{S}_h^n is the *hollow* subspace in \mathcal{S}^n defined by $\mathcal{S}_h^n := \{A \in \mathcal{S}^n : \text{diag}(A) = 0\}$. The nonconvexity of (5) is caused by the rank constraint. Ignoring the rank constraint leads to the convex relaxation of (5):

$$\min \|Y - D\|^2/2, \quad \text{s.t. } Y \in \mathcal{S}_h^n \quad \text{and} \quad -JYJ \succeq 0. \quad (7)$$

The feasible region of (7) is actually the convex cone of all $n \times n$ EDMs, denoted by \mathcal{E}^n . The convex relaxation problem (7) has a serious drawback: Its (generalized) Slater condition does not hold because JYJ has *zero* as its eigenvalue (e.g., $(JYJ)e = 0$). Two important convex reformulations emerged to rectify this drawback.

One is based on the fact that there exists a one-to-one linear transformation $\mathcal{L} : \mathcal{S}_+^{n-1} \mapsto \mathcal{E}^n$ (see [1, Eq.(8)]). Therefore, (7) is equivalent to

$$\min \|\mathcal{L}(X) - D\|^2/2 \quad \text{s.t. } X \in \mathcal{S}_+^{n-1}. \quad (8)$$

This reformulation establishes an important link to SDP and was first studied by Alfakih et al. [1]. It was further studied as a prototype of convex quadratic SDPs by Toh [46], where a regularization term is added to encourage a low-rank solution of (8). The other reformulation is based on the fact that

$$-JYJ \succeq 0 \iff -Y \in \mathcal{K}_+^n := \left\{ A \in \mathcal{S}^n : x^T A x \geq 0, \quad x \in e^\perp \right\},$$

where $e^\perp := \{x \in \mathbb{R}^n : e^T x = 0\}$. Therefore, (7) is equivalent to

$$\min \|Y - D\|^2/2, \quad \text{s.t. } Y \in \mathcal{S}_h^n \quad \text{and} \quad -Y \in \mathcal{K}_+^n. \quad (9)$$

A nice feature of this reformulation is that it can be treated as a projection problem onto the intersection of the subspace \mathcal{S}_h^n and the closed convex cone $(-\mathcal{K}_+^n)$. Hence, the Method of Alternating Projections (MAP) of Dykstra-Han type [13, 22] is a choice. In fact, Glunt et al. [17] and Gaffke and Mathar [14] independently studied such MAPs, yet based on different projection formulae onto $(-\mathcal{K}_+^n)$. A drawback of MAPs is their slow convergence. Recently, Qi [43] proposed a semismooth Newton-CG method, which enjoys the quadratic convergence rate. Another progress is [26] by Jiang et al. on (convex) semidefinite least squares problems, which include (8) as a special case. We note that for both reformulations (8) and (9), the Slater condition holds.

(d) Our approach and main contributions. Little success has been achieved towards the solution of (6). Earlier attempts include a (complicated) two-phase method and its variants in [18, 19], where no rigorous convergence analysis was given. However, we begin to see important progresses are being made. For example, Mishra et al. [40] proposed a Manifold-Based-Optimization (MBO for short) method for (8) with \mathcal{L} (see [40, Problem (2)]) being a linear mapping from \mathcal{S}_+^n to \mathcal{E}^n , which is further restricted on a rank-constrained manifold. The MBO works extremely well for a class of problems, but has its limitations. We will report our numerical experience of it in Sect. 6.

In this paper, we try to fill the gap by continuing our effort initiated in [43] to tackle the problem (5), which in terms of the terminologies in (9), is equivalent to

$$\min \|Y - D\|^2/2, \quad \text{s.t. } Y \in \mathcal{S}_h^n, \quad -Y \in \mathcal{K}_+^n, \quad \text{and} \quad \text{rank}(JYJ) \leq r. \quad (10)$$

Casting (5) as (10) is more revealing of the problem structure than its original formulation. To demonstrate this point, we compare it with the low-rank positive semidefinite least squares problem recently studied by Gao [15] and Gao and Sun [16]:

$$\min \|X - C\|^2/2, \quad \text{s.t. } X \in \Omega \quad \text{and} \quad X \in \mathcal{S}_+^n(r), \quad (11)$$

where $C \in \mathcal{S}^n$ is given, Ω is a closed convex set in \mathcal{S}^n defined by linear equalities and/or inequalities, and the objective function can include the H -weights. A majorized penalty approach proposed in [15, 16] proves to be extremely efficient for benchmarks of the nearest low-rank correlation matrix problems. The success of the majorization approach is largely due to the nice properties of the projection operator to the nonconvex set $\mathcal{S}_+^n(r)$. We would like to ask what the corresponding set of $\mathcal{S}_+^n(r)$ is in (5). It turns out that it is the set

$$\mathcal{K}_+^n(r) := \{Y \in \mathcal{S}^n : Y \in \mathcal{K}_+^n \quad \text{and} \quad \text{rank}(JYJ) \leq r\}$$

(not $\mathcal{E}^n(r)$ in (5)!) that plays a similar role in (10) as $\mathcal{S}_+^n(r)$ does in (11).

Therefore, our research starts from investigating the properties of $\mathcal{K}_+^n(r)$, followed by a detailed study of the Lagrangian dual problem of (10) (see (32))

$$\min_{y \in \mathbb{R}^n} \theta(y) := \|\Pi_{\mathcal{K}_+^n(r)}(-D + \text{Diag}(y))\|^2/2 - \|D\|^2/2, \quad (12)$$

where $\Pi_{\mathcal{K}_+^n(r)}(X)$ is the orthogonal projection onto $\mathcal{K}_+^n(r)$ and $\text{Diag}(y)$ is the diagonal matrix with y being its diagonal. Although the projection may be not unique due to the nonconvexity of $\mathcal{K}_+^n(r)$, the function $\theta(y)$ is well defined and is irrelevant to the choice of the projection. Those results serve the basis for extending the majorized penalty method of Gao and Sun [16] to the problem (6). The extension is built in such a way that the Newton method developed in [43] is well suited to solving the subproblems encountered. The convergence properties of the original approach of Gao and Sun are inherited and delicate issues such as the choice of penalty (concave) functions are dealt with so as to improve the numerical performance of the method. We conducted extensive numerical tests on problems (5) and (6) arising from three important fields:

Euclidean embedding of Network coordinate systems, Sensor network localization with/without anchors, and Molecular conformations. Our numerical results show that the proposed method is very promising.

(e) Organization of the paper. Sect. 2 includes some basic facts related to \mathcal{K}_+^n and $\mathcal{S}_+^n(r)$. We investigate the properties of $\Pi_{\mathcal{K}_+^n(r)}(\cdot)$ in Sect. 3. In Sect. 4, we study the Lagrangian dual problem (12). We characterize the subdifferential $\partial\theta(y)$ solely based on the information of the first r largest eigenvalues of $(J(-D + \text{Diag}(y))J)$ and the corresponding eigenvectors (see Eq. (33) and Prop. 3.3). The boundedness of the level set of $\theta(\cdot)$ is proved in (Prop. 4.1). We are also able to characterize when there is no duality gap (Thm. 4.2). This result clarifies the situations when (5) can be solved to its global optimality. Sect. 5 is devoted to the H weighted problem (6), where the majorized penalty approach of Gao and Sun [16] is extended. Using the zero duality gap result (Thm. 4.2), Prop. 5.4 establishes the rationale of the majorized penalty approach. We conduct extensive numerical tests in Sect. 6, and finally make some conclusions in Sect. 7.

2 Preliminaries

In this section, we prepare some facts for our nonsmooth analysis of the metric projection onto $\mathcal{K}_+^n(r)$. It turns out that the metric projection onto the convex set \mathcal{K}_+^n plays a very important role.

Given a closed set $\mathcal{C} \subset \mathcal{S}^n$ and $X \in \mathcal{S}^n$, let $\Pi_{\mathcal{C}}(X)$ denote an optimal solution of the following metric projection problem:

$$\min \|Z - X\|^2 \quad \text{s.t. } Z \in \mathcal{C}.$$

When \mathcal{C} is convex, $\Pi_{\mathcal{C}}(X)$ is unique. When \mathcal{C} is not convex, there may be multiple solutions and we let $\Pi_{\mathcal{C}}^B(X)$ denote the set of all projections of X onto \mathcal{C} .

Let $Q \in \mathcal{S}^n$ be the Householder matrix defined by

$$Q := I - \frac{2}{v^T v} v v^T \quad \text{with } v := [1, \dots, 1, \sqrt{n+1}]^T \in \mathbb{R}^n.$$

We note that Q is symmetric and orthogonal (i.e., $Q^2 = I$). For $X \in \mathcal{S}^n$, denote

$$QXQ =: \begin{bmatrix} \widehat{X}_1 & \widehat{x} \\ \widehat{x}^T & \widehat{x}_0 \end{bmatrix} \quad \text{with } \widehat{X}_1 \in \mathcal{S}^{n-1}. \quad (13)$$

According to [23, Thm. 2.1] (see also [17, Eq. (3.16)]), we have the Hayden-Wells formula

$$\Pi_{\mathcal{K}_+^n}(X) = Q \begin{bmatrix} \Pi_{\mathcal{S}_+^{n-1}}(\widehat{X}_1) & \widehat{x} \\ \widehat{x}^T & \widehat{x}_0 \end{bmatrix} Q. \quad (14)$$

Recall that $\mathcal{S}_+^n(r)$ is the set of all positive semidefinite matrices whose ranks are not greater than r . The following are basic identities that we are going to use frequently.

Lemma 2.1 *For X having the decomposition (13), let*

$$X_0 := \begin{bmatrix} 0 & \widehat{x} \\ \widehat{x}^T & \widehat{x}_0 \end{bmatrix}.$$

The following hold.

(i) $J\Pi_{\mathcal{K}_+^n}(X)J = \Pi_{\mathcal{S}_+^n}(JXJ).$

$$(ii) \quad \Pi_{\mathcal{K}_+^n}(X) = \Pi_{\mathcal{S}_+^n}(JXJ) + QX_0Q = \Pi_{\mathcal{S}_+^n}(JXJ) + (X - JXJ).$$

$$(iii) \quad X = JXJ + QX_0Q \quad \text{and} \quad \langle JXJ, X - JXJ \rangle = 0.$$

$$(iv) \quad X \in \mathcal{K}_+^n \text{ if and only if } \widehat{X}_1 \succeq 0. \text{ Moreover, } X \in \mathcal{K}_+^n(r) \text{ if and only if } \widehat{X}_1 \in \mathcal{S}_+^{n-1}(r).$$

Proof. The matrix J defined in (2) has a nice relationship with Q :

$$J = Q \begin{bmatrix} I_{n-1} & 0 \\ 0 & 0 \end{bmatrix} Q. \quad (15)$$

(i) It follows from the Hayden-Wells formula (14) that

$$\begin{aligned} J\Pi_{\mathcal{K}_+^n}(X)J &= Q \begin{bmatrix} I_{n-1} & 0 \\ 0 & 0 \end{bmatrix} Q^2 \begin{bmatrix} \Pi_{\mathcal{S}_+^{n-1}}(\widehat{X}_1) & \hat{x} \\ \hat{x}^T & \hat{x}_0 \end{bmatrix} Q^2 \begin{bmatrix} I_{n-1} & 0 \\ 0 & 0 \end{bmatrix} Q \\ &= Q \begin{bmatrix} \Pi_{\mathcal{S}_+^{n-1}}(\widehat{X}_1) & 0 \\ 0 & 0 \end{bmatrix} Q. \end{aligned}$$

We also note that

$$JXJ = Q \begin{bmatrix} I_{n-1} & 0 \\ 0 & 0 \end{bmatrix} QXQ \begin{bmatrix} I_{n-1} & 0 \\ 0 & 0 \end{bmatrix} Q = Q \begin{bmatrix} \widehat{X}_1 & 0 \\ 0 & 0 \end{bmatrix} Q. \quad (16)$$

The fact of Q being symmetric and orthogonal proves (i).

(ii) This is a consequence of (i). By the Hayden-Wells formula (14), we have

$$\begin{aligned} \Pi_{\mathcal{K}_+^n}(X) &= Q \begin{bmatrix} \Pi_{\mathcal{S}_+^{n-1}}(\widehat{X}_1) & 0 \\ 0 & 0 \end{bmatrix} Q + Q \begin{bmatrix} 0 & \hat{x} \\ \hat{x}^T & \hat{x}_0 \end{bmatrix} Q \\ &= J\Pi_{\mathcal{K}_+^n}(X)J + Q \begin{bmatrix} 0 & \hat{x} \\ \hat{x}^T & \hat{x}_0 \end{bmatrix} Q \\ &= \Pi_{\mathcal{S}_+^n}(JXJ) + Q \begin{bmatrix} 0 & \hat{x} \\ \hat{x}^T & \hat{x}_0 \end{bmatrix} Q. \end{aligned}$$

(iii) It follows from (13) that

$$X = Q \begin{bmatrix} \widehat{X}_1 & \hat{x} \\ \hat{x}^T & \hat{x}_0 \end{bmatrix} Q = Q \begin{bmatrix} \widehat{X}_1 & 0 \\ 0 & 0 \end{bmatrix} Q + Q \begin{bmatrix} 0 & \hat{x} \\ \hat{x}^T & \hat{x}_0 \end{bmatrix} Q = JXJ + QX_0Q.$$

Moreover, we have

$$\langle JXJ, X - JXJ \rangle = \langle JXJ, QX_0Q \rangle = \langle QJXJQ, X_0 \rangle = 0.$$

The last equality uses the structure of X_0 and the fact (by (16))

$$QJXJQ = \begin{bmatrix} \widehat{X}_1 & 0 \\ 0 & 0 \end{bmatrix}.$$

(iv) Under the condition of (iv), it follows from (13) and the Hayden-Wells formula (14) that

$$X \in \mathcal{K}_+^n \iff \Pi_{\mathcal{K}_+^n}(X) = X \iff Q\Pi_{\mathcal{K}_+^n}(X)Q = QXQ \iff \widehat{X}_1 = \Pi_{\mathcal{S}_+^{n-1}}(\widehat{X}_1) \iff \widehat{X}_1 \succeq 0.$$

This proves the first part. It follows from (16) that $\text{rank}(JXJ) = \text{rank}(\widehat{X}_1)$. Hence, $X \in \mathcal{K}_+^n(r)$ is equivalent to $\widehat{X}_1 \in \mathcal{S}_+^{n-1}(r)$. \square

The result in (iii) can be regarded as an orthogonal decomposition of matrix X . Therefore, we have

$$\|X\|^2 = \|JXJ\|^2 + \|X - JXJ\|^2. \quad (17)$$

It is clear from Lemma 2.1 that the projection $\Pi_{\mathcal{K}_+^n}$ is related to the projection $\Pi_{\mathcal{S}_+^n}$. Such a relationship can be extended to the metric projections $\Pi_{\mathcal{K}_+^n(r)}$ and $\Pi_{\mathcal{S}_+^n(r)}$. To establish this relationship, we need the following results from [15, Sect. 2.4.1].

