

Euclidean Distance Matrix Correction with a Single Corrupted Element

by

Tina Chenrui Xu

A thesis
presented to the University of Waterloo
in fulfillment of the
thesis requirement for the degree of
Master of Mathematics
in
Combinatorics and Optimization

Waterloo, Ontario, Canada, 2024

© Tina Chenrui Xu 2024

Author's Declaration

This thesis consists of material all of which I authored or co-authored: see Statement of Contributions included in the thesis. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

I understand that my thesis may be made electronically available to the public.

Statement of Contribution

This thesis is largely based on the paper [5], which is joint work with Abdo Alfakih, Woosuk L. Jung, and Henry Wolkowicz. The research group worked on developing the algorithms together.

Abstract

In this thesis, we study the problem of correcting the error from a noisy Euclidean distance matrix (EDM). An EDM is a matrix where elements are squared distance between points in $\mathbb{R}^d$. We consider the special case where only one of the distance is corrupted.

We develop efficient algorithms to solve this problem, initially assuming that the points are in general position, and solve the problem using three different types of facial reduction: exposing vector, facial vectors, and Gale transform. Furthermore, we investigate yielding elements of an EDM and develop an algorithm for identifying one small principal submatrix with the embedding dimension d when many of the points are in a linear manifold of dimension smaller than d , allowing us to handle a more general problem. We present numerical experiments implemented in MATLAB, demonstrating the effectiveness of our solutions.

Acknowledgements

I express my greatest gratitude to my advisors, Prof. Henry Wolkowicz and Prof. Walaa Moursi, for their guidance throughout my studies. I would like to especially thank Prof. Wolkowicz for his patience and wisdom. I would also like to thank Prof. Abdo Alfakih for the recommendation he provided to me on this project. Finally, I would like to thank my partner Roger Gu. His support and love made this project possible.

Contents

Author's Declaration	ii
Statement of Contribution	iii
Abstract	iv
Acknowledgements	v
1 Introduction	1
1.1 Problem Description	2
1.2 Main Contributions and Outline	2
1.3 Preliminary Notation	3
2 Preliminaries	4
2.1 Affine and Convex Sets	4
2.2 Background in Linear Algebra	6
2.2.1 Generalized Inverse	7
2.2.2 Inertia and Interlacing Theorem	8
2.3 Semidefinite Optimization	9
2.3.1 Faces of $\mathbb{S}_+^n$	10
3 Euclidean Distance Matrix	12
3.1 The Basic of Euclidean Distance Matrices	12
3.1.1 Connection Between EDM and Positive Semidefinite Matrices	12
3.2 Gale Transform and Affine Dependency	16
3.2.1 Affine Dependency	16
3.2.2 General Position	18
3.3 Euclidean Distance Matrix Completion Problem	19

3.3.1	Solving EDMCP Using Facial Reduction	19
3.4	Yielding Entry of an EDM	25
3.4.1	Positive Semidefinite Interval of a Matrix Pencil	25
3.4.2	Yielding Interval of an EDM	26
3.4.3	Yielding Interval with Fixed Embedding Dimension	27
4	Solving the problem	31
4.1	Solving the Problem Under General Position Assumption	32
4.1.1	Bisection Method With Exposing Vector (BVE)	36
4.1.2	Multi-Block Method With Facial Vector (MBFV)	37
4.1.3	Small Blocks With Gale Transform (SBGT)	39
4.2	Solving the Problem Without General Position Assumption	40
4.2.1	Identifying a Good Block	42
4.2.2	Hard Case MBFV	46
4.2.3	An Example of a Problem with Multiple Solutions	47
5	Time Complexity and Numerics	49
5.1	Time Complexity	49
5.1.1	Runtime for BEV	49
5.1.2	Runtime for MBFV	50
5.1.3	Runtime for SBGT	50
5.1.4	Runtime for Identifying a Good Block	50
5.2	Numerics	51
6	Conclusion	54
	References	55

List of Tables

5.1	Average of 200 problems; $n = 1000$ to $20,000$, $d = 5$	51
5.2	Average of 1000 problems; n from 100 to 10000, d random positive integer between 2 and 10	52

List of Algorithms

1	Algorithm Framework	32
2	Small Block Algorithm	33
3	Completing D From the Facial Vector	34
4	Bisection Method With Exposing Vector	36
5	Multi-Block Facial Vector Algorithm	38
6	Gale Transform Algorithm	39
7	Hard Case Algorithm Framework - No General Position Assumption	41
8	Hard Case Small Block Algorithm - No General Position Assumption	42
9	Finding a Small Good Block	43
10	Hard Case Facial Vector Algorithm- No General Position Assumption	46

Chapter 1

Introduction

A Euclidean distance matrix (EDM) is a matrix where each element represents the squared distance between a pair of points in a Euclidean space. See Definition 1.1.1 below. This structure arises in various applications. For example, in sensor network localization the distances between wireless sensors can be measured when they are within a certain radio range. If enough distances are known, the position of the sensors in the network can then be determined using only the information of distances between the sensors, e.g., see [14, 20]. Another application is molecular confirmation where the distance between pairs of atoms are extracted using nuclear magnetic resonance, and the distances can be used to determine the structure of the molecule, e.g., see [12]. These problems fall into the area called *distance geometry*. Techniques in distance geometry have also been applied in machine learning, for example, in [29]. In machine learning, one can find an EDM in a low embedding dimension to detect the low dimensional structure from high dimensional data sets of images; and this is done while preserving the distance property between nearby images.

However, in practical scenarios, the data that populate these matrices can be noisy or incomplete, leading to inaccuracies that need addressing. This leads to the question being studied in this thesis:

Suppose we are given n points in $\mathbb{R}^d$ along with the distances between pairs of points but with exactly one of the distances *corrupted*/noisy. Can we identify the pair of points with the corrupted distance, and can we identify the magnitude of the noise?

Many related problems have been extensively studied. In [25], the fundamental result connecting positive semidefinite matrices with Euclidean distance matrices is established. A closely related problem is the Euclidean distance matrix completion problem (EDMCP), which focuses on filling in missing or unknown distances in a partially specified EDM. If we know the corrupted entry, then our problem can also be considered as an EDMCP. Classical results of EDMCP are studied [7]. Related results are given in [1, 6, 17]. In [20], the EDMCP and its application to sensor network localization problem is studied, and an algorithm is presented for solving EDMCP using facial reduction. Our algorithm is largely based on the result presented in [20], and we discuss results from it in detail in Section 3.3.1. In [22], the problem of locating the source of a single signal where distance between other points to this node is noisy has been studied. Results from this study are applicable to sensor network localization when the unknown node is a cellular phone and the other nodes are cellular towers. This has application in situations such as emergency response. Our problem is applicable to the case when the distance between one of the cellular towers and the phone is obstructed.

The special aspect of the problem we wish to solve is: firstly we do not know the position of the corrupted entry, and secondly we wish to find the magnitude of the noise exactly.

1.1 Problem Description

For $x \in \mathbb{R}^d$, we work with the Euclidean norm $\|x\| := \sqrt{\sum_{i=1}^d x_i^2}$. The Euclidean distance between two points $x, y \in \mathbb{R}^d$ is

$$\|x - y\| = \sqrt{\sum_{i=1}^d (x_i - y_i)^2}.$$

Let $\mathbb{S}^n \subseteq \mathbb{R}^n$ denote the set of $n \times n$ symmetric matrices.

Definition 1.1.1. $D \in \mathbb{S}^n$ is a Euclidean distance matrix (EDM) if

$$\exists p_1, \dots, p_n \in \mathbb{R}^d \text{ such that } D_{ij} = \|p_i - p_j\|^2, \forall i, j \in \{1, \dots, n\}. \quad (1.1)$$

If D is an EDM, then the smallest d for which the above condition is possible is called the embedding dimension. $P := [p_1 \ \dots \ p_n]^T$ is called the configuration matrix of D .

Let $\mathbb{E}^n \subseteq \mathbb{S}^n$ denote the set of n by n Euclidean distance matrices.

Problem 1 (Find the position and magnitude of noise in an EDM with single corrupted element). Let $D \in \mathbb{E}^n$ with embedding dimension d , and let n be much greater than d . Let $i, j \in \{1, \dots, n\}$, $i \neq j$, and $\epsilon \neq 0$. Let $E_{ij} := e_i e_j^T + e_j e_i^T$ where e_i, e_j are the standard basis vector. Let

$$D_n = D + \epsilon E_{ij}.$$

The problem being addressed in this thesis is

Given the noisy EDM D_n and the embedding dimension d , find i, j and ϵ such that $D_n - \epsilon E_{ij}$ is an EDM with embedding dimension d .

1.2 Main Contributions and Outline

In this thesis we provide a solution to problem of identifying the position and magnitude of noise exactly in an EDM with a single corrupted entry with efficient algorithms. The main algorithms appear in Chapter 4. We first solve the problem under an assumption that points generating the original EDM are in general position in Section 4.1. Then, we relax this assumption and present a modified algorithm in Section 4.2, that solves the problem in a more general setting.

In Chapter 2, we present the relevant background in convex geometry, linear algebra, and semidefinite optimization.

In Chapter 3, we present the relevant theorems for solving the problem. Section 3.1 introduces the classical results in the study of EDM. Section 3.2 introduces the concept of Gale transform, and presents a way of identifying affine dependency using the Gale matrix. This section provides

the theoretical basis for algorithms presented in Section 4.2. In Section 3.3, we review some known results in solving EDMCP using facial reduction techniques, that provides the theoretical foundation for our algorithms, and connects Gale matrices with exposing vectors. In Section 3.4, we discuss the concept of yielding, which addresses the issue of when our problem is unsolvable. We survey results for matrix pencil (see Definition 3.4.2) and yielding interval of an EDM, and provide our own characterization.

Finally, in Chapter 5, we analyse the theoretical runtime in Section 5.1. We implement the algorithms in MATLAB, and we demonstrate the performance of our algorithms in numerical testing in Section 5.2.

1.3 Preliminary Notation

Let $\mathbb{S}^n \subseteq \mathbb{R}^{n \times n}$ denote the set of n by n real symmetric matrices. For $x, y \in \mathbb{R}^n$, the inner product between x and y is $\langle x, y \rangle := \sum_{i=1}^n x_i y_i = x^T y$. The Euclidean norm is the distance induced by this inner product. $\|x\| := \sqrt{\langle x, x \rangle}$. For matrices $A, B \in \mathbb{S}^n$, define the inner product as $\langle A, B \rangle := \text{trace}(AB)$.

A Euclidean space is a finite dimensional vector space equipped with inner product over the real numbers. We use $\mathbb{V}$ to denote a Euclidean space. Let e denote the vector of all ones of appropriate dimension, and let I represent the identity matrix.

For this thesis, we let $D_n \in \mathbb{S}^n$ denote the noisy EDM with exactly one noisy entry, and we use d to denote the embedding dimension. We use i_c, j_c to denote the corrupted index such that $i_c < j_c$, and let ϵ denote the noise such that $D_n - \epsilon E_{i_c, j_c}$ is an EDM with embedding dimension d .

For $i, j, n \in \mathbb{N}$ with $i \leq j$, we use $[i : j]$ to denote the list $i, i+1, \dots, j$, and we use $[n]$ to denote $\{1, \dots, n\}$. Let $\mathcal{I}, \mathcal{J}$ be lists of indices. We use the notation, $[\mathcal{I}, \mathcal{J}]$ to denote the list of indices in $\mathcal{I}$ followed by the list of indices in $\mathcal{J}$.

Let $A \in \mathbb{R}^{n \times n}$. A submatrix of A is a matrix obtained by removing some rows and some columns from A . Let $\mathcal{I}, \mathcal{J}$ be lists of indices with no repeated elements. We use $A(\mathcal{I}, \mathcal{J})$ to denote the submatrix of A using rows indexed by $\mathcal{I}$ and columns indexed by $\mathcal{J}$. Thus, $A(\mathcal{I}, \mathcal{J})$ is the submatrix obtained by deleting the rows with indices $[n] \setminus \mathcal{I}$, and deleting the columns with indices $[n] \setminus \mathcal{J}$. We use $A(\mathcal{I}, :)$ to denote the submatrix of A with rows indexed by $\mathcal{I}$ and all the columns. Similarly, $A(:, \mathcal{J})$ denote the submatrix of A with all the rows, and columns indexed by $\mathcal{J}$.

A *principal submatrix* of A is a square submatrix obtained from A by deleting the same indexed rows and columns. We use $A(\mathcal{I})$ to indicate a principal submatrix where both the rows and columns are indexed by $\mathcal{I}$. Thus, $A(\mathcal{I})$ is the submatrix obtained by deleting rows and columns with indices $[n] \setminus \mathcal{I}$. In this thesis, if $\mathcal{I}$ is a list of consecutive integers, so $\mathcal{I} = i, i+1, \dots, i+j$ for some $i, j \in \mathbb{N}$, then we call $A(\mathcal{I})$ a *consecutive principal submatrix*. We introduce further notation as needed.

Chapter 2

Preliminaries

In this chapter, we introduce the relevant background in convex geometry, linear algebra, and semidefinite programming.

2.1 Affine and Convex Sets

In this section, we introduce some basic definitions for affine spaces and convex geometry. For reference, see the textbook[23].

A set A is *affine* if the line passing through any two points of A lies entirely in A . In other words, for all $x, y \in A$ and $\lambda \in \mathbb{R}$, $\lambda x + (1 - \lambda)y \in A$. All linear subspaces are affine sets, and affine sets which contain the origin are linear subspaces of $\mathbb{R}^n$.

If $A \subseteq \mathbb{R}^n$ is an affine set and $x \in \mathbb{R}^n$, then the translation of A by x , given by $A + x = \{a + x : a \in A\}$, is an affine set as well. Two affine sets A, B are said to be *parallel* if $B = A + x$ for some $x \in \mathbb{R}^n$.

Every nonempty affine set A is parallel to a unique linear subspace L given by $L = \{x - y : x \in A, y \in A\}$. The dimension of a nonempty affine set is the dimension of the unique subspace parallel to it.

Affine sets with dimension 0, 1, and 2 are called points, lines, and planes, respectively. An affine set with dimension $n - 1$ in $\mathbb{R}^n$ is called a *hyperplane*, and can be characterized as

$$H = \{x : \langle x, b \rangle = \beta\}$$

for some $0 \neq b \in \mathbb{R}^n, \beta \in \mathbb{R}$. Every hyperplane may be represented in this way with b and β unique up to a common nonzero multiple.

The *affine hull* of A , denoted by $\text{affhull}(A)$ is the smallest affine set containing A , and can be represented by

$$\text{affhull}(A) = \left\{ \sum_{i=1}^m \lambda_i a_i : m \geq 1, a_i \in A, \lambda_i \in \mathbb{R}, \sum_{i=1}^m \lambda_i = 1 \right\}. \quad (2.1)$$

The vector sum $\sum_{i=1}^m \lambda_i a_i$ where $\sum_{i=1}^m \lambda_i = 1$ is called an *affine combination* of $a_1, \dots, a_m$.

A set of points $\{a_1, \dots, a_m\}$ is *affinely independent* if $\dim(\text{affhull}(\{a_1, \dots, a_m\})) = m - 1$. We say $\{a_1, \dots, a_m\}$ is *affinely dependent*, if it is not affinely independent. Note that

$$\text{affhull}(\{a_1, \dots, a_m\}) = L + a_1,$$

where

$$L = \text{affhull}(\{0, a_2 - a_1, \dots, a_m - a_1\}) = \text{span}(\{0, a_2 - a_1, \dots, a_m - a_1\}).$$

Thus, $\{a_1, \dots, a_m\}$ is affinely independent if, and only if, $\{a_2 - a_1, \dots, a_m - a_1\}$ is linearly independent. This connection implies that many facts about linear independence can be applied to affine independence. For example, any m -dimensional affine set can be expressed as the affine hull of $m + 1$ points.

There is another connection between affine independence and linear independence by the following proposition.

Proposition 2.1.1. *Let $a_1, \dots, a_m \in \mathbb{R}^n$. $\{a_1, \dots, a_m\}$ is affinely independent if, and only if, $\left\{\begin{pmatrix} a_1 \\ 1 \end{pmatrix}, \dots, \begin{pmatrix} a_m \\ 1 \end{pmatrix}\right\}$ are linearly independent in $\mathbb{R}^{n+1}$.*

Proof. $\{a_1, \dots, a_m\}$ is affinely dependent if, and only if, $\{a_2 - a_1, \dots, a_m - a_1\}$ is linearly dependent, which is true if, and only if, there exists $c_2, \dots, c_m \in \mathbb{R}$ not all equal to zero such that

$$\begin{aligned} & c_2(a_2 - a_1) + \dots + c_m(a_m - a_1) = 0 \\ \iff & (-c_2 - \dots - c_m)a_1 + c_2a_2 + \dots + c_ma_m = 0 \\ \iff & \exists c_1, \dots, c_m \text{ not all zero, } c_1a_1 + c_2a_2 + \dots + c_ma_m = 0, \text{ such that } \sum_{i=1}^m c_i = 0 \\ \iff & \exists c_1, \dots, c_m \text{ not all zero, } c_1 \begin{pmatrix} a_1 \\ 1 \end{pmatrix} + c_2 \begin{pmatrix} a_2 \\ 1 \end{pmatrix} + \dots + c_m \begin{pmatrix} a_m \\ 1 \end{pmatrix} = 0, \end{aligned}$$

which is equivalent to $\left\{\begin{pmatrix} a_1 \\ 1 \end{pmatrix}, \dots, \begin{pmatrix} a_m \\ 1 \end{pmatrix}\right\}$ is linearly dependent. □

Corollary 2.1.2. *Let $A \subseteq \mathbb{R}^n$ be nonempty, and $\bar{a} \in A$. Then,*

$$\dim(\text{affhull}(A)) = \dim(\text{span}(\{a - \bar{a} : a \in A\})) = \dim\left(\text{span}\left(\left\{\begin{pmatrix} a \\ 1 \end{pmatrix} : a \in A\right\}\right)\right) - 1.$$

Proof. $\text{affhull}(A) = L + \bar{a}$ where $L = \text{span}(\{a - \bar{a} : a \in A\})$. Thus, $\dim(\text{affhull}(A)) = \dim(\text{span}(\{a - \bar{a} : a \in A\}))$. We know that for any $\bar{a}, a_1, \dots, a_m \in \mathbb{R}^n$,

$$\begin{aligned} & \{a_1 - \bar{a}, \dots, a_m - \bar{a}\} \text{ is linearly independent} \\ \iff & \{\bar{a}, a_1, \dots, a_m\} \text{ is affinely independent} \\ \iff & \left\{\begin{pmatrix} \bar{a} \\ 1 \end{pmatrix}, \begin{pmatrix} a_1 \\ 1 \end{pmatrix}, \dots, \begin{pmatrix} a_m \\ 1 \end{pmatrix}\right\} \text{ is linearly independent,} \end{aligned}$$

by Proposition 2.1.1. Thus, $\{a_1 - \bar{a}, \dots, a_k - \bar{a}\}$ is a maximal size linearly independent set in $\text{span}(\{a - \bar{a} : a \in A\})$ if, and only if, $\left\{\begin{pmatrix} \bar{a} \\ 1 \end{pmatrix}, \begin{pmatrix} a_1 \\ 1 \end{pmatrix}, \dots, \begin{pmatrix} a_k \\ 1 \end{pmatrix}\right\}$ is a maximal size linearly independent

set in $\text{span} \left(\left\{ \begin{pmatrix} a \\ 1 \end{pmatrix} : a \in A \right\} \right)$. Thus, $\dim(\text{span}(\{a - \bar{a} : a \in A\})) = \dim \left(\text{span} \left(\left\{ \begin{pmatrix} a \\ 1 \end{pmatrix} : a \in A \right\} \right) \right) - 1$. $\square$

A set $A \subseteq \mathbb{V}$ is *convex* if for all $x, y \in A$ and $0 < \lambda < 1$, $\lambda x + (1 - \lambda)y \in A$. The convex hull of A denoted $\text{conv}(A)$ is the smallest convex set containing A and can be expressed as

$$\text{conv}(A) = \left\{ \sum_{i=1}^m \lambda_i a_i : m \geq 1, a_i \in A, \sum_{i=1}^m \lambda_i = 1, \lambda_i \geq 0 \right\}. \quad (2.2)$$

The vector sum $\lambda_1 a_1 + \dots, \lambda_m a_m$ where $\sum_{i=1}^m \lambda_i = 1, \lambda_i \geq 0$ is called a *convex combination* of $a_1, \dots, a_m$.

Let C be a convex set, the *relative interior* of C is

$$\text{relint}(C) := \{x \in \text{affhull}(C) : \exists \epsilon > 0, \mathcal{B}(x, \epsilon) \cap \text{affhull}(C) \subseteq C\},$$

where $\mathcal{B}(x, \epsilon) := \{y \in \mathbb{V} : \|x - y\| < \epsilon\}$ is the open ball centred at x with radius ϵ .

A set $K \subseteq \mathbb{V}$ is called a *cone* if for all $x \in K$ and $\lambda \geq 0$, $\lambda x \in K$. Note that some definition only require a cone to be closed under positive scalar multiplication, but we will require the origin to be in a cone as well. A convex cone is a cone that is convex. A set $K \subseteq \mathbb{V}$ is a convex cone if, and only if, it is closed under addition and non negative scalar multiplication.

Let C be a convex set, a *supporting hyperplane* of C is a hyperplane

$$H = \{x : \langle x, b \rangle = \beta\},$$

with $b \neq 0$, where $\langle x, b \rangle \geq \beta$ for every $x \in C$ and $H \cap C \neq \emptyset$.

2.2 Background in Linear Algebra

Let $\mathbb{V}_1, \mathbb{V}_2$ be Euclidean spaces. For a linear transformation $\mathcal{A} : \mathbb{V}_1 \rightarrow \mathbb{V}_2$, its *adjoint* is $\mathcal{A}^* : \mathbb{V}_2 \rightarrow \mathbb{V}_1$ such that

$$\langle \mathcal{A}^*(y), x \rangle_{\mathbb{V}_1} = \langle y, \mathcal{A}(x) \rangle_{\mathbb{V}_2}, \forall x \in \mathbb{V}_1, y \in \mathbb{V}_2.$$

Let $\mathbb{V}$ be a Euclidean space, and $S \subseteq \mathbb{V}$. The orthogonal complement of S is

$$S^\perp := \{x \in \mathbb{V} : \langle x, s \rangle = 0 \forall s \in S\}.$$

For a vector $a \in \mathbb{V} \setminus \{0\}$, we also use $a^\perp$ to denote $\{x \in \mathbb{V} : \langle x, a \rangle = 0\}$.

For $S \subseteq \mathbb{V}$ be an linear subspace, then $(S^\perp)^\perp = S$. Furthermore, $\mathbb{V}$ can be decomposed as $\mathbb{V} = S \oplus S^\perp$. Any $x \in \mathbb{V}$ can be uniquely decomposed as $x = x_S + x_{S^\perp}$ where $x_S \in S$, $x_{S^\perp} \in S^\perp$. We define the *orthogonal projection* $\mathcal{P}_S$ onto S by a linear map such that $\mathcal{P}_S(x) = x_S$. The orthogonal projection has property that $\mathcal{P}_S(x) = x$ if $x \in S$, and $\mathcal{P}_S(x) = 0$ if $x \in S^\perp$. For any orthogonal projection $\mathcal{P}$, $\mathcal{P} = \mathcal{P}_{\text{range}(\mathcal{P})}$, $\mathcal{P} = \mathcal{P}^2$, $\mathcal{P} = \mathcal{P}^*$, and $I - \mathcal{P}$ is the projection onto $\text{range}(\mathcal{P})^\perp$.

Proposition 2.2.1 (Fundamental theorem of linear algebra, [27]). *Let $\mathcal{A} : \mathbb{V}_1 \rightarrow \mathbb{V}_2$ be a linear transformation, then*

$$\text{range}(\mathcal{A})^\perp = \text{null}(\mathcal{A}^*) \quad \text{null}(\mathcal{A})^\perp = \text{range}(\mathcal{A}^*),$$

and $\text{rank}(\mathcal{A}) = \text{rank}(\mathcal{A}^*)$.