Let X have the spectral decomposition

$$X = P\Lambda(X)P^T,$$

where $\Lambda(X) := \text{Diag}(\lambda(X))$, $\lambda_1(X) \geq \dots \geq \lambda_n(X)$ are the eigenvalues of X being arranged in the non-increasing order and $P \in \mathcal{O}^n$ with \mathcal{O}^n being the set of all $n \times n$ orthogonal real matrices. Define

$$\alpha := \{i \mid \lambda_i(X) > \lambda_r(X)\} \quad \text{and} \quad \beta := \{i \mid \lambda_i(X) = \lambda_r(X)\}.$$

For an index set $\mathcal{I} \subseteq \{1, \dots, n\}$, $X_{\mathcal{I}}$ denotes the submatrix consisting of the columns in X indexed by \mathcal{I} .

Lemma 2.2 [15, Lemma 2.4] *The set $\Pi_{\mathcal{S}_+^n(r)}^B(X)$ can be characterized as follows.*

$$\Pi_{\mathcal{S}_+^n(r)}^B(X) = \left\{ [P_\alpha, P_\beta V_\beta] \text{Diag}(v) [P_\alpha, P_\beta V_\beta]^T \mid V_\beta \in \mathcal{O}^{|\beta|} \right\}$$

where $v := ((\lambda_1(X))_+, \dots, (\lambda_r(X))_+, 0, \dots, 0)^T \in \mathbb{R}^{|\alpha|+|\beta|}$.

We let $\Pi_{\mathcal{S}_+^n(r)}(X)$ denote a particular solution in $\Pi_{\mathcal{S}_+^n(r)}^B(X)$. The distance $\|X - \Pi_{\mathcal{S}_+^n(r)}(X)\|$ is irrelevant to the choice of the solution. Define

$$\Xi_r(X) := \frac{1}{2} \left(\|X\|^2 - \|X - \Pi_{\mathcal{S}_+^n(r)}(X)\|^2 \right).$$

Then $\Xi_r(\cdot)$ is a convex function of X (see [24, Chp. IV: Example 2.1.4]). We let $\partial\Xi_r(X)$ denote the subdifferential of Ξ_r at X .

Lemma 2.3 [15, Prop. 2.5] *$\partial\Xi_r(X)$ is the convex hull of $\Pi_{\mathcal{S}_+^n(r)}^B(X)$, i.e.,*

$$\partial\Xi_r(X) = \text{conv} \Pi_{\mathcal{S}_+^n(r)}^B(X).$$

An important property of $\Pi_{\mathcal{S}_+^n(r)}(X)$ is that it provides an orthogonal decomposition of X in the sense of

$$X = \Pi_{\mathcal{S}_+^n(r)}(X) + (X - \Pi_{\mathcal{S}_+^n(r)}(X)) \quad \text{and} \quad \langle \Pi_{\mathcal{S}_+^n(r)}(X), X - \Pi_{\mathcal{S}_+^n(r)}(X) \rangle = 0. \quad (18)$$

This can be verified directly by Lemma 2.2. Consequently we have

$$\|X\|^2 = \|\Pi_{\mathcal{S}_+^n(r)}(X)\|^2 + \|X - \Pi_{\mathcal{S}_+^n(r)}(X)\|^2,$$

and hence $\Xi_r(X) = \|\Pi_{\mathcal{S}_+^n(r)}(X)\|^2/2$.

3 Properties of Metric Projection $\Pi_{\mathcal{K}_+^n(r)}$

It has been widely known that the set of positive semidefinite matrices and the set of Euclidean distance matrices tend to share similar properties (see, e.g., [27, 31, 32]). In this section, we extend such similarities to the nonconvex sets $\mathcal{S}_+^n(r)$ and $\mathcal{K}_+^n(r)$. Much has been known about the projections onto $\mathcal{S}_+^n(r)$, see [48, 15]. This makes our extension much easier. We continue to use the matrix partition of QXQ as in (13). Our first result is about $\Pi_{\mathcal{S}_+^n(r)}^B(JXJ)$.

Lemma 3.1

$$\Pi_{\mathcal{S}_+^n(r)}^B(JXJ) = Q \begin{bmatrix} \Pi_{\mathcal{S}_+^{n-1}(r)}^B(\widehat{X}_1) & 0 \\ 0 & 0 \end{bmatrix} Q \quad \text{and} \quad J\Pi_{\mathcal{S}_+^n(r)}^B(JXJ)J = \Pi_{\mathcal{S}_+^n(r)}^B(JXJ).$$

Proof. We note that $\widehat{X}_1 \in \mathcal{S}^{n-1}$. Let \widehat{X}_1 have the following spectral decomposition:

$$\widehat{X}_1 = U\widehat{\Lambda}_1U^T,$$

with $\widehat{\Lambda}_1 := \text{Diag}(\widehat{\lambda}_1, \dots, \widehat{\lambda}_{n-1})$, $\widehat{\lambda}_1 \geq \dots \geq \widehat{\lambda}_{n-1}$ being eigenvectors of \widehat{X}_1 and $U \in \mathcal{O}^{n-1}$ being a corresponding orthogonal matrix of orthonormal eigenvectors of \widehat{X}_1 . Let

$$\widehat{\alpha} := \{i \mid \widehat{\lambda}_i > \widehat{\lambda}_r\} \quad \text{and} \quad \widehat{\beta} := \{i \mid \widehat{\lambda}_i = \widehat{\lambda}_r\}.$$

Let

$$\overline{U} := \begin{bmatrix} U & 0 \\ 0 & 1 \end{bmatrix}.$$

From (16), we have

$$JXJ = Q\overline{U} \begin{bmatrix} \widehat{\Lambda}_1 & \\ & 0 \end{bmatrix} (Q\overline{U})^T$$

and $P := Q\overline{U}$ is orthogonal. Define $\widehat{v} := ((\widehat{\lambda}_1)_+, \dots, (\widehat{\lambda}_r)_+, 0, \dots, 0)^T \in \mathbb{R}^{|\widehat{\alpha}|+|\widehat{\beta}|}$. It follows from Lemma 2.2 that

$$\begin{aligned} \Pi_{\mathcal{S}_+^n(r)}^B(JXJ) &= \left\{ [P_{\widehat{\alpha}}, P_{\widehat{\beta}}V_{\widehat{\beta}}] \text{Diag}(\widehat{v}) [P_{\widehat{\alpha}}, P_{\widehat{\beta}}V_{\widehat{\beta}}]^T \mid V_{\widehat{\beta}} \in \mathcal{O}^{|\widehat{\beta}|} \right\} \\ &= \left\{ Q \begin{bmatrix} [U_{\widehat{\alpha}}, U_{\widehat{\beta}}V_{\widehat{\beta}}] \text{Diag}(\widehat{v}) [U_{\widehat{\alpha}}, U_{\widehat{\beta}}V_{\widehat{\beta}}]^T & 0 \\ & 0 \end{bmatrix} Q \mid V_{\widehat{\beta}} \in \mathcal{O}^{|\widehat{\beta}|} \right\} \\ &= Q \begin{bmatrix} \Pi_{\mathcal{S}_+^{n-1}(r)}^B(\widehat{X}_1) & 0 \\ 0 & 0 \end{bmatrix} Q. \end{aligned} \tag{19}$$

The second part of the lemma comes from the first part, the relationships (15), (19), and the fact $Q^2 = I$. \square

Define

$$Z^* := \Pi_{\mathcal{S}_+^n(r)}(\widehat{X}_1) + Q \begin{bmatrix} 0 & \widehat{x} \\ \widehat{x}^T & \widehat{x}_0 \end{bmatrix} Q.$$

Then Lemma 3.1 implies that there exists an element $\Pi_{\mathcal{S}_+^{n-1}(r)}(\widehat{X}_1)$ in $\Pi_{\mathcal{S}_+^{n-1}(r)}^B(\widehat{X}_1)$ such that

$$\begin{aligned} Z^* &= Q \begin{bmatrix} \Pi_{\mathcal{S}_+^{n-1}(r)}(\widehat{X}_1) & 0 \\ 0 & 0 \end{bmatrix} Q + Q \begin{bmatrix} 0 & \widehat{x} \\ \widehat{x}^T & \widehat{x}_0 \end{bmatrix} Q \\ &= Q \begin{bmatrix} \Pi_{\mathcal{S}_+^{n-1}(r)}(\widehat{X}_1) & \widehat{x} \\ \widehat{x}^T & \widehat{x}_0 \end{bmatrix} Q \\ &= \Pi_{\mathcal{K}_+^n}(Z^*) \quad (\text{by Hayden-Wells formula (14)}) \end{aligned}$$

Hence, $Z^* \in \mathcal{K}_+^n$. Moreover, we have

$$\text{rank}(JZ^*J) = \text{rank}(\Pi_{\mathcal{S}_+^{n-1}(r)}(\widehat{X}_1)) \leq r.$$

Therefore, $Z^* \in \mathcal{K}_+^n(r)$. In fact, we have the following result.

Lemma 3.2

$$Z^* \in \Pi_{\mathcal{K}_+^n(r)}^B(X).$$

Proof. Let Z be an arbitrary matrix in $\mathcal{K}_+^n(r)$. Denote

$$\begin{bmatrix} \widehat{Z}_1 & \widehat{z} \\ \widehat{z}^T & \widehat{z}_0 \end{bmatrix} := QZQ.$$

Then Lemma 2.1(iv) implies $\widehat{Z}_1 \in \mathcal{S}_+^{n-1}(r)$. We have the following chain of inequalities.

$$\begin{aligned} \|Z - X\|^2 &= \|QZQ - QXQ\|^2 \\ &= \left\| \begin{bmatrix} \widehat{Z}_1 & \widehat{z} \\ \widehat{z}^T & \widehat{z}_0 \end{bmatrix} - \begin{bmatrix} \widehat{X}_1 & \widehat{x} \\ \widehat{x}^T & \widehat{x}_0 \end{bmatrix} \right\|^2 \\ &= \|\widehat{Z}_1 - \widehat{X}_1\|^2 + \left\| \begin{bmatrix} 0 & \widehat{z} \\ \widehat{z}^T & \widehat{z}_0 \end{bmatrix} - \begin{bmatrix} 0 & \widehat{x} \\ \widehat{x}^T & \widehat{x}_0 \end{bmatrix} \right\|^2 \\ &\geq \min_{Z_1 \in \mathcal{S}_+^{n-1}(r)} \|Z_1 - \widehat{X}_1\|^2 + \min_{[z^T \ z_0]^T \in \mathbb{R}^n} \left\| \begin{bmatrix} 0 & z \\ z^T & z_0 \end{bmatrix} - \begin{bmatrix} 0 & \widehat{x} \\ \widehat{x}^T & \widehat{x}_0 \end{bmatrix} \right\|^2 \end{aligned} \quad (20)$$

$$= \|\Pi_{\mathcal{S}_+^{n-1}(r)}(\widehat{X}_1) - \widehat{X}_1\|^2. \quad (21)$$

Apparently, the lower bound of (21) is reached when $Z = Z^*$. This proves our result. \square

The following result gives a complete characterization of $\Pi_{\mathcal{K}_+^n(r)}^B(X)$.

Proposition 3.3 *It holds*

$$\Pi_{\mathcal{K}_+^n(r)}^B(X) = \Pi_{\mathcal{S}_+^n(r)}^B(JXJ) + (X - JXJ). \quad (22)$$

Moreover, for any pair $(\Pi_{\mathcal{K}_+^n(r)}(X), \Pi_{\mathcal{S}_+^n(r)}(JXJ))$ satisfying

$$\Pi_{\mathcal{K}_+^n(r)}(X) = \Pi_{\mathcal{S}_+^n(r)}(JXJ) + (X - JXJ), \quad (23)$$

we have

$$\|\Pi_{\mathcal{K}_+^n(r)}(X)\|^2 = \|\Pi_{\mathcal{S}_+^n(r)}(JXJ)\|^2 + \|X - JXJ\|^2, \quad (24)$$

$$J\Pi_{\mathcal{K}_+^n(r)}(X)J = \Pi_{\mathcal{S}_+^n(r)}(JXJ), \quad (25)$$

and

$$\langle \Pi_{\mathcal{K}_+^n(r)}(X), X - \Pi_{\mathcal{K}_+^n(r)}(X) \rangle = 0. \quad (26)$$

Proof. We first note that from Lemma 2.1 (iii) that

$$Q \begin{bmatrix} 0 & \widehat{x} \\ \widehat{x}^T & \widehat{x}_0 \end{bmatrix} Q = X - JXJ, \quad \forall X \in \mathcal{S}^n. \quad (27)$$

We then note that $\Pi_{\mathcal{S}_+^n(r)}(JXJ)$ in the definition of Z^* above is arbitrarily chosen from $\Pi_{\mathcal{S}_+^n(r)}^B(JXJ)$. Lemma 3.2 implies that the right-hand side of (22) is included in the left-hand side of (22). Now we prove the reverse direction.

Let $Z \in \Pi_{\mathcal{K}_+^n(r)}^B(X)$. It follows from the last statement in the proof of Lemma 3.2 that the lower bound in (21) is achieved at Z . As a consequence, we must have

$$\widehat{Z}_1 \in \Pi_{\mathcal{S}_+^{n-1}(r)}^B(\widehat{X}_1) \quad \text{and} \quad \widehat{z} = \widehat{x}, \quad \widehat{z}_0 = \widehat{x}_0,$$

which implies the existence of $\Pi_{\mathcal{S}_+^{n-1}(r)}(\widehat{X}_1) \in \Pi_{\mathcal{S}_+^{n-1}(r)}^B(\widehat{X}_1)$ such that $\widehat{Z}_1 = \Pi_{\mathcal{S}_+^{n-1}(r)}(\widehat{X}_1)$. It follows that

$$QZQ = \begin{bmatrix} \widehat{Z}_1 & \widehat{z} \\ \widehat{z}^T & \widehat{z}_0 \end{bmatrix} = \begin{bmatrix} \Pi_{\mathcal{S}_+^{n-1}(r)}(\widehat{X}_1) & \widehat{x} \\ \widehat{x}^T & \widehat{x}_0 \end{bmatrix}$$

and

$$\begin{aligned} Z &= Q \begin{bmatrix} \Pi_{\mathcal{S}_+^{n-1}(r)}(\widehat{X}_1) & \widehat{x} \\ \widehat{x}^T & \widehat{x}_0 \end{bmatrix} Q = Q \begin{bmatrix} \Pi_{\mathcal{S}_+^{n-1}(r)}(\widehat{X}_1) & 0 \\ 0 & 0 \end{bmatrix} Q + Q \begin{bmatrix} 0 & \widehat{x} \\ \widehat{x}^T & \widehat{x}_0 \end{bmatrix} Q \\ &= \Pi_{\mathcal{S}_+^n(r)}^B(JXJ) + (X - JXJ). \quad (\text{by Lemma 3.1}) \end{aligned}$$

This proves (22).

For any pair $(\Pi_{\mathcal{K}_+^n(r)}(X), \Pi_{\mathcal{S}_+^n(r)}(JXJ))$ satisfying (23), we consider the inner product of $\Pi_{\mathcal{S}_+^n(r)}(JXJ)$ and $(X - JXJ)$.

$$\begin{aligned} &\langle X - JXJ, \Pi_{\mathcal{S}_+^n(r)}(JXJ) \rangle \\ &= \langle X_0, Q \Pi_{\mathcal{S}_+^n(r)}(JXJ) Q \rangle = \text{trace} \left(X_0 \begin{bmatrix} \Pi_{\mathcal{S}_+^{n-1}(r)}^B(\widehat{X}_1) & 0 \\ 0 & 0 \end{bmatrix} \right) \quad (\text{using (19)}) \\ &= 0, \quad (\text{using the structure of } X_0 \text{ in Lemma 2.1}) \end{aligned} \quad (28)$$

which implies (24).

By noticing $JQX_0QJ = 0$, we have from Lemma 3.1 and (23) that

$$J\Pi_{\mathcal{K}_+^n(r)}(X)J = J\Pi_{\mathcal{S}_+^n(r)}(JXJ)J + JQX_0QJ = \Pi_{\mathcal{S}_+^n(r)}(JXJ),$$

which proves (25). Finally, we have

$$\begin{aligned} &\langle \Pi_{\mathcal{K}_+^n(r)}(X), X - \Pi_{\mathcal{K}_+^n(r)}(X) \rangle \\ &= \langle \Pi_{\mathcal{S}_+^n(r)}(JXJ) + X - JXJ, -\Pi_{\mathcal{S}_+^n(r)}(JXJ) + JXJ \rangle \quad (\text{by (22)}) \\ &= -\langle X - JXJ, \Pi_{\mathcal{S}_+^n(r)}(JXJ) \rangle \quad (\text{by (18) and Lemma 2.1(iii)}) \\ &= 0. \quad (\text{by (28)}) \end{aligned}$$

□

Define the function $\Xi_r^e : \mathcal{S}^n \mapsto \mathbb{R}$ by

$$\begin{aligned} \Xi_r^e(Z) &:= \frac{1}{2} \|Z\|^2 - \frac{1}{2} \|\Pi_{\mathcal{K}_+^n(r)}(Z) - Z\|^2 \quad \forall Z \in \mathcal{S}^n \\ &= \frac{1}{2} \|\Pi_{\mathcal{K}_+^n(r)}(Z)\|^2. \quad (\text{by (26)}) \end{aligned} \quad (29)$$

We note that $\Xi_r^e(Z)$ is irrelevant to the choice of $\Pi_{\mathcal{K}_+^n(r)}(Z)$. It follows from [24, Chp. IV: Example 2.1.4] that Ξ_r^e is convex. Let $\partial\Xi_r^e(X)$ denote the subdifferential of Ξ_r^e at X . We have

Proposition 3.4

$$\partial\Xi_r^e(X) = \text{conv } \Pi_{\mathcal{K}_+^n(r)}^B(X).$$

Proof. It follows from Prop. 3.3 that there exists an element $\Pi_{\mathcal{S}_+^n(r)}(JXJ)$ such that

$$\Pi_{\mathcal{K}_+^n(r)}(X) = \Pi_{\mathcal{S}_+^n(r)}(JXJ) + QX_0Q,$$

where X_0 is defined as in Lemma 2.1, which also proved

$$X = JXJ + QX_0Q.$$

We hence have

$$\|\Pi_{\mathcal{K}_+^n(r)}(X) - X\|^2 = \|\Pi_{\mathcal{S}_+^n(r)}(JXJ) - JXJ\|^2, \quad (30)$$

and

$$\begin{aligned} \Xi_r^e(X) &= \frac{1}{2}\|X\|^2 - \frac{1}{2}\|\Pi_{\mathcal{K}_+^n(r)}(X) - X\|^2 && \text{(by definition of } \Xi_r^e\text{)} \\ &= \frac{1}{2}\|JXJ\|^2 + \frac{1}{2}\|X - JXJ\|^2 - \frac{1}{2}\|\Pi_{\mathcal{K}_+^n(r)}(X) - X\|^2 && \text{(by (17))} \\ &= \frac{1}{2}\|JXJ\|^2 - \frac{1}{2}\|\Pi_{\mathcal{S}_+^n(r)}(JXJ) - JXJ\|^2 + \frac{1}{2}\|X - JXJ\|^2 && \text{(by (30))} \\ &= \Xi_r(JXJ) + \frac{1}{2}\|X - JXJ\|^2. && \text{(by definition of } \Xi_r\text{)} \end{aligned}$$

Since both Ξ_r^e and Ξ_r are convex, we have (with $Y := JXJ$) that

$$\begin{aligned} \partial\Xi_r^e(X) &= J\partial\Xi_r(Y)J + X - JXJ && \text{(by the chain rule of subdifferentials)} \\ &= J(\text{conv } \Pi_{\mathcal{S}_+^n(r)}^B(JXJ))J + X - JXJ && \text{(by Lemma 2.3)} \\ &= \text{conv } \Pi_{\mathcal{S}_+^n(r)}^B(JXJ) + X - JXJ && \text{(by Lemma 3.1)} \\ &= \text{conv } \Pi_{\mathcal{K}_+^n(r)}^B(X). && \text{(by Prop. 3.3)} \end{aligned}$$

This proves our result. \square

This result also implies a necessary and sufficient condition for Ξ_r^e being differentiable.

Corollary 3.5 *Let $\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_n$ be the eigenvalues of JXJ arranged in nonincreasing order. Then, $\Xi_r^e(\cdot)$ is differentiable at X if and only if $\lambda_{r+1} \leq 0$ or in the case of $\lambda_{r+1} > 0$ one must have $\lambda_r > \lambda_{r+1}$.*

Proof. Since $\Xi_r^e(\cdot)$ is convex, $\Xi_r^e(\cdot)$ is differentiable at X if and only if $\partial\Xi_r^e(X)$ contains a singleton. By Prop. 3.4, $\partial\Xi_r^e(X)$ is a singleton if and only if $\Pi_{\mathcal{K}_+^n(r)}^B(X)$ is a singleton. By Prop. 3.3, this holds if and only if $\Pi_{\mathcal{S}_+^n(r)}^B(JXJ)$ is a singleton. It then follows from Lemma 2.2 that $\Pi_{\mathcal{S}_+^n(r)}^B(JXJ)$ is a singleton if and only if the stated condition holds (noting that zero eigenvalues among $\{\lambda_i\}$ do not contribute to the calculation of $\Pi_{\mathcal{S}_+^n(r)}(JXJ)$). \square

We note that a similar characterization holds for the function Ξ_r [15, Remark 2.6].

4 Lagrangian Dual Approach for (5)

In this section, we study the Lagrangian dual approach to the problem (5), which is equivalent to (10). We note that \mathcal{S}_h^n is a subspace, hence (10) is equivalent to (via replacing Y by $(-Y)$)

$$v_p := \min \|Y + D\|^2/2 \quad \text{s.t.} \quad Y \in \mathcal{S}_h^n \quad \text{and} \quad Y \in \mathcal{K}_+^n(r). \quad (31)$$

The Lagrangian function $L : \mathcal{S}^n \times \mathbb{R}^n \mapsto \mathbb{R}$ is defined by

$$\begin{aligned} L(Y, y) &:= \|Y + D\|^2/2 - \langle \text{diag}(Y), y \rangle \\ &= \|Y - (-D + \text{Diag}(y))\|^2/2 + \|D\|^2/2 - \|-D + \text{Diag}(y)\|^2/2. \end{aligned}$$

Let

$$\begin{aligned} \theta(y) &:= - \min_{Y \in \mathcal{K}_+^n(r)} L(Y, y) \\ &= -\frac{1}{2} \|\Pi_{\mathcal{K}_+^n(r)}(-D + \text{Diag}(y)) - (-D + \text{Diag}(y))\|^2 + \frac{1}{2} \|-D + \text{Diag}(y)\|^2 - \frac{1}{2} \|D\|^2 \\ &= \frac{1}{2} \|\Pi_{\mathcal{K}_+^n(r)}(-D + \text{Diag}(y))\|^2 - \frac{1}{2} \|D\|^2 \quad (\text{by (29)}) \\ &= \Xi_r^e(-D + \text{Diag}(y)) - \frac{1}{2} \|D\|^2. \end{aligned}$$

The Lagrangian dual problem is then defined by

$$v_d := \max_{y \in \mathbb{R}^n} -\theta(y),$$

which is equivalent to (in the form of minimization)

$$-v_d = \min_{y \in \mathbb{R}^n} \theta(y) = \Xi_r^e(-D + \text{Diag}(y)) - \|D\|^2/2. \quad (32)$$

This is a convex problem because $\Xi_r^e(\cdot)$ is convex and $\theta(\cdot)$ is a composition of Ξ_r^e with a linear mapping. Hence, it follows from Prop. 3.4 that the subdifferential of θ at y is given by

$$\partial\theta(y) = \text{diag}(\partial\Xi_r^e(Y)) = \text{diag}\left(\text{conv}(\Pi_{\mathcal{K}_+^n(r)}^B(Y))\right), \quad (33)$$

where $Y := -D + \text{Diag}(y)$.

It is known that $v_p \geq v_d$ and the quantity $(v_p - v_d)$ is called the duality gap. The key questions for the Lagrangian dual approach are when the duality gap vanishes and whether the dual objective value v_d is attainable. We will answer those questions below.

Proposition 4.1 *The level set $L_c := \{y \in \mathbb{R}^n : \theta(y) \leq c\}$ is bounded for any constant $c \in \mathbb{R}$. Consequently, problem (32) has an optimal solution and the dual optimal objective value v_d is attained.*

Proof. We prove the conclusion of this result by contradiction. Suppose that there exists a constant c such that L_c is unbounded. There must be a sequence $\{y^k\}$ satisfying $\|y^k\| \rightarrow \infty$ and $\theta(y^k) \leq c$ for all y^k . Without loss of generality, we assume that $y^k/\|y^k\| \rightarrow \bar{y}$. Denote $B^k := (-D + \text{Diag}(y^k))/\|y^k\|$. We have $\lim_{k \rightarrow \infty} B^k = \text{Diag}(\bar{y})$. Let

$$Q(\text{Diag}(\bar{y}))Q =: \begin{bmatrix} \hat{B}_1 & \hat{b} \\ \hat{b}^T & \hat{b}_0 \end{bmatrix} \quad \text{with } \hat{B}_1 \in \mathcal{S}^{n-1}.$$

It follows from Lemma 2.2 and Prop. 3.3 that $\Pi_{\mathcal{K}_+^n(r)}^B(X)$ is positively homogeneous, i.e.,

$$\Pi_{\mathcal{K}_+^n(r)}^B(\rho X) = \rho \Pi_{\mathcal{K}_+^n(r)}^B(X), \quad \forall \rho > 0.$$

We have

$$\frac{c}{\|y^k\|^2} \geq \frac{\theta(y^k)}{\|y^k\|^2} = \frac{1}{2} \|\Pi_{\mathcal{K}_+^n(r)}(-D/\|y^k\| + \text{Diag}(y^k/\|y^k\|))\|^2 - \frac{1}{2} \|D\|^2/\|y^k\|^2.$$

Taking limits on both sides of the above inequality yields (due to the continuity of $\theta(\cdot)$)

$$\|\Pi_{\mathcal{K}_+^n(r)}(\text{Diag}(\bar{y}))\|^2 = 0,$$

which because of (24) implies

$$\|\Pi_{\mathcal{S}_+^n(r)}(J\text{Diag}(\bar{y})J)\|^2 + \|\text{Diag}(\bar{y}) - J\text{Diag}(\bar{y})J\|^2 = 0.$$

This further implies

$$\text{Diag}(\bar{y}) - J\text{Diag}(\bar{y})J = 0.$$

It follows from (27) (take $X = \text{Diag}(\bar{y})$) that

$$\bar{b} := \begin{bmatrix} \hat{b} \\ \hat{b}_0 \end{bmatrix} = 0,$$

where \bar{b} denotes the last column of $(Q\text{Diag}(\bar{y})Q)$. We thus have

$$0 = (Q\text{Diag}(\bar{y})Q)e_n = -\frac{1}{\sqrt{n}}Q\text{Diag}(\bar{y})e = -\frac{1}{\sqrt{n}}Q\bar{y},$$

where we used $Qe_n = -e/\sqrt{n}$. The nonsingularity of Q implies $\bar{y} = 0$, contradicting $\|\bar{y}\| = 1$. This proves our results. \square

Theorem 4.2 *Suppose that \bar{y} is an optimal solution of the dual problem (32). Then we must have*

$$0 \in \text{diag} \left(\text{conv} \Pi_{\mathcal{K}_+^n(r)}^B(-D + \text{Diag}(\bar{y})) \right). \quad (34)$$

Moreover, if there exists a matrix $\bar{Y} \in \Pi_{\mathcal{K}_+^n(r)}^B(-D + \text{Diag}(\bar{y}))$ such that $\text{diag}(\bar{Y}) = 0$, then \bar{Y} globally solves the primal problem (5) and there is no duality gap between the primal and dual problems.