Proof. Since $y \in \text{range}(\mathcal{A})^\perp$ if, and only if, $\langle y, \mathcal{A}x \rangle = \langle \mathcal{A}^*y, x \rangle = 0$ for all $x \in \mathbb{V}_1$, which occurs if, and only if, $\mathcal{A}^*y = 0$. Similarly, $x \in \text{range}(\mathcal{A}^*)^\perp$ if, and only if, $\langle x, \mathcal{A}^*y \rangle = \langle \mathcal{A}x, y \rangle = 0$ for all $y \in \mathbb{V}_2$, which occurs if, and only if, $\mathcal{A}x = 0$. Therefore, $\text{rank}(\mathcal{A}^*) = \dim(\mathbb{V}_2) - \text{nullity}(\mathcal{A}^*) = \text{rank}(\mathcal{A})$. $\square$

Let $X \in \mathbb{S}^n$, then all of its eigenvalues are real. Let $\lambda(X) := [\lambda_1(X), \dots, \lambda_n(X)]^T$ denote the vector formed by eigenvalues of X , where $\lambda_1(X) \geq \lambda_2(X) \geq \dots \geq \lambda_n(X)$ are the eigenvalues of X .

Let $\text{diag} : \mathbb{S}^n \rightarrow \mathbb{R}^n$ denote the linear map given by

$$\text{diag}(X) := [X_{11}, \dots, X_{nn}]^T.$$

Its adjoint is $\text{diag}^* = \text{Diag} : \mathbb{R}^n \rightarrow \mathbb{S}^n$ given by

$$\text{Diag}(x) := \begin{bmatrix} x_1 & 0 & \cdots & 0 \\ 0 & x_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & x_n \end{bmatrix}.$$

For every $X \in \mathbb{S}^n$, there exists orthogonal $Q \in \mathbb{R}^{n \times n}$ ($Q^T Q = Q Q^T = I$) such that

$$X = Q \text{Diag}(\lambda(X)) Q^T.$$

Moreover, we define the map $\mathcal{P}_{\mathcal{I}} : \mathbb{S}^n \rightarrow \mathbb{S}^{|\mathcal{I}|}$,

$$\mathcal{P}_{\mathcal{I}}(A) := A(\mathcal{I}).$$

The inverse image is

$$\mathcal{P}_{\mathcal{I}}^{-1}(\bar{A}) := \{A \in \mathbb{S}^n : A(\mathcal{I}) = \bar{A}\}.$$

The adjoint map is $\mathcal{P}_{\mathcal{I}}^* : \mathbb{S}^{|\mathcal{I}|} \rightarrow \mathbb{S}^n$ given by

$$[\mathcal{P}_{\mathcal{I}}^*(\bar{A})]_{ij} = \begin{cases} \bar{A}_{ij}, & \text{if } i \in \mathcal{I}, j \in \mathcal{I} \\ 0, & \text{otherwise.} \end{cases}$$

2.2.1 Generalized Inverse

In this part, we introduce the concept of a *generalized inverse*, or the Moore-Penrose pseudoinverse. For reference on generalized inverse, see [10].

Definition 2.2.2 (Moore-Penrose pseudoinverse). *Let $\mathcal{A} : \mathbb{V}_1 \rightarrow \mathbb{V}_2$ be a linear map. The Moore-Penrose pseudoinverse of $\mathcal{A}$ denoted by $\mathcal{A}^\dagger$ is the linear map $\mathcal{A}^\dagger : \mathbb{V}_2 \rightarrow \mathbb{V}_1$ satisfying*

$$1. \mathcal{A}\mathcal{A}^\dagger = \mathcal{P}_{\text{range}(\mathcal{A})}$$

$$2. \mathcal{A}^\dagger\mathcal{A} = \mathcal{P}_{\text{range}(\mathcal{A}^\dagger)}$$

Every linear map $\mathcal{A}$ has a unique map $\mathcal{A}^\dagger$ that satisfy the above equations. Furthermore, the generalized inverse of $\mathcal{A}$ can also be equivalently defined as the linear transformation $\mathcal{A}^\dagger : \mathbb{V}_2 \rightarrow \mathbb{V}_1$ such that $\mathcal{A}^\dagger x = 0$ if $x \in \text{range}(\mathcal{A})^\perp$, and $\mathcal{A}^\dagger x = (\mathcal{A}|_{\text{range}(\mathcal{A}^*)})^{-1} x$ if $x \in \text{range}(\mathcal{A})$.

Then, $\mathcal{A}\mathcal{A}^\dagger x = 0$ if $x \in \text{range}(\mathcal{A})^\perp$, and $\mathcal{A}\mathcal{A}^\dagger x = x$ if $x \in \text{range}(\mathcal{A})$. Similarly, $\mathcal{A}^\dagger\mathcal{A}x = 0$ if $x \in \text{null}(\mathcal{A}) = \text{range}(\mathcal{A}^*)^\perp$, and $\mathcal{A}^\dagger\mathcal{A}x = x$ if $x \in \text{range}(\mathcal{A}^*) = \text{range}(\mathcal{A}^\dagger)$.

In case of real matrices, we can calculate the Moore-Penrose pseudoinverse of A by the singular value decomposition. Let $A = UDV$ be the singular value decomposition of A , so D is a pseudo-diagonal matrix. Then, the pseudoinverse of A can be calculated as $A^\dagger = V^T D^\dagger U^T$. $D^\dagger$ is obtained by replacing the nonzero entries in D with their multiplicative inverses.

The above definition was given by Moore in 1935. An equivalent definition by Penrose in 1955 is the map $\mathcal{A}^\dagger$ that satisfies the Penrose equations:

$$1. \mathcal{A}\mathcal{A}^\dagger\mathcal{A} = \mathcal{A}.$$

$$2. \mathcal{A}^\dagger\mathcal{A}\mathcal{A}^\dagger = \mathcal{A}^\dagger.$$

$$3. \mathcal{A}\mathcal{A}^\dagger = (\mathcal{A}\mathcal{A}^\dagger)^*.$$

$$4. \mathcal{A}^\dagger\mathcal{A} = (\mathcal{A}^\dagger\mathcal{A})^*.$$

Proposition 2.2.3. *Let $\mathcal{A} : \mathbb{V}_1 \rightarrow \mathbb{V}_2$ be a linear operator. If $\mathcal{A}$ is injective, then $\mathcal{A}^\dagger$ is a left inverse of $\mathcal{A}$. If $\mathcal{A}$ is surjective, then $\mathcal{A}^\dagger$ is a right inverse of $\mathcal{A}$.*

Proof. If $\mathcal{A}$ is injective, then $\text{null}(\mathcal{A}) = \{0\}$ and $\text{range}(\mathcal{A}^\dagger) = \text{range}(\mathcal{A}^*) = \mathbb{V}_1$. Thus,

$$\mathcal{A}^\dagger\mathcal{A} = \mathcal{P}_{\text{range}(\mathcal{A}^\dagger)} = \mathcal{P}_{\mathbb{V}_1} = \text{Id}_{\mathbb{V}_1},$$

where $\text{Id}_{\mathbb{V}_1}$ is the identity operator on $\mathbb{V}_1$. If $\mathcal{A}$ is surjective, then

$$\mathcal{A}\mathcal{A}^\dagger = \mathcal{P}_{\text{range}(\mathcal{A})} = \mathcal{P}_{\mathbb{V}_2} = \text{Id}_{\mathbb{V}_2},$$

where $\text{Id}_{\mathbb{V}_2}$ is the identity operator on $\mathbb{V}_2$. □

2.2.2 Inertia and Interlacing Theorem

In this section, we introduce several useful results in matrix analysis. See [19] for a detailed reference on the topic.

Definition 2.2.4. *Let $A, B \in \mathbb{S}^n$. If there exists a nonsingular matrix Q such that $B = QAQ^T$, then B is said to be congruent with A , and we write $A \cong B$.*

Definition 2.2.5 (Inertia). *Let $A \in \mathbb{S}^n$, the inertia of A is the ordered triple*

$$i(A) = (i_+(A), i_-(A), i_0(A)),$$

where $i_+(A)$ is the number of positive eigenvalues of A ; $i_-(A)$ is the number of negative eigenvalues of A , and $i_0(A)$ is the number of zero eigenvalues of A .

Theorem 2.2.6 (Sylvester's law of inertia). *$A, B \in \mathbb{S}^n$ are congruent if, and only if, they have the same inertia.*

Definition 2.2.7 (Schur complement). *Let $A \in \mathbb{S}^n$ and partition A as*

$$A = \begin{bmatrix} B & C \\ C^T & D \end{bmatrix}$$

where B is nonsingular. $D - C^T B^{-1} C$ is the Schur complement of B in A .

Let A be the same as in the above definition. Notice that

$$\begin{bmatrix} B & C \\ C^T & D \end{bmatrix} = \begin{bmatrix} I & 0 \\ C^T B^{-1} & I \end{bmatrix} \begin{bmatrix} B & 0 \\ 0 & D - C^T B^{-1} C \end{bmatrix} \begin{bmatrix} I & B^{-1} C \\ 0 & I \end{bmatrix}. \quad (2.3)$$

Therefore, $A \cong \begin{bmatrix} B & 0 \\ 0 & D - C^T B^{-1} C \end{bmatrix}$.

Theorem 2.2.8 (Eigenvalue interlacing theorem). *Let $A \in \mathbb{S}^n$ and partition A as*

$$A = \begin{bmatrix} B & C \\ C^T & D \end{bmatrix}$$

where $B \in \mathbb{S}^m$. Then for all $i = 1, \dots, m$,

$$\lambda_{i+n-m}(A) \leq \lambda_i(B) \leq \lambda_i(A).$$

2.3 Semidefinite Optimization

In this section, we provide the definition for semidefinite optimization, and some facts about positive semidefinite matrices. Additionally, we introduce the well known structure of the positive semidefinite cone. For reference, see the following textbook on semidefinite optimization [28].

A symmetric matrix $A \in \mathbb{S}^n$ is *positive semidefinite* if for all $x \in \mathbb{R}^n$, $x^T A x \geq 0$. It is *positive definite* if for all $x \in \mathbb{R}^n$ such that $x \neq 0$, $x^T A x > 0$. We use $\mathbb{S}_+^n$ and $\mathbb{S}_{++}^n$ to denote the sets of positive semidefinite and positive definite matrices in $\mathbb{S}^n$, respectively. For $A, B \in \mathbb{S}^n$, we write $A \succeq B$ if $A - B \in \mathbb{S}_+^n$. The set of positive semidefinite matrices is a cone of the vector space $\mathbb{S}^n$.

Proposition 2.3.1 (Schur's complement theorem). *Let $A = \begin{bmatrix} B & C \\ C^T & D \end{bmatrix}$, where B is positive definite. A is positive definite if and only if the Schur complement $D - C^T B^{-1} C$ is positive definite. A is positive semidefinite if and only if $D - C^T B^{-1} C$ is positive semidefinite.*

Proof. This follows from equation (2.3). □

Proposition 2.3.2 (Proposition 1.11, [28]). *For $X \in \mathbb{S}^n$, the following are equivalent*

1. X is positive semidefinite;
2. $\lambda_i(X) \geq 0$ for every $i \in \{1, \dots, n\}$;

3. there exists $\mu \in \mathbb{R}_+^n$, $h^{(i)} \in \mathbb{R}^n$ for $i \in \{1, \dots, n\}$ such that

$$X = \sum_{i=1}^n \mu_i h^{(i)} h^{(i)T};$$

4. there exists $B \in \mathbb{R}^n$ such that $X = BB^T$;

5. for every nonempty $\mathcal{J} \subseteq \{1, \dots, n\}$, $\det(X(\mathcal{J})) \geq 0$;

6. for every $S \in \mathbb{S}_+^n$, $\langle X, S \rangle \geq 0$.

The number of nonzero eigenvalues of $X \in \mathbb{S}_+^n$ is equal to the rank of X . In part 4., if $\text{rank}(X) = r$, then there exists $B \in \mathbb{R}^{n \times r}$ such that $X = BB^T$.

Let $C \in \mathbb{S}^n$, $b \in \mathbb{R}^m$, $\mathcal{A} : \mathbb{S}^n \rightarrow \mathbb{R}^m$ be linear. The semidefinite programming problem in standard form is

$$(P) \quad \begin{aligned} & \inf \quad \langle C, X \rangle \\ & \text{subject to} \quad \mathcal{A}(X) = b \ . \\ & \quad \quad \quad X \succeq 0 \end{aligned}$$

Its dual is

$$(D) \quad \begin{aligned} & \sup \quad b^T y \\ & \text{subject to} \quad \mathcal{A}^*(y) + S = C \ . \\ & \quad \quad \quad S \succeq 0 \end{aligned}$$

2.3.1 Faces of $\mathbb{S}_+^n$

If $K \subseteq \mathbb{V}$ is a convex cone, we say that $F \subseteq K$ is a face of K , denoted $F \trianglelefteq K$ if F is a convex cone, and for all $x, y \in K$ such that $x + y \in F$, we have that $x, y \in F$. If $F \neq \emptyset$ and $F \neq K$, then we say F is a *proper face* of K , and denote it by $F \triangleleft K$.

For $A \subseteq K$, the minimal face of K containing A is denoted by $\text{face}_K(A)$. Sometime we may drop the subscript and write $\text{face}(A)$ if the context is clear.

$F \trianglelefteq K$ is an *exposed face* if $F = K \cap H$ for some supporting hyperplane H of K . Thus, F is exposed if there exists $b \neq 0$ such that

$$F = \{x \in K, \langle x, b \rangle = 0\}.$$

In this case, we say that b exposes F , or that b is an *exposing vector*. Notice that

$$F = K \cap b^\perp.$$

A cone K is *facially exposed* if every face of K is exposed.

Proposition 2.3.3. For $i = 1, \dots, n$, let $F_i \trianglelefteq \mathbb{S}_+^n$ and suppose Y_i exposes F_i . Then $\sum_{i=1}^n Y_i$ exposes $\cap_{i=1}^n F_i$.

Proof. If $X \in \cap_{i=1}^n F_i$, then $\forall i \in \{1, \dots, n\}$, $\langle X, Y_i \rangle = 0$, thus

$$\left\langle X, \sum_{i=1}^n Y_i \right\rangle = 0.$$

Conversely, suppose for some $X \in \mathbb{S}_+^n$, $\langle X, \sum_{i=1}^n Y_i \rangle = 0$. Since $Y_i, X \in \mathbb{S}_+^n$, $\langle X, Y_i \rangle \geq 0$ for all i . Thus,

$$\langle X, Y_i \rangle = 0 \quad \forall i \in \{1, \dots, n\}.$$

Therefore, $\cap_{i=1}^n F_i = \{X \in \mathbb{S}_+^n : \langle X, \sum_{i=1}^n Y_i \rangle = 0\}$, and $\sum_{i=1}^n Y_i$ exposes $\cap_{i=1}^n F_i$, as claimed. $\square$

It is well known that faces of the positive semidefinite cone can be characterized by the null space or the column space of matrices in the face. Some reference include [8, 30]. The following theorems list several properties about faces of the semidefinite cone. Proof can be found in [16, 21].

Theorem 2.3.4. *Let $Q = [U \ V] \in \mathbb{R}^{n \times n}$ be orthogonal, where $U \in \mathbb{R}^{n \times k}$, and let*

$$F = U\mathbb{S}_+^k U^T = \left\{ Q \begin{bmatrix} A & 0 \\ 0 & 0 \end{bmatrix} Q^T : A \in \mathbb{S}_+^k \right\}.$$

Then F is a face of $\mathbb{S}_+^n$, and

1. $F = \{X \in \mathbb{S}_+^n : \text{range}(X) \subseteq \text{range}(U)\}.$
2. $F = \{X \in \mathbb{S}_+^n : \text{null}(X) \supseteq \text{range}(V)\}.$
3. $\text{relint}(F) = U\mathbb{S}_{++}^k U^T = \{X \in \mathbb{S}_+^n : \text{range}(X) = \text{range}(U)\}.$
4. $F = \mathbb{S}^n \cap Y^\perp$ for any $Y \in \mathbb{S}_+^n$ with $\text{range}(Y) = \text{range}(V)$. Thus, Y is an exposing vector.

Theorem 2.3.5. *Suppose $F \trianglelefteq \mathbb{S}_+^n$, and $X \in \text{relint}(F)$. Let $\text{range}(U) = \text{range}(X)$, $\text{range}(V) = \text{null}(X)$, $Q = [U \ V] \in \mathbb{R}^{n \times n}$ be orthogonal. Then,*

$$F = U\mathbb{S}_+^k U^T = \left\{ Q \begin{bmatrix} A & 0 \\ 0 & 0 \end{bmatrix} Q^T : A \in \mathbb{S}_+^k \right\}. \quad (2.4)$$

Thus, every nonempty face $F \trianglelefteq \mathbb{S}_+^n$ has the form $F = U\mathbb{S}_+^k U^T$ for some $U \in \mathbb{R}^{n \times k}$. The face F can also be written as $\mathbb{S}_+^n \cap Y^\perp$, with $Y \in \mathbb{S}_+^n$ such that $\text{range}(Y) = \text{range}(V)$. Thus, $\mathbb{S}_+^n$ is facially exposed.

Theorem 2.3.6. *For any $X \in \mathbb{S}_+^n$, let $k = \text{rank}(X)$, $U \in \mathbb{R}^{n \times k}$ be a full column rank matrix with $\text{range}(U) = \text{range}(X)$. Then we have*

$$\text{faces}_{\mathbb{S}_+^n}(X) = U\mathbb{S}_+^k U. \quad (2.5)$$

U is called a facial vector for the face $F = \text{faces}_{\mathbb{S}_+^n}(X)$.

Chapter 3

Euclidean Distance Matrix

In this chapter we introduce the relevant results in the study of Euclidean distance matrices.

3.1 The Basic of Euclidean Distance Matrices

Recall Definition 1.1.1 that $D \in \mathbb{S}^n$ is an EDM if there exists $p_1, \dots, p_n$ in a Euclidean space such that $D_{ij} = \|p_i - p_j\|^2$ for all $i, j \in [n]$. The embedding dimension d is the smallest integer d for the above condition to hold.

Note that $\dim(\text{affhull}(\{p_1, \dots, p_n\})) = d$. If $\dim(\text{affhull}(\{p_1, \dots, p_n\})) = \dim(\text{span}(\{0, p_2 - p_1, \dots, p_n - p_1\})) = k < d$, then $\text{span}(\{0, p_2 - p_1, \dots, p_n - p_1\})$ is a Euclidean space with dimension k . Thus, $\text{span}(\{0, p_2 - p_1, \dots, p_n - p_1\})$ is isomorphic to $\mathbb{R}^k$, and we can embed $p_1, \dots, p_n$ in $\mathbb{R}^k$.

Let $P = [p_1 \ \dots \ p_n]^T$ be the configuration matrix for the EDM D . If $P^T e = \sum_{i=1}^n p_i = 0$, then we say P is *centered*.

Notice that if we translate all the points by the same constant vector, the EDM formed by them remains the same. That is, if $\bar{P}^T = P^T - ve^T$, then both $\bar{P}$ and P are configuration matrices for D . In particular, if we let $v := \frac{1}{n}P^T e$, then

$$\begin{aligned}\bar{P}^T e &= P^T e - \frac{1}{n}P^T e(e^T e) \\ &= P^T e - \frac{n}{n}P^T e \\ &= 0.\end{aligned}$$

Therefore, without loss of generality, we assume that the configuration matrix of any EDM is centered. The vector $v = \frac{1}{n}P^T e$ is known as the *barycenter* of the points.

3.1.1 Connection Between EDM and Positive Semidefinite Matrices

There is an interesting connection between EDMs and positive semidefinite matrices.

Definition 3.1.1. Let $P = [p_1 \ \dots \ p_n]^T$. Then $G := PP^T$ is the corresponding Gram matrix.

Since $P^T e = 0$, we get $Ge = 0$, i.e., G is *centered*.

The Operator $\mathcal{K}$

Note that $G_{ij} = p_i^T p_j$. Let D be the EDM formed by P , then

$$\begin{aligned} D_{ij} &= \|p_i - p_j\|^2 \\ &= p_i^T p_i + p_j^T p_j - 2p_i^T p_j \\ &= G_{ii} + G_{jj} - 2G_{ij}. \end{aligned}$$

Therefore, we make the following definition.

Definition 3.1.2. *The Lindenstrauss operator $\mathcal{K} : \mathbb{S}^n \rightarrow \mathbb{S}^n$, is given by*

$$\mathcal{K}(G) := \text{diag}(G)e^T + e \text{diag}(G)^T - 2G. \quad (3.1)$$

$\mathcal{K}$ is clearly linear. If D is an EDM with configuration matrix P , then $\mathcal{K}(PP^T) = \mathcal{K}(G) = D$.

Let

$$\mathbb{S}_C^n := \{X \in \mathbb{S}^n : Xe = 0\}, \quad \mathbb{S}_H^n := \{X \in \mathbb{S}^n : \text{diag}(X) = 0\},$$

denote the *centered subspace* and the *hollow subspace*, respectively. Also define

$$\mathcal{D}_e(X) := \text{diag}(X)e^T + e \text{diag}(X)^T. \quad (3.2)$$

Hence, we can also write $\mathcal{K}$ as $\mathcal{K}(X) = \mathcal{D}_e(X) - 2X$. The adjoint of $\mathcal{K}$ is

$$\mathcal{K}^*(X) := 2(\text{Diag}(Xe) - X). \quad (3.3)$$

Proposition 3.1.3. *[1] We have*

$$\text{range}(\mathcal{K}) = \mathbb{S}_H^n, \quad \text{null}(\mathcal{K}) = \text{range}(\mathcal{D}_e), \quad \mathcal{K}(\mathbb{S}_C^n) = \mathbb{S}_H^n.$$

$$\text{range}(\mathcal{K}^*) = \text{null}(\mathcal{K})^\perp = \mathbb{S}_C^n \quad \text{null}(\mathcal{K}^*) = \text{range}(\mathcal{K})^\perp = \text{range}(\text{Diag}).$$

Proof. We provide a proof here for completeness.

Let $X \in \mathbb{S}^n$. Then $\mathcal{D}_e(X) = \text{diag}(X)e^T + e \text{diag}(X)^T$, so $\text{diag}(\mathcal{D}_e(X)) = 2 \text{diag}(X)$. Hence,

$$\mathcal{K}(\mathcal{D}_e(X)) = 2 \text{diag}(X)e^T + 2e \text{diag}(X)^T - 2(\text{diag}(X)e^T + e \text{diag}(X)^T) = 0.$$

Therefore, $\text{range}(\mathcal{D}_e) \subseteq \text{null}(\mathcal{K})$. Furthermore, $\mathcal{K}(X) = 0$ if, and only if, $2X_{ij} = X_{ii} + X_{jj}$ for all i, j . Thus, $\mathcal{K}(X) = 0$ if, and only if, $X = xe^T + ex^T$ where $x = \frac{1}{2} \text{diag}(X)$. This proves $\text{null}(\mathcal{K}) = \text{range}(\mathcal{D}_e)$.

For any $X \in \mathbb{S}^n$, $\mathcal{K}(X)_{ii} = X_{ii} + X_{ii} - 2X_{ii} = 0$. Therefore, $\text{range}(\mathcal{K}) \subseteq \mathbb{S}_H^n$. For any $Y \in \mathbb{S}_H^n$,

$$\mathcal{K}\left(-\frac{1}{2}Y\right) = \text{diag}\left(-\frac{1}{2}Y\right)e^T + e \text{diag}\left(-\frac{1}{2}Y\right)^T + Y = 0 + 0 + Y = Y.$$

This proves $\text{range}(\mathcal{K}) = \mathbb{S}_H^n$.

Let $Y \in \mathbb{S}_H^n$, consider $v = \frac{1}{4n}Ye$, and $X = -\frac{1}{2}Y + ve^T + ev^T$. Then $\mathcal{K}(X) = \mathcal{K}(-\frac{1}{2}Y) + \mathcal{K}(ve^T +$

$ev^T) = Y + 0 = Y$. Moreover,

$$\begin{aligned} \left(-\frac{1}{2}Y + ve^T + ev^T\right)e &= -\frac{1}{2}Ye + n\left(\frac{1}{4n}Ye\right) + e\left(\frac{1}{4n}Ye\right)^T e \\ &= -\frac{1}{2}Ye + \frac{1}{4}Ye + \frac{1}{4}Ye = 0. \end{aligned}$$

Thus, $(-\frac{1}{2}Y + ve^T + ev^T) \in \mathbb{S}_C^n$, proving $\mathcal{K}(\mathbb{S}_C^n) = \mathbb{S}_H^n$.