Proof. Since the dual problem is convex, \bar{y} is its optimal solution if and only if $0 \in \partial\theta(\bar{y})$, which is equivalent to (34) by (33). Moreover, \bar{Y} is feasible to the primal problem (5). We also have

$$-\theta(\bar{y}) = L(\bar{Y}, \bar{y}) = \frac{1}{2}\|\bar{Y} + D\|^2 - \langle \text{diag}(\bar{Y}), \bar{y} \rangle = \frac{1}{2}\|\bar{Y} + D\|^2,$$

which is the primal objective function value at the feasible solution \bar{Y} . That is, there is no duality gap and hence \bar{Y} solves the primal problem (5). \square

Thm. 4.2 will be used to justify the majorized penalty approach to be introduced in the next section (see Prop. 5.4). Cor. 3.5 has its application to (34).

Corollary 4.3 *Let $\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_n$ be the eigenvalues of $(J(-D + \text{Diag}(\bar{y}))J)$. If $\lambda_{r+1} \leq 0$ or in the case of $\lambda_{r+1} > 0$, $\lambda_r > \lambda_{r+1}$, then $\bar{Y} \in \Pi_{\mathcal{K}_+^n(r)}^B(-D + \text{Diag}(\bar{y}))$ globally solves problem (5).*

Proof. We note that under the conditions stated in the corollary, $\theta(\cdot)$ is differentiable at \bar{y} and $\Pi_{\mathcal{K}_+^n(r)}^B(-D + \text{Diag}(\bar{y}))$ contains just one element \bar{Y} , which globally solves problem (5) by Thm. 4.2. \square

5 A Majorized Penalty Approach for (6)

When there exists H -weights, the Lagrangian dual approach studied in Sect. 4 is not possible because the orthogonal projection onto $\mathcal{K}_+^n(r)$ under the H -weights cannot be calculated easily. Therefore, other numerical methods have to be developed for the H -weighted problem (6). The purpose of this section is to extend the majorized penalty approach of Gao and Sun [16] to the problem (6). We will first outline this approach and state its nice convergence properties. We then focus on some implementation issues when applying this approach to (6). Finally, we justify the approach using the zero duality gap result established in Thm. 4.2.

5.1 The Gao-Sun Proposal

In [16], Gao and Sun proposed a majorized penalty approach for the following problem:

$$v_* := \min f(X), \quad \text{s.t. } X \in \Omega, \quad X \succeq 0, \quad \text{and} \quad \text{rank}(X) \leq r, \quad (35)$$

where $f : \mathcal{S}^n \mapsto \mathbb{R}$ is convex, and Ω is a closed convex set in \mathcal{S}^n defined by a finite number of linear equalities and/or inequalities on \mathcal{S}^n .

(a) **The penalty function.** Define

$$p(X) := \langle I, X \rangle - \sum_{i=1}^r \lambda_i(X), \quad \forall X \in \mathcal{S}^n$$

where $\lambda_1(X) \geq \lambda_2(X) \geq \dots \geq \lambda_n(X)$ are the eigenvalues of X . Obviously, one has

$$X \succeq 0 \quad \implies \quad p(X) \geq 0$$

and

$$X \succeq 0 \quad \text{and} \quad \text{rank}(X) \leq r \quad \iff \quad p(X) = 0, \quad X \succeq 0.$$

Therefore, the concave function $p(X)$ serves as a penalty function of the rank constraint over \mathcal{S}_+^n . This yields the following penalty problem, where $c > 0$ is the penalty parameter.

$$\min f(X) + cp(X), \quad \text{s.t. } X \in \Omega \quad \text{and} \quad X \succeq 0. \quad (36)$$

The following results on the relationship between the original problem (35) and its penalty problem (36) are the combination of Prop. 3.1 and Prop. 3.2 in [16].

Proposition 5.1 *Let X_c^* denote a global optimal solution of (36), X_r be a feasible solution of (35), and X^* be an optimal solution of the following convex problem (i.e., the rank constraint in (35) is dropped)*

$$\min f(X), \quad \text{s.t. } X \in \Omega \quad \text{and} \quad X \succeq 0.$$

(i) *If $\text{rank}(X_c^*) \leq r$, then X_c^* already solves (35).*

(ii) *If the penalty parameter c is chosen to satisfy $c \geq (f(X_r) - f(X^*))/\epsilon$, for some given $\epsilon > 0$, then we have*

$$p(X_c^*) \leq \epsilon \quad \text{and} \quad f(X_c^*) \leq v_* - cp(X_c^*).$$

The result in (ii) means that when the rank error measured by $p(\cdot)$ at X_c^* is less than ϵ , the corresponding objective value comes very close to the optimal value v_* . Such a solution is referred to as an ϵ -optimal solution in [16].

(b) The majorization functions. Due to the nature of the objective function in the penalty problem (36) (e.g., $f(\cdot)$ being H -weighted and $p(\cdot)$ concave), a majorization scheme is proposed by Gao and Sun. A function, denoted by $m_k^f(\cdot)$, is a majorization of $f(\cdot)$ at X^k if

$$m_k^f(X) \geq f(X) \quad \forall X \in \mathcal{S}^n \quad \text{and} \quad m_k^f(X^k) = f(X^k).$$

For $f(\cdot)$ being $\|H \circ (X - D)\|^2/2$, $m_k^f(\cdot)$ can be chosen to be

$$m_k^f(X) := f(X^k) + \langle H \circ H \circ (X - D), X - X^k \rangle + \frac{1}{2} \|W^{1/2}(X - X^k)W^{1/2}\|^2, \quad (37)$$

where $W := \text{Diag}(w)$ and $0 < w \in \mathbb{R}^n$ is defined by

$$w_i := \max\{\tau, \max\{H_{ij} : j = 1, \dots, n\}\}, \quad i = 1, \dots, n \quad (38)$$

for some $\tau > 0$. Other choices of W can be found in [25].

Since $p(\cdot)$ is concave, we have

$$p(X) \leq p(X^k) + \langle U^k, X - X^k \rangle, \quad \forall U^k \in \partial p(X^k).$$

A natural majorization $m_k^p(\cdot)$ for $p(\cdot)$ is then given by

$$m_k^p(X) := p(X^k) + \langle U^k, X - X^k \rangle. \quad (39)$$

Define

$$m_k(X) := m_k^f(X) + cm_k^p(X), \quad \forall X \in \mathcal{S}^n. \quad (40)$$

The majorization subproblem to be solved is

$$\min m_k(X), \quad \text{s.t.} \quad X \in \Omega \quad \text{and} \quad X \succeq 0. \quad (41)$$

(c) The majorized penalty approach. The following algorithm is then used to solve the penalty problem (36).

Algorithm 5.2 (*Majorized Penalty Algorithm (MPA)*)

(S.1) Choose $0 \preceq X^0 \in \Omega$. Set $k := 0$.

(S.2) Define the majorization function $m_k(\cdot)$ as in (40) and solve the subproblem (41) to get X^{k+1} .

(S.3) If $X^{k+1} = X^k$, stop; otherwise, set $k := k + 1$ and go to (S.2).

We note that $p(\cdot)$ is concave and hence it is differentiable almost everywhere. Let $\partial_B p(X)$ be the B -subdifferential of $p(\cdot)$ at X :

$$\partial_B p(X) := \left\{ \lim_{X^j \rightarrow X} \nabla p(X^j) : p(\cdot) \text{ is differentiable at } X^j \right\}.$$

It is known that $\partial_B p(X) \subseteq \partial p(X)$. Let $\bar{\Omega} := \Omega \cap \mathcal{S}_+^n$. For any $X \in \bar{\Omega}$, let $\mathcal{N}_{\bar{\Omega}}(X)$ denote the normal cone of $\bar{\Omega}$ at X :

$$\mathcal{N}_{\bar{\Omega}}(X) := \{Z \in \mathcal{S}^n : \langle Z, Y - X \rangle \leq 0, \quad \forall Y \in \bar{\Omega}\}.$$

A point $X \in \bar{\Omega}$ is said to be a B -stationary point of (36) if

$$(\nabla f(X) + \mathcal{N}_{\bar{\Omega}}(X)) \cap (c\partial_B p(X)) \neq \emptyset.$$

A B -stationary point is usually the best that a numerical method can find for (36) as it is nonconvex. The following convergence result of Alg. 5.2 is the application of [16, Thm. 3.4] to the case where W is being chosen as in (38).

Proposition 5.3 *Let $\{X^k\}$ be the sequence generated by Alg. 5.2. Then $\{f(X^k)\}$ is a monotonically decreasing sequence. If $X^{k+1} = X^k$ for some X^k , then X^k is an optimal solution of (36). Otherwise, the infinite sequence $\{X^k\}$ satisfies*

$$\frac{1}{2} \|W^{1/2}(X^{k+1} - X^k)W^{1/2}\|^2 \leq f(X^k) - f(X^{k+1}), \quad k = 0, 1, \dots$$

Moreover, the sequence $\{X^k\}$ is bounded and any accumulation point is a B -stationary point of (36).

5.2 Application to (6)

The problem (6) that we are facing in this paper is different from (35) because our variable Y is EDM and X in (35) is PSD. Critical issues in applying Alg. 5.2 (denoted as MPA) to (6) including choosing proper penalty functions and effectively solving the resulting subproblems. This subsection is patterned after the preceding one to highlight those critical issues as well as to save us from restating the properties of MPA in our context. Heavy calculations are often omitted below.

(a) The penalty function. Problem (6) is equivalent to (by replacing Y by $(-Y)$)

$$\min f(Y) := \|H \circ (Y + D)\|^2/2, \quad \text{s.t. } Y \in \mathcal{S}_h^n, \quad Y \in \mathcal{K}_+^n, \quad \text{and } \text{rank}(JYJ) \leq r. \quad (42)$$

Given that $JYJ \succeq 0$ for any feasible point of (42), a natural penalty function for the rank constraint is $q(Y) := p(JYJ)$. However, we have a better choice for reasons to be given.

Let $s \in \mathbb{R}^n$ be such that $e^T s = 1$. Define $J_s := I - es^T$. It follows from [20, Thm. 2] that

$$J_s Y J_s^T \succeq 0, \quad \forall Y \in \mathcal{S}_h^n \cap \mathcal{K}_+^n.$$

Moreover, $\text{rank}(J_s Y J_s^T)$ is the same for all such s . To see why the rank is constant for such s , we use the identities $J_s J = J_s$ and $J J_s = J$. Therefore, we have

$$\text{rank}(J_s Y J_s^T) = \text{rank}(J_s J Y J J_s^T) \leq \text{rank}(J Y J) = \text{rank}(J J_s Y J_s^T J) \leq \text{rank}(J_s Y J_s^T).$$

We further require $s > 0$. That is, s is a positive vector. Let $S := \text{Diag}(s)$. Then, $\text{rank}(J_s Y J_s^T) = \text{rank}(S^{1/2} J_s Y J_s^T S^{1/2})$. Recall the positive vector w defined in (38). Let s be the normalized vector of w :

$$s := w / \sum_{i=1}^n w_i. \quad (43)$$

The penalty function that we are going to use is

$$q(Y) := \left(\sum_{i=1}^n w_i \right) p(S^{1/2} J_s Y J_s^T S^{1/2}). \quad (44)$$

The penalty problem that we are going to solve is

$$\min \frac{1}{2} \|H \circ (Y + D)\|^2 + cq(Y), \quad \text{s.t. } Y \in \mathcal{S}_h^n \quad \text{and} \quad Y \in \mathcal{K}_+^n. \quad (45)$$

We note that when $w = e$, $q(Y)$ reduces to $p(JYJ)$.

(b) The majorization functions. The majorization function for $f(Y)$ is the same as in (37):

$$m_k^f(Y) := f(Y^k) + \langle H \circ H \circ (Y^k + D), Y - Y^k \rangle + \frac{1}{2} \|W^{1/2}(Y - Y^k)W^{1/2}\|^2. \quad (46)$$

We now derive the majorization function for $q(Y)$ at Y^k .

Let \sqrt{w} be the componentwise square root of w . We also define

$$J^w := I - \sqrt{w}\sqrt{w}^T / \sum_{i=1}^n w_i \quad \text{and} \quad \tilde{Y} := W^{1/2}YW^{1/2} \quad \forall Y \in \mathcal{S}^n.$$

It is easy to see that $(J^w)^2 = J^w$ and

$$J_s W^{-1/2} = (I - e s^T) W^{-1/2} = W^{-1/2} J^w,$$

which implies

$$S^{1/2} J_s Y J_s^T S^{1/2} = S^{1/2} J_s W^{-1/2} \tilde{Y} W^{-1/2} J_s^T S^{1/2} = J^w \tilde{Y} J^w / \sum_{i=1}^n w_i. \quad (47)$$

Let $\tilde{Y}^k := W^{1/2}Y^k W^{1/2}$ and \tilde{Y}^k admit the following spectral decomposition:

$$J^w \tilde{Y}^k J^w = P^k \text{Diag}(\lambda_1^k, \dots, \lambda_n^k) (P^k)^T, \quad (48)$$

where $\lambda_1^k \geq \dots \geq \lambda_n^k$ are eigenvalues of $J^w \tilde{Y}^k J^w$ and $(P^k)^T P^k = I$. Let P_1^k be the submatrix consisting of the first r columns of P^k and define

$$U^k := P_1^k (P_1^k)^T. \quad (49)$$

With some calculation, we can verify that

$$W^{1/2} \left(J^w (I - U^k) J^w \right) W^{1/2} \in \partial q(Y^k).$$

The majorization function for $q(Y)$ is then given by

$$m_k^q(Y) := q(Y^k) + \langle W^{1/2} (J^w (I - U^k) J^w) W^{1/2}, Y - Y^k \rangle. \quad (50)$$

Let

$$m_k(Y) := m_k^f(Y) + c m_k^q(Y).$$

The subproblem corresponding to (41) is

$$\min m_k(Y), \quad \text{s.t.} \quad Y \in \mathcal{S}_h^n \cap \mathcal{K}_+^n. \quad (51)$$

The only remaining issue concerning the application of Alg. 5.2 is how to solve the subproblem (51). We address this issue below.

(c) Solving the majorization subproblem (51). By ignoring the constant terms in the objective function, the majorized problem (51) is equivalent to the following problem:

$$\min \|W^{1/2}(Y + D^k)W^{1/2}\|^2/2, \quad \text{s.t.} \quad Y \in \mathcal{S}_h^n \text{ and } Y \in \mathcal{K}_+^n, \quad (52)$$

where $D^k := -Y^k + W^{-1}(H \circ H \circ (Y^k + D))W^{-1} + cW^{-1/2}J^w(I - U^k)J^wW^{-1/2}$, whose calculation needs no more than $6n^2$ operations. Problems of the type (52) are known as the diagonally weighted least squares problems [15]. Problem (52) can be effectively solved by the well developed Semismooth Newton-CG method in [43, Eq. (66)]. Moreover, the solution obtained will also justify why we choose the penalty function (44) over others (e.g., $p(JYJ)$). Below we validate those claims.

Let $\widetilde{D}^k := W^{1/2}D^k W^{1/2}$. It is easy to see that

$$Y \in \mathcal{S}_h^n \iff \widetilde{Y} = W^{1/2}Y W^{1/2} \in \mathcal{S}_h^n$$

and

$$\begin{aligned} Y \in \mathcal{K}_+^n &\iff W^{-1/2}\widetilde{Y}W^{-1/2} \in \mathcal{K}_+^n \\ &\iff W^{-1/2}\widetilde{Y}W^{-1/2} \succeq 0 \text{ on } \{e\}^\perp \\ &\iff \widetilde{Y} \succeq 0 \text{ on } \{W^{1/2}e\}^\perp \\ &\iff \widetilde{Y} \in \mathcal{K}_w^n \end{aligned} \tag{53}$$

where \mathcal{K}_w^n is the closed convex cone defined by

$$\mathcal{K}_w^n := \left\{ A \in \mathcal{S}^n \mid A \succeq 0 \text{ on } \{W^{1/2}e\}^\perp \right\}.$$

Problem (52) is equivalent to the following problem.

$$\min \|\widetilde{Y} + \widetilde{D}^k\|^2/2, \quad \text{s.t. } \widetilde{Y} \in \mathcal{S}_h^n \quad \text{and} \quad \widetilde{Y} \in \mathcal{K}_w^n. \tag{54}$$

By applying the Lagrangian dual approach studied in Sect. 4, we obtain the following dual problem:

$$\min \theta_w(y) := \|\Pi_{\mathcal{K}_w^n}(-\widetilde{D}^k + \text{Diag}(y))\|^2/2 - \|\widetilde{D}^k\|^2/2, \tag{55}$$

which can be effectively solved by the Semismooth Newton-CG method [43, Eq. (66)]. Let y^k denote the optimal solution of (55). Then

$$Y^{k+1} = W^{-1/2}\Pi_{\mathcal{K}_w^n}(-\widetilde{D}^k + \text{Diag}(y^k))W^{-1/2} \tag{56}$$

is the optimal solution of (52). We need to calculate $\Pi_{\mathcal{K}_w^n}(-\widetilde{D}^k + \text{Diag}(y^k))$.

Let Q_w be the Householder transformation that maps the vector $W^{1/2}e$ to $[0, \dots, 0, -\|W^{1/2}e\|]^T$. Then

$$Q_w = I - \frac{2}{u^T u} u u^T,$$

where

$$u := \left[\sqrt{w_1}, \dots, \sqrt{w_{n-1}}, \sqrt{w_n} + \sqrt{\sum_{i=1}^n w_i} \right]^T.$$

Clearly, if $w = e$, we have $Q_w = Q$. It is easy to verify that

$$J^w = I - \frac{1}{\sum_{i=1}^n w_i} \sqrt{w} \sqrt{w}^T = Q_w \begin{bmatrix} I_{n-1} & 0 \\ 0 & 0 \end{bmatrix} Q_w.$$

By making use of the property above, we can prove the following results by following the proof for Lemma 2.1 (i) and (ii)

$$\left. \begin{aligned} \Pi_{\mathcal{K}_w^n}(A) &= \Pi_{\mathcal{S}_+^n}(J^w A J^w) + (A - J^w A J^w) \\ J^w \Pi_{\mathcal{K}_w^n}(A) J^w &= \Pi_{\mathcal{S}_+^n}(J^w A J^w) \end{aligned} \right\} \quad \forall A \in \mathcal{S}^n. \tag{57}$$

When applied to (56), (57) means that we need to compute the spectral decomposition $J^w A^k J^w$, where $A^k := -\widetilde{D}^k + \text{Diag}(y^k)$. We now return to the penalty function (44). In order

to construct the majorization function $m_k^q(Y)$ at Y^{k+1} , we need to calculate U^{k+1} by (49), which further requires the spectral decomposition (48) for $J^w \widetilde{Y^{k+1}} J^w$. We have

$$\begin{aligned} J^w \widetilde{Y^{k+1}} J^w &= J^w W^{1/2} Y^{k+1} W^{1/2} J^w && \text{(by definition of } \widetilde{Y^{k+1}} \text{)} \\ &= J^w \Pi_{\mathcal{K}_w^n}(A^k) J^w && \text{(by (56))} \\ &= \Pi_{\mathcal{S}_+^n}(J^w A^k J^w) && \text{(by (57)).} \end{aligned}$$

This means that the spectral decomposition $J^w \widetilde{Y^{k+1}} J^w$ is a by-product of computing Y^{k+1} in (56). In other words, the choice of the penalty function (44) saves us from doing a new spectral decomposition (48) in order to construct the majorization function $m_k^q(Y)$ at Y^k .

(d) The rationale of the majorized penalty approach. In this part, we explain the rationale behind the majorized penalty approach for the problem (42). For simplicity, we only consider the case $H = E$. But the results below can be extended to the diagonally weighted case, hence adequate to cover the general case. Therefore, the problem that we consider in this part is

$$\min f(Y) := \|Y + D\|^2/2, \quad \text{s.t. } Y \in \mathcal{S}_h^n, \quad Y \in \mathcal{K}_+^n, \quad \text{and } \text{rank}(JYJ) \leq r. \quad (58)$$

The corresponding penalty problem is

$$\min \|Y + D\|^2/2 + cq(Y), \quad \text{s.t. } Y \in \mathcal{S}_h^n \quad \text{and } Y \in \mathcal{K}_+^n. \quad (59)$$

Let

$$\overline{\Omega} := \{Y \in \mathcal{S}^n : Y \in \mathcal{S}_h^n \text{ and } Y \in \mathcal{K}_+^n\}.$$

When the majorized penalty approach Alg. 5.2 is applied to (59), we obtain a B -stationary point \overline{Y} (see Prop. 5.3):

$$(\nabla f(\overline{Y}) + \mathcal{N}_{\overline{\Omega}}(\overline{Y})) \cap (c\partial q_B(\overline{Y})) \neq \emptyset. \quad (60)$$

We assume that \overline{Y} satisfies the rank constraint $\text{rank}(J\overline{Y}J) \leq r$. This assumption is reasonable as one of our stopping criteria for Alg. 5.2 is to ensure that the rank error, measured by $q(Y)$, is small (see (62)). This assumption implies $\overline{Y} \in \mathcal{K}_+^n(r)$. We have the following key result.

Proposition 5.4 *We further assume that $\text{rank}(J\overline{Y}J) = r$ (i.e., the rank constraint is tight). Then, there exists $\overline{y} \in \mathbb{R}^n$ such that*

$$\overline{Y} \in \Pi_{\mathcal{K}_+^n(r)}^B(-D + \text{Diag}(\overline{y})).$$

Consequently, \overline{y} solves the Lagrangian dual problem (32) and \overline{Y} solves the primal problem (58).

Proof. Denote

$$Q\overline{Y}Q =: \begin{bmatrix} Z & z \\ z^T & z_0 \end{bmatrix} \quad \text{with } Z \in \mathcal{S}^{n-1}.$$

Because $\overline{Y} \in \mathcal{K}_+^n(r)$, we have $\overline{Y} \in \Pi_{\mathcal{K}_+^n(r)}^B(\overline{Y})$. We also note that

$$J\overline{Y}J = Q \begin{bmatrix} Z & 0 \\ 0 & 0 \end{bmatrix} Q.$$

It follows from Prop. 3.3 that

$$Q \begin{bmatrix} Z & z \\ z^T & z_0 \end{bmatrix} Q = \overline{Y} \in \Pi_{\mathcal{K}_+^n(r)}^B(\overline{Y}) = \Pi_{\mathcal{S}_+^n(r)}^B(J\overline{Y}J) + \overline{Y} - J\overline{Y}J = Q \begin{bmatrix} \Pi_{\mathcal{S}_+^{n-1}(r)}^B(Z) & z \\ z^T & z_0 \end{bmatrix} Q.$$

This means that $Z \in \Pi_{\mathcal{S}_+^{n-1}(r)}^B(Z)$, implying that $Z \in \mathcal{S}_+^{n-1}(r)$. Therefore, we assume that Z has the following spectral decomposition:

$$Z = V \begin{bmatrix} \Lambda_1 & 0 \\ 0 & 0 \end{bmatrix} V^T,$$

where $\Lambda_1 = \text{Diag}(\lambda_1, \dots, \lambda_r)$ with $\lambda_1 \geq \dots \geq \lambda_r > 0$ being the eigenvalues of Z ; and $V^T V = I_{n-1}$. We write $V = [V_1, V_2]$ and the columns of V_1 being the eigenvectors corresponding to $\lambda_1, \dots, \lambda_r$.

Then we have

$$\bar{Y} = Q \begin{bmatrix} V \begin{bmatrix} \Lambda_1 & 0 \\ 0 & 0 \end{bmatrix} V^T & z \\ z^T & z_0 \end{bmatrix} Q \quad \text{and} \quad J\bar{Y}J = Q \begin{bmatrix} V \begin{bmatrix} \Lambda_1 & 0 \\ 0 & 0 \end{bmatrix} V^T & 0 \\ 0 & 0 \end{bmatrix} Q.$$

It follows from [17, Thm 3.1] that

$$\mathcal{N}_{\mathcal{K}_+^n}(\bar{Y}) = \left\{ Q \begin{bmatrix} V \begin{bmatrix} 0 & 0 \\ 0 & M \end{bmatrix} V^T & 0 \\ 0 & 0 \end{bmatrix} Q : -M \in \mathcal{S}_+^{n-r-1} \right\},$$

and from [17, Thm. 3.2] that

$$\mathcal{N}_{\bar{\Omega}}(\bar{Y}) = \{\text{Diag}(y) : y \in \mathbb{R}^n\} + \mathcal{N}_{\mathcal{K}_+^n}(\bar{Y}).$$

We further note that in this case (i.e., $w = e$) $q(Y) = p(JYJ)$. Under the assumption $\text{rank}(J\bar{Y}J) = r$, we have

$$\partial_B q(\bar{Y}) = \left\{ J - Q \begin{bmatrix} V_1 \\ 0 \end{bmatrix} [V_1^T \ 0] Q \right\} = \left\{ Q \begin{bmatrix} V \begin{bmatrix} 0 & 0 \\ 0 & I_{n-r-1} \end{bmatrix} V^T & 0 \\ 0 & 0 \end{bmatrix} Q \right\}.$$

The B -stationary point condition (60) implies that there exist \bar{y} and $M \in -\mathcal{S}_+^{n-r-1}$ such that

$$\bar{Y} + D - \text{Diag}(\bar{y}) + Q \begin{bmatrix} V \begin{bmatrix} 0 & 0 \\ 0 & M \end{bmatrix} V^T & 0 \\ 0 & 0 \end{bmatrix} Q = cQ \begin{bmatrix} V \begin{bmatrix} 0 & 0 \\ 0 & I_{n-r-1} \end{bmatrix} V^T & 0 \\ 0 & 0 \end{bmatrix} Q.$$

Rearranging the similar terms in the above equation yields

$$-D + \text{Diag}(\bar{y}) = Q \begin{bmatrix} V \begin{bmatrix} \Lambda_1 & 0 \\ 0 & M - cI_{n-r-1} \end{bmatrix} V^T & z \\ z^T & z_0 \end{bmatrix} Q.$$

It follows from Prop. 3.3 that

$$\begin{aligned} & \Pi_{\mathcal{K}_+^n(r)}^B(-D + \text{Diag}(\bar{y})) \\ &= \Pi_{\mathcal{S}_+^n(r)}^B(J(-D + \text{Diag}(\bar{y}))J) + (-D + \text{Diag}(\bar{y})) - J(-D + \text{Diag}(\bar{y}))J \\ &= Q \begin{bmatrix} V \Pi_{\mathcal{S}_+^{n-1}(r)}^B \left(\begin{bmatrix} \Lambda_1 & 0 \\ 0 & M - cI_{n-r-1} \end{bmatrix} \right) V^T & z \\ z^T & z_0 \end{bmatrix} Q \\ &\ni Q \begin{bmatrix} V \begin{bmatrix} \Lambda_1 & 0 \\ 0 & 0 \end{bmatrix} V^T & z \\ z^T & z_0 \end{bmatrix} Q \\ &= \bar{Y}. \end{aligned}$$

The penultimate line used the fact that M is negative semidefinite.

We note that \bar{Y} is feasible to problem (58). Hence,

$$0 = \text{diag}(\bar{Y}) = \text{diag}\left(\Pi_{\mathcal{K}_+^n(r)}^B(-D + \text{Diag}(\bar{y}))\right).$$

It follows from Thm. 4.2 that \bar{y} solves the dual problem (32) and \bar{Y} solves the primal problem (58). \square

We make following brief remarks about Prop. 5.4.