For any $Y \in \mathbb{S}^n$, $\mathcal{K}^*(Y)e = 2(\text{Diag}(Ye) - Y)e = 2(Ye - Ye) = 0$, so $\mathbb{S}_C^n \supseteq \text{range}(\mathcal{K}^*)$. By Proposition 2.2.1, $\text{range}(\mathcal{K}^*) = \text{null}(\mathcal{K})^\perp$. Hence, for any $X \in \mathbb{S}_C^n$ and $Y \in \mathbb{S}^n$,

$$\begin{aligned} \langle X, \text{diag}(Y)e^T + e \text{diag}(Y)^T \rangle &= \text{trace}(X \text{diag}(Y)e^T) + \text{trace}(Xe \text{diag}(Y)^T) \\ &= \text{trace}(e^T X \text{diag}(Y)) + \text{trace}(Xe \text{diag}(Y)^T) \\ &= 0 \end{aligned}$$

since $Xe = e^T X = 0$. This shows $\mathbb{S}_C^n \subseteq \text{range}(\mathcal{D}_e)^\perp = \text{range}(\mathcal{K}^*)$. Thus, $\text{range}(\mathcal{K}^*) = \mathbb{S}_C^n = \text{range}(\mathcal{D}_e)^\perp$.

Additionally, we observe that $\langle X, H \rangle = \text{trace}(XH) = 0$ for all $H \in \mathbb{S}_H^n$ if, and only if, X is a diagonal matrix. Thus, $\text{null}(\mathcal{K}^*) = \text{range}(\text{Diag})$, $\square$

The Generalized Inverse of $\mathcal{K}$

Now, we develop the generalized inverse for $\mathcal{K}$. Let

$$J := I - \frac{1}{n}ee^T \tag{3.4}$$

be the projection onto $e^\perp$, the orthogonal complement of e .

Let $\text{offDiag} : \mathbb{S}^n \rightarrow \mathbb{S}^n$

$$\text{offDiag}(D) := D - \text{Diag}(\text{diag}(D)).$$

In other words,

$$\text{offDiag}(D)_{ij} = \begin{cases} D_{ij} & \text{if } i \neq j, \\ 0 & \text{if } i = j. \end{cases}$$

which is the orthogonal projection onto $\mathbb{S}_H^n$.

Proposition 3.1.4. *The Moore-Penrose pseudoinverse of $\mathcal{K}$ is given by $\mathcal{K}^\dagger : \mathbb{S}^n \rightarrow \mathbb{S}^n$ [1],*

$$\mathcal{K}^\dagger(D) := -\frac{1}{2}J \text{offDiag}(D)J. \tag{3.5}$$

Proof. We include the proof for completeness, and demonstrate $-\frac{1}{2}J \text{offDiag}(D)J$ is indeed the generalized inverse of $\mathcal{K}$.

First, we show $\mathcal{K}|_{\mathbb{S}_C^n}$ and $\mathcal{K}^\dagger|_{\mathbb{S}_H^n}$ are inverses of each other [1], [13].

For $X \in \mathbb{S}_C^n$,

$$\begin{aligned}
JK(X)J &= \mathcal{K}(X) - \frac{1}{n}ee^T\mathcal{K}(X) - \frac{1}{n}\mathcal{K}(X)ee^T + \frac{1}{n^2}ee^T\mathcal{K}(X)ee^T \\
&= (\text{diag}(X)e^T + e\text{diag}(X)^T - 2X) - (\text{diag}(X)e^T + e\text{diag}(X)^T) \\
&\quad - \frac{2}{n}(\text{diag}(X)^Te)ee^T + \frac{2}{n}(\text{diag}(X)^Te)ee^T \\
&= -2X.
\end{aligned}$$

Thus, $\mathcal{K}^\dagger\mathcal{K}(X) = -\frac{1}{2}JK(X)J = X$.

For $Y \in \mathbb{S}_H^n$,

$$\begin{aligned}
\mathcal{K}^\dagger(Y) &= -\frac{1}{2}\left(Y - \frac{1}{n}Yee^T - \frac{1}{n}ee^TY + \frac{1}{n^2}ee^TYee^T\right) \\
\text{diag}(\mathcal{K}^\dagger(Y)) &= -\frac{1}{2}\left(0 - \frac{1}{n}Ye - \frac{1}{n}Ye + \frac{1}{n^2}(e^TYe)e\right) \\
&= \frac{1}{n}Ye - \frac{1}{2n^2}(e^TYe)e.
\end{aligned}$$

Thus,

$$\begin{aligned}
\mathcal{K}\mathcal{K}^\dagger(Y) &= \frac{1}{n}Yee^T - \frac{1}{2n^2}(e^TYe)ee^T + \frac{1}{n}e(Ye)^T - \frac{1}{2n^2}e(e^TYe)e^T - 2\mathcal{K}^\dagger(Y) \\
&= \frac{1}{n}Yee^T + \frac{1}{n}ee^TY - \frac{1}{n^2}ee^TYee^T - 2\mathcal{K}^\dagger(Y) \\
&= Y.
\end{aligned}$$

Therefore, $\mathcal{K}$ and $\mathcal{K}^\dagger$ are inverses when restricted to their ranges. We have $\mathcal{K}^\dagger\mathcal{K}(X) = X$ for every $X \in \text{range}(\mathcal{K}^*) = \mathbb{S}_C^n$, and $\mathcal{K}\mathcal{K}^\dagger(Y) = Y$ for every $Y \in \mathbb{S}_H^n$.

Next, $Je = 0$ so $\mathbb{S}_C^n \supseteq \text{range}(\mathcal{K}^\dagger)$, and the above calculation showed $\mathbb{S}_C^n \subseteq \text{range}(\mathcal{K}^\dagger)$. Thus, $\text{range}(\mathcal{K}^\dagger) = \mathbb{S}_C^n$. By Proposition 3.1.3, $\mathbb{S}_C^n^\perp = \text{null}(\mathcal{K})$, thus for any $X \in \mathbb{S}_C^n^\perp$, $\mathcal{K}^\dagger\mathcal{K}(X) = 0$.

Furthermore, for any $Y \in \text{range}(\text{Diag}) = \text{range}(\mathcal{K})^\perp$, $-\frac{1}{2}J\text{offDiag}(Y)J = 0$. Thus, $\mathcal{K}\mathcal{K}^\dagger(Y) = 0$ for every $Y \in \text{range}(\mathcal{K})^\perp = \mathbb{S}_H^n = \text{range}(\text{Diag})$.

We have $\mathcal{K}\mathcal{K}^\dagger = \mathcal{P}_{\text{range}(\mathcal{K})}$ and $\mathcal{K}^\dagger\mathcal{K} = \mathcal{P}_{\text{range}(\mathcal{K}^\dagger)}$. By Definition 2.2.2, $-\frac{1}{2}J\text{offDiag}(D)J$ is indeed the Moore-Penrose pseudoinverse of $\mathcal{K}$. $\square$

Moreover, note that $\mathcal{K}^\dagger\mathcal{K} = \mathcal{P}_{\text{range}(\mathcal{K}^\dagger)} = \mathcal{P}_{\mathbb{S}_C^n}$, $\mathcal{K}^\dagger\mathcal{K}(X) = JXJ$ for all $X \in \mathbb{S}^n$.

Proposition 3.1.5. [1] *We have that $\mathcal{K}$ and $\mathcal{K}^\dagger$ are bijective between $\mathbb{E}^n \subset \mathbb{S}_H^n$ and $\mathbb{S}_+^n \cap \mathbb{S}_C^n$. We have*

$$\mathcal{K}^\dagger(\mathbb{E}^n) = \mathbb{S}_+^n \cap \mathbb{S}_C^n, \quad \mathcal{K}(\mathbb{S}_+^n \cap \mathbb{S}_C^n) = \mathbb{E}^n. \quad (3.6)$$

Let $V \in \mathbb{R}^{n \times (n-1)}$ be such that columns of V forms an orthonormal basis of $e^\perp$. For every matrix in $X \in \mathbb{S}_C^n \cap \mathbb{S}_+^n$, $\text{range}(X) \subseteq \text{range}(V)$, thus, $\mathbb{S}_C^n \cap \mathbb{S}_+^n = V\mathbb{S}_+^{n-1}V^T$. Define $\mathcal{K}_V : \mathbb{S}^{n-1} \rightarrow \mathbb{S}^n$

$$\mathcal{K}_V(X) := \mathcal{K}(VXV^T), \quad (3.7)$$

then $\mathcal{K}_V(\mathbb{S}_+^{n-1}) = \mathbb{E}^n$.

The following theorem connects EDM with the Gram matrix through the operator $\mathcal{K}$.

Theorem 3.1.6. [17, 25] *A matrix $D \in \mathbb{S}_H^n$ is a Euclidean distance matrix if, and only if, $\mathcal{K}^\dagger(D) \in \mathbb{S}_+^n$. Moreover, the embedding dimension of D is given by $\text{rank}(\mathcal{K}^\dagger(D))$.*

3.2 Gale Transform and Affine Dependency

In this section, we introduce the concept of Gale transform. Gale transform is important in the study of convex polytopes [18], and is also valuable in the study of EDMs. In [3], the role Gale transform plays in studying of EDMs is discussed in details. In addition, Gale transform provides information about affine dependency between points in the configuration matrix.

Definition 3.2.1. *Let $D \in \mathbb{E}^n$, and let d be its embedding dimension, P be its configuration matrix. The Gale space of D , denoted by $\text{gal}(D)$ is*

$$\text{gal}(D) := \text{null} \left(\begin{bmatrix} P^T \\ e^T \end{bmatrix} \right) = \text{null} \left(\begin{bmatrix} | & & | \\ p_1 & \cdots & p_n \\ | & & | \\ 1 & \cdots & 1 \end{bmatrix} \right) = \text{null}(P^T) \cap \text{null}(e^T). \quad (3.8)$$

Since $\dim(\text{affhull}(\{p_1, \dots, p_n\})) = d$, by Corollary 2.1.2, $\begin{bmatrix} P^T \\ e^T \end{bmatrix}$ has full row rank of $d+1$. Thus, the dimension of $\text{gal}(D)$ is $n - d - 1$.

Let $G = \mathcal{K}^\dagger(D)$ be the Gram matrix. Recall that $G = PP^T$, so $\text{null}(G) = \text{null}(P^T)$. Thus, the Gale space can also be defined in terms of the null space of the Gram matrix

$$\text{gal}(D) = \text{null} \left(\begin{bmatrix} G \\ e^T \end{bmatrix} \right).$$

Definition 3.2.2. *Let $D \in \mathbb{E}^n$. A $n \times n - d - 1$ matrix N whose columns forms a basis of $\text{gal}(D)$ is the Gale matrix of D . Let*

$$N = \begin{bmatrix} -n_1^T - \\ \vdots \\ -n_n^T - \end{bmatrix}.$$

The vectors $n_1, \dots, n_n$ are called the Gale transform of $p_1, \dots, p_n$.

3.2.1 Affine Dependency

For $p_1, \dots, p_n \in \mathbb{R}^d$, the Gale transform $n_1, \dots, n_n$ encode information about affine dependency among $p_1, \dots, p_n$.

Proposition 3.2.3. [3] *Let $n_1, \dots, n_n \in \mathbb{R}^{n-d-1}$ be Gale transforms of $p_1, \dots, p_n \in \mathbb{R}^d$. Let $S \subseteq [n]$ such that $|S| = d+1$, and let $\bar{S} = [n] \setminus S$. Then, $\{p_i : i \in S\}$ is affinely dependent if, and only if, $\{n_i : i \in \bar{S}\}$ is linearly dependent.*

Theorem 3.2.4. *Let $n \geq 3$, and let $D \in \mathbb{E}^n$ with embedding dimension $d \leq n - 2$. Let N, P be a Gale matrix and a configuration matrix of D such that $P^T e = 0$. Then the following hold.*

1. *If $n_i = 0$ for some $i \in [n]$, then $p_i \neq 0$ and $p_i \notin \text{affhull}(\{p_1, \dots, p_n\} \setminus \{p_i\})$ [4].*
2. *If $p_i \notin \text{affhull}(\{p_1, \dots, p_n\} \setminus \{p_i\})$, then $n_i = 0$.*
3. *Let $\mathcal{I} = \{i \in [n] : n_i = 0\}$, then $\{p_i : i \in \mathcal{I}\}$ is linearly independent.*

Proof. 1. Suppose for the contrary that $n_i = 0$ but $p_i \in \text{affhull}(\{p_1, \dots, p_n\} \setminus \{p_i\})$. Without loss of generality, suppose $i = 1$. Then there exists $\lambda_2, \dots, \lambda_n$ such that

$$\begin{bmatrix} p_1 \\ 1 \end{bmatrix} = \sum_{k=2}^n \lambda_k \begin{bmatrix} p_k \\ 1 \end{bmatrix}.$$

Thus, $(-1, \lambda_2, \dots, \lambda_n)^T \in \text{null} \left(\begin{bmatrix} p_1 & \dots & p_n \\ 1 & \dots & 1 \end{bmatrix} \right)$. Hence there exists $0 \neq \mu \in \mathbb{R}^{n-d-1}$ such that $N\mu = (-1, \lambda_2, \dots, \lambda_n)^T$. However, in the first row, $n_1^T \mu = 0^T \mu = 0 \neq -1$, which is a contradiction.

Further suppose for contradiction that $p_1 = 0$. Then, since P is centered, $\sum_{i=1}^n p_i = 0$. Thus, $p_1 = 0 = p_2 + \dots + p_n$, implying $p_1 \in \text{affhull}(\{p_2, \dots, p_n\})$ which is a contradiction.

2. Suppose p_i is not in the affine hull of $\{p_1, \dots, p_n\} \setminus \{p_i\}$. If for $(\lambda_1, \dots, \lambda_n) \neq 0$,

$$0 = \begin{bmatrix} p_1 & \dots & p_n \\ 1 & \dots & 1 \end{bmatrix} \begin{bmatrix} \lambda_1 \\ \vdots \\ \lambda_n \end{bmatrix},$$

then

$$0 = \lambda_i \begin{bmatrix} p_i \\ 1 \end{bmatrix} + \sum_{k \neq i} \lambda_k \begin{bmatrix} p_k \\ 1 \end{bmatrix}.$$

This implies $\lambda_i = 0$. Otherwise, we can write p_i as an affine combination of $\{p_1, \dots, p_n\} \setminus \{p_i\}$. Thus, the i th row of the Gale matrix must be zero, which correspond to $n_i = 0$.

3. Let $\mathcal{I} = \{i \in [n] : n_i = 0\}$. If $|\mathcal{I}| = 1$, then the claim is proven by 1., so assume $|\mathcal{I}| > 1$. Without loss of generality, suppose $n_1, \dots, n_r = 0$ and $n_{r+1}, \dots, n_n \neq 0$. By 1., $p_1, \dots, p_r \neq 0$. Suppose for contradiction that $\{p_1, \dots, p_r\}$ is linearly dependent. Thus, there exists $\lambda_1, \lambda_2, \dots, \lambda_r$ not all zero such that $\sum_{k=1}^r \lambda_k p_k = 0$. Without loss of generality, suppose $\lambda_1 \neq 0$. Additionally, $P^T e = 0$ so $\sum_{k=1}^n p_k = 0$. Therefore, for any $\beta \in \mathbb{R}$,

$$\begin{aligned} 0 &= \sum_{k=1}^r \lambda_k p_k + \beta \sum_{k=1}^n p_k \\ \Rightarrow 0 &= \sum_{k=1}^r (\beta + \lambda_k) p_k + \sum_{k=r+1}^n \beta p_k \end{aligned}$$

Let $x = (\beta + \lambda_1, \dots, \beta + \lambda_r, \beta, \dots, \beta)^T$, so $P^T x = 0$. $e^T x = n\beta + \sum_{k=1}^r \lambda_k$.

Let $\beta = -\frac{1}{n} \sum_{k=1}^r \lambda_k$, then $e^T x = 0$, and $x \in \text{null} \left(\begin{bmatrix} p^T \\ e^T \end{bmatrix} \right)$. Therefore, there exists $\mu \in \mathbb{R}^{n-d-1}$ such that $N\mu = x$. For all $i = 1, \dots, r$, $0 = (n_i)^T \mu = x_i = \beta + \lambda_i$. Hence, $\lambda_i = \frac{1}{n} \sum_{k=1}^r \lambda_k$ for all $i = 1, \dots, r$, which is only possible if $\lambda_i = 0$ for all $i = 1, \dots, r$. However, $\lambda_1, \dots, \lambda_r$ are not all zero, leading to a contradiction $\square$

3.2.2 General Position

Next, we introduce the concept of general position. The assumption of general position can significantly simplify the problem we are aiming to solve.

Definition 3.2.5. $p_1, \dots, p_n \in \mathbb{R}^d$ are in general position if every $d+1$ of these points are affinely independent. An EDM $D \in \mathbb{E}^n$ with embedding dimension d is said to be in general position, if the points generating D are in general position in $\mathbb{R}^d$.

For example, if $d = 2$, then $p_1, \dots, p_n$ are in general position if no 3 points are on a line.

Proposition 3.2.6. $p_1, \dots, p_n \in \mathbb{R}^d$ are in general position if, and only if, for all $k \leq d+1$, and $S \subseteq \{p_1, \dots, p_n\}$ with $|S| = k$,

$$\dim(\text{affhull}(\{p_i : i \in S\})) = k - 1$$

Proof. Suppose for contradiction, for some $k \leq d+1$, $S \subseteq \{p_1, \dots, p_n\}$ with $|S| = k$, $\dim(\text{affhull}(\{p_i : i \in S\})) \leq k - 2$. Without loss of generality, let $S = \{p_1, \dots, p_k\}$. Let $T = \{p_{t_1}, \dots, p_{t_{d+1-k}}\} \subseteq (\{p_1, \dots, p_n\} \setminus S)$. Consider $S \cup T$, $|S \cup T| = d+1$. $\dim(\text{affhull}(\{p_1, \dots, p_k, p_{t_1}, \dots, p_{t_{d+1-k}}\})) \leq k - 2 + (d+1 - k) = d - 1$. Therefore, $\{p_1, \dots, p_n\}$ is not in general position. $\square$

Proposition 3.2.7. $p_1, \dots, p_n \in \mathbb{R}^d$ are in general position, if, and only if, every square submatrix of $\begin{bmatrix} P^T \\ e^T \end{bmatrix}$ is nonsingular. Furthermore, if $D \in \mathbb{E}^n$ is in general position in $\mathbb{R}^d$, if, and only if, for every $\mathcal{I} \subseteq [n]$ such that $|\mathcal{I}| \geq d+1$,

$$\text{rank}(\mathcal{K}^\dagger(D(\mathcal{I}))) = d.$$

Proof. The first claim follows from the definition of general position and Proposition 2.1.1. For the second claim, by Theorem 3.1.6, $\text{rank}(\mathcal{K}^\dagger(D(\mathcal{I})))$ is the embedding dimension of $\{p_i : i \in \mathcal{I}\}$. If $\text{rank}(\mathcal{K}^\dagger(D(\mathcal{I}))) < d$, then $\{p_i : i \in \mathcal{I}\}$ can be embedded in a lower dimension, implying $\dim(\text{affhull}(\{p_i : i \in \mathcal{I}\})) < d$, so $\{p_1, \dots, p_n\}$ is not in general position. $\square$

Proposition 3.2.8. [2] Let $D \in \mathbb{E}^n$ with embedding dimension $d \leq n - 2$. Let N be the Gale matrix of D . Then D is in general position if, and only if, every submatrix of N with size $n - d - 1$ is nonsingular.

3.3 Euclidean Distance Matrix Completion Problem

A matrix D is a *partial Euclidean distance matrix* if every entry of D is either specified or unspecified, and every fully specified principal submatrix of D is an EDM. We can associate a weighted undirected graph $G = (V, E, w)$ with the partial EDM $D \in \mathbb{S}^n$, such that $V = \{1, \dots, n\}$, $E = \{(i, j) : i \neq j, D_{ij} \text{ is specified}\}$, and $w_{ij} = \sqrt{D_{ij}}$. Let H be the adjacency matrix of the graph, so that $H_{ij} = 1$ if D_{ij} is specified, and $H_{ij} = 0$ otherwise. The *Euclidean distance matrix completion problem* (EDMCP) is

$$\begin{aligned} \text{find} \quad & \bar{D} \in \mathbb{E}^n \\ \text{s.t.} \quad & H \circ \bar{D} = H \circ D \end{aligned}$$

where $\circ$ is the entry wise Hadamard product. We can also formulate this problem as

$$\begin{aligned} \text{find} \quad & G \in \mathbb{S}_+^n \cap \mathbb{S}_C^n \\ \text{s.t.} \quad & H \circ \mathcal{K}(G) = H \circ D \end{aligned}$$

A natural question to consider is when does a partial EDM admit a completion. It has been shown by Bakonyi and Johnson [7] that any partial EDM with a chordal graph can be completed to an EDM, and every nonchordal graph has a partial EDM that does not admit any EDM completion. Note that a graph is *chordal* if every cycle of the graph with length at least four has a chord.

If we restrict the embedding dimension of the completion, and ask whether there exists an EDM completion with embedding dimension d , then the *low dimensional EDMCP* is

$$\begin{aligned} \text{find} \quad & \bar{D} \in \mathbb{E}^n \\ \text{s.t.} \quad & H \circ \bar{D} = H \circ D \\ & \text{embdim}(\bar{D}) = d \end{aligned}$$

The low dimensional EDMCP is NP-hard [24].

EDMCP is closely related to the problem we are trying to solve. If we know the position of the error, then the problem can be modelled as a low dimensional EDMCP.

3.3.1 Solving EDMCP Using Facial Reduction

The technique of *facial reduction* is pioneered by Borwein and Wolkowicz [9]. Let (P) be a standard semidefinite optimization problem. Suppose we can find the minimal face F containing the feasible region of (P) . Then, we can modify (P) into the new problem $(\tilde{P}) : \inf\{\langle C, X \rangle : \mathcal{A}(X) = b, X \in F\}$, thus reducing the dimension of the problem.

In general, the minimal face containing the feasible region is hard to identify. However, the special structure of EDMCP allow us to identify the minimal face containing positive semidefinite matrices where entries of some principal submatrices are specified. If we consider the graph associated with the partial EDM, then every fully specified principal submatrix is a clique in the graph.

Let $D \in \mathbb{E}^n$, and $G = \mathcal{K}^\dagger(D) \in \mathbb{S}_+^n \cap \mathbb{S}_C^n$. By Theorem 2.3.5, we can write $\text{face}(G) = U\mathbb{S}_+^d U^T = \mathbb{S}_+^n \cap Z^\perp$. For simplicity, we say U is a facial vector for G or U is a facial vector for D to mean

$\text{face}(G) = \text{face}(\mathcal{K}^\dagger(D)) = U\mathbb{S}_+^d U^T$. Similarly, we say Z is an exposing vector for G or for D to mean Z is an exposing vector for $\text{face}(G) = \text{face}(\mathcal{K}^\dagger(D))$.

Recall that when $V \in \mathbb{R}^{n \times n-1}$ is chosen so that $\begin{bmatrix} V & e \end{bmatrix}$ is orthogonal. We have that

$$\mathbb{S}_C^n \cap \mathbb{S}_+^n = V\mathbb{S}^{n-1}V^T = \mathcal{K}^\dagger(\mathbb{E}^n).$$

Facial Vector Approach

The following results presented in [20] give an approach to solving EDMCP by building a facial vector from intersecting cliques.

Theorem 3.3.1 (Single Clique Facial Reduction. Theorem 2.3,[20]). *Let $D \in \mathbb{E}^n$, $d = \text{embdim}(D)$, $\mathcal{I} = [1 : k]$, and $\bar{D} = D(\mathcal{I})$, $t = \text{embdim}(\bar{D})$. Let*

$$\bar{G} = \mathcal{K}^\dagger(\bar{D}) = \bar{U}_G S \bar{U}_G^T,$$

where $S \in \mathbb{S}_{++}^t$, $\bar{U}_G \in \mathbb{R}^{k \times t}$, $\bar{U}_G^T \bar{U}_G = I_t$. Furthermore, let $U_G = \begin{bmatrix} \bar{U}_G & \frac{1}{\sqrt{k}}e \end{bmatrix}$, $U = \begin{bmatrix} U_G & 0 \\ 0 & I_{n-k} \end{bmatrix}$ and $\begin{bmatrix} V & \frac{U^T e}{\|U^T e\|} \end{bmatrix} \in \mathbb{R}^{(n-k+t+1) \times (n-k+t+1)}$ be an orthogonal matrix. Then

$$\text{face}\left(\mathcal{K}^\dagger(\mathcal{P}_{\mathcal{I}}^{-1}(\bar{D}) \cap \mathbb{E}^n)\right) = \left(U\mathbb{S}_+^{n-k+t+1}U^T\right) \cap \mathbb{S}_C^n = (UV)\mathbb{S}_+^{n-k+t}(UV)^T. \quad (3.9)$$

Thus, UV is a facial vector for the minimal face of all EDMs $\tilde{D}$ with the $\tilde{D}(\mathcal{I}) = \bar{D}$.

When the intersection between two principal submatrices has embedding dimension d , we can find the facial vector for the intersection of the two faces efficiently.

Lemma 3.3.2 (Lemma 2.9, [20]). *Let*

$$U_1 = \begin{matrix} & d+1 \\ s_1 & \begin{bmatrix} U_1' \\ U_1'' \end{bmatrix} \\ k & \end{matrix}, \quad U_2 := \begin{matrix} & d+1 \\ s_2 & \begin{bmatrix} U_2'' \\ U_2' \end{bmatrix} \\ k & \end{matrix}, \quad \hat{U}_1 := \begin{matrix} & d+1 & s_2 \\ s_1 & \begin{bmatrix} U_1' & 0 \\ U_1'' & 0 \\ 0 & I \end{bmatrix} \\ k & \end{matrix}, \quad \hat{U}_2 := \begin{matrix} & s_1 & d+1 \\ s_2 & \begin{bmatrix} I & 0 \\ 0 & U_2'' \\ 0 & U_2' \end{bmatrix} \\ k & \end{matrix}$$

be appropriately blocked with $U_1'', U_2'' \in \mathbb{R}^{k \times (d+1)}$ full column rank, and $\text{range}(U_1'') = \text{range}(U_2'')$. Furthermore, let

$$\bar{U}_1 := \begin{matrix} & d+1 \\ s_1 & \begin{bmatrix} U_1' \\ U_1'' \\ U_2'(U_2'')^\dagger U_1'' \end{bmatrix} \\ k & \end{matrix}, \quad \bar{U}_2 := \begin{matrix} & d+1 \\ s_1 & \begin{bmatrix} U_1'(U_1'')^\dagger U_2'' \\ U_2'' \\ U_2' \end{bmatrix} \\ k & \end{matrix}. \quad (3.10)$$

Then, $\bar{U}_1, \bar{U}_2$ are full column rank and satisfy

$$\text{range}(\hat{U}_1) \cap \text{range}(\hat{U}_2) = \text{range}(\bar{U}_1) = \text{range}(\bar{U}_2).$$

Moreover, if $e_{d+1} \in \mathbb{R}^{d+1}$ is the $(d+1)$ th standard unit vector, and $U_i e_{d+1} = \alpha_i e$ for some $\alpha_i \neq 0$ for $i = 1, 2$, then $\bar{U}_i e_{d+1} = \alpha_i e$, for $i = 1, 2$.

Theorem 3.3.3 (Theorem 2.7, [20]). *Let $D \in \mathbb{E}^n$ with embedding dimension d . Let*

$$\mathcal{I}_1 = [1 : \bar{k}_1 + \bar{k}_2], \quad \mathcal{I}_2 = [\bar{k}_1 + 1 : \bar{k}_1 + \bar{k}_2 + \bar{k}_3] \subseteq [n],$$

$$k_1 = |\mathcal{I}_1| = \bar{k}_1 + \bar{k}_2, \quad k_2 = |\mathcal{I}_2| = \bar{k}_2 + \bar{k}_3, \quad k = |\mathcal{I}_1 \cup \mathcal{I}_2| = \bar{k}_1 + \bar{k}_2 + \bar{k}_3.$$

Thus, $D(\mathcal{I}_1), D(\mathcal{I}_2)$ are two intersecting principal submatrices, where the intersection is $D([\bar{k}_1 + 1 : \bar{k}_1 + \bar{k}_2])$.

For $i = 1, 2$, let $D_i = D(\mathcal{I}_i) \in \mathbb{E}^{k_i}$ with embedding dimension t_i . Let $G_i = \mathcal{K}^\dagger(D_i) = \bar{U}_i S_i \bar{U}_i^T$, where $\bar{U}_i \in \mathbb{R}^{k_i \times t_i}$ is orthogonal, and $S_i \in \mathbb{S}_{++}^{t_i}$. Let $U_i = \begin{bmatrix} \bar{U}_i & \frac{1}{\sqrt{k_i}} e \end{bmatrix}$. Let t and $\bar{U} \in \mathbb{R}^{k \times (t+1)}$ satisfy $\bar{U}^T \bar{U} = I_{t+1}$, and

$$\text{range}(\bar{U}) = \text{range} \left(\begin{bmatrix} U_1 & 0 \\ 0 & I_{\bar{k}_3} \end{bmatrix} \right) \cap \text{range} \left(\begin{bmatrix} I_{\bar{k}_1} & 0 \\ 0 & U_2 \end{bmatrix} \right). \quad (3.11)$$

Let $U = \begin{bmatrix} \bar{U} & 0 \\ 0 & I_{n-k} \end{bmatrix} \in \mathbb{R}^{n \times (n-k+t+1)}$ and $\begin{bmatrix} V & \frac{U^T e}{\|U^T e\|} \end{bmatrix} \in \mathbb{R}^{(n-k+t+1) \times (n-k+t+1)}$ be orthogonal. Then

$$\cap_{i=1}^2 \text{face } \mathcal{K}^\dagger(\mathcal{P}_{\mathcal{I}_i}^{-1}(D_i) \cap \mathbb{E}^n) = \left(U \mathbb{S}_+^{n-k+t+1} U^T \right) \cap \mathbb{S}_C^n = (UV) \mathbb{S}_+^{n-k+t} (UV)^T. \quad (3.12)$$

Lemma 3.3.4. [20] *Let the hypothesis of Theorem 3.3.3 hold. Let $G = \mathcal{K}^\dagger(D(\mathcal{I}_1 \cap \mathcal{I}_2))$, and suppose $\text{rank}(G) = d$. Since $\text{embdim}(D(\mathcal{I}_1 \cap \mathcal{I}_2)) = d$, $\text{embdim}(D_1) = \text{embdim}(D_2) = d$ as well, so $U_i \in \mathbb{R}^{k_i \times d}$ for $i = 1, 2$. Partition U_1, U_2 as*

$$U_1 := \begin{matrix} & d+1 \\ k_1 & \begin{bmatrix} U_1' \\ U_1'' \end{bmatrix} \\ k_2 & \end{matrix}, \quad U_2 := \begin{matrix} & d+1 \\ k_2 & \begin{bmatrix} U_2'' \\ U_2' \end{bmatrix} \\ k_3 & \end{matrix},$$

Then, U_1'', U_2'' have full column rank, and $\text{range}(U_1'') = \text{range}(U_2'')$.

Let d be the embedding dimension of $D \in \mathbb{E}^n$, and let $\mathcal{I}_1, \mathcal{I}_2 \subseteq [n]$. If the intersection $D(\mathcal{I}_1 \cap \mathcal{I}_2)$ also has embedding dimension d , we say the intersection is *rigid*. When the intersection is rigid, there is a unique completion for $D(\mathcal{I}_1 \cup \mathcal{I}_2)$, and the following theorem provides a method of determining it.

Theorem 3.3.5 (Nonsingular Facial Reduction with Rigid Intersection. Theorem 2.10, [20]). *Let the hypothesis of Theorem 3.3.3 hold. Let*

$$\mathcal{I} \subseteq \mathcal{I}_1 \cap \mathcal{I}_2, \quad \bar{D} = D(\mathcal{I}), \quad G = \mathcal{K}^\dagger(\bar{D}), \quad \bar{U}_{\mathcal{I}} = \bar{U}(\mathcal{I}, :),$$

where $\bar{U} \in \mathbb{R}^{k \times (t+1)}$ satisfy (3.11). Let $\begin{bmatrix} \bar{V} & \frac{\bar{U}^T e}{\|\bar{U}^T e\|} \end{bmatrix} \in \mathbb{R}^{(t+1) \times (t+1)}$ be orthogonal. Let

$$R := (J\bar{U}_{\mathcal{I}}\bar{V})^\dagger G \left((J\bar{U}_{\mathcal{I}}\bar{V})^\dagger \right)^T. \quad (3.13)$$

If the embedding dimension for $\bar{D}$ is d , then $t = d$, $R \in \mathbb{S}_{++}^d$ is the unique solution of the equation

$$(J\bar{U}_{\mathcal{I}}\bar{V})R(J\bar{U}_{\mathcal{I}}\bar{V})^T = G, \quad (3.14)$$

and

$$D(\mathcal{I}_1 \cup \mathcal{I}_2) = \mathcal{K}((\bar{U}\bar{V})R(\bar{U}\bar{V})^T). \quad (3.15)$$

The following corollary shows if we know the minimal face of $\mathbb{S}_+^n \cap \mathbb{S}_C^n$ containing $\mathcal{K}^\dagger(D)$, then we can recover D by solving a small equation using a principal submatrix of D with embedding dimension d . Result of this corollary is used extensively in our algorithm.

Corollary 3.3.6 ([20]). *Let $D \in \mathbb{E}^n$ with embedding dimension d . Let $\mathcal{I} \subseteq [n]$ such that $|\mathcal{I}| \geq d+1$, and $D(\mathcal{I})$ has embedding dimension d . Let $G_{\mathcal{I}} = \mathcal{K}^\dagger(D(\mathcal{I}))$. Let V be a centered facial vector for D , so $\mathcal{K}^\dagger(D) = VZV^T$ for some $Z \in \mathbb{R}_{++}^d$. Let $V_{\mathcal{I}} = V(\mathcal{I}, :)$. Then,*

$$(JV_{\mathcal{I}})R(JV_{\mathcal{I}})^T = G_{\mathcal{I}}$$

has a unique solution $R \in \mathbb{S}_{++}^d$, with $R = Z$. Thus, $D = \mathcal{K}(VRV^T)$.

Proof. The corollary follows from Theorem 3.3.5 by letting $\mathcal{I}_1 = \mathcal{I}_2 = [n]$. We also provide a direct proof.

Since V is the facial vector for G , $\text{range}(G) = \text{range}(V)$, and $PP^T = G = VZV^T$ for some $Z \in \mathbb{S}_{++}^d$. Let $Z = SS^T$, then $P = VS$ gives a centered configuration matrix for G . This is a consistent but overdetermined system. Consider only the rows corresponding to $\mathcal{I}$, then

$$P(\mathcal{I}, :) = V_{\mathcal{I}}S.$$

We have $\text{range}(P(\mathcal{I}, :)) = \text{range}(V_{\mathcal{I}})$, since S is nonsingular. $\text{rank}(P(\mathcal{I}, :)) = \text{rank}(V_{\mathcal{I}}) = d$, thus the system $P(\mathcal{I}, :) = V_{\mathcal{I}}S$ has a unique solution, which is also the solution to $P = VS$. Let S be this unique solution. Clearly, it is also a solution to

$$JP(\mathcal{I}, :) = JV_{\mathcal{I}}S.$$

Notice that

$$\begin{aligned} JP(\mathcal{I}, :)P(\mathcal{I}, :)^T J &= \mathcal{P}_{\mathbb{S}_C^n}(P(\mathcal{I}, :)P(\mathcal{I}, :)^T) \\ &= G_{\mathcal{I}}. \end{aligned}$$

In other word, $JP(\mathcal{I}, :)$ is a centered configuration matrix for $D(\mathcal{I})$.

Let $G_{\mathcal{I}} = P_{\mathcal{I}}P_{\mathcal{I}}^T$ so $P_{\mathcal{I}}$ is any centered configuration matrix for $D(\mathcal{I})$. We show the system $P_{\mathcal{I}} = JV_{\mathcal{I}}X$ has a unique solution. $\text{rank}(P_{\mathcal{I}}) = \text{rank}(G_{\mathcal{I}}) = d$, and we show $\text{nullity}(JV_{\mathcal{I}}) = 0$. If for some $y \in \mathbb{R}^d$, $y \in \text{null}(JV_{\mathcal{I}})$,

$$JV_{\mathcal{I}}y = 0 \Rightarrow V_{\mathcal{I}}y = \frac{e^T V_{\mathcal{I}}y}{n} e.$$

Since $\text{rank}(G_{\mathcal{I}}) = d$, there exists $d+1$ points in $\mathcal{I}$ such that the dimension of their affine hull is d . By Corollary 2.1.2, $\text{rank}([P(\mathcal{I}, :) \ e]) = d+1$, therefore, $e \notin \text{range}(P_{\mathcal{I}}) = \text{range}(V_{\mathcal{I}})$. Hence, $y = 0$, $\text{rank}(JV_{\mathcal{I}}) = d$ as well. Therefore, the system $P_{\mathcal{I}} = JV_{\mathcal{I}}X$ has a unique solution.

In particular, this shows $JP(\mathcal{I}, :) = JV_{\mathcal{I}}S$ and $P(\mathcal{I}, :) = V_{\mathcal{I}}S$ has the same unique solution S .

Furthermore, this shows $JV_{\mathcal{I}}$ is injective, so $(JV_{\mathcal{I}})^\dagger$ is a left inverse of $JV_{\mathcal{I}}$. Thus, $(JV_{\mathcal{I}})^T$ is surjective and $((JV_{\mathcal{I}})^T)^\dagger$ is a right inverse of $(JV_{\mathcal{I}})^T$.

Now, let Q be the unique solution to $P_{\mathcal{I}} = JV_{\mathcal{I}}X$, then

$$G_{\mathcal{I}} = P_{\mathcal{I}}P_{\mathcal{I}}^T = JV_{\mathcal{I}}QQ^TV_{\mathcal{I}}^TJ,$$

hence

$$QQ^T = (JV_{\mathcal{I}})^\dagger G_{\mathcal{I}} (V_{\mathcal{I}}^T J)^\dagger.$$

On the other hand, $JP(\mathcal{I}, :)P(\mathcal{I}, :)^T J = G_{\mathcal{I}}$, and $JP(\mathcal{I}, :) = JV_{\mathcal{I}}S$, thus,

$$QQ^T = (JV_{\mathcal{I}})^\dagger JP(\mathcal{I}, :)P(\mathcal{I}, :)^T J (V_{\mathcal{I}}^T J)^\dagger = (JV_{\mathcal{I}})^\dagger (JV_{\mathcal{I}})SS^T (V_{\mathcal{I}}^T J)(V_{\mathcal{I}}^T J)^\dagger = SS^T = Z.$$

Let $R = QQ^T$, then $R = Z$. Hence, we can recover D as $D = \mathcal{K}(VRV^T)$. $\square$

If $D_{\mathcal{I}_1}$ and $D_{\mathcal{I}_2}$ are two intersecting blocks that are specified and the intersection has embedding dimension d , then, using Lemma 3.3.4 and Lemma 3.3.2, we can efficiently determine a $\bar{U}$ satisfying (3.11) in Theorem 3.3.3. Thus, we can compute a facial vector for $D_{\mathcal{I}_1 \cup \mathcal{I}_2}$ from facial vector of $D_{\mathcal{I}_1}$ and $D_{\mathcal{I}_2}$. Then, using Theorem 3.3.5, we can uniquely complete $D_{\mathcal{I}_1 \cup \mathcal{I}_2}$.

Exposing Vector Approach

Another approach to solving EDMCP with facial reduction is to consider the exposing vector instead of the facial vector. An algorithm for solving EDMCP adding exposing vectors from overlapping cliques was presented in [14], Algorithm 1.

Theorem 3.3.7 (Theorem 4.12, [15]). *Let $D \in \mathbb{E}^n, d = \text{embdim}(D), \mathcal{I} \subseteq [n]$, $k = |\mathcal{I}|$, and $\bar{D} = D(\mathcal{I})$. Then, for any matrix Z exposing $\text{face}_{\mathbb{S}_+^k \cap \mathbb{S}_C^k}(\mathcal{K}^\dagger(\bar{D}))$, $\mathcal{P}_{\mathcal{I}}^*(Z)$ exposes*

$$\text{face}_{\mathbb{S}_+^n \cap \mathbb{S}_C^n} \left(\mathcal{K}^\dagger(P_{\mathcal{I}}^{-1}(\bar{D}) \cap \mathbb{E}^n) \right)$$

where $[\mathcal{P}_{\mathcal{I}}^*(Z)]_{ij} = Z_{ij}$ for all $i, j \in \mathcal{I}$, and 0 otherwise.

Recall Proposition 2.3.3 that for $F_i \trianglelefteq \mathbb{S}_+^n$, if Y_i exposes F_i . Then $\sum_{i=1}^n Y_i$ exposes $\cap_{i=1}^n F_i$.

Let $D \in \mathbb{E}^n$ with embedding dimension d , $\mathcal{I}_1, \mathcal{I}_2 \subseteq [n]$. For $i = 1, 2$, let $D_i = D(\mathcal{I}_i)$, $G_i = \mathcal{K}^\dagger(D_i)$, and $\text{face}(G_i) = \mathbb{S}_+^{|\mathcal{I}_i|} \cap \mathbb{S}_C^{|\mathcal{I}_i|} \cap \hat{Z}_i^\perp$. Let $F_i = \text{face} \mathcal{K}^\dagger(P_{\mathcal{I}_i}^{-1}(D_i) \cap \mathbb{E}^n)$, and let Z_i be centered exposing vector for F_i . Then, $Z_1 + Z_2$ is a centered exposing vector for $F_1 \cap F_2$.

Let V be orthogonal, full column rank, and span $\text{null}([Z \quad e]^T)$. We find the completion for $D(\mathcal{I}_1 \cup \mathcal{I}_2)$ by solving for the optimal solution for

$$\min_{R \succeq 0} \|H_{\mathcal{I}_1 \cup \mathcal{I}_2} \circ \mathcal{K}(VRV^T) - H_{\mathcal{I}_1 \cup \mathcal{I}_2} \circ D\|. \quad (3.16)$$

If the intersection $D(\mathcal{I}_1 \cap \mathcal{I}_2)$ is rigid, the solution to (3.16) is unique. Theorem 3.3.5 demonstrated that we can explicitly find the unique completion $D(\mathcal{I}_1 \cap \mathcal{I}_2)$, and we can find the completion by solving a small system of equations using Corollary 3.3.6.

Therefore, we can building an exposing vector for D by adding up exposing vectors from overlapping cliques. Then, we can find the centered facial vector V for D , and we have an efficient way of uniquely determining $R = QQ^T \in \mathbb{R}_{++}^d$ such that $D = \mathcal{K}(VRV^T)$.

Gale Matrix Approach

Theorem 3.3.8. *Let $D \in \mathbb{E}^n$, $G = \mathcal{K}^\dagger(D)$. If N is a Gale matrix for D , then $Z = NN^T$ is a centered exposing vector for G . Conversely, if Z is a centered exposing vector for G , then for any factorization $Z = NN^T$, $N \in \mathbb{R}^{n \times n-d-1}$, N is a Gale matrix of D .*

Proof. A centered exposing vector for G is a matrix $Z \in \mathbb{S}_+^n \cap \mathbb{S}_C^n$ such that $GZ = 0, Ze = 0$. Factor $Z = NN^T$ then $\text{range}(N) = \text{null}(G) \cap \text{null}(e)$ so N is a Gale matrix of D . On the other hand, by definition, $\text{range}(N) = \text{null}(G) \cap \text{null}(e^T)$, thus $NN^T \in \mathbb{S}_+^n$, $NN^T G = 0$, and $NN^T e = 0$. Therefore, NN^T is a centered exposing vector of G . $\square$

Theorem 3.3.9. *Let $D \in \mathbb{E}^n$ with embedding dimension d , Let $\mathcal{I}_1, \mathcal{I}_2 \subseteq [n]$, such that $\text{embdim}(D(\mathcal{I}_1 \cap \mathcal{I}_2)) = d$. Let $|\mathcal{I}_1| = k_1, |\mathcal{I}_2| = k_2$ and $|\mathcal{I}_1 \cap \mathcal{I}_2| = k$.*

$$N_1 = \begin{matrix} & k_1-d-1 \\ & A \\ k_1-k & \left[\begin{array}{c} \\ B \end{array} \right] \\ & k \end{matrix}, \quad N_2 = \begin{matrix} & k_2-d-1 \\ & C \\ k_2-k & \left[\begin{array}{c} \\ D \end{array} \right] \\ & k \end{matrix}$$

be Gale matrices of $D(\mathcal{I}_1)$ and $D(\mathcal{I}_2)$ respectively. Then,

$$N = \begin{matrix} & k_1-d-1 & k_2-d-1 \\ & A & 0 \\ k_1-k & \left[\begin{array}{cc} B & C \end{array} \right] \\ & k & \\ k_2-k & \left[\begin{array}{cc} 0 & D \end{array} \right] \end{matrix}$$

is a Gale matrix for $D(\mathcal{I}_1 \cup \mathcal{I}_2)$.

Proof. Since $N_1 = \begin{matrix} & k_1-d-1 \\ & A \\ k_1-k & \left[\begin{array}{c} \\ B \end{array} \right] \\ & k \end{matrix}$ is a Gale matrix for $D(\mathcal{I}_1)$,

$$Z_1 = N_1 N_1^T = \begin{bmatrix} AA^T & AB^T \\ BA^T & BB^T \end{bmatrix}$$

is an exposing vector for $D(\mathcal{I}_1)$. Similarly,

$$Z_2 = N_2 N_2^T = \begin{bmatrix} CC^T & CD^T \\ DC^T & DD^T \end{bmatrix}$$

is an exposing vector for $D(\mathcal{I}_2)$. By Theorem 3.3.7, and Proposition 2.3.3,

$$\begin{matrix} k_1 & k_2-k \\ k_2-k & \left[\begin{array}{cc} Z_1 & 0 \\ 0 & 0 \end{array} \right] \end{matrix} + \begin{matrix} k_1-k & k_2 \\ k_2 & \left[\begin{array}{cc} 0 & 0 \\ 0 & Z_2 \end{array} \right] \end{matrix} = \begin{matrix} k_1-k & k & k_2-k \\ k & \left[\begin{array}{ccc} AA^T & AB^T & 0 \\ BA^T & BB^T + CC^T & CD^T \\ 0 & DC^T & DD^T \end{array} \right] \\ k_2-k \end{matrix} = NN^T$$

is an exposing vector for $\cap_{i=1}^2 \text{face} \left(\mathcal{K}^\dagger(P_{\mathcal{I}_i}^{-1}(D(\mathcal{I}_i) \cap \mathbb{E}^n)) \right)$. Since the overlap is rigid, N is a Gale matrix for $D(\mathcal{I}_1 \cup \mathcal{I}_2)$. $\square$

3.4 Yielding Entry of an EDM

In order to consider the question of identifying the corrupted entry in an EDM, we must consider the question of whether perturbing the value of one entry will cause a proper EDM to not be an EDM. Let $D \in \mathbb{E}^n$, $\epsilon \in \mathbb{R}$, and $i, j \in [n]$. Recall that $E_{ij} = e_i e_j^T + e_j e_i^T$. If for the corrupted entry i, j , $D_n = D + \epsilon E_{ij}$ remains a proper EDM of embedding dimension d , then we cannot distinguish the corrupted EDM from a proper EDM. Then, the problem of recovering i, j and ϵ cannot be solved.

Definition 3.4.1. Let $D \in \mathbb{E}^n$, $i, j \in [n]$. Let $l_{ij} \leq 0, u_{ij} \geq 0$ be two scalars such that $D + \epsilon E_{ij}$ is an EDM if, and only if, $l_{ij} \leq \epsilon \leq u_{ij}$. The entry D_{ij} is called *unyielding* if $l_{ij} = u_{ij} = 0$. Otherwise, the entry D_{ij} is called *yielding*. If every entry of D is *unyielding*, we say D is *unyielding*. If any entry of D is *yielding*, we say D is *yielding*.

3.4.1 Positive Semidefinite Interval of a Matrix Pencil

First, we introduce the concept of a matrix pencil. The question of when an entry is yielding is also related to the question of when a matrix pencil is positive semidefinite.

Definition 3.4.2. Let $A, B \in \mathbb{R}^{n \times n}$, the *matrix pencil* is the function $t \mapsto A + tB$, $t \in \mathbb{R}$.

When $A \succeq 0$ and B is rank 1 or 2, the positive semidefinite interval for the matrix pencil $A + tB$ is studied in [11]. The positive semidefinite interval for the matrix pencil $A + tB$ in a wide range of cases is summarized in [26].

Theorem 3.4.3. [11] Suppose $\text{rank}(B) \leq 2$. Without loss of generality, we can suppose $B = uu^T + \lambda vv^T$, $\lambda = -1, 0$, or 1 . Let the positive semidefinite interval be $[l, r]$, so $A + tB \succeq 0$ if and only if $t \in [l, r]$.