- (i) When r (e.g., $r = 2, 3$) is small, which is often the case in real applications, the assumption $\text{rank}(J\bar{Y}J) = r$ (tight rank) always holds. When $\text{rank}(J\bar{Y}J)$ is strictly less than r , \bar{Y} has to be a global optimal solution (instead of being just a B -stationary point) of (59) in order for it to be the optimal solution of (58) (see Prop. 5.1(i)). Prop. 5.4 together with Prop. 5.1 provides the rationale behind the majorized penalty approach. Other rationale in the context of the low-rank positive semidefinite optimization can be found in [15, Sect. 4.2], [38], and the recent PhD thesis [39].
- (ii) In Alg. 5.2, the subproblem is solved by the semismooth Newton-CG method [43]. The Lagrange multiplier y calculated by the method can be used as an approximate to \bar{y} in the result.
- (iii) The result can be extended to the diagonally weighted problem

$$\min \|W^{1/2}(Y + D)W^{1/2}\|/2, \quad \text{s.t. } Y \in \mathcal{S}_h^n \text{ and } Y \in \mathcal{K}_+^n(r).$$

6 Numerical Results

In this section, we report extensive numerical results on both problems (5) and (6), which arise from three important fields: Euclidean Embedding of Network Coordinate Systems, Sensor Network Localization (SNL), and Molecular Conformation. This section is organized as follows. We first describe the solvers we have used and their parameters setting. We then describe the applications with comments on the obtained numerical results.

6.1 Solvers and Their Parameters Setting

(a) Solvers. The following solvers are tested: SDP solver of [4] (Semidefinite Programming based solver); MBO solver of [40] (Manifold-Based-Optimization method with its optimization algorithm option being the trust-region algorithm); and Alg. 5.2 applied to (6). We call the resulting algorithm EMBED, which stands for “EMbedding the Best Euclidean Distances”.

(b) Initialization. When the matrix D is partially known, it can be used to constructed a graph having edge (i, j) if d_{ij} is known and d_{ij} is treated as the corresponding edge weight of (i, j) . The graph is connected if for any two nodes i and j , there exists a chain of edges that link i and j . Therefore, the shortest path can be calculated between any two nodes if the graph is connected. For a disconnected graph, it can be decomposed into small sized connected graphs, which can be dealt with separately. The famous `Isomap` method of Tenenbaum et al. [45] is a novel variant of cMDS in that the distances used in cMDS are the shortest path distances of a connected graph constructed in a certain way based on the partially known distances.

Motivated by `Isomap`, in this paper, we replace the missing distance d_{ij} by its shortest path distance and keep those known d_{ij} unchanged. We used the `MATLAB` built-in function

`graphallshortestpaths` to calculate the shortest path distances to replace all the missing distances. We denote the resulting distance matrix by D_s . We then solve the convex problem (9) with D being replaced by D_s by the Semismooth Newton-CG method [43] to obtain Y^0 .

We will see that this initialization step is vital to certain problems with many missing distances (e.g., D is sparse). For more comments, see the second paragraph of Sect. 6.3(f).

(c) Parameters setting. For MBO [40], we set `params.tol` = `params.vtol` = `1.0e-5` and the optimization method used is the trust-region method. For SDP [4] solver, we set `OPTIONS.tolrel` = `3.0e-4` (all others are default values). For EMBED (Alg. 5.2 applied to (6)), we terminate it when the following two conditions are met. The first condition is on the objective function value.

$$f_{\text{frog}} := \frac{|\sqrt{f(Y^k)} - \sqrt{f(Y^{k-1})}|}{\max\{100, \sqrt{f(Y^{k-1})}\}} \leq \text{tol}, \quad (61)$$

where $f(Y) = 0.5\|H \circ (Y - D)\|^2$ and `tol` is a small tolerance level (e.g., 1.0×10^{-4}). In other words, whenever there is lack of the relative progress on the successive objective function values, we believe that the current iterate is a good candidate subject to the second condition below. This stopping criterion was suggested by Gao and Sun [16] for the low-rank nearest correlation matrix problems.

The second condition is on the rank of the current iterate Y^k . There are two ways to monitor the rank. One is to compute the absolute value of the eigenvalue residue:

$$\text{rankerror} := \sum_{i=r+1}^n \lambda_i(-JY^k J) \leq \text{ranktol}, \quad (62)$$

where `ranktol` is a small tolerance (e.g., 10^{-2}) and $\lambda_1 \geq \dots \geq \lambda_n$ are the eigenvalues of $(-JY^k J)$, which is positive semidefinite. This quantity does not scale well with the magnitude of $(-JY^k J)$. To rectify this drawback, we also calculate the percentage of the first r eigenvalues of $(-JY^k J)$ out of all the eigenvalues.

$$\text{Eigenratio} := \sum_{i=1}^r \lambda_i(-JY^k J) / \sum_{i=1}^n \lambda_i(-JY^k J) \leq \text{Eigentol}, \quad (63)$$

`Eigentol` is a high percentage (e.g., 90%).

EMBED solves the penalty problem (45) for a fixed penalty parameter c . In practical implementation, we may start from c_0 and increase c a few times before we can find a good solution. The initial $c = c_0$ is calculated as follows. We first compute `rankerror` by (62) with $Y^k = Y^0$. Let

$$c_0 := \begin{cases} \sqrt{\text{rankerror}}, & \text{if } \text{rankerror} > 10^3 \\ \max\{10, \text{rankerror}\}, & \text{otherwise.} \end{cases}$$

Update c_k ($k \geq 1$) as follows

$$c_k := \begin{cases} c_{k-1}, & \text{if } \text{rank}(JY^{k-1} J) \leq r \\ 4c_{k-1}, & \text{otherwise.} \end{cases}$$

That is, we keep the penalty parameter unchanged if the current iterate has the desired embedding dimension. Otherwise, it is increased by 4 times.

We emphasize that the main reason for using a large value (10^3) in choosing c_0 is that for large scale problems, the initial point often has large rank errors. Choosing a large c_0 would quickly bring the rank of first few iterates close to r . Obviously, this heuristic choice can be fine tuned depending on problems at hand. All tests were carried out using the 64-bit version of MATLAB R2011b on a Windows 7 desktop with 64-bit operating system having Intel(R) Core(TM) 2 Duo CPU of 3.16GHz, and 4.0 GB of RAM. The companion MATLAB package EMBED is available from <http://www.personal.soton.ac.uk/hdqi>.

6.2 Application 1: Euclidean Embedding of Network Coordinate Systems

One important approach to many networked problems is to embed the Internet nodes (e.g., network of hosts) in a Euclidean space with an appropriately chosen dimension based on the pairwise distance (e.g., measured by round-trip time) matrix. However, there is always a bias on the distance measured. This is due to a number of reasons. For example, during the time that the measurements are being taken, the nodes may not have full Internet connectivity or they may experience local connection issues. Real experiences on measuring the distances over a global network are documented in [42, Sect. V (A): Data Collection].

Consequently, the distance matrix would have missing distances and the embedding has to be carried out based on the obtained partial distance matrix. For example, the matrix **King1740**¹ is a distance matrix among 1740 nodes on the Internet with 13284 missing pairwise distances. The matrix **Meridian2500**² is a distance matrix among 2500 nodes with 1766 missing distances. The matrix **Ng19**³ is a 19×19 distance matrix with no missing distances. Lee et al. [34] studied the suitability of the Euclidean embedding. One restriction of this study is that the network has to be fully connected. In other words, they assume no missing distances among the network. In [34], the nodes with missing measurements are removed from the network. For example, **King1740** is reduced to only 462 nodes (i.e., **King462**) after the removal.

One of the standard metrics to measure the overall fitness of embedding is the **stress** [5] (also see [34]) defined by

$$\text{stress} := \sqrt{\frac{\sum_{i,j} (d_{ij} - \hat{d}_{ij})^2}{\sum_{i,j} d_{ij}^2}},$$

where d_{ij} is the actual measured distance between node i and j and \hat{d}_{ij} is the estimated one. In this application, we deal with the original matrices with missing distances. We note that the smaller the **stress** is, the better the overall estimation will be. Let $D = (d_{ij}^2)$ be the (original) squared distance matrix and $H_{ij} = 1$ if d_{ij} is available, and 0 otherwise. We calculate the nearest (squared) distance matrix, denoted by \hat{D} , to D through (6). We then take $\hat{d}_{ij} = \sqrt{\hat{D}_{ij}}$. The embedding dimension used is 7, which is suggested by [34, Sect. B].

In Table 1, we report the following data obtained by **MBO** and **EMBED** (the **SDP** solver is not suitable here as the problems **King1740** and **Meridian2500** are too large and both have a large number of distances available with less than 1% missing distances): **cpu** time is in seconds; **obj** := $\sqrt{\sum (H_{ij}(\hat{d}_{ij} - d_{ij})^2)}$; and the **stress** (the most important data in the table).

Problems	MBO			EMBED		
Name	cpu	obj	stress	cpu	obj	stress
Ng19	0.6	232.81	6.24e-2	0.3	272.12	3.10e-2
King1740	6678	92612525	0.18	501	95965495	0.19
Meridian2500	17784	90865440	0.17	2215	93423287	0.18

Table 1: Comparison between **MBO** and **EMBED** on three partial distance matrices from Network coordinate systems.

The **stress** data were also computed in [34, Fig.2(a)] by 4 well-known methods in computer sciences, but based on the fully connected distance matrices as mentioned before. Applying those methods here is not straightforward. But the data reported in [34] is sufficient to show that our approach is very promising in this application. By comparing the corresponding stress

¹Available from <http://pdos.lcs.mit.edu/p2psim/Kingdata>.

²Available from <http://www.cs.cornell.edu/People/egs/meridian/data.php>.

³Available from <http://www.cs.rice.edu/~eugeneng/research/gnp>.

data between [34, Fig.2(a)] and Table 1, it is clearly seen that the results by `EMBED` show a significant improvement on all problems. For example, the goodness-of-fit (i.e., the stress) of `Meridian2500` is at least 4 times better than that of `King2305`, which was obtained by removing the 1766 missing distances from `Meridian2500`.

We feel that this simple application of our approach to the Network coordinate systems opens a new embedding method and its full potential needs to be assessed over a large set of data. However, such a process of assessment, which is more on the assessment methodologies than optimization methods, is beyond the scope of this paper.

6.3 Application 2: Sensor Network Localization and Molecular Conformation

The first application does not involve actual embedding. We stopped at calculating the `stress` measure of the goodness-of-fit. In this application, we calculate the actual embedding and use another goodness-of-fit measure: the Root Mean Square Distance (RMSD). The problems in this part are from the Sensor Network Localization (SNL) and Molecular Conformation.

(a) SNL and Molecular conformation. Generally speaking, SNL describes a class of problems that aim to estimate the location (i.e., Cartesian coordinates) of some points based on a set of incomplete Euclidean pairwise distances between those points. As in [47], suppose there are n distinct points in \mathbb{R}^r , denoted by $[\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n]$. We know the Cartesian coordinates of the first m points (“anchors”), and the (inaccurate or contaminated) Euclidean distances d_{ij} between “neighboring” points i and j for $(i, j) \in \mathcal{B}$, where \mathcal{B} is a set of index pairs (i, j) with $1 \leq i < j \leq n$. We wish to estimate the Cartesian coordinates of the last $(n - m)$ points (“sensors”). When $m = 0$, it is known as anchor-free SNL. There are a large number of publications on SNL. We do not intend to give a review on this subject, but instead refer to the survey paper [30] and many references therein. As stated in [30], SNL can be formulated as (6) with $H_{ij} = 1$ and $D_{ij} = d_{ij}^2$ for (i, j) or $(j, i) \in \mathcal{B}$; and $H_{ij} = 0$ otherwise.

The molecular conformation considered in this application refers to any one of the infinite number of possible spatial arrangements of atoms in a molecule that result from rotation of its constituent groups of atoms about single bounds. We intend to construct a molecular conformation from a set of observed distances among atoms (denoted as $\{d_{ij} : (i, j) \in \mathcal{B}\}$), which can be obtained through an experimental technique such as NMR (Nuclear Magnetic Resonance). The problem can then be formulated as (6) as we do for SNL.

In the test problems below, Examples 6.1 and 6.5 are examples of molecular conformation and the rest are of SNL.

(b) Test problems. We tested three groups of problems that can all be modelled as SNL or molecular conformation. The true coordinates are known in these problems. The first group consists of deterministic examples.

Example 6.1 [7, 40] (Helix structure) Consider the 3-dimensional helix structure defined by

$$(x, y, z) = (4 \cos(3t), 4 \sin(3t), 2t), \quad 0 \leq t \leq 2\pi.$$

We sample n equal-paced points on $[0, 2\pi]$ and compute all the distances between those points.

Example 6.2 (The data set of cities⁴) We choose to test 3 instances. Data of `uscaps` contains distances of 50 US capitals and they are computed from their 2 dimensional coordinates. Data of `kn57` contains distances (in integer) of miles between 57 cities. Since they are “highway miles”, the distances are difficult to use to construct an exact planar map. Data of `wg59` contains

⁴Data available from <http://people.sc.fsu.edu/~jburkardt/datasets/cities/cities.html>.

distances (in integer) of 59 cities in western Germany. But it is not clear how those distances were calculated. The two-dimensional coordinates are also provided in those examples. Hence, the embedding dimension is $r = 2$.

The second group are randomly generated examples of SNL with anchors.

Example 6.3 [4] (denoted as *SNLa*) First, 4 anchors are placed at the positions $(\pm 0.45, \pm 0.45)$. Then $(n - 4)$ points are randomly generated in the unit square $[-0.5, 0.5] \times [-0.5, 0.5]$ via the MATLAB command: $\mathbf{x} = \mathbf{rand}(2, 1)$. The distances between the n points are then calculated.

Example 6.4 [3, 47] (denoted as *SNLb*) The n points are generated as in Example 6.3. But instead of fixed positions of anchors, we choose 10% of the points (i.e., $10n\%$) to be anchors.

Example 6.5 [2] The last group are the Protein Data Bank (PDB) problems [2]. They are 1GM2, 1PBM, 1PTQ, 1CTF, 1AU6, 1HOE, 1PHT, 1POA, 1AX8, 1F39, 1GPV, 1RGS.