1. If $\lambda = 0$, or 1 .

(a) If $u \in \text{range}(A)$ and $v \in \text{range}(A)$, then

$$l = \frac{2}{-u^T x - v^T y - \sqrt{(u^T x - v^T y)^2 + 4(u^T y)^2}},$$

$$r = +\infty.$$

where $x, y \in \mathbb{R}^n$ such that $Ax = u, Ay = v$.

(b) If $u \notin \text{range}(A)$ or $v \notin \text{range}(A)$, then $l = 0, r = +\infty$.

2. If $\lambda = -1$.

(a) If $u \in \text{range}(A)$ and $v \in \text{range}(A)$, then

$$l = \frac{2}{-u^T x - v^T y - \sqrt{(u^T x + v^T y)^2 - 4(u^T x)^2}},$$

$$r = \frac{2}{-u^T x + v^T y + \sqrt{(u^T x + v^T y)^2 - 4(u^T x)^2}}.$$

where $x, y \in \mathbb{R}^n$ such that $Ax = u, Ay = v$.

- (b) If $u \in \text{range}(A)$, $v \notin \text{range}(A)$, then $l = -\frac{1}{u^T x}$, $r = 0$, where $x \in \mathbb{R}^n$ such that $Ax = u$.
- (c) If $u \notin \text{range}(A)$, $v \in \text{range}(A)$, then $l = 0$, $r = \frac{1}{v^T y}$, where $y \in \mathbb{R}^n$ such that $Ay = v$.
- (d) If $u \notin \text{range}(A)$, $v \notin \text{range}(A)$, but $v \in \text{range} \left(\begin{bmatrix} A & u \end{bmatrix} \right)$. Let $x \in \mathbb{R}^n$, $\alpha \in \mathbb{R}$ such that $Ax + \alpha u = v$. Then,

$$1 - \alpha^2 = 0 \Rightarrow l = r = 0.$$

$$1 - \alpha^2 > 0 \Rightarrow l = 0 \text{ and } r = \frac{1 - \alpha^2}{(v^T - \alpha u^T)x}.$$

$$1 - \alpha^2 < 0 \Rightarrow l = \frac{1 - \alpha^2}{(v^T - \alpha u^T)x} \text{ and } r = 0.$$

- (e) If $u \notin \text{range}(A)$, $v \notin \text{range}(A)$, and $v \notin \text{range} \left(\begin{bmatrix} A & u \end{bmatrix} \right)$, then $l = r = 0$.

In our case, let $D \in \mathbb{E}^n$ with embedding dimension d , $i, j \in [n]$, let

$$D(\epsilon) := D + \epsilon E_{ij}.$$

Let $G_E = \mathcal{K}^\dagger(E_{ij})$, and define the matrix pencil

$$G(\epsilon) = \mathcal{K}^\dagger(D(\epsilon)) = G + \epsilon G_E. \quad (3.17)$$

$D(\epsilon) \in \mathbb{E}^n$ if, and only if, $G + \epsilon G_E$ is positive semidefinite. This is a rank 2 update of G . However, we also have the additional condition that the rank of $G + \epsilon G_E$ remains d .

3.4.2 Yielding Interval of an EDM

Next, we present result from [4] that characterizes yielding entries of an EDM using concepts from Gale transform.

Lemma 3.4.4 (Proposition 2.2, [4]). *Let a and $b \in \mathbb{R}^d$ be two nonzero, nonparallel vectors, $d \geq 2$. Let $X = ab^T + ba^T$, then X has one positive eigenvalue λ_1 and one negative eigenvalue λ_d , where*

$$\lambda_1 = a^T b + \|a\| \|b\|, \quad \lambda_d = a^T b - \|a\| \|b\|.$$

Theorem 3.4.5 (Theorem 3.1, [4]). *Let $D \in \mathbb{E}^n$ with embedding dimension $d = n - 1$. Then every entry of D is yielding.*

Theorem 3.4.6 (Theorem 3.2, [4]). *Let $D \in \mathbb{E}^n$ with embedding dimension $d \leq n - 2$. Let $n_1, \dots, n_n$ be Gale transforms of generating points of D . Then, entry D_{kl} is unyielding if, and only if, n_k is not parallel to n_l .*

If n_k and n_l are parallel, then there are two cases: $n_k = n_l = 0$ or $n_k \neq 0$ and $n_l \neq 0$. If $n_k = n_l = 0$, then Theorem 3.2.4 implies $p_k \neq 0, p_l \neq 0$, $\{p_k, p_l\}$ is linearly independent, and $p_k, p_l \notin \text{affhull}(\{p_1, \dots, p_n\} \setminus \{p_k, p_l\})$. This implies $\dim(\text{affhull}(\{p_1, \dots, p_n\} \setminus \{p_k, p_l\})) = d - 2$. All but two points lie in a affine space of dimension $d - 2$ comes into effect in Theorem 3.4.9, when we characterize yielding entry but add the additional requirement that the embedding dimension of the EDM does not change.

The following theorem characterizes the yielding interval.

Theorem 3.4.7 (Theorem 4.2 & Theorem 4.3 [4]). *Let assumption of Theorem 3.4.6 hold, and further let $S = P(P^T P)^{-1}$, where P is the configuration matrix of D . Let s_i^T be the i th row of S .*

1. *If $n_k = n_l = 0$, then the yielding interval of D_{kl} is given by*

$$\left[\frac{2}{\lambda_d}, \frac{2}{\lambda_1} \right], \quad (3.18)$$

where $\lambda_1 = s_k^T s_l + \|s_k\| \|s_l\|$, $\lambda_d = s_k^T s_l - \|s_k\| \|s_l\|$.

2. *If n_k, n_l are both nonzero and $n_k = cn_l$ for some nonzero scalar c , then the yielding interval of D_{kl} is given by*

$$\left[\frac{-4c}{\|s_k - cs_l\|^2}, 0 \right] \quad \text{if } c > 0, \text{ and } \left[0, \frac{4|c|}{\|s_k - cs_l\|^2} \right] \quad \text{if } c < 0. \quad (3.19)$$

Theorem 3.4.8 (Corollary 3.2, [4]). *Let $D \in \mathbb{E}^n$ with embedding dimension $d \leq n - 3$. If D is in general position in $\mathbb{R}^d$, then every entry is unyielding.*

3.4.3 Yielding Interval with Fixed Embedding Dimension

Note that in the definition of yielding, $D + \epsilon D_{ij}$ is not required to stay at the same embedding dimension. We further characterize the condition for $G(\epsilon) = G + \epsilon G_E$ to not only be positive semidefinite, and also $\text{rank}(G(\epsilon)) = d$.

Let $D \in \mathbb{E}^n$ with embedding dimension d . If for some $i, j \in [n]$, there exists an interval $(-\delta, \delta)$ with $\delta > 0$ such $D + \epsilon D_{ij}$ is an EDM with embedding dimension d for all $\epsilon \in (-\delta, \delta)$, then we say D_{ij} is *yielding with fixed rank*. If for all $i \neq j \in [n]$, D_{ij} is not yielding with fixed rank, then we say D is *unyielding with fixed rank*.

Theorem 3.4.9. [5] *Let $D \in \mathbb{E}^n$ with embedding dimension $d \leq n - 2$. Let $\epsilon > 0$. Let $G = \mathcal{K}^\dagger(D) = Q\Lambda Q^T$ be the Gram matrix with its spectral decomposition. Suppose*

$$\Lambda = \begin{bmatrix} 0 & 0 \\ 0 & \Lambda_+ \end{bmatrix}, \quad \Lambda_+ \in \mathbb{S}_{++}^d.$$

Let $G_E := \mathcal{K}^\dagger(E_{ij})$, and define

$$\bar{G}_E := Q^T G_E Q = \begin{bmatrix} \bar{G}_{11} & \bar{G}_{12} \\ \bar{G}_{12}^T & \bar{G}_{22} \end{bmatrix},$$

where $\bar{G}_{22} \in \mathbb{S}^d$.

$\Lambda_+ + \epsilon \bar{G}_{22} \succ 0$ if, and only if, $I + \epsilon \Lambda_+^{-1/2} \bar{G}_{22} \Lambda_+^{-1/2} \succ 0$. Let

$$I_\epsilon = \left(-\frac{1}{\lambda_{\max}(\Lambda_+^{-1/2} \bar{G}_{22} \Lambda_+^{-1/2})}, -\frac{1}{\lambda_{\min}(\Lambda_+^{-1/2} \bar{G}_{22} \Lambda_+^{-1/2})} \right) \quad (3.20)$$

where a zero in the denominator results in $-\infty/+ \infty$ for the left/right bound of the interval. Define

the EDM condition:

$$\epsilon \in I_\epsilon \text{ and } \epsilon \left(\bar{G}_{11} - \epsilon \bar{G}_{12} (\Lambda_+ + \epsilon \bar{G}_{12})^{-1} \bar{G}_{21} \right) \succeq 0. \quad (3.21)$$

Then:

1. If the EDM condition holds, then

$$D(\epsilon) \in \mathbb{E}^n, \text{embdim}(D(\epsilon)) \geq d. \quad (3.22)$$

If $\bar{G}_{11} = 0$ or $\bar{G}_{11} - \epsilon \bar{G}_{12} (\Lambda_+ + \epsilon \bar{G}_{22})^{-1} \bar{G}_{21} = 0$, then

$$D(\epsilon) \in \mathbb{E}^n, \text{embdim}(D(\epsilon)) = d. \quad (3.23)$$

2. Suppose there exists $\delta \in \mathbb{R}_{++}$ such that for all $\epsilon \in (-\delta, \delta)$,

$$D(\epsilon) \in \mathbb{E}^n, \text{embdim}(D(\epsilon)) = d.$$

Then the configuration matrix P of D has $n - 2 \geq d$ points that are in a linear manifold L of dimension $d - 2$. Furthermore, the yielding entries i, j correspond to the two points outside the linear manifold.

Proof. We define the matrix pencil $G(\epsilon) = G + \epsilon G_E$, where $G_E = -\frac{1}{2} J E_{ij} J$. An orthogonal pair of eigenvector of E_{ij} is $e_i + e_j$ and $e_i - e_j$ with eigenvalues 1 and -1 respectively. $J E_{ij} J = \mathcal{P}_{\mathbb{S}_C^n}(E_{ij})$, thus $J E_{ij} J$ and $G_E = -\frac{1}{2} J E_{ij} J$ also have exactly two nonzero eigenvalues, one with a positive eigenvalue and another with a negative eigenvalue. Specifically, G_E has two orthogonal eigenvectors

$$e_{ij} := e_i - e_j, \quad e_{ijc} := \frac{2}{k} e - (e_i + e_j),$$

and their corresponding eigenvalues are

$$\lambda_1(G_E) = \frac{1}{2} > 0 > \lambda_n(G_E) = \frac{2-k}{2k}.$$

By Theorem 2.2.6 and Theorem 2.2.8, $\lambda_{\max}(\Lambda_+^{-1/2} \bar{G}_{22} \Lambda_+^{-1/2}) \geq 0 \geq \lambda_{\min}(\Lambda_+^{-1/2} \bar{G}_{22} \Lambda_+^{-1/2})$. Therefore, $0 \in I_\epsilon$. Clearly,

$$\epsilon \in I_\epsilon \iff I + \epsilon \Lambda_+^{-1/2} \bar{G}_{22} \Lambda_+^{-1/2} \succ 0 \iff \Lambda_+ + \epsilon \bar{G}_{22} \succ 0,$$

and

$$G + \epsilon G_E = Q(\Lambda_+ + \epsilon \bar{G}_E) Q^T \succeq 0 \iff \Lambda + \epsilon \bar{G}_E \succeq 0.$$

1. Suppose ϵ and $\bar{G}_E$ satisfy the EDM condition:

$$\epsilon \in I_\epsilon \text{ and } \epsilon \left(\bar{G}_{11} - \epsilon \bar{G}_{12} (\Lambda_+ + \epsilon \bar{G}_{12})^{-1} \bar{G}_{21} \right) \succeq 0.$$

By Schur's complement theorem (Proposition 2.3.1),

$$\epsilon \left(\bar{G}_{11} - \epsilon \bar{G}_{12} (\Lambda_+ + \epsilon \bar{G}_{22})^{-1} \bar{G}_{21} \right) \succeq 0 \iff G(\epsilon) \succeq 0.$$

Therefore, $G(\epsilon) \succeq 0$, and $\text{rank}(G(\epsilon)) \geq d$, implying (3.22).

Furthermore, by (2.3).

$$\begin{aligned} \Lambda + \epsilon \bar{G}_E &= \begin{bmatrix} \epsilon \bar{G}_{11} & \epsilon \bar{G}_{12} \\ \epsilon \bar{G}_{12}^T & \Lambda_+ + \epsilon \bar{G}_{22} \end{bmatrix} \\ &\cong \begin{bmatrix} \epsilon \left(\bar{G}_{11} - \epsilon \bar{G}_{12} (\Lambda_+ + \epsilon \bar{G}_{22})^{-1} \bar{G}_{21} \right) & 0 \\ 0 & \Lambda_+ + \epsilon \bar{G}_{22} \end{bmatrix}. \end{aligned}$$

Suppose $\left(\bar{G}_{11} - \epsilon \bar{G}_{12} (\Lambda_+ + \epsilon \bar{G}_{22})^{-1} \bar{G}_{21} \right) = 0$. Since $\text{rank}(\Lambda_+ + \epsilon \bar{G}_{22}) = d$, we have

$$\text{rank}(\Lambda + \epsilon \bar{G}_E) = d.$$

Similarly if $\bar{G}_{11} = 0$, then $\bar{G}_{12} = 0$, so we get $\text{rank}(\Lambda + \epsilon \bar{G}_E) = d$ as well, proving (3.23).

2. Suppose there exists $\delta \in \mathbb{R}_{++}$ such that for all $\epsilon \in (-\delta, \delta)$, $D(\epsilon) \in \mathbb{E}^n$, and $\text{embdim}(D(\epsilon)) = d$. This is equivalent to

$$G + \epsilon G_E \succeq 0, \text{rank}(G + \epsilon G_E) = d.$$

We wish to show this implies that $\text{range}(G_E) \subseteq \text{range}(G)$. Recall that

$$\Lambda + \epsilon \bar{G}_E \cong \begin{bmatrix} \epsilon \left(\bar{G}_{11} - \epsilon \bar{G}_{12} (\Lambda_+ + \epsilon \bar{G}_{22})^{-1} \bar{G}_{21} \right) & 0 \\ 0 & \Lambda_+ + \epsilon \bar{G}_{22} \end{bmatrix}.$$

Since $\text{rank}(\Lambda_+ + \epsilon \bar{G}_{22}) = d$, so if $\text{rank}(\Lambda + \epsilon \bar{G}_E) = d$, we need

$$\text{rank} \left(\epsilon \left(\bar{G}_{11} - \epsilon \bar{G}_{12} (\Lambda_+ + \epsilon \bar{G}_{22})^{-1} \bar{G}_{21} \right) \right) = 0.$$

Thus, it must be that $\left(\bar{G}_{11} - \epsilon \bar{G}_{12} (\Lambda_+ + \epsilon \bar{G}_{22})^{-1} \bar{G}_{21} \right) = 0$.

For all $0 \neq \epsilon \in I_\epsilon \cap (-\delta, \delta)$, $(\Lambda_+ + \epsilon \bar{G}_{22}) \succ 0$, thus the above equation only hold when $\bar{G}_{11} = 0$. Then, $\bar{G}_{12} = 0$ as well, and we have $\text{range}(G_E) \subseteq \text{range}(G)$.

Let V be a facial vector for G , then $\text{range}(V)$ must contain the span of the two eigenvectors of G_E . Thus, there exists a facial vector for G of the form.

$$V = [e_{ij} \quad e_{ijc} \quad \bar{V}]$$

such that V is orthogonal. Without loss of generality, suppose $i = n - 1, j = n$, let

$$V = \begin{bmatrix} V_1 \\ V_2 \end{bmatrix}$$

where $V_1 \in \mathbb{R}^{(n-2) \times d}$, $V_2 \in \mathbb{R}^{2 \times d}$. The first column of V_1 is all 0, and the second column of V_1 is $\frac{2}{n}e$. $\text{rank}(V_1) = d - 2$, since $V_1 e = 0$.

$$G = PP^T = VRV^T = VR^{1/2}R^{1/2}V^T$$

for some $R \in \mathbb{R}_{++}^d$.

$$P = VR^{1/2} = \begin{bmatrix} V_1 R^{1/2} \\ V_2 R^{1/2} \end{bmatrix}.$$

Since $\text{rank}(V_1 R^{1/2})$ is the same as $\text{rank}(V_1)$, the first $n - 2$ rows of $P = VR^{1/2}$ are in a $d - 2$ dimension manifold.

□

Corollary 3.4.10. *Let $D \in \mathbb{E}^n$ with embedding dimension $d = 2 < n - 1$. If there exists $\delta > 0$ and $i, j \in [n]$ such that $D + \epsilon E_{ij}$ is an EDM with embedding dimension 2 for all $\epsilon \in (-\delta, \delta)$, then, $D_{st} = (D_n)_{st} = 0$ for $s \neq i, t \neq j$.*

Thus, when $d = 2$, if all off-diagonal entry of D are nonzero, then D is unyielding with fixed rank.

Proof. By 2. of Theorem 3.4.9, if $D + \epsilon E_{ij}$ is an EDM with embedding dimension 2, then all other points are in a linear manifold of dimension $d - 2 = 0$. Therefore, all other points are the same point and have distance 0 with each other. □

Corollary 3.4.11. *If $D \in \mathbb{E}^n$ with embedding dimension $d \leq n - 2$ is generated from points in general position, then D is unyielding with fixed rank.*

Proof. Let $P = [p_1, \dots, p_n]^T$ be the configuration matrix of D . If D is generated from points in general position, then every $d + 1$ points of $\{p_1, \dots, p_n\}$ is affinely independent. Thus, for all $k \leq d + 1$, the dimension of affine hull of every k points has dimension $k - 1$.

Suppose $D + \epsilon E_{ij}$ is an EDM with embedding dimension d for small ϵ , then $\dim(\text{affhull}(\{p_1, \dots, p_n\} \setminus \{p_i, p_j\})) = d - 2$. Since $n \geq d + 2$, by Proposition 3.2.6 $|\{p_1, \dots, p_n\} \setminus \{p_i, p_j\}| \geq d$, thus the dimension of $\text{affhull}(\{p_1, \dots, p_n\} \setminus \{p_i, p_j\})$ should be $\geq d - 1$, leading to a contradiction. □

Chapter 4

Solving the problem

Recall the problem of identifying the position and magnitude of noise in an EDM with a single corrupted entry. Problem 1:

Let $D \in \mathbb{E}^n$ with embedding dimension d , and let n be much greater than d . Let $i_c, j_c \in [n]$, $i_c < j_c$, and $\epsilon \neq 0$. Let

$$D_n = D + \epsilon E_{i_c j_c}.$$

Given the noisy EDM D_n and the embedding dimension d , find i, j and ϵ such that $D_n - \epsilon E_{ij}$ is an EDM with embedding dimension d .

We will also work under the assumption that n is much greater than d , and let $i_c < j_c$ to distinguish i_c and j_c . We also give the following definition.

Definition 4.0.1. Let $\mathcal{I} \subset [n]$, $|\mathcal{I}| \geq d + 1$ be an index set and suppose (i_c, j_c) is the corrupted entry. Let $G_{\mathcal{I}} = K^{\dagger}(D_n(\mathcal{I}))$.

1. We say that $D_n(\mathcal{I})$ is a good block if $G_{\mathcal{I}}$ is positive semidefinite with rank d .
2. We say that $D_n(\mathcal{I})$ is an uncorrupted good block if in addition to being a good block, we also have $i_c \notin \mathcal{I}$ and $j_c \notin \mathcal{I}$.
3. We say that $D_n(\mathcal{I})$ is a bad block if $G_{\mathcal{I}}$ is not positive semidefinite, or $\text{rank}(G_{\mathcal{I}}) > d$.
4. We say that $D_n(\mathcal{I})$ is an undetermined block if $G_{\mathcal{I}}$ is positive semidefinite, but $\text{rank}(G_{\mathcal{I}}) < d$.

In this definition, we see that if $D_n(\mathcal{I})$ is a bad block, then $\{i_c, j_c\} \subseteq \mathcal{I}$. Conversely if $\{i_c, j_c\} \subseteq \mathcal{I}$, $D_n(\mathcal{I})$ is bad block when $D(\mathcal{I})$ is unyielding. If $D(\mathcal{I})$ is yielding, then $D_n(\mathcal{I})$ may be a bad block, an undetermined block or a good block.

If points generating D are not in general position, then some principal submatrix $D(\mathcal{I})$ may have embedding dimension less than d . In this case, $D_n(\mathcal{I})$ may be uncorrupted, but be an undetermined block.

If only i_c or j_c is in $\mathcal{I}$, then $D_n(\mathcal{I})$ is a proper EDM. While we make the distinction between a good block and an uncorrupted good block, we cannot determine whether $D_n(\mathcal{I})$ is uncorrupted by checking the Gram matrix $G_{\mathcal{I}}$. This definition is introduced to assist with the analysis in Section 4.2.

4.1 Solving the Problem Under General Position Assumption

In this section, we assume $D \in \mathbb{E}^n$ is generated from points in general position in $\mathbb{R}^d$. Assume n is much greater than d . This assumption hold true in many real world applications of EDM. For example, in sensor network localization, d is often 2 or 3. n is the number of sensors, and can be a very large number.

Let $i_c, j_c \in [n]$, and let D_{i_c, j_c} be the corrupted entry, and let $\epsilon \neq 0$ be the noise. Let $i_c < j_c$ to distinguish i_c and j_c . Let $D_n = D + \epsilon E_{i_c j_c}$.

Under the assumption that D is in general position, by Proposition 3.2.7, every principal submatrix of D with size at least $d + 1$ is an EDM with embedding dimension d . Corollary 3.4.11 implies all entries are unyielding, thus D_n is not a proper EDM. Furthermore, every principal submatrix of D with size $n \geq d + 2$ is unyielding with fixed rank.

Let $\mathcal{I} \subseteq [n]$. If $\{i_c, j_c\} \not\subseteq \mathcal{I}$, then $D_n(\mathcal{I})$ is a proper EDM. Suppose $|\mathcal{I}| \geq d + 2$. If $\{i_c, j_c\} \subseteq \mathcal{I}$, then $D(\mathcal{I})$ is not a proper EDM with embedding dimension d , and we can identify it by checking $\mathcal{K}^\dagger(D_n(\mathcal{I}))$ is not positive semidefinite or $\text{rank}(\mathcal{K}^\dagger(D_n(\mathcal{I}))) \neq d$.

We present three different facial reduction methods using each of the exposing vector, facial vector, and Gale transform approaches. All three methods, follow this general idea:

Algorithm 1: Algorithm Framework

- 1 Divide the $\mathcal{I} = [n]$ into several overlapping blocks : $\mathcal{I}_1, \dots, \mathcal{I}_r$, such that $|\mathcal{I}_i| \geq d + 2$ for each i , and $|\mathcal{I}_i \cap \mathcal{I}_{i+1}| \geq d + 1$ for each $i = 1, \dots, r - 1$.
 - 2 $D_i = D_n(\mathcal{I}_i)$, $G_i = \mathcal{K}^\dagger(D_i)$.
 - 3 Check if each D_i is an EDM with embedding dimension d by checking if $G_i \succeq 0$ and $\text{rank}(G_i) = d$.
 - 4 **if** All D_i are EDMs with embedding dimension d **then**
 - 5 Use the **exposing vector/facial vector/gale matrix** of each clique D_i to form the **exposing vector/facial vector/gale matrix** of D .
 - 6 Obtain D_{complete} using Corollary 3.3.6.
 - 7 Find i_s, j_s, ϵ by checking the nonzero entry of $D_n - D_{\text{complete}}$.
 - 8 **else**
 - 9 /* There exist a D_b that is not an EDM with embedding dimension d */
 - 10 **if** The size of D_b is big, for example $|D_b| > 2d + 2$ **then**
 - 11 Go back to Line 1 and divide D_b into small blocks.
 - 12 **else**
 - 13 Search for the corrupted position i_s, j_s in the small block D_b
 - 14 Permute points so that i_s, j_s is outside of the principal diagonal blocks $D_1, \dots, D_r$.
 - 15 Go back to Line 1 and complete D to find the true value of $D_{i_s j_s}$.
-

We now provide more explanation for the algorithm above. The first step is to divide $\mathcal{I}$ into several overlapping blocks. We require the size of each block to be at least $d + 2$ to prevent yielding with fixed rank, and the overlap between each block is required to be rigid. Under the general position assumption, rigid intersection is equivalent to the overlap between two blocks being at least size $d + 1$.

In the actual implementation of the algorithm, we sometime take the overlap to be slightly greater than $d + 1$. With random data, there might be points close to being affinely dependent in some blocks, and taking a slightly larger overlap provide better stability.

If the noise is located outside of these principal diagonal blocks, then the problem can be solved as an EDMCP. With $\mathcal{I}_1, \dots, \mathcal{I}_r$ being overlapping cliques in the known partial EDM, we can use the information of these cliques to explicitly complete the EDM D_{complete} .

If we found a bad block among any of the principal diagonal blocks $D_1, \dots, D_r$, then we have narrowed the range on the position of the noise. Say D_b is found to be a bad block, then we can repeat the divide and conquer method when the size of D_b is big, or we can search each entry in this small block D_b to find the position of the noise. In the algorithmic framework, we gave an example of using $|D_b| > 2d + 2$ as a condition to continue with the subdivision, but a different condition can be used as well. As long as D_b can still be divided into at least 2 blocks of size at least $d + 2$ with an overlap of size at least $d + 1$, the block can be further subdivided. On the other hand, as long as the size of D_b is small relative to n , we can stop the subdividing procedure, and use the following small block algorithm to find the position of the noise. In this chapter, we say a block is *small* if it has size that is a linear function of d .

This leads to Line 13 in Algorithm 1. We provide the details on how we search for the small block in the following algorithm. We frequently use it as a subroutine in all methods.

Algorithm 2: Small Block Algorithm

Data: Corrupted EDM $D_n = D + \epsilon E_{i_c j_c}$, where $D \in \mathbb{E}^n$ has embedding dimension d and is in general position. $\mathcal{I}_b \subseteq \{1, \dots, n\}$, $|\mathcal{I}_b| \geq d + 1$ such that $D_n(\mathcal{I}_b)$ is not an EDM with embedding dimension d

Result: i_s, j_s such that $\{i_s, j_s\} = \{i_c, j_c\}$

```

1  $\mathcal{I}_g = (\{1, \dots, n\} \setminus \mathcal{I}_b)[1 : 2]$ .
2  $D_c = D_n(\mathcal{I}_g)$ .
3  $n_c = |\mathcal{I}_g|$ .
4  $\mathcal{I}_{\text{found}} = \emptyset$ .
5 for  $i = 1$  to  $|\mathcal{I}_b|$  do
6    $\mathcal{I}_i = (\mathcal{I}_b \setminus \{i\}) \cup \mathcal{I}_g$ .
7    $D_i = D_n(\mathcal{I}_i)$ .
8   if  $D_i$  is a good block then
9      $\mathcal{I}_{\text{found}} = \mathcal{I}_{\text{found}} \cup \{i\}$ .
10 /*  $\mathcal{I}_{\text{found}}$  must contain exactly two points that are the corrupted entries */
11  $i_s = \min(\mathcal{I}_{\text{found}})$ ,  $j_s = \max(\mathcal{I}_{\text{found}})$ .
12 return  $i_s, j_s$ .
```

Theorem 4.1.1. Given $D_n = D + \epsilon E_{i_c j_c}$ where $D \in \mathbb{E}^n$ has embedding dimension d and in general position, $i_c, j_c \in [n]$ such that $i_c < j_c$, and $\mathcal{I}_b \subseteq \{1, \dots, n\}$, $|\mathcal{I}_b| \geq d + 1$ such that $D_n(\mathcal{I}_b)$ is not an EDM with embedding dimension d , Algorithm 2 returns i_s, j_s such that $i_s = i_c$, $j_s = j_c$.

Proof. In the algorithm, we first obtain $\mathcal{I}_g = (\{1, \dots, n\} \setminus \mathcal{I}_b)[1 : 2]$ as two points known to be good. We remove points from $\mathcal{I}_b \cup \mathcal{I}_g$ one at a time and check whether the remaining points give a good block. Let $i \in \mathcal{I}_b$. We append the two points to ensure the size of $(\mathcal{I}_b \setminus \{i\}) \cup \mathcal{I}_g$ is at least $d + 2$, so that it is unyielding with fixed rank.

For any $i \in \mathcal{I}_b$, $D_i = D((\mathcal{I}_b \setminus \{i\}) \cup \mathcal{I}_g)$ does not contain the corrupted distance if, and only if, $i = i_c$ or j_c . Therefore, after iterating through all $i \in \mathcal{I}_b$, $\mathcal{I}_{\text{found}}$ must contain exactly two points i_c and j_c . Thus, the algorithm correctly output two points i_s, j_s such that $i_s = i_c, j_s = j_c$. $\square$

In the actual implementation of this algorithm in MATLAB, we append $d + 1$ points from $(\{1, \dots, n\} \setminus \mathcal{I}_b)$ for better stability. Having a slightly larger block ensure the block less likely to be almost having embedding dimension less than d .

In addition, note that this method of finding the position of noise works for a block of any size. If we apply this method on the entire matrix D_n , we can find the position of the noise, however, it will be inefficient. Under the assumption that n is much greater than d , and that the size of $\mathcal{I}_b$ is small, Algorithm 2 has asymptotic $O(1)$ runtime as $n \rightarrow \infty$. Time complexity of the algorithm is further discussed in Section 5.1

Finally, we provide the detail on Line 6 in Algorithm 1. This is also a commonly used subroutine in all methods.

Algorithm 3: Completing D From the Facial Vector

Data: $V \in \mathbb{R}^{n \times d}$ or $Z \in \mathbb{R}^{n \times n}$ such that $\text{face } \mathcal{K}^\dagger(D) = V\mathbb{S}_+^d V^T = \mathbb{S}_+^n \cap Z^\perp$, $\mathcal{I} \subseteq [n]$ such that $D_n(\mathcal{I}) \in \mathbb{E}^{|\mathcal{I}|}$ and $\text{embdim}(D(\mathcal{I})) = d$

Result: $R \in \mathbb{S}_{++}^d$ and $D_{\text{complete}} = \mathcal{K}(VRV^T)$ such that $D_{\text{complete}} = D$

- 1 $D_1 = D_n(\mathcal{I})$, $G_1 = \mathcal{K}^\dagger(D_1)$.
 - 2 $P_1 \in \mathbb{R}^{|\mathcal{I}| \times d}$ such that $P_1 P_1^T = G_1$.
 - 3 **if** We know the exposing vector instead of the facial vector **then**
 - 4 $\left[\begin{array}{l} V \in \mathbb{R}^{n \times d} \text{ whose columns forms a basis of null } \left(\begin{bmatrix} Z^T \\ e^T \end{bmatrix} \right) \end{array} \right]$
 - 5 $V_1 = V(\mathcal{I}, :)$.
 - 6 $Q = (JV_1)^\dagger P_1$.
 - 7 $R = QQ^T$.
 - 8 $D_{\text{complete}} = \mathcal{K}(VRV^T)$.
 - 9 **return** D_{complete} .
-

Theorem 4.1.2. Let $D \in \mathbb{E}^n$ with embedding dimension d . Given $V \in \mathbb{R}^{n \times d}$ such that $\text{face } \mathcal{K}^\dagger(D) = V\mathbb{S}_+^d V^T$ and given $\mathcal{I} \subseteq [n]$ such that $D_n(\mathcal{I}) \in \mathbb{E}^{|\mathcal{I}|}$ with $\text{embdim}(D(\mathcal{I})) = d$, then Algorithm 3 finds R such that $D_{\text{complete}} = \mathcal{K}(VRV^T) = D$.

If instead we are given a exposing vector $Z \in \mathbb{R}^{n \times n}$ such that $\text{face } \mathcal{K}^\dagger(D) = \mathbb{S}_+^n \cap Z^\perp$, Algorithm 3 first finds a facial vector V such that $\text{face } \mathcal{K}^\dagger(D) = V\mathbb{S}_+^d V^T = \mathbb{S}_+^n \cap Z^\perp$.

Proof. It directly follows from Corollary 3.3.6 that D_{complete} equals D . $\square$

We can choose $\mathcal{I}$ to be any set such that $D_n(\mathcal{I})$ is an EDM with embedding dimension d . In particular, we can choose a block with good condition number. For example, we can calculate the condition number of the principal submatrices $D_1, \dots, D_r$ and choose the block with minimum condition number for Algorithm 3.

For an EDM D , we use the value of the maximum eigenvalue of $\mathcal{K}^\dagger(D)$ divided by the minimum positive eigenvalue of $\mathcal{K}^\dagger(D)$ as its condition number. Thus, a large condition number implies the

minimum positive eigenvalue is close to 0, implying D is close to having embedding dimension less than d .

In the next few sections, we apply the above algorithmic framework using each of the three different facial reduction methods. Furthermore, we experiment with several methods of dividing the blocks:

1. Bisection. We divide $\mathcal{I}$ into two blocks overlapping in a block with size $d + 1$.
2. Maximum number of block with minimal overlap. We divide blocks into $\mathcal{I}_1, \dots, \mathcal{I}_r$ such that $|\mathcal{I}_i| = 2(d + 1)$ and $|\mathcal{I}_i \cap \mathcal{I}_{i+1}| = d + 1$. Each overlapped section is only used in two blocks, thus we call this the “multi-block” method.
3. Minimum block size. We divide each block into block of size $d + 2$, the minimum possible size.

While we can pair any division method with any facial reduction method, we will see in Section 5.1 that there is a benefit to runtime for dividing blocks into small size when using the facial vector method.

4.1.1 Bisection Method With Exposing Vector (BVE)

We use a bisection division method and solves the problem using exposing vectors in this section.

Algorithm 4: Bisection Method With Exposing Vector

Data: $D_n = D + \epsilon E_{i_c j_c}$. D an EDM with embedding dimension d . Points that form D are in general position

Result: i_s, j_s and ϵ_s such that $D_n - \epsilon_s E_{i_s j_s}$ is an EDM with embedding dimension d .

```

1  $\mathcal{I}_c = [1 : n]$ .
2  $D_c = D_n(\mathcal{I}_c)$ .
3  $n_c = |\mathcal{I}|$ .
4  $\mathcal{I}_1 = [1 : \lceil \frac{n_c+d+2}{2} \rceil]$ ,  $\mathcal{I}_2 = [\lfloor \frac{n_c-d-2}{2} \rfloor : n]$ . for  $i = 1, 2$  do
5    $D_i = D_c(\mathcal{I}_i)$ ,  $G_i = \mathcal{K}^\dagger(D_i)$ .
6   if  $D_i$  is a bad block then
7     if  $|\mathcal{I}_i| \leq 2d + 2$  then
8       Run small block algorithm Algorithm 2, and obtain the corrupted index  $i_s, j_s$ .
9        $\mathcal{I}_g = \mathcal{I}_c \setminus \{i_s, j_s\}$ .
10       $\mathcal{I}_c = [i_s, \mathcal{I}_g, j_s]$ .
11      Run Line 2 to Line 24 to obtain  $D_{\text{complete}}$ .
12       $\epsilon_s = \text{nonzero entry in } (D_c - D_{\text{complete}})$ .
13      return  $i_s, j_s, \epsilon_s$ .
14   else
15      $\mathcal{I}_c = \mathcal{I}_i$ .
16     Go back to Line 2.
17   else
18      $Z'_i$  = matrix formed from a centered set of orthonormal basis vector for  $\text{null}(G_i)$ .
19     i.e.,  $Z'_i G_i = 0$ ,  $Z_i'^T e = 0$ ,  $Z_i'^T Z'_i = I$ .
20      $Z_i = Z'_i Z_i'^T$  is the centered exposing vector.
21      $W_i = \mathcal{P}_{\mathcal{I}_i}^*(Z_i)$ , so  $W_i(\mathcal{I}_i) = Z_i$  and zero everywhere else.
22  $Z = W_1 + W_2$ .
23  $V \in \mathbb{R}^{n \times d}$  whose column forms a basis of  $\text{null} \left( \begin{bmatrix} Z^T \\ e^T \end{bmatrix} \right)$ .
24 Run Algorithm 3 and obtain  $D_{\text{complete}}$ .
25 Find  $i_s, j_s$  to be the index of none zero entry in  $D_c - D_{\text{complete}}$ , and
     $\epsilon_s = (D_c - D_{\text{complete}})_{i_s, j_s}$ .
26 return  $i_s, j_s, \epsilon_s$ .
```

Theorem 4.1.3. Let $D \in \mathbb{E}^n$ with embedding dimension d and be in general position. Let $i_c, j_c \in [n]$ such that $i_c < j_c$, $\epsilon \in \mathbb{R}$, and let $D_n = D + \epsilon E_{i_c j_c}$. Algorithm 4 returns i_s, j_s and ϵ_s such that $D_n - \epsilon E_{i_s j_s} = D$.

Proof. Line 4 divides $\mathcal{I}_c$ into two overlapping blocks with an overlap of size $d + 1$, and the size of each block is at least $d + 2$. Since D is in general position, if both blocks are good, then the overlap is good as well. Thus, the overlap between D_1 and D_2 is rigid. The algorithm calculates the exposing vector for each block, and it follows from Theorem 3.3.7 and the analysis in Section 3.3.1

that the above algorithm builds an exposing vector for D , and so we can obtain $D_{\text{complete}} = D$ by Theorem 4.1.2. Since $D_{\text{complete}} = D$, by looking at the nonzero entry of $D_n - D_{\text{complete}}$, we obtain the position and magnitude of the noise.

If one of the two blocks is bad, we can reduce the size of the problem and repeat the process on the bad block. If the size of the bad block is too small to further subdivide, we use Algorithm 2 to find the position of the noise. Then, permuting the index according to Line 9 ensures the corrupted entry is outside of the principal diagonal blocks. Then, we can repeat the process and obtain the exposing vector for D and calculate D_{complete} . $\square$

4.1.2 Multi-Block Method With Facial Vector (MBFV)

In this section, we divide D_n into the maximum number of blocks with minimal overlap. We divide each block to have size $2d + 2$, and the overlap with the adjacent has size $d + 1$. Since the starting size of each block is already small, we do not further subdivide and call on Algorithm 2 when we

encounter a bad block.

Algorithm 5: Multi-Block Facial Vector Algorithm

Data: $D_n = D + \epsilon E_{i_c j_c}$. D an EDM with embedding dimension d . D is in general position.

Result: i_s, j_s and ϵ_s such that $D_n - \epsilon_s E_{i_s j_s}$ is an EDM with embedding dimension d .

```

1  $\mathcal{I}_c = [n]$ .
2  $n_c = |\mathcal{I}_c|$ .
3  $s_1 = 1, e_1 = 2d + 2$ .
4  $\mathcal{I}_1 = \mathcal{I}_c([s_1 : e_1])$ ,  $D_1 = D_n(I_1)$ ,  $G_1 = \mathcal{K}^\dagger(D_1)$ .
5 if  $D_1$  is a good block then
6    $V_1$  = matrix formed eigenvector associated with positive eigenvalues (This is the facial
   vector for the block).
7 else
8   Record we have found a bad block with index  $\mathcal{I}_b = \mathcal{I}_1$ .
9 while have not found a bad block and  $e_1 < n_c$  do
10    $s_2 = e_1 - d, e_2 = \min(n_c, s_2 + 2d + 1)$ .
11    $\mathcal{I}_2 = \mathcal{I}_c([s_2 : e_2])$ .
12    $D_2 = D_n(I_2)$ .
13   if  $D_2$  is a good block then
14      $V_2$  = matrix formed eigenvector associated with positive eigenvalues.
15     Append the vector of all ones to  $V_1$  and  $V_2$ .
16     Partition  $V_1, V_2$  as  $V_1 := \begin{matrix} d+1 \\ d+1 \end{matrix} \begin{bmatrix} V_1' \\ V_1'' \end{bmatrix}$ ,  $V_2 := \begin{matrix} d+1 \\ |\mathcal{I}_2| - d - 1 \end{matrix} \begin{bmatrix} V_2'' \\ V_2' \end{bmatrix}$ .
17     Form the facial vector for  $D_n(I_1 \cup I_2)$  by  $V = \begin{bmatrix} V_1'' \\ V_1' \\ V_2'(V_2'')^\dagger V_1'' \end{bmatrix}$  or  $V = \begin{bmatrix} V_2'' \\ V_1'(V_1'')^\dagger V_2'' \\ V_2' \end{bmatrix}$ .
     Remove the column of all ones. Update  $V_1 = V$ ,  $e_1 = e_2$ .
18   else
19     Record we found a bad block with index  $\mathcal{I}_b = \mathcal{I}_2$ .
20 if no bad block is found then
21   /*  $V$  is the facial vector for  $D$ . */
22    $D_{\text{complete}}$  using Algorithm 3 Find  $i_s, j_s$  to be the index of nonzero entry in
    $D_n - D_{\text{complete}}$ ,  $\epsilon_s = (D_n - D_{\text{complete}})_{i_s, j_s}$ .
23 else
24   /* a bad block is found */
25   Run Algorithm 2, obtain the corrupted index  $i_s, j_s$ .
26    $\mathcal{I}_g = \mathcal{I}_c \setminus \{i_s, j_s\}$ .
27    $\mathcal{I}_c = [i_s, \mathcal{I}_g, j_s]$ .
28   Run Line 2 to Line 22 and obtain  $D_{\text{complete}}$ .
29   Recover  $\epsilon_s$  as the nonzero entry in  $(D_n - D_{\text{complete}})$ .
30 return  $i_s, j_s, \epsilon_s$ .
```

Theorem 4.1.4. Let $D \in \mathbb{E}^n$ with embedding dimension d and in general position, $i_c, j_c \in [n]$ such that $i_c < j_c$, and $\epsilon \in \mathbb{R}$. Let $D_n = D + \epsilon E_{i_c j_c}$. Algorithm 5 finds i_s, j_s , and ϵ_s such that $D_n - \epsilon E_{i_s j_s} = D$.

Proof. Algorithm 5 first divides D_n into many overlapping blocks of size $2d + 2$, and every two consecutive blocks overlap in a block of size $d + 1$. If any of the small blocks is bad, Algorithm 2 finds the position of the error. Then, we permute it so that the corrupted position is not in any of the small principal submatrices.

Suppose all the small blocks are good. Since D is in general position, the overlap between any two consecutive block is rigid. By Lemma 3.3.4, we can apply Lemma 3.3.2 and Theorem 3.3.3 repeatedly to build the facial vector for D . Then, we use Algorithm 3 to obtain $D_{\text{complete}} = D$, so we can obtain the position and magnitude of the noise. $\square$

In numerical testing, we observed that increasing this overlap size slightly can increase the accuracy of the algorithm, and use an overlap of size $d + 3$. When the overlap size is only $d + 1$, the overlap may be close to being in embedding dimension $d - 1$, leading to a higher error in the final result.

4.1.3 Small Blocks With Gale Transform (SBGT)

The minimum block size we can divide into is having each block with size $d + 2$.

Algorithm 6: Gale Transform Algorithm

Data: $D_n = D + \epsilon E_{i_c j_c}$, where D an EDM with embedding dimension d and D is in general position

Result: i_s, j_s and ϵ such that $D_n - \epsilon_s E_{i_s j_s}$ is an EDM with embedding dimension d .

```

1  $\mathcal{I}_c = [1 : n]$ .
2  $N \in \mathbb{R}_{n \times n-d-1}$ .
3 while  $i = 1$  to  $n - d$  and have not found a bad block do
4    $D_i = D_n(\mathcal{I}_c[i : d + i + 1])$ ,  $G_i = \mathcal{K}^\dagger(D_i)$ .
5   if  $D_i$  is a bad block then
6     Run the Algorithm 2 and obtain output  $i_s, j_s$ .
7      $\mathcal{I}_g = \mathcal{I}_c \setminus \{i_s, j_s\}$ .
8      $\mathcal{I}_c = [i_s, \mathcal{I}_g, j_s]$ .
9     Run Line 2 to Line 17 of this algorithm to obtain  $D_{\text{complete}}$ .
10    Recover  $\epsilon_s$  as the nonzero entry in  $(D_n - D_{\text{complete}})$ .
11    return  $i_s, j_s, \epsilon_s$ .
12  else
13    Calculate the Gale matrix  $N_i$  for  $D_i$ .
14     $N([i : i + d + 1], i) = N_i$ .
15 /* Built the full Gale matrix successfully */
16  $V \in \mathbb{R}^{n \times d}$  whose columns forms a basis of null  $\left( \begin{bmatrix} N^T \\ e^T \end{bmatrix} \right)$ .
17 Run Algorithm 3 to obtain  $D_{\text{complete}}$ .
18 Find  $i_s, j_s$  to be the index of none zero entry in  $D_n - D_{\text{complete}}$ , and
    $\epsilon_s = (D_n - D_{\text{complete}})_{i_s, j_s}$ .
19 return  $i_s, j_s, \epsilon_s$ .
```

Theorem 4.1.5. *Let $D \in \mathbb{E}^n$ with embedding dimension d and in general position, $i_c, j_c \in [n]$ such that $i_c < j_c$, and $\epsilon \in \mathbb{R}$. Let $D_n = D + \epsilon E_{i_c j_c}$. Algorithm 6 finds i_s, j_s , and ϵ_s such that $D_n - \epsilon E_{i_s j_s} = D$.*

Proof. For each $i = 1, \dots, n - d$, let $D_i = D_n([i : d + i + 1])$. Thus, each D_i has size $d + 2$ and overlaps with D_{i-1} and D_{i+1} is a block of size $d + 1$. Since D is in general position, the overlap between any two consecutive blocks is rigid.

If any of the small blocks is bad, Algorithm 2 obtains the position of the error. Then, we permute the points so the corrupted position is not in any of the principal submatrices.

Suppose all D_i s are good blocks, then we apply Theorem 3.3.9 repeatedly and obtain the Gale matrix for D . Recall that the Gale matrix is related to the exposing vector by Theorem 3.3.8. Thus, this process is equivalent to adding up exposing vectors in Algorithm 4. From the Gale matrix, we can obtain the facial vector, and use Algorithm 3 to obtain $D_{\text{complete}} = D$. Thus, we can obtain the position and magnitude of the noise. $\square$

Notice that we can also choose a single clique $D(\mathcal{I})$ with the best condition number, and build the minimum sized blocks as $D(\mathcal{I} \cup \{i\})$ for each $i \notin \mathcal{I}$. In numerical testing, we find this method provides higher accuracy.

4.2 Solving the Problem Without General Position Assumption

We now remove the assumption that points are in general position, and adjust our algorithm to solve the problem. We call the problem without general position assumption the *hard* case. We relax the assumption of general position to that there exists one uncorrupted good principal diagonal block of size $d + 1$. Note that if $D \in \mathbb{E}^n$ with embedding dimension d is in general position, then the corrupted EDM D_n has an uncorrupted good principal diagonal block of size $d + 1$.

In addition, we impose the restriction that points generating D are distinct. Thus, all nondiagonal elements of D are nonzero. This is a reasonable restriction to make on the data. If two points generating D are the same, we may consider them as one single point. By Corollary 3.4.10, when the embedding dimension is 2, this assumption ensures any principal submatrix of D is unyielding with fixed rank.

For an EDM $D \in \mathbb{E}^n$ to contain a yielding with fixed rank entry D_{ij} , all but points i and j must be in a affine space of dimension $d - 2$. Thus, if the D_n contains one uncorrupted good principal diagonal block of size $d + 1$, the case of yielding with fixed rank cannot occur.

Furthermore, all three facial reduction algorithms require overlapping cliques with rigid intersection, and rigid intersection is the only requirement on the cliques. Therefore, with this assumption that there exists an uncorrupted good block, we can always solve the problem with the facial reduction algorithms.