(c) Selection of distances. We select a subset of distances to use according to the following two rules. The first rule is to randomly select a percentage p (e.g., 15%) of all available distances. Let the vector \mathbf{v} denote those selected distances and $\sigma(\mathbf{v})$ be the standard deviation of \mathbf{v} . Then the selected distances are perturbed by

$$\widehat{d}_{ij} := d_{ij} + \mathbf{nf} \cdot \sigma(\mathbf{v}) \cdot \epsilon_{ij}, \quad (64)$$

where d_{ij} is the true distance between point i and j ; \mathbf{nf} is the noise factor (e.g., $\mathbf{nf} = 0.1$ corresponds 10% of noise level); and ϵ_{ij} are independent standard normal random variables. This type of perturbation is known to be “additive”.

The second rule is to use the “unit ball” model to define \mathcal{B} .

$$\mathcal{B} := \{(i, j) : d_{ij} \leq \mathbf{radiatorange}\},$$

where $\mathbf{radiatorange} > 0$. Let

$$\widehat{d}_{ij} := d_{ij} \cdot |1 + \epsilon_{ij} \cdot \mathbf{nf}|, \quad \forall (i, j) \in \mathcal{B}. \quad (65)$$

This type of perturbation is known to be “multiplicative”. Let $D = (\widehat{d}_{ij}^2)$.

(d) The goodness-of-fit measure: RMSD. All three solvers are able to provide the estimated positions $\{\widehat{\mathbf{x}}_i, i = 1, \dots, n\}$. We will use the Root Mean Square Distance (RMSD) to measure the accuracy of the estimated positions

$$RMSD := \frac{1}{\sqrt{n}} \left(\sum_{i=1}^n \|\widehat{\mathbf{x}}_i - \mathbf{x}_i\|^2 \right)^{1/2},$$

where $\{\mathbf{x}_i\}$ are the actual positions. In order to calculate RMSD, the Procrustes analysis (see, e.g. [9, Chp. 5]) has to be carried out on the obtained solutions by each solver. Since the SDP solver has already included this matching procedure, there is no need to describe it here.

For EMBED, the positions of sensors estimated are further refined to obtain $\{\widehat{\mathbf{x}}_i, i = 1, \dots, n\}$ by applying the gradient descent method proposed in [4, Sect. V]. This post-processing step is necessary as the embedding positions obtained from solving (5) or (6) are based on the nearest EDM to a contaminated distance matrix. The embedding positions are results of all calculated nearest distances. Consequently, there is room for improvement when only those known distances are used via the gradient method. For MBO solver, this post refinement does not lead to

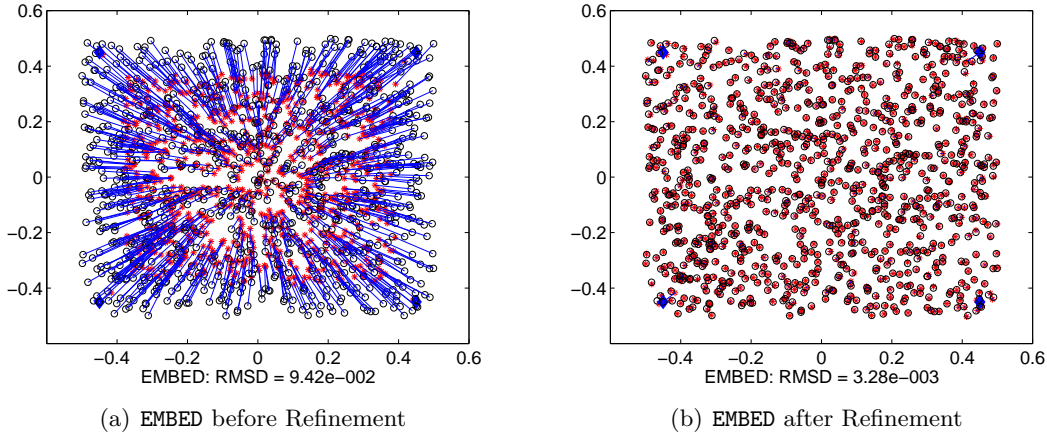


Figure 1: Example 6.3 with the multiplicative noisy rule (65) solved by `EMBED` ($n = 1000$, `radiorange` = 0.4, and `nf` = 0.1). Anchors are shown by \diamond . Each sensor position found (“*”) is joined to its true position (“o”) by a line. The left figure (a) shows the result before the gradient-descent method refinement and the right figure (b) shows the result after the refinement.

much improvement and it sometimes leads to worse embedding positions. Therefore, we will not apply this refinement on `MBO`.

(e) The effect of the gradient-descent method refinement. The necessity and the effect of the refinement step after solving (6) is clearly demonstrated in Fig. 1, where Example 6.3 ($n = 1000$) was solved by `EMBED`. The found sensor positions after solving (6) have a higher RMSD and they tend to be attracted to the center of the graph (they formed a “ring” around the center). Those found positions are the best one can get from the calculated nearest EDM. Furthermore, the nearest EDM contains noises because it was obtained from a incomplete set of noisy distances. In order for those positions to be “closer” to the true positions, they would have to be embedded in a higher dimensional space. The found positions can then be seen as the projections from this higher dimensional embedding. This explains why they tend to be “crowded” around the center. We also note that they are not far from their respective true positions. The refinement step brings them very close to their true positions (low RMSD). This “crowding” phenomenon is also observed in the `SDP` approach, see [4]. We note that the refinement step of [4] does not always improve RMSD.

(f) Numerical comparison. The `SDP` solver of [4] is known to be robust and is capable of providing localization of high quality. Therefore, we choose it to be our benchmark for comparison. However, it is only suitable for n not very big. Hence, our first test is on small sized problems in order to learn the solution qualities of `EMBED`. We then test medium to large scale problems without comparing to the `SDP` solver.

We use the `PDB` problem `1GM2` to conduct our numerical comparison and draw a few observations. Figure 2 plots the time and RMSD against the `radiorange` obtained from the three solvers. For this problem, `SDP` used much more time than both `MBO` and `EMBED` to terminate while the latter two used a similar amount of time (about 1 second). However, `EMBED` performed best in terms of RMSD. The accuracy of localizations increases as the radio range increases for all three solvers. A large radio range means high density, which is the ratio between the number of distances known and the total number of the distances. For `radiorange` = 4, both `SDP` and `MBO` failed to produce a satisfactory localization ($\text{RMSD} \geq 5$), whereas `EMBED` yielded a RMSD (≈ 1) of much better quality. `SDP` works well from 5, `MBO` from 7, and `EMBED` from 4.

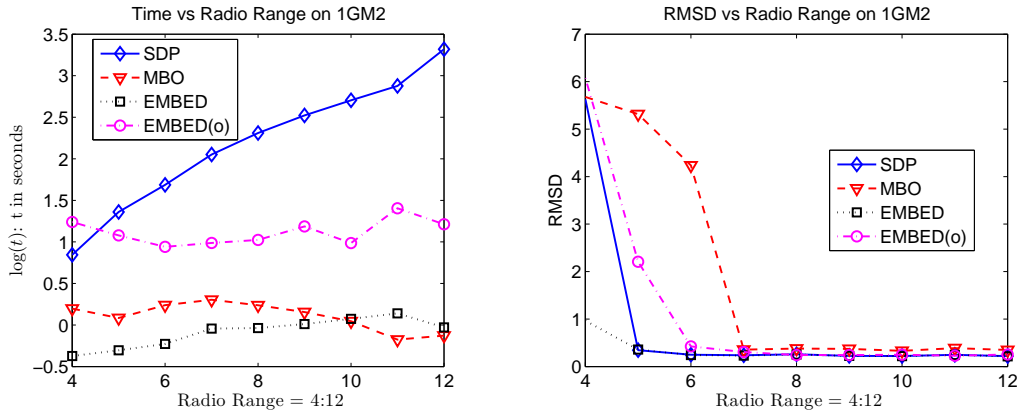


Figure 2: Comparison between Time, RMSD and Radio Range on PDB problem: 1GM2 with the multiplicative noisy rule (65) (random state is 0, $n = 166$, $\text{nf} = 0.1$). Results are average on 10 repeated runs.

The key to the success of **EMBED** is its use of the shortest path in the Initialization step. To see this, we also include the behavior of **EMBED(o)**, which is **EMBED** without using the shortest path. We used a tighter `ranketo1` = 10^{-8} (see (62)) in order for **EMBED(o)** to achieve comparable results. It can be seen that **EMBED(o)** only worked well from `radiorange` ≥ 6 and used longer time than **EMBED**. Therefore, using the shortest path distances to replace the missing distances is vital for the performance of **EMBED**. Further evidence that supports this observation came from that, without using the shortest path distances, **EMBED** would have failed all PDB test problems except 1PBM, 1AU6 in Table 4.

Problems		SDP			MBO			EMBED		
Name	n	cpu	obj	rmsd	cpu	obj	rmsd	cpu	obj	rmsd
helix	121	221	492.54	8.19e-2	0.5	488.27	9.71e-2	0.2	492.12	8.09e-2
uscap	50	10	20293488	48.91	2	18696367	103.85	0.1	20172903	46.51
kn57	57	6	9484235	22.01	1.2	9061894	33.69	0.1	9414059	20.88
wg59	59	7	53685	1.42	0.3	52284	1.92	0.1	53554	1.39
SNLa	104	103	3.07	5.80e-3	–	–	–	0.5	3.09	5.80e-3
SNLb	100	83	2.83	5.79e-3	–	–	–	0.4	3.01	5.80e-3

Table 2: Average results on 20 instances of each problem when 100% distances are available. The distances (may not be true distances) are further perturbed according to (64) with $\text{nf} = 0.1$. Problems **SNLa**, **SNLb** are not suitable for MBO.

A particular scenario that has higher densities of known distances happens in sampling a function (e.g., Example 6.1) and in geospatial data (e.g., Example 6.2). Table 2 reports results on CPU time (in seconds); objective value ($\sqrt{\sum H_{ij}(\|\hat{\mathbf{x}}_i - \mathbf{x}_i\|^2 - d_{ij}^2)^2}$); and RMSD when all 100% distances are used. The distances are perturbed according to (64). For each problem, we generated 20 randomly perturbed distances and the results in Table 2 are average on those 20 instances. It can be seen that the SDP solver and **EMBED** produced localizations of similar quality in terms of RMSD and **EMBED** only used a fraction of the time used by the SDP solver. While MBO performed well on **helix** and **wg59** examples, it yielded much worse localizations on **uscap**, **kn57** than SDP and **EMBED**. As MBO is designed for anchor-free SNL problems, the examples of **SNLa**, **SNLb** (with anchors) are not suitable for MBO to compare with the other two solvers. Hence, we will not test **SNLa**, **SNLb** on MBO here and hereafter.

We now present three more tables of results with $n \geq 500$. We only compare the results

between MBO and EMBED. Table 3 is on problems with only 15% distances available (i.e., 85% distances are uniformly randomly removed). These distances are further perturbed according to (64). For the PDB problems, both MBO and EMBED return comparable RMSD, but MBO is significantly faster. Because MBO is designed for anchor-free localization problems, it is not suitable for SNLa, SNLb, on which EMBED returns localizations of high accuracy (small RMSD). Table 4 is on problems with the known distances being selected by the “unit ball” model and being further perturbed according to (65). We also include the problem information such as density (Density column) and radio range (R column). It is known that the realistic `radiorange` values for the PDB data is 6 and acceptable `rmsd` value is 2 (see [25] for more comments on reasonable `rmsd`). EMBED performed satisfactorily on all tested problems except 1F39. By contrast, MBO failed on all tested problems.

Table 5 is on PDB problems with the known distances being selected by the “unit ball” model and being kept exact (no further noise is added). This time, MBO was successful with 1AU6, 1PHT (RMSD is about 2), but failed all other problems. Like in Table 4, EMBED was able to return satisfactory RMSD (with higher accuracy when compared to Table 4) for all problems except 1F39.

Our numerical results reported above seem to suggest that, when the known distances (exact or contaminated) are uniformly randomly selected, MBO tends to be faster than EMBED with both returning satisfactory RMSD. When the known distances are selected by the unit ball model, MBO tends to fail while EMBED is still capable of returning satisfactory RMSD (except 1F39 in the PDB problems).

Problems		MBO			EMBED		
Name	n	cpu	obj	rmsd	cpu	obj	rmsd
helix	501	2	778.68	1.31e-1	7	788.82	1.10e-1
SNLa	504	-	-	-	59	11.12	7.16e-3
SNLa	1004	-	-	-	277	45.67	5.19e-3
SNLa	1504	-	-	-	805	103.77	3.71e-3
SNLb	500	-	-	-	86	10.86	7.17e-3
SNLb	1000	-	-	-	349	45.38	5.56e-3
SNLb	1500	-	-	-	917	102.39	6.17e-3
1GM2	166	0.4	385.64	5.38e-1	0.8	438.62	6.98e-1
1PBM	388	1.3	1460.73	2.51e-1	6	1490.42	2.08e-1
1PTQ	402	1.3	2294.89	3.29e-1	6	2343.30	2.64e-1
1CTF	487	2	3367.91	3.16e-1	7	3423.92	2.56e-1
1AU6	506	2	2616.75	3.01e-1	8	2664.66	2.31e-1
1HOE	558	3	4221.12	3.28e-1	10	4285.64	2.60e-1
1PHT	814	7	6907.63	2.59e-1	28	6978.64	2.10e-1
1POA	914	8	10763.52	3.14e-1	35	10864.17	2.49e-1
1AX8	1003	11	11799.47	2.78e-1	49	11894.95	2.25e-1
1F39	1534	61	116488.38	9.72e-1	135	117304.86	6.25e-1
1GPV	1842	78	38232.52	2.89e-1	305	38423.28	2.28e-1
1RGS	2015	96	51295.98	3.21e-1	416	51531.39	2.45e-1

Table 3: Average results on 20 instances of each problem when 15% randomly chosen distances are available. The distances (may not be true distances) are further perturbed according to (64) with `nf` = 0.1. Problems SNLa, SNLb are not suitable for MBO.