We have the adjusted algorithm framework:

Algorithm 7: Hard Case Algorithm Framework - No General Position Assumption

- 1 Search for a good block of size at least $d + 1$: $\mathcal{I}_{\text{good}} \mathcal{I}_{\text{rest}} = [n] \setminus \mathcal{I}_{\text{good}}$.
 - 2 Divide the $\mathcal{I}_{\text{rest}}$ into several disjoint blocks : $\mathcal{I}_1, \dots, \mathcal{I}_r$, such that $|\mathcal{I}_i| \geq 1$ for each i .
 - 3 $\mathcal{I}'_i = \mathcal{I}_i \cup \mathcal{I}_{\text{good}}$ (Then, $D_n(\mathcal{I}'_i \cap \mathcal{I}'_j)$ is a good block for each i, j).
 - 4 $D_i = D_n(\mathcal{I}'_i)$, $G_i = \mathcal{K}^\dagger(D'_i)$.
 - 5 Check if each D_i is an EDM with embedding dimension d by checking if $G_i \succeq 0$ and $\text{rank}(G_i) = d$.
 - 6 **if** all D_i are EDMs with with embedding dimension d **then**
 - 7 Use the **exposing vector/facial vector/gale matrix** of each clique D_i to form the **exposing vector/facial vector/gale matrix** of D .
 - 8 Obtain D_{complete} using Corollary 3.3.6 . Find i_s, j_s, ϵ by checking the nonzero entry of $D_n - D_{\text{complete}}$.
 - 9 **else**
 - 10 /* There exist a $\mathcal{I}'_b$ such that $D_n(\mathcal{I}'_b)$ that is not an EDM with embedding dimension d */
 - 11 **if** The size of D_b is big, for example $|D_b| > 2d + 2$ **then**
 - 12 Go back to Line 2 and divide $\mathcal{I}_b \setminus \mathcal{I}_{\text{good}}$ into small blocks.
 - 13 **else**
 - 14 Search for the position of the corrupted entry i_s, j_s in the small block.
 $\mathcal{I}' = [n] \setminus \{i_s, j_s\}$ Search for a new good block $\mathcal{I}'_{\text{good}}$ from $D_n(\mathcal{I}')$, this new good block is an uncorrupted good block.
 - 15 Permute points so that i_s, j_s is outside of the principal diagonal blocks.
 - 16 Go back to Line 2 and complete D to find the true value of $D_{i_s j_s}$.
-

Details on how to find a good block are discussed in Section 4.2.1. We wish to find an uncorrupted good block as the rigid intersection between blocks. However, as noted in the beginning of Chapter 4, we cannot distinguish a good block from an uncorrupted good block by checking the rank of the Gram matrix.

Thus, we first find a good block $\mathcal{I}_{\text{good}}$ using Algorithm 9, which we discuss in more details in the next section. Then, we divide the $\mathcal{I}_{\text{rest}}$ into several disjoint blocks : $\mathcal{I}_1, \dots, \mathcal{I}_r$, such that $|\mathcal{I}_i| \geq 1$ for each i , and let $\mathcal{I}'_i = \mathcal{I}_i \cup \mathcal{I}_{\text{good}}$. Hence, $\mathcal{I}'_i \cap \mathcal{I}'_j = \mathcal{I}_{\text{good}}$ for all i, j , and the intersection is rigid.

If $\mathcal{I}_{\text{good}}$ is not an uncorrupted good block, without loss of generality, say $i_c \in \mathcal{I}_{\text{good}}$. Then, let $\mathcal{I}_k$ be the block with $j_c \in \mathcal{I}_k$, then $\mathcal{I}'_k$ would be a bad block. Then, we can search the small bad block and identify i_c, j_c . Finally, we can exclude i_c, j_c and search for an uncorrupted good block. Using this as the intersection with all other points, we can obtain $D_{\text{complete}} = D$ and solve the problem.

Another adjustment we need to make is the algorithm we use to find the position of the noise in a small block. The original algorithm Algorithm 2 rely on the fact that any block of size $d + 2$

not containing the corrupted entry is a good block.

Algorithm 8: Hard Case Small Block Algorithm - No General Position Assumption

Data: Corrupted EDM $D_n = D + \epsilon E_{i_c j_c}$, where $D \in \mathbb{E}^n$ has embedding dimension d and contains an uncorrupted good block. $\mathcal{I}_b \subseteq \{1, \dots, n\}$ such that $\{i_c, j_c\} \subseteq \mathcal{I}_b$

Result: i_s, j_s such that $\{i_s, j_s\} = \{i_c, j_c\}$

```

1  $\mathcal{I}' = \{1, \dots, n\} \setminus \mathcal{I}_b$ .
2  $D_c = D_n(\mathcal{I}')$ .
3  $n_c = |\mathcal{I}'|$ .
4  $\mathcal{I}_{found} = \emptyset$ .
5 while  $i = 1$  to  $n_c - d$  and good block not found do
6    $D_i = D_c([i : d + i])$ .
7   if  $D_i$  is a good block then
8      $\mathcal{I}_g = [i : d + i]$ .
9 if Cannot find a good block then
10    $\mathcal{I}_g = \mathcal{I}'$ .
11 for  $i = 1$  to  $|\mathcal{I}_b|$  do
12    $\mathcal{I}_i = (\mathcal{I}_b \setminus \{i\}) \cup \mathcal{I}_g$ .  $D_i = D_n(\mathcal{I}_i)$ .
13   if  $D_i$  is a good block then
14      $\mathcal{I}_{found} = \mathcal{I}_{found} \cup \{i\}$ .
15 /*  $\mathcal{I}_{found}$  must contain exactly two points that are the corrupted entries */
16 return  $\mathcal{I}_{found} = \{i_s, j_s\}$ .
```

Theorem 4.2.1. Let $D_n = D + \epsilon E_{i_c j_c}$ where $D \in \mathbb{E}^n$ has embedding dimension d and in contains an uncorrupted good block. Let $i_c, j_c \in [n]$ such that $i_c < j_c$, and $\mathcal{I}_b \subseteq \{1, \dots, n\}$, $|\mathcal{I}_b| \geq d + 1$ such that $D_n(\mathcal{I}_b)$ is not an EDM with embedding dimension d . Algorithm 8 returns i_s, j_s such that $i_s = i_c, j_s = j_c$.

Proof. First, we check all consecutive principal diagonal blocks of size $d + 1$ in the remaining entries. If we find a good block $\mathcal{I}_g$, then $\mathcal{I}_g$ is an uncorrupted good block. We remove points in $\mathcal{I}_b$ from $\mathcal{I}_b \cup \mathcal{I}_g$ one at a time. $D((\mathcal{I}_b \setminus \{i\}) \cup \mathcal{I}_g)$ always contains an uncorrupted good block and hence is unyielding with fixed rank. The same argument as the original small block algorithm applies. If removing i causes $D_n((\mathcal{I}_b \setminus \{i\}) \cup \mathcal{I}_g)$ to become a proper EDM, then i is a part of the corrupted entry.

If none of the consecutive principal diagonal blocks of size $d + 1$ has embedding dimension d , then we use the entire $D_n(\{1, \dots, n\} \setminus \mathcal{I}_b)$ as the uncorrupted good block. This means we need to compute the eigenvalue decomposition for a large matrix. However, $|\mathcal{I}_b|$ is small, we do not need to perform this computation often, so it does not lead to a increase in runtime. $\square$

4.2.1 Identifying a Good Block

Assuming there exists an uncorrupted good block, we use the following algorithm to find a good block. In the process of finding a small good block, we may also encounter a bad block, in which

case, we return this bad block, and proceed to Line 14 in Algorithm 7.

Algorithm 9: Finding a Small Good Block

Data: $D_n = D + \epsilon E_{i_c j_c}$. $D \in \mathbb{E}^n$ an EDM with embedding dimension d . D_n contains a principal submatrix of size $d + 1$ that is a uncorrupted good block.

Result: Either a good block of size $d + 1$ or a bad block of size $d + 1$

```

1  $\mathcal{I}_c = [1 : n]$ .
2  $D_c = D_n(\mathcal{I}_c)$ .
3  $n_c = |\mathcal{I}_c|$ .
4  $\mathcal{I} = \emptyset$ .
5 while the first block is not chosen and have not found a bad block do
6   while  $i = 1$  to  $n_c - d$  and have not found a bad block do
7      $D_i = D_c([i : d + i])$ .
8      $G_i = \mathcal{K}^\dagger(D_i)$ .
9     if  $D_i$  is a good block then
10       Record we found a good block.
11       Record the condition number of  $G_i$ .
12     else if  $D_i$  is a bad block then
13       Record we found a bad block with index  $\mathcal{I}_b$  and exit.
14     else
15       Calculate the Gale matrix  $N_i = \begin{bmatrix} -n_i^T - \\ \vdots \\ -n_{i+d}^T - \end{bmatrix}$  for  $G_i$ .
16        $\mathcal{I} = \mathcal{I} \cup \{s : n_s = 0 \text{ in } N_i\}$ .
17   if there is a good block then
18     Pick  $D_i$  with the best condition number.
19     Record that we have chosen a good block and exist the loop.
20   else if there is no good block and no bad block then
21      $\mathcal{I}_r = \mathcal{I}_c([1 : d + 1]) \setminus \mathcal{I}$ .
22      $\mathcal{I}_c = [\mathcal{I}, \mathcal{I}_r]$ .
23      $\mathcal{I} = \emptyset$ .
24      $D_c = D_n(\mathcal{I}_c)$ .
25      $n_c = |\mathcal{I}_c|$ .
26     Continue with the loop.
```

Theorem 4.2.2. Let $D_n = D + \epsilon E_{i_c j_c}$, where $D \in \mathbb{E}^n$ an EDM with embedding dimension d . Suppose D contains a principal submatrix of size $d + 1$ that is a uncorrupted good block. Then, Algorithm 9 either returns a bad block of size $d + 1$, or a good block of size $d + 1$.

Proof. First, we check all consecutive principal submatrices of size $d + 1$. If any of the consecutive principal submatrices has embedding dimension d , we have found a good block.

Otherwise, all consecutive principal submatrices of size $d + 1$ have affine dependency among their generating points. By Theorem 3.2.4, by selecting the index set corresponding to rows of zeros in Line 14, we identify points that are not affine combinations of the other points in the block.

Conduct the procedure on all consecutive principal diagonal blocks of size $d + 1$. $\mathcal{I}$ is an index list with no element repeated, where $s \in \mathcal{I}$ if there is some $i \in \{1, \dots, n_c - d\}$ such that $s \in \{i, i + 1, \dots, i + d\}$ and $n_s = 0$ in the corresponding Gale matrix N_i . The order of this index list is in the order we add into it.

Next, consider Line 21 of the algorithm. We let $\mathcal{I}_r$ be the points in $\mathcal{I}_c(1 : d + 1)$ that are not in $\mathcal{I}$, and append it at the end of $\mathcal{I}$. Let $\mathcal{I}_c = [\mathcal{I}, \mathcal{I}_r]$. Then, we repeat the process on $D_n(\mathcal{I}_c)$. Through this process, if we do not encounter any bad blocks, we must eventually obtain at least $d + 1$ points that are affinely independent.

First, we note that $\mathcal{I} \neq \emptyset$. If $\mathcal{I} = \emptyset$, but no good block was found, then all points are in a linear manifold spanned by the first $d + 1$ points, which has dimension less than the embedding dimension d . Therefore, the minimum length for $\mathcal{I}_c$ is $d + 1$.

To prove the correctness of the algorithm, we show that:

1. At any iteration, D_c contains a principal submatrix of size $d + 1$ such that the corresponding submatrix in the uncorrupted D is an EDM with embedding dimension d .
2. The algorithm terminates.

Proof of 1: By assumption, claim 1 holds for the first iteration. Inductively, suppose that at the beginning of some iteration, claim 1 holds, and we have not found any good block in this iteration. Let $\mathcal{I}'_c$ denote the index list from the previous iteration, and $\mathcal{I}_c$ the new index list built in the current iteration. $\mathcal{I}_c$ is $\mathcal{I}$ plus the remaining points among the first $d + 1$ points of $\mathcal{I}'_c$. In particular, $\mathcal{I}'_c(1 : d + 1) \subseteq \mathcal{I}_c$. Since we have not found a good block, the embedding dimension of the first $d + 1$ points is less than d .

For $k \geq d + 1$, consider $D(\mathcal{I}'_c([1 : k]))$. Suppose $D(\mathcal{I}'_c([1 : k]))$ has embedding dimension $r < d$, then $\text{embdim}(D(\mathcal{I}'_c([1 : k + 1]))) = r + 1$ or r .

In the first case, we call $k + 1$ a *dimension increasing point*. In this case, we have

$$p_{\mathcal{I}'_c(k+1)} \notin \text{affhull}(\{p_i : i \in \mathcal{I}'_c([1 : k])\}).$$

Thus,

$$p_{\mathcal{I}'_c(k+1)} \notin \text{affhull}(\{p_i : i \in \mathcal{I}'_c([k + 1 - d : k])\}),$$

so $n_{k+1} = 0$ in N_{k+1-d} and $p_{\mathcal{I}'_c(k+1)} \in \mathcal{I} \subseteq \mathcal{I}_c$. In addition, $p_{\mathcal{I}'_c(k+1)} \notin \text{affhull}(\mathcal{I}_c \cap \{p_i : i \in \mathcal{I}'_c([1 : k])\})$. Thus, $D(\mathcal{I}_c)$ contains an EDM with embedding dimension $r + 1$.

By the inductive hypothesis, there is a good block, so eventually there must be some $k' > d + 1$ such that $D(\mathcal{I}'_c([1 : k']))$ is an EDM with embedding dimension d . Let k' be the first index for this to happen. Therefore, $D(\mathcal{I}'_c([1 : k' - 1]))$ has embedding dimension less than d . The argument above shows $k \in \mathcal{I}_c$ and that $D(\mathcal{I}_c)$ contains points in a manifold of dimension d .

Next we show that the algorithm will terminate. *Proof of 2:* Let $n'_c = |\mathcal{I}'_c|$. We have that $\mathcal{I}_c = [\mathcal{I}, \mathcal{I}_r]$, where $\mathcal{I}_r = \mathcal{I}'_c([1 : d + 1]) \setminus \mathcal{I}$. Suppose $|\mathcal{I}_c| = |\mathcal{I}'_c|$, so the size does not decrease in this iteration, and never decreases in future iterations.

The only case for this to happen is when all of $\mathcal{I}'_c([d + 2 : n'_c])$ are in $\mathcal{I}$. Let $k = \max\{d + 2, n'_c - d\}$, and consider $\mathcal{I}'_c([k : n'_c])$. If $n_c \geq 2(d + 1)$, then $\mathcal{I}'_c([k : n'_c])$ is the last $d + 1$ points, otherwise, it is the set of points after $d + 1$.

We claim that

$$\forall i \in \mathcal{I}_c([k : n'_c]), p_i \notin \text{affhull}(\{p_j : j \in I_c([k : n'_c]) \setminus i\}).$$

The base case is $i = n'_c$. Since the last point is in $\mathcal{I}$, it must be selected from the last block. Thus, it is not in the affine hull of $\{p_j : j \in I_c[k : n'_c - 1])\}$.

Inductively, suppose for some $k \leq i < n'_c$, the property holds for all $j > i$. Then, by the inductive hypothesis,

$$p_{\mathcal{I}(i)} \notin \text{affhull}(\{p_j : j \in \mathcal{I}'_c([i + 1, n'_c])\}).$$

Suppose for contradiction that $p_{\mathcal{I}(i)} = \sum_{j=i+1}^{n'_c} \lambda_j p_{\mathcal{I}'_c(j)}$ for some $\lambda_{i+1}, \dots, \lambda_{n'_c}$ such that $\sum_{j=i+1}^{n'_c} \lambda_j = 1$. Then for some r such that $\lambda_r \neq 0$, $p_{\mathcal{I}(i)} = \frac{1}{\lambda_r} \left(p_{\mathcal{I}(i)} - \sum_{j \neq r} \lambda_j p_{\mathcal{I}'_c(j)} \right)$, contradicting the inductive hypothesis.

Furthermore, any consecutive block of size $d + 1$ containing i must also contain all points in $[n'_c - d : i - 1]$. Since i is selected, it must be that

$$p_{\mathcal{I}'_c(i)} \notin \text{affhull}(\{p_j : j \in \mathcal{I}'_c([n'_c - d, i - 1])\}).$$

Therefore,

$$p_{\mathcal{I}'_c(i)} \notin \text{affhull}(\{p_j : j \in \mathcal{I}'_c([k : n'_c]) \setminus i\}).$$

Thus, $\{p_i : i \in \mathcal{I}'_c([k : n'_c])\}$ is affinely independent. But the last $d + 1$ points cannot be affinely independent, otherwise it becomes a good block. Thus, $k = d + 2 > n'_c - d$. If the size does not decrease in this iteration, then $n'_c < 2(d + 2)$.

At this point, since the block size is small, we can check whether $\mathcal{K}^\dagger(D_n(\mathcal{I}_c))$ is a good/bad block and return this block as a good/bad block.

However, we can also continue with the process and must be able to obtain a good/bad block of size $d + 1$ after a finite amount of iteration. Let $h = n'_c - d - 1$. Split $\mathcal{I}'_c$ into $\mathcal{I}_1 = \mathcal{I}'_c([1 : h])$, $\mathcal{I}_2 = \mathcal{I}'_c([h + 1 : d + 1])$, and $\mathcal{I}_3 = \mathcal{I}'_c([d + 2 : n'_c])$. Thus, $|\mathcal{I}_1, \mathcal{I}_2| = d + 1$ and $|\mathcal{I}_2, \mathcal{I}_3| = d + 1$. $\text{embdim}(D([\mathcal{I}_2, \mathcal{I}_3])) < d$. We already demonstrated $\{p_i : i \in \mathcal{I}_3\}$ is affinely independent, therefore, $\{p_i : i \in \mathcal{I}_2\}$ must be affinely dependent. Furthermore, any block of size $d + 1$ containing any index in $\mathcal{I}_2$ must contain all indices in $\mathcal{I}_2$. Therefore, not all $i \in \mathcal{I}_2$ can be in $\mathcal{I}$.

Next, consider the dimension increasing points in $\mathcal{I}_3$. If i is a dimension increasing point, then $p_i \notin \text{affhull}(\{p_j : j \in \mathcal{I}'_c([1 : i - 1])\})$, and we know $p_i \notin \text{affhull}(\{p_j : j \in \mathcal{I}'_c([i + 1 : n'_c])\})$. Therefore, the dimension increasing points in one iteration must always be selected in $\mathcal{I}$ in future iterations.

If there are no dimension increasing points in $\mathcal{I}_3$. Then it must be that $\text{embdim}(D(\mathcal{I}'_c([1 : d + 1]))) = d$, otherwise $\text{embdim}(D(\mathcal{I}'_c)) < d$ contradicting that $D(\mathcal{I}_c)$ always contains a good block. When a point is selected in $\mathcal{I}$, it moves to the front of $\mathcal{I}_c$. Combining the fact that the dimension increasing points are always in $\mathcal{I}$ and not all points in $\mathcal{I}_2$ can be in $\mathcal{I}$, all dimension increasing points in $\mathcal{I}_3$ must eventually move to the first $d + 1$ positions in at most h steps. Then, the first block is a good block of size $d + 1$ and the algorithm terminates. $\square$

4.2.2 Hard Case MBFV

We provide an example of adjusting the MBFV algorithm to solve the problem without the general position assumption. The other methods can also be adjusted based on Algorithm 7.

Algorithm 10: Hard Case Facial Vector Algorithm- No General Position Assumption

Data: $D_n = D + \epsilon E_{i_c j_c}$. D an EDM with embedding dimension d and contains a principal submatrix of size $d + 1$ that is a uncorrupted good block.

Result: i_s, j_s and ϵ_s such that $D_n - \epsilon_s E_{i_s j_s} = D$.

```

1 Run Algorithm 9.
2 if  $A$  bad block is found then
3   Run Algorithm 8, and obtain the corrupted position  $i_s, j_s$ .
4   Exclude  $i_s, j_s$  and run Algorithm 9 to find an uncorrupted good block  $\mathcal{I}_g$ .
5   Let  $\mathcal{I}' = [1 : n] \setminus [\mathcal{I}_g, i_s, j_s]$ , and permute points into the order  $[\mathcal{I}_g, i_s, \mathcal{I}', j_s]$ .
6 else
7   Permute the points so the good block is at the beginning.
8  $I_{\text{rest}} = [d + 1 : n]$ .
9  $s_1 = 1, e_1 = d + 1$ .
10  $\mathcal{I}_1 = [\mathcal{I}_g, \mathcal{I}_{\text{rest}}(s_1 : e_1)]$ ,  $D_1 = D_n(I_1)$ ,  $G_1 = \mathcal{K}^\dagger(D_1)$ .
11 if  $D_1$  is a good block then
12    $V_1$  =matrix formed eigenvector associated with positive eigenvalues.
13 else
14   Record that we have found a bad block with index  $I_b = I_1$ .
15 while have not found a bad block and  $e_1 < n - d - 1$  do
16    $s_2 = e_1 + 1, e_2 = \min(n - d - 1, s_2 + d)$ .
17    $\mathcal{I}_2 = [\mathcal{I}_g, I_{\text{rest}}(s_2 : e_2)]$ ,  $D_2 = D_n(I_2)$ .
18   if  $D_2$  is a good block then
19      $V_2$  =matrix formed eigenvector associated with positive eigenvalues.
20     Append the vector of all ones to  $V_1$  and  $V_2$ .
21     Partition  $V_1, V_2$  as  $V_1 := \begin{matrix} d+1 \\ d+1 \end{matrix} \begin{bmatrix} V_1' \\ V_1'' \end{bmatrix}$ ,  $V_2 := \begin{matrix} d+1 \\ |\mathcal{I}_2| - d - 1 \end{matrix} \begin{bmatrix} V_2'' \\ V_2' \end{bmatrix}$ .
22     Form the facial vector for  $D_n(I_1 \cup I_2)$  by  $V = \begin{bmatrix} V_1'' \\ V_1' \\ V_2'(V_2'')^\dagger V_1'' \end{bmatrix}$  or  $V = \begin{bmatrix} V_2'' \\ V_1'(V_1'')^\dagger V_2'' \\ V_2' \end{bmatrix}$ .
23     Remove the column of all ones. Update  $V_1 = V$ ,  $e_1 = e_2$ .
24   else
25     Record we found a bad block with index  $\mathcal{I}_b = \mathcal{I}_2$ .
26 if no bad block is found then
27   Run Algorithm 3 to obtain  $D_{\text{complete}}$ .
28   Find  $i_s, j_s$  to be the nonzero entry in  $D_n - D_{\text{complete}}$ ,  $\epsilon_s = (D_n - D_{\text{complete}})_{i_s, j_s}$ .
29 else
30   Run Algorithm 8, obtain the corrupted index  $i_s, j_s$ .
31   Run Line 4 to Line 32 of Algorithm 10.
```

Theorem 4.2.3. *Let $D \in \mathbb{E}^n$ with embedding dimension d and contains a principal submatrix of size $d + 1$ that is a uncorrupted good block. Let $i_c, j_c \in [n]$ such that $i_c < j_c$ and let $\epsilon \in \mathbb{R}$. Given $D_n = D + \epsilon E_{i_c j_c}$, Algorithm 10 returns i_s, j_s, ϵ_s such that $i_s = i_c, j_s = j_c, \epsilon_s = \epsilon$.*

Proof. Algorithm 10 first calls Algorithm 9 to obtain a small good/bad block. If a bad block is obtained, Algorithm 8 finds the corrupted position, and we can use Algorithm 9 to obtain a small good blocks from the remaining correct positions.

Once we have a small good block, it can be used as the rigid intersection for all the blocks. The correctness of the rest of the algorithm follows from Theorem 4.1.4. $\square$

4.2.3 An Example of a Problem with Multiple Solutions

In this section, we provide an interesting example we noticed when the EDM $D \in \mathbb{E}^n$ is unyielding with fixed rank, and but there does not exist a unique solution to the problem.

Note that the following example does not satisfy the assumption that there exists an uncorrupted good block. Let $d = 3$.

$$D_n = \begin{bmatrix} 0 & 4 & 16 & 8 & 6 & 14 \\ 4 & 0 & 4 & 4 & 6 & 6 \\ 16 & 4 & 0 & 8 & 14 & 6 \\ 8 & 4 & 8 & 0 & 18 & 14 \\ 6 & 6 & 14 & 18 & 0 & 20 \\ 14 & 6 & 6 & 14 & 20 & 0 \end{bmatrix}.$$

Let $G_n = \mathcal{K}^\dagger(D_n)$, we see that $\lambda(G_n) \approx (-0.7252, 0, 0, 4.7157, 7.4003, 13.2759)^T$. Therefore, D_n is not an EDM, as its corresponding $G_n = \mathcal{K}^\dagger(D_n)$ is not positive semidefinite.

However, there are 6 ways we can correct D_n to an EDM with embedding dimension 3. We can do so by changing $D_{4,5}, D_{4,6}$ or $D_{5,6}$, and for each position, there are two possible magnitude for the noise. Therefore, it is impossible to determine the actual corruption.

We can change $D_{4,5}$ in two different ways to obtain either of the following possible uncorrupted EDM:

$$D_1 = \begin{bmatrix} 0 & 4 & 16 & 8 & 6 & 14 \\ 4 & 0 & 4 & 4 & 6 & 6 \\ 16 & 4 & 0 & 8 & 14 & 6 \\ 8 & 4 & 8 & 0 & \boxed{14} & 14 \\ 6 & 6 & 14 & \boxed{14} & 0 & 20 \\ 14 & 6 & 6 & 14 & 20 & 0 \end{bmatrix}, \quad D_2 = \begin{bmatrix} 0 & 4 & 16 & 8 & 6 & 14 \\ 4 & 0 & 4 & 4 & 6 & 6 \\ 16 & 4 & 0 & 8 & 14 & 6 \\ 8 & 4 & 8 & 0 & \boxed{\frac{6}{5}} & 14 \\ 6 & 6 & 14 & \boxed{\frac{6}{5}} & 0 & 20 \\ 14 & 6 & 6 & 14 & 20 & 0 \end{bmatrix}.$$

Or we can change the (4,6) entry in the following ways:

$$D_3 = \begin{bmatrix} 0 & 4 & 16 & 8 & 6 & 14 \\ 4 & 0 & 4 & 4 & 6 & 6 \\ 16 & 4 & 0 & 8 & 14 & 6 \\ 8 & 4 & 8 & 0 & 18 & \boxed{\frac{42}{5}} \\ 6 & 6 & 14 & 18 & 0 & 20 \\ 14 & 6 & 6 & \boxed{\frac{42}{5}} & 20 & 0 \end{bmatrix}, \quad D_4 = \begin{bmatrix} 0 & 4 & 16 & 8 & 6 & 14 \\ 4 & 0 & 4 & 4 & 6 & 6 \\ 16 & 4 & 0 & 8 & 14 & 6 \\ 8 & 4 & 8 & 0 & 18 & \boxed{2} \\ 6 & 6 & 14 & 18 & 0 & 20 \\ 14 & 6 & 6 & \boxed{2} & 20 & 0 \end{bmatrix}.$$

Similarly, we can change the (5,6) entry:

$$D_5 = \begin{bmatrix} 0 & 4 & 16 & 8 & 6 & 14 \\ 4 & 0 & 4 & 4 & 6 & 6 \\ 16 & 4 & 0 & 8 & 14 & 6 \\ 8 & 4 & 8 & 0 & 18 & 14 \\ 6 & 6 & 14 & 18 & 0 & \boxed{14} \\ 14 & 6 & 6 & 14 & \boxed{14} & 0 \end{bmatrix}, \quad D_6 = \begin{bmatrix} 0 & 4 & 16 & 8 & 6 & 14 \\ 4 & 0 & 4 & 4 & 6 & 6 \\ 16 & 4 & 0 & 8 & 14 & 6 \\ 8 & 4 & 8 & 0 & 18 & 14 \\ 6 & 6 & 14 & 18 & 0 & \boxed{6} \\ 14 & 6 & 6 & 14 & \boxed{6} & 0 \end{bmatrix}.$$

This situation occurs in general when all but 3 points are in a manifold of dimension $d - 2$. In this case, points 1-3 are on a line in $\mathbb{R}^3$, and point 4-6 are off the line. For example, a configuration matrix for D_1 is

$$P_1 = \begin{bmatrix} 0 & 0 & 2 \\ 0 & 0 & 0 \\ 0 & 0 & -2 \\ -2 & 0 & 0 \\ 1 & 2 & 1 \\ 1 & -2 & -1 \end{bmatrix}.$$

Points 1,2,3 are in a line on the z -axis in $\mathbb{R}^3$, thus are in an affine space of dimension 1. Points 4,5,6 are off the line and in general position. If we perturb the distance between points 4 and 5, then we cannot find a position for point 6 that can preserve both its original distances with 4 and 5. However, there exists two locations for point 6 such that it can maintain the distance with point 5. Then, we can take the error to be in the distance between point 4 and 6 and can fix the corrupted EDM by changing the (4,6) entry instead.

Even though this example is only $n = 6$, we can increase the size of this problem by adding any number of points on the z -axis in the generating matrix P_1 .

This example illustrates a new class of problem, in addition to yielding, where the problem of identifying the position and location of the noise in a corrupted EDM with a single error is impossible.

In Section 4.2.1 and Section 4.2.2, we showed that if there exists an uncorrupted good block, then we can solve the problem exactly. For future study, it would be interesting to see if problems that do not belong to the two unsolvable cases, and do not contain an uncorrupted good block can be solved, and how to solve them.

Chapter 5

Time Complexity and Numerics

We analyse the theoretical complexity and demonstrate results from numerical experimentation on our algorithms in this chapter.

5.1 Time Complexity

We analyse the big O runtime for each algorithm in Section 4.1, and the runtime to find a good block in Section 4.2.1 when we don't assume the EDM is in general position. Suppose the corrupted EDM is $D_n \in \mathbb{S}^n$ and has embedding dimension d . Recall that n is assumed to be much greater than d , so we only consider the asymptotic behaviour as $n \rightarrow \infty$ and treat d as a constant.

5.1.1 Runtime for BEV

In each iteration, we divide an $n \times n$ matrix into two blocks of size $\frac{n+d+2}{2}$, and compute the spectral decomposition of their Gram matrices to obtain an exposing vector. The runtime is $2O\left(\left(\frac{n+d+2}{2}\right)^3\right) = 2O(n^3)$. If both of the principal diagonal blocks are proper EDMs, then we add them to obtain the exposing vector for D .

After obtaining the centered exposing vector Z , we calculate the facial vector V as an orthogonal matrix whose column forms a basis of $\text{null}\left(\begin{bmatrix} Z^T \\ e^T \end{bmatrix}\right)$. This step again has a runtime of $O(n^3)$.

Finally, Algorithm 3 runs in constant time since we are only solving a system of equation where the size of the system is a function of d . Therefore, $O(n^3)$ is the runtime for one iteration.

There are $\frac{n(n-1)}{2}$ choices for the position of the corrupted entry. There are $\left(\frac{n-d-2}{2}\right)^2$ points in the top right block outside of the two principal submatrices. Therefore, the probability the corrupted point is in one of the principal submatrices is

$$1 - \left(\frac{n-d-2}{2}\right)^2 \bigg/ \frac{n(n-1)}{2} \approx \frac{1}{2}$$

when n is large. Therefore, with probability almost $\frac{1}{2}$, we continue with the divide and conquer algorithm, and work with the smaller matrix of size $\frac{n+d+2}{2}$, reducing the size of the problem by

approximately a half. After i divisions, the size of the matrix is $\frac{n+2^{i-1}(d+2)}{2^i}$. Let $T(n)$ be the total runtime of the algorithm, and let $f(n)$ be the runtime of one iteration. We have the recurrence

$$T(n) = \frac{1}{2}T\left(\frac{n}{2}\right) + f(n) = \frac{1}{2}T\left(\frac{n}{2}\right) + O(n^3).$$

$O(n^3)$ dominate the recurrence relationship. Therefore, the total runtime is $O(n^3)$.

5.1.2 Runtime for MBFV

In MBFV method, we calculate the spectral decomposition of the Gram matrix for each consecutive principal diagonal block of size $2d + 2$. Building the facial vector from two overlapping blocks only requires calculating the generalized inverse of a small matrix. Therefore, calculating the facial vector of small blocks is done in constant time, and the work to build the full facial vector V has runtime $O(n)$. Calculating R such that $D = \mathcal{K}(VRV^T)$ is also done in constant time, as we only need to solve a small system of equations. Therefore, the total runtime is $O(n)$.

We note that if we were to use a bisection method, then we need to calculate the facial vector of a matrix size $\frac{n+d+2}{2}$, and we would have a runtime of $O(n^3)$. Therefore, it is more efficient to divide into blocks with small size.

5.1.3 Runtime for SBGT

We calculate the Gale matrix for each consecutive principal diagonal block of size $d + 2$, and build up the Gale matrix N for D . This step can be built in constant time $O(n)$. As noted before, the Gale matrix N and the exposing vector Z for D are connected by $Z = NN^T$. Therefore, we see that by dividing into small blocks, we can build up the exposing vector/Gale matrix more efficiently compared to dividing into large blocks. Furthermore, there is a higher probability the corrupted distance is outside of the principal diagonal blocks when the size of each block is small.

After obtaining the Gale matrix N , we calculate the facial vector V of D . This is the most time consuming step, with runtime of $O(n^3)$. Thus, the SBGT algorithm has a total runtime of $O(n^3)$.

5.1.4 Runtime for Identifying a Good Block

To identify a good block, we calculate the Gale matrix for all consecutive principal diagonal blocks of size $d + 1$. This step has runtime $O(n)$.

We identify points that are not in the affine hull of other points in the small block, and then repeat the process on these points. In analysing the algorithm, we show the size of the index list we work with must decrease or reach a small block size and will terminate in a small number of steps. Therefore, the runtime for the total algorithm is $O(n^2)$.

In numerical testing, we usually can identify a small block with embedding dimension d after only 1 or 2 iterations, and solving the hard case problem is only slightly slower than solving the problem under the general position assumption. It is unclear whether there is an upper bound tighter than $O(n^2)$ for this algorithm.

5.2 Numerics

In this section, we tests the performance of these algorithm using MATLAB. First, we generate random problems and compare the three methods of facial reduction. Randomly generated problems are expected to be in general position.

All tests on conducted in MATLAB R2024a on a Dell PowerEdge R640 machine with two Intel Xeon Gold 6244 8-core 3.6 GHz (Cascade Lake) CPU, and 192 GB Memory, using the fast linux server of the math faculty computing facility at the University of Waterloo.

In the tables, we run 1000 problems under each data specification. The output includes:

- tol-met: Percentage of problems we find the position of the noise and the relative error of between the noise we found and the actual noise met the tolerance of 10^{-6} .
- rel-error: The average relative error of the magnitude of the noise.
- time: The average time it took to solve one problem.

Data specifications			BEV			MBFV			SBGT		
n	d	noise	tol-attained	rel-error	time(s)	tol-attained	rel-error	time(s)	tol-attained	rel-error	time(s)
1000	5	0.196	100%	1.58e-10	0.077	100%	8.86e-12	0.019	100%	9.82e-04	0.080
2000	5	-0.214	100%	1.22e-10	0.350	100%	1.35e-11	0.060	100%	8.25e-12	0.322
3000	5	-0.273	100%	1.58e-10	0.880	100%	1.43e-11	0.152	100%	2.96e-11	0.759
4000	5	0.051	100%	2.73e-10	1.659	100%	1.84e-11	0.253	99%	5.13e-04	1.311
5000	5	0.219	100%	3.50e-10	2.876	100%	1.77e-11	0.378	100%	1.43e-11	2.094
6000	5	-0.077	100%	1.90e-10	4.613	100%	1.84e-11	0.530	100%	1.08e-11	3.154
7000	5	0.481	100%	2.56e-10	7.212	100%	2.44e-11	0.706	100%	1.18e-11	4.525
8000	5	0.185	100%	3.50e-10	10.158	100%	2.54e-11	0.923	100%	1.38e-11	6.331
9000	5	-0.019	100%	3.89e-10	14.037	100%	2.37e-11	1.177	100%	1.24e-11	8.509
10000	5	-0.108	100%	5.26e-10	19.257	100%	2.54e-11	1.455	100%	1.60e-11	11.144
11000	5	-0.157	100%	2.90e-10	23.665	100%	2.87e-11	1.742	100%	1.50e-11	14.079
12000	5	0.229	100%	2.85e-10	32.461	100%	3.08e-11	2.107	100%	1.58e-11	17.961
13000	5	-0.061	100%	4.82e-10	41.664	100%	3.27e-11	2.543	100%	1.34e-11	22.221
14000	5	-0.440	100%	4.74e-10	49.687	100%	3.27e-11	2.781	100%	3.77e-11	26.696
15000	5	-0.102	100%	4.12e-10	61.680	100%	3.49e-11	3.273	100%	1.72e-11	32.358
16000	5	0.238	100%	8.05e-10	62.119	100%	3.21e-11	3.672	100%	1.68e-11	37.935
17000	5	-0.318	100%	6.44e-10	72.432	100%	3.73e-11	4.271	100%	2.05e-11	44.683
18000	5	-0.325	100%	7.99e-10	82.122	100%	3.53e-11	4.521	100%	1.66e-11	51.527
19000	5	0.032	100%	5.72e-10	90.706	100%	3.26e-11	4.981	100%	1.84e-11	58.178
20000	5	0.032	100%	4.43e-10	99.551	100%	4.16e-11	4.803	100%	1.99e-11	62.434

Table 5.1: Average of 200 problems; $n = 1000$ to $20,000$, $d = 5$

We successfully recovered the original EDM accurately, with the solution times consistent with the theoretical runtime. Among the three method, only SBGT do not met the required tolerance in some cases. Since SBGT uses the least amount of information from D_n to recover D , it is subject to more variability, and can be less accurate sometime.

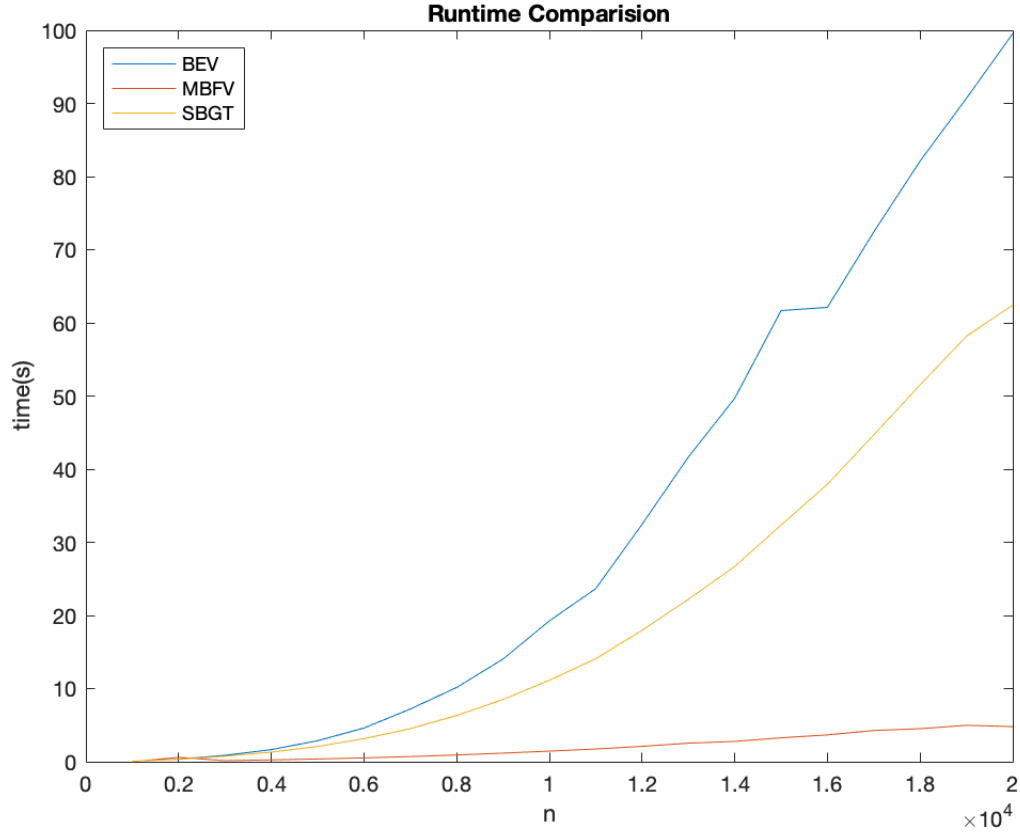


Figure 5.1: Runtime vs. the problem size n

In terms of runtime, as expected, BEV is slightly slower than SGBT, and MBFV is extremely fast. As shown in the table, MBFV solves a 20,000 by 20,000 problem in under 5 seconds.

In the following table, we compare the MBFV method with the adjusted method for solving problems not in general position. We generated random problems and shifted many points to lie in a lower dimensional linear manifold for the adjusted MBFV method.

Data specifications			General Position MBFV			Adjusted MBFV		
n	d	noise	tol-attained	rel-error	time(s)	tol-attained	rel-error	time(s)
100	8	-0.157	100%	3.26e-12	0.002	100%	4.09e-13	0.005
200	4	0.229	100%	2.56e-12	0.003	100%	4.89e-14	0.006
300	4	-0.061	100%	3.34e-12	0.004	100%	4.71e-14	0.010
400	6	-0.440	100%	5.08e-12	0.005	100%	1.39e-13	0.014
500	8	-0.102	100%	7.88e-12	0.006	100%	2.47e-13	0.022
2000	5	0.238	100%	1.21e-11	0.047	100%	6.46e-14	0.091
4000	10	-0.318	100%	3.20e-11	0.217	100%	3.13e-13	0.376
6000	8	-0.325	100%	3.24e-11	0.442	100%	1.91e-13	0.654
8000	6	0.032	100%	2.70e-11	0.748	100%	1.05e-13	0.990
10000	5	0.032	100%	2.63e-11	1.256	100%	6.15e-14	1.559

Table 5.2: Average of 1000 problems; n from 100 to 10000, d random positive integer between 2 and 10

We solved all problem accurately and efficiently. The hard case MBFV method is still highly efficient, and only has a slight increase in runtime compare to the original MBFV. Thus, the method we developed to identify a small good block is very efficient.

Chapter 6

Conclusion

In this thesis, we explored the problem of identifying and correcting a single corrupted entry in a Euclidean Distance Matrix (EDM). Using results from solving the Euclidean Distance Matrix Completion Problem (EDMCP), we developed efficient divide-and-conquer algorithms using facial reduction techniques to detect the position and magnitude of the noise. We introduced three different facial reduction methods: exposing vector, facial vector, and Gale transform.

Additionally, we investigated yielding entries in an EDM and addressed the case where the points generating the problem are not in general position. This involved developing an algorithm to search for a small principal diagonal block with the desired rank.

We performed thorough numerical experiments using MATLAB, demonstrating that our methods are highly efficient even on large-scale problems.

References

- [1] S. AL-HOMIDAN AND H. WOLKOWICZ, *Approximate and exact completion problems for Euclidean distance matrices using semidefinite programming*, Linear Algebra Appl., 406 (2005), pp. 109–141. [1](#), [13](#), [14](#), [15](#)
- [2] A. ALFAKIH, *On dimensional rigidity of bar-and-joint frameworks*, Discrete Appl. Math., 155 (2007), pp. 1244–1253. [18](#)
- [3] ———, *Euclidean distance matrices and their applications in rigidity theory*, Springer, Cham, 2018. [16](#)
- [4] ———, *On yielding and jointly yielding entries of Euclidean distance matrices*, Linear Algebra Appl., 556 (2018), pp. 144–161. [17](#), [26](#), [27](#)
- [5] A. ALFAKIH, W. L. JUNG, H. WOLKOWICZ, AND T. XU, *Single element error correction in a Euclidean distance matrix*, 2024. [iii](#), [27](#)
- [6] A. ALFAKIH, A. KHANDANI, AND H. WOLKOWICZ, *Solving Euclidean distance matrix completion problems via semidefinite programming*, Comput. Optim. Appl., 12 (1999), pp. 13–30. [1](#)
- [7] M. BAKONYI AND C. R. JOHNSON, *The Euclidean distance matrix completion problem*, SIAM J. Matrix Anal. Appl., 16 (1995), pp. 646–654. [1](#), [19](#)
- [8] G. BARKER AND D. CARLSON, *Cones of diagonally dominant matrices*, Pacific J. of Math., 57 (1975), pp. 15–32. [11](#)
- [9] J. BORWEIN AND H. WOLKOWICZ, *Facial reduction for a cone-convex programming problem*, J. Austral. Math. Soc. Ser. A, 30 (1980/81), pp. 369–380. [19](#)
- [10] S. L. S. L. V. CAMPBELL AND C. D. C. D. MEYER, *Generalized inverses of linear transformations*, Classics in applied mathematics ; 56, Society for Industrial and Applied Mathematics SIAM, 3600 Market Street, Floor 6, Philadelphia, PA 19104, Philadelphia, Pa, 2009. [7](#)
- [11] R. J. CARON AND N. GOULD, *Finding a positive semidefinite interval for a parametric matrix*, Linear Algebra Appl., 76 (1986), pp. 19–29. [25](#)
- [12] G. CRIPPEN AND T. HAVEL, *Distance geometry and molecular conformation*, vol. 15 of Chemometrics Series, Research Studies Press Ltd., Chichester, 1988. [1](#)
- [13] F. CRITCHLEY, *On certain linear mappings between inner-product and squared distance matrices*, Linear Algebra Appl., 105 (1988), pp. 91–107. [14](#)

- [14] D. DRUSVYATSKIY, N. KRISLOCK, Y.-L. C. VORONIN, AND H. WOLKOWICZ, *Noisy Euclidean distance realization: robust facial reduction and the Pareto frontier*, SIAM J. Optim., 27 (2017), pp. 2301–2331. [1](#), [23](#)
- [15] D. DRUSVYATSKIY, G. PATAKI, AND H. WOLKOWICZ, *Coordinate shadows of semidefinite and Euclidean distance matrices*, SIAM J. Optim., 25 (2015), pp. 1160–1178. [23](#)
- [16] D. DRUSVYATSKIY AND H. WOLKOWICZ, *The many faces of degeneracy in conic optimization*, Foundations and Trends® in Optimization, 3 (2017), pp. 77–170. [11](#)
- [17] J. GOWER, *Properties of Euclidean and non-Euclidean distance matrices*, Linear Algebra and its Applications, 67 (1985), pp. 81–97. [1](#), [16](#)
- [18] B. GRÜNBAUM, *Convex polytopes*, Interscience Publishers John Wiley & Sons, Inc., New York, 1967. With the cooperation of Victor Klee, M. A. Perles and G. C. Shephard. Pure and Applied Mathematics, Vol. 16. [16](#)
- [19] R. A. HORN AND C. R. JOHNSON, *Matrix Analysis*, Cambridge University Press, 2 ed., 2012. [8](#)
- [20] N. KRISLOCK AND H. WOLKOWICZ, *Explicit sensor network localization using semidefinite representations and facial reductions*, SIAM J. Optim., 20 (2010), pp. 2679–2708. [1](#), [20](#), [21](#), [22](#)
- [21] KRISLOCK, NATHAN, *Semidefinite Facial Reduction for Low-Rank Euclidean Distance Matrix Completion*, PhD thesis, 2010. [11](#)
- [22] L. PATTERSSON, S. SREMAC, F. WANG, AND H. WOLKOWICZ, *Noisy Euclidean distance matrix completion with a single missing node*, J. Global Optim., 75 (2019), pp. 973–1002. [1](#)
- [23] R. ROCKAFELLAR, *Convex analysis*, Princeton Landmarks in Mathematics, Princeton University Press, Princeton, NJ, 1997. Reprint of the 1970 original, Princeton Paperbacks. [4](#)
- [24] J. SAXE, *Embeddability of weighted graphs in k -space is strongly NP-hard*, in Seventeenth Annual Allerton Conference on Communication, Control, and Computing, Proceedings of the Conference held in Monticello, Ill., October 10–12, 1979, Urbana, 1979, University of Illinois Department of Electrical Engineering, pp. xiv+1036. Proceedings of the International School of Physics “Enrico Fermi”, LXX*. [19](#)
- [25] I. SCHOENBERG, *Remarks to Maurice Fréchet’s article “Sur la définition axiomatique d’une classe d’espace distanciés vectoriellement applicable sur l’espace de Hilbert”*, Ann. Math., 36 (1935), pp. 724–732. [1](#), [16](#)
- [26] H. SONG, *Positive semidefinite intervals for matrix pencils*, Master’s thesis, University of Windsor, 2004. [25](#)
- [27] G. STRANG, *The fundamental theorem of linear algebra*, The American mathematical monthly, 100 (1993), pp. 848–. [7](#)
- [28] L. TUNCEL, *Polyhedral and semidefinite programming methods in combinatorial optimization*, Fields Institute monographs ; 27, American Mathematical Society, Providence, RI, 2010. [9](#)
- [29] K. Q. WEINBERGER AND L. K. SAUL, *Unsupervised learning of image manifolds by semidefinite programming*, CVPR’04, USA, 2004, IEEE Computer Society, p. 988–995. [1](#)

- [30] H. WOLKOWICZ, R. SAIGAL, AND L. VANDENBERGHE, eds., *Handbook of semidefinite programming*, International Series in Operations Research & Management Science, 27, Kluwer Academic Publishers, Boston, MA, 2000. Theory, algorithms, and applications. [11](#)