Problems				MBO			EMBED		
Name	n	Density	R	cpu	obj	rmsd	cpu	obj	rmsd
helix	501	23.79%	5	8	988.92	5.31	7	753.91	1.64e-1
SNLa	504	36.23%	0.4	-	-	-	9	27.42	5.51e-3
SNLa	1004	33.99%	0.4	-	-	-	67	148.94	4.28e-3
SNLa	1504	34.98%	0.4	-	-	-	155	184.29	5.83e-3
SNLb	500	33.91%	0.4	-	-	-	8	30.85	4.46e-3
SNLb	1000	33.19%	0.4	-	-	-	50	82.29	3.11e-3
SNLb	1500	35.50%	0.4	-	-	-	174	79.67	2.52e-3
1GM2	166	27.06%	6	1.6	406.93	2.73	1.2	338.12	2.94e-1
1PBM	388	13.87%	6	5	711.61	4.38	4	581.29	2.43e-1
1PTQ	402	8.79%	6	5	821.93	8.07	6	491.87	3.75e-1
1CTF	487	7.25%	6	6	890.55	9.36	7	541.29	3.47e-1
1AU6	506	12.65%	6	8	954.64	4.99	7	760.04	2.02e-1
1HOE	558	6.55%	6	8	1023.82	10.01	10	591.56	3.16e-2
1PHT	814	5.35%	6	22	1213.77	8.53	22	790.04	9.44e-1
1POA	914	4.07%	6	17	1295.71	11.83	28	764.35	3.64e-1
1AX8	1003	3.74%	6	26	1292.56	11.17	41	803.88	3.58e-1
1F39	1534	2.16%	6	145	791.15	27.72	53	892.44	12.05
1GPV	1842	2.24%	6	89	1765.46	14.44	246	1148.53	1.19
1RGS	2015	1.87%	6	65	1999.56	18.63	303	1161.10	5.67e-1

Table 4: Average results on 20 instances of each problem when the unit ball model is used to select the known distances, which (may not be true distances) are further perturbed according to (65) with $\text{nf} = 0.1$. Problems SNLa, SNLb are not suitable for MBO.

Problems				MBO			EMBED		
Name	n	Density	R	cpu	obj	rmsd	cpu	obj	rmsd
1GM2	166	27.06%	6	2	318.54	3.82	1	1.65e-1	3.42e-3
1PBM	388	13.87%	6	5	589.13	5.81	6	1.33e-2	1.03e-3
1PTQ	402	8.79%	6	4	715.82	8.74	7	5.36e-1	7.88e-3
1CTF	487	7.25%	6	7	820.37	9.25	10	2.63e-1	4.51e-3
1AU6	506	12.65%	6	8	374.56	1.97	10	2.20e-2	4.37e-4
1HOE	558	6.55%	6	8	847.96	10.27	13	6.67e-2	1.11e-3
1PHT	814	5.35%	6	24	285.13	2.18	21	56.46	8.06e-1
1POA	914	4.07%	6	32	946.75	8.82	26	11.02	1.17e-1
1AX8	1003	3.74%	6	29	946.76	8.97	36	6.63	7.08e-2
1F39	1534	2.16%	6	354	154.07	21.93	53	420.45	12.16
1GPV	1842	2.24%	6	81	1263.38	15.16	218	73.92	9.84e-1
1RGS	2015	1.87%	6	83	1639.43	17.82	313	31.65	3.57e-1

Table 5: Average results on 20 instances of each problem when the unit ball model is used to select the known distances, which are kept exact (no further noise added).

7 Conclusion

EDMs with low embedding dimensions appear in many multi-disciplinary subjects. We touched three such applications: Euclidean embedding of Network coordinate systems, Sensor network localization, and Molecular conformation. In those applications, distances are missing (e.g., due to unavailability of network connections) or are inaccurate (e.g., due to measurement bias). A key problem in such situations is to calculate the nearest EDM to the known one under the required low embedding dimension constraint. This paper conducted both theoretical and numerical study of this nonconvex problem.

In theory, we studied the Lagrangian dual approach of the problem when the weights are uniform (i.e., $H = E$). A crucial geometrical object in this study is $\mathcal{K}_+^n(r)$, a counterpart of $\mathcal{S}_+^n(r)$. The metric projection onto $\mathcal{K}_+^n(r)$ plays a vital role in characterizing the *zero* duality gap result, which was further used to justify the majorization penalty approach adopted from Gao and Sun [16].

Numerically, we addressed a few key issues in applying the majorized penalty approach to our problem. In particular, a specific choice of the penalty function allows us to recycle the computational results of the semismooth Newton-CG method, which is used to solve the subproblem of the majorized penalty approach. We emphasize that the high efficiency of the Newton-CG method is the key to the numerical performance of the majorization method.

In algorithmic implementation, we took the useful experiences from the SDP solver. For example, the post-refinement step by the gradient descent method is taken from the SDP solver and it worked very well for the test problems. Most of them are from real applications. Another important technique used in the SDP solver is the regularization, which is a linear term added to the objective function in order to alleviate the “crowding” phenomenon also experienced with our method. However, our preliminary study on the exact form of the regularization term in our context is not conclusive. The technique is potentially extremely important in improving the overall embedding quality. We will explore this issue in our future work.

Acknowledgments. The authors would like to thank Prof. Kim-Chuan Toh for his help on using the SDP solver as well as on the PDB problems. They also like to thank Bamdev Mishra for useful discussions on the manifold-based-optimization solver [40] (denoted by MBO solver in this paper). The authors also wish to thank the three referees as well as the AE for their valuable comments and constructive suggestions, which have greatly improved the quality and the presentation of the paper.

References

- [1] A.Y. ALFAKIH, A. KHANDANI, AND H. WOLKOWICZ, *Solving Euclidean distance matrix completion problems via semidefinite programming*, Comput. Optim. Appl., 12 (1999), pp. 13–30.
- [2] H.M. BERMAN, J. WESTBROOK, Z. FENG, G. GILLILAN, T.N. BHAT, H. WEISSIG, I.N. SHINDYALOV, AND P.E. BOURNE, *The protein data bank*, Nucleic Acids Research 28 (2000), pp. 235–242.
- [3] P. BISWAS AND Y. YE, *Semidefinite programming for ad hoc wireless sensor network localization*, in Proceedings of the 3rd IPSN, Berkeley, CA, 2004, pp. 46–54.
- [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 Trans. on Automation Science and Engineering, 3 (2006), pp. 360–371.

- [5] I. BORG AND P.J.F. GROENEN, *Modern Multidimensional Scaling: Theory and Applications* (2nd ed.) Springer Series in Statistics, Springer, 2005.
- [6] L. CAYTON AND S. DASGUPTA, *Robust Euclidean embedding*, Proceeding of the 23rd International Conference on Machine Learning, pp. 169–176, Pittsburgh, PA, 2006.
- [7] D.I. CHU, H.C. BROWN, AND M.T. CHU, *On least squares Euclidean distance matrix approximation and completion*, Department of Mathematics, North Carolina State University, Tech. Rep., 2003.
- [8] F.H. CLARKE, *Optimization and Nonsmooth Analysis*, John Wiley & Sons, New York, 1983.
- [9] T.F. COX AND M.A.A. COX, *Multidimensional Scaling*, 2nd Ed, Chapman and Hall/CRC, 2001.
- [10] G. CRIPPEN AND T. HAVEL, *Distance Geometry and Molecular Conformation*, New York: Wiley.
- [11] J. DATTORRO, *Convex Optimization and Euclidean Distance Geometry*, Meboo Publishing USA, 2005.
- [12] J. DE LEEUW, *An alternating least squares approach to squared distance scaling*, unpublished manuscript, Department of Data Theory, University of Leiden, Leiden, the Netherlands, 1975.
- [13] R.L. DYKSTRA, *An algorithm for restricted least squares regression*, J. Amer. Statist. Assoc., 78 (1983), pp. 839–842.
- [14] N. GAFFKE AND R. MATHAR, *A cyclic projection algorithm via duality*, *Metrika*, 36 (1989), pp. 29–54.
- [15] Y. GAO, *Structured Low Rank Matrix Optimization Problems: a Penalty Approach*, PhD Thesis (2010), National University of Singapore.
- [16] Y. GAO AND D.F. SUN, *A majorized penalty approach for calibrating rank constrained correlation matrix problems*. Technical Report, Department of Mathematics, National University of Singapore, March 2010.
- [17] W. GLUNT, T.L. HAYDEN, S. HONG, AND J. WELLS, *An alternating projection algorithm for computing the nearest Euclidean distance matrix*, *SIAM J. Matrix Anal. Appl.*, 11 (1990), pp. 589–600.
- [18] W. GLUNT, T.L. HAYDEN, AND W.-M. LIU, *The embedding problem for predistance matrices*, *Bulletin of Mathematical Biology*, 53 (1991), pp. 769–796.
- [19] W. GLUNT, T.L. HAYDEN, AND R. RAYDAN, *Molecular conformations from distance matrices*, *J. Computational Chemistry*, 14 (1993), pp. 114–120.
- [20] J.C. GOWER, *Euclidean distance geometry*, *Math. Scientist*, 7 (1982), pp. 1–14.
- [21] J.C. GOWER, *Properties of Euclidean and non-Euclidean distance matrices*, *Linear Algebra Appl.*, 67 (1985), pp. 81–97.
- [22] S.P. HAN, *A successive projection method*, *Math. Programming*, 40 (1988), pp. 1–14.

- [23] T.L. HAYDEN AND J. WELLS, *Approximation by matrices positive semidefinite on a subspace*, Linear Algebra Appl., 109 (1988), pp. 115–130.
- [24] J.-B. HIRIART-URRUTY AND C. LEMARÉCHAL, *Convex Analysis and Minimization Algorithms I*, Springer-Verlag, Berlin 1993.
- [25] K.F. JIANG, D.F.SUN, AND K.-C. TOH, *An inexact accelerated proximal gradient method for large scale linearly constrained convex SDP*, SIAM J. Optim., 22 (2012), pp. 1042–1064.
- [26] K. JIANG, D. SUN, AND K.-C. TOH, *Solving nuclear norm regularized and semidefinite matrix least squares problems with linear quality constraints*, Fields Institute Communications Series on Discrete Geometry and Optimization, K. Bezdek, A. Deze, and Y. Ye eds (2013), pp. 133–162.
- [27] C.R. JOHNSON AND P. TARAZAGA, *Connections between the real positive semidefinite and distance matrix completion problems*, Linear Algebra Appl. 223/224 (1995), pp. 375–391.
- [28] S. KIM, M. KOJIMA, AND H. WAKI, *Exploiting sparsity in SDP relaxation for sensor network localization*. SIAM J. Optimization 20 (2009), pp. 192–215.
- [29] N. KRISLOCK AND H. WOLKOWICZ, *Explicit sensor network localization using semidefinite representations and facial reductions*, SIAM J. Optim. 20 (2010), pp. 2679–2708.
- [30] N. KRISLOCK AND H. WOLKOWICZ, *Euclidean distance matrices and applications*, In *Handbook of Semidefinite, Cone and Polynomial Optimization*, M. Anjos and J. Lasserre, eds (2012), pp. 879–914.
- [31] M LAURENT, *A connection between positive semidefinite and Euclidean distance matrix completion problems*, Linear Algebra Appl. 273 (1998), pp. 9–22.
- [32] M LAURENT, *Matrix completion problem*, in: C.A. Floudas, P.M. Pardalos (Eds.), *The Encyclopedia of Optimization*, vol. III, Kluwer, 2001, pp. 221–229.
- [33] C. LAVOR, L. LIBERTI, N. MACULAN, AND A. MUCHERINO, *Recent advances on the discretization molecular distance geometry problem*, European J. Operational Research 219 (2012), pp. 698–706.
- [34] S. LEE, Z. ZHANG, S. SAHU, AND D. SAHA, *On suitability of Euclidean embedding for host-based network coordinate systems*, IEEE/ACM Trans. on Networking, 18 (1), 2010, pp. 27–40.
- [35] L. LIBERTI, C. LAVOR, N. MACULAN, AND A. MUCHERINO, *Euclidean distance geometry and applications*, available from arXiv:1205:0349v1. May 3, 2012. To appear in: SIAM Review.
- [36] L. LIBERTI, C. LAVOR, A. MUCHERINO, AND N. MACULAN, *Molecular distance geometry methods: from continuous to discrete*, International Transactions in Operational Research 18 (2011), pp. 33–51.
- [37] R. MATHAR, *The best Euclidean fit to a given distance matrix in prescribed dimensions*, Linear Algebra and Its Applications 67 (1985), pp. 1–6.
- [38] W. MIAO, S. PAN, AND D. SUN, *A rank-corrected procedure for matrix completion with fixed basis coefficients*, Arxiv preprint arXiv:1210.3709, 2012.

- [39] W. MIAO, *Matrix completion procedure with fixed basis coefficients and rank regularized problems with hard constraints*, PhD thesis (2013), Department of Mathematics, National University of Singapore.
- [40] B. MISHRA, G. MEYER, AND R. SEPULCHRE, *Low-rank optimization for distance matrix completion*. In Proc. of 50th IEEE Con. Decis. Cont., December 2011.
- [41] J.J. MORÉ AND Z. WU, *Distance geometry optimization for protein structures*, J. Global Optimization 15 (1999), pp. 219–234.
- [42] T.E. NG AND H. ZHANG *Predicting Internet network distance with co-ordinates-based approaches*, In Proc. IEEE INFOCOM, New York, June 2002, 170–179.
- [43] H.-D. QI, *A semismooth Newton method for the nearest Euclidean distance matrix problem*, SIAM J. Matrix Analysis and Applications 34 (2013), pp. 67–93.
- [44] I.J. SCHOENBERG, *Remarks to Maurice Fréchet’s article “Sur la définition axiomatique d’une classe d’espaces vectoriels distanciés applicables vectoriellement sur l’espace de Hilbert”*, Ann. Math. 36 (1935), pp. 724–732.
- [45] J.B. TENENBAUM, V. DE SILVA, AND J.C. LANGFORD, *A global geometric framework for nonlinear dimensionality reduction*, Science 290 (2000), pp. 2319–2323.
- [46] K.C. TOH, *An inexact path-following algorithm for convex quadratic SDP*, Mathematical Programming, 112 (2008), pp. 221–254.
- [47] P. TSENG, *Second-order cone programming relaxation of sensor network localization*, SIAM J. Optimization 18 (2007), pp. 156–185.
- [48] M.W. TROSSET, *Distance matrix completion by numerical optimization*, Comput. Optim. Appl., 17 (2000), pp. 11–22.
- [49] G. YOUNG AND A.S. HOUSEHOLDER, *Discussion of a set of points in terms of their mutual distances*, Psychometrika 3 (1938), pp. 19–22